



CENTRO UNIVERSITÁRIO CARIOCA

GABRIEL MATHEUS DE CARVALHO FERNANDES

RENATO NASCIMENTO DA SILVA

APLICAÇÃO PARA EDUCAÇÃO FINANCEIRA DO INVESTIDOR INICIANTE

RIO DE JANEIRO

2020

GABRIEL MATHEUS DE CARVALHO FERNANDES

RENATO NASCIMENTO DA SILVA

APLICAÇÃO PARA EDUCAÇÃO FINANCEIRA DO INVESTIDOR INICIANTE

Trabalho de Conclusão de Curso apresentado
ao Centro Universitário Carioca, como
requisito parcial para obtenção do grau de
Bacharel em Ciência da Computação

Orientador: Prof. D.Sc Sérgio Assunção
Monteiro

RIO DE JANEIRO

2020

S 586a

Silva, Renato Nascimento da.

Aplicação para educação financeira do investidor
Iniciante / Renato Nascimento da Silva; Gabriel
Matheus de Carvalho Fernandes. Rio de
Janeiro, 2020.

64 f.

Orientador: Sérgio Assunção Monteiro.

Trabalho de Conclusão de Curso (Graduação em
Ciência da Computação) – Centro Universitário
Unicarioca. Rio de Janeiro, 2020.

1. Educação financeira. 2. Investidor iniciante. I.
Fernandes, Gabriel Matheus de Carvalho. II.
Monteiro, Sérgio Assunção. III. Título.

CDD 332

GABRIEL MATHEUS DE CARVALHO FERNANDES

RENATO NASCIMENTO DA SILVA

APLICAÇÃO PARA EDUCAÇÃO FINANCEIRA DO INVESTIDOR INICIANTE

Trabalho de Conclusão de Curso
apresentado ao Centro
Universitário Carioca, como
requisito parcial para obtenção do
grau de Bacharel em Ciência da
Computação

BANCA EXAMINADORA

Prof. Sérgio Monteiro Assunção, D.Sc - Orientador
Centro Universitário Carioca

Prof. André Luiz Avelino Sobral, M.Sc
Centro Universitário Carioca

Prof. Alberto Tavares da Silva, D.Sc
Centro Universitário Carioca

AGRADECIMENTOS

Gabriel

Agradecimento ao amor da minha vida e melhor amiga Julia Mendonça, por ter me ajudado em todos os momentos difíceis e me motivado sempre que necessário. Se eu consegui concluir esse trabalho foi também por sua causa.

Renato

Agradecimento a minha noiva e companheira para a vida inteira, Gizelda, por sempre estar ao meu lado, me apoiando a cada passo dado. Também agradeço a minha amada mãe, Lenira, que com todo o seu amor e dedicação me possibilitou ser a pessoa que sou hoje.

Agradecemos ao orientador Sérgio Monteiro por seu auxílio em todo o processo de desenvolvimento do projeto, bem como por estar sempre disponível para nos aconselhar sobre a melhor forma de agir em todas as situações.

RESUMO

No Brasil, muitas pessoas deixam de investir seu dinheiro em locais mais lucrativos por desconhecimento ou medo de ter prejuízo. Neste contexto, se faz necessário ferramentas que auxiliem o possível investidor a transformar este receio de investir, em segurança. Este trabalho procura fazer com que o investidor consiga unir a teoria necessária para dar seus primeiros passos neste território que é muitas vezes incerto, e ao mesmo tempo aplique na prática seus conhecimentos adquiridos. A partir do momento em que o investidor iniciante tiver todas as informações necessárias para começar ele terá plena certeza de que investir não é algo difícil, e com isso romperá a barreira que o impedia de dar início a sua vida investidora

Palavras chave: investidor, iniciante, educação, primeiros passos.

ABSTRACT

In Brazil, many people stop investing their money in more profitable places due to lack of knowledge or fear of the money loss. In this context, it is necessary to have tools that help the possible investor to transform this fear of investment, in a security feeling. This work seeks to make the investor able to unite the necessary theory to take his first steps in this territory that is often uncertain, and at the same time apply in practice his acquired knowledge. As soon as the novice investor has all the necessary information to start, he will be completely sure that investing is not something difficult, and thus breaking through the barrier that prevented him from starting his investing life.

Keywords: investor, beginner, education, first steps

LISTA DE ILUSTRAÇÕES

Figura 1 - Trecho do código de extração: CDB, LCI, LCA.....	22
Figura 2 - Exemplo da estrutura da tabela em extração: CDB, LCI, LCA.....	22
Figura 3 -Trecho do código de extração: Tesouro Direto	23
Figura 4 - Exemplo da estrutura da tabela em extração: Tesouro Direto	23
Figura 5 - Trecho do código de extração: Carteira recomendada	24
Figura 6 - Exemplo da estrutura da tabela em extração: Carteira recomendada	24
Figura 7 - Trecho do código de transformação: CDB, LCI, LCA.....	25
Figura 8 - Exemplo da estrutura da tabela em transformação: CDB, LCI, LCA	25
Figura 9 - Trecho do código de transformação: Tesouro direto.....	26
Figura 10 - Exemplo da estrutura da tabela em transformação: Tesouro direto	26
Figura 11 - Trecho do código de transformação: Carteira recomendada	27
Figura 12 - Exemplo da estrutura da tabela em transformação: Carteira recomendada	27
Figura 13 - Trecho do código de carga das tabelas	28
Figura 14 - Função download_opcoesRF	28
Figura 15 - Função download_opcoesRV	29
Figura 16 - Função ir_regressivo.....	29
Figura 17 - Função consulta_opcoes_CDB.....	31
Figura 18 - Exemplo da estrutura da tabela gerada pela função consulta_opcoes_CDB	31
Figura 19 - Função consulta_opcoes_LCI.....	32
Figura 20 - Exemplo da estrutura da tabela gerada pela função consulta_opcoes_LCI	32
Figura 21 - Função consulta_opcoes_LCA	32
Figura 22 - Exemplo da estrutura da tabela gerada pela função consulta_opcoes_LCA	32
Figura 23 - Função consulta_opcoes_tesouro	33
Figura 24 - Exemplo da estrutura da tabela gerada pela função consulta_opcoes_tesouro	33
Figura 25 - Função criar_resultado_RF_excel	35
Figura 26 - Função criar_conta	37
Figura 27 - Tela inicial	40

Figura 28 - Tela de cadastro	41
Figura 29 - Tela de formulário	42
Figura 30 - Tela de login	43
Figura 31 - Tela de estudo sobre renda fixa.....	44
Figura 32 - Tela de estudo sobre CDB.....	45
Figura 33 - Tela de estudo sobre LCI.....	46
Figura 34 - Tela de estudo sobre LCA	47
Figura 35 - Tela de estudo sobre tesouro direto.....	48
Figura 36 - Tela de estudo sobre FGC.....	49
Figura 37 - Tela de estudo sobre indexadores	50
Figura 38 - Tela de opções de investimento em renda fixa	52
Figura 39 - Tela de estudo sobre renda variável	53
Figura 40 - Tela de estudo sobre como começar a investir na bolsa	54
Figura 41 - Tela de estudo sobre como analisar uma ação	55
Figura 42 - Tela de estudo sobre Stop	56
Figura 43 - Tela de bibliografia para estudo sobre renda variável.....	57
Figura 44 - Tela de avisos sobre renda variável.....	58
Figura 45 - Tela de considerações finais.....	59
Figura 46 - Tabela de opções adequadas em renda fixa: CDB.....	60
Figura 47 - Tabela de opções adequadas em renda fixa: LCI.....	60
Figura 48 - Tabela de opções adequadas em renda fixa: LCA	61
Figura 49 - Tabela de opções adequadas em renda fixa: tesouro	61
Figura 50 - Tabela de opções adequadas em renda variável: carteira recomendada	61
Figura 51 – Protótipo de tela	62

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVO DO TRABALHO	13
1.2	ESTRUTURA DO TRABALHO	13
2	TECNOLOGIAS, TERMOS TÉCNICOS E LEIS	14
2.1	PYTHON	14
2.2	PYQT	14
2.3	QT DESIGNER	14
2.4	PANDAS	14
2.5	INVESTIDOR.....	14
2.5.1	Investidor Conservador	15
2.5.2	Investidor Moderado	15
2.5.3	Investidor Arrojado.....	15
2.6	EMISSORES.....	15
2.7	RENTA FIXA	15
2.7.1	Letra de Crédito do Agronegócio	16
2.7.2	Letra de Crédito Imobiliário	16
2.7.3	Certificado de Depósito Bancário.....	16
2.8	FUNDO GARANTIDOR DE CRÉDITO.....	16
2.9	TESOURO DIRETO.....	17
2.9.1	Tesouro Prefixado.....	17
2.9.2	Tesouro Ligado a Variação da Inflação ou Taxa de Juros Básica	17
2.9.3	Tesouro SELIC	17
2.10	SELIC.....	17
2.11	INFLAÇÃO.....	18
2.12	RENTA VARIÁVEL.....	18
2.13	AÇÕES.....	18
2.14	BOLSA DE VALORES.....	18
2.15	BOVESPA	18
2.16	COMISSÃO DE VALORES MOBILIÁRIOS	19
2.17	ASSOCIAÇÃO BRASILEIRA DAS ENTIDADES DOS MERCADOS FINANCEIRO E DE CAPITAIS	19
2.18	CÓDIGO ANBIMA DE REGULAÇÃO E MELHORES PRÁTICAS	19

3	DESENVOLVIMENTO DA METODOLOGIA DO TRABALHO	20
3.1	O PROBLEMA	20
3.2	PROPOSTA.....	20
3.3	ANÁLISE DOS CÓDIGOS DESENVOLVIDOS.....	21
3.3.1	Processo de ETL	21
3.3.1.1	Extração	21
3.3.1.1.1	Extração CDB, LCI, LCA	21
3.3.1.1.2	Extração Tesouro Direto	22
3.3.1.1.3	Extração carteira sugerida.....	23
3.3.1.2	Transformação	24
3.3.1.2.1	Transformação CDB, LCI, LCA	24
3.3.1.2.2	Transformação Tesouro direto	25
3.3.1.2.3	Transformação carteira recomendada	26
3.3.1.3	Carga.....	27
3.3.2	Função download_opcoesRF	28
3.3.3	Função download_opcoesRV	28
3.3.4	Função ir_regressivo	29
3.3.5	Função consulta_opcoes_CDB.....	30
3.3.6	Funções consulta_opcoes_LCI e consulta_opcoes_LCA	31
3.3.7	Função consulta_tesouro.....	33
3.3.8	Função criar_resultado_RF_excel	33
3.3.9	Função criar_conta	36
3.4	FLUXOGRAMA DE EXECUÇÃO E MODELO DE DADOS.....	37
3.4.1	Fluxograma de Execução	38
3.4.2	Modelo de dados	39
4	INTERAÇÃO DO USUÁRIO COM A APLICAÇÃO.....	40
4.1	TELA INICIAL	40
4.2	CADASTRO.....	41
4.3	FORMULÁRIO.....	42
4.4	LOGIN.....	43
4.5	ESTUDO SOBRE RENDA FIXA	44
4.5.1	Estudo sobre CDB	45

4.5.2	Estudo sobre LCI	46
4.5.3	Estudo sobre LCA.....	47
4.5.4	Estudo sobre tesouro direto	48
4.5.5	Estudo sobre FGC	49
4.5.6	Estudo sobre indexadores	50
4.6	OPÇÕES DE INVESTIMENTO EM RENDA FIXA	51
4.7	ESTUDO SOBRE RENDA VARIÁVEL.....	53
4.7.1	ESTUDO SOBRE COMO COMEÇAR A INVESTIR NA BOLSA.....	54
4.7.2	ESTUDO SOBRE COMO ANALISAR UMA AÇÃO	55
4.7.3	ESTUDO SOBRE STOP	56
4.7.4	SUGESTÃO DE APROFUNDAMENTO SOBRE RENDA VARIÁVEL	57
4.8	AVISOS SOBRE RENDA VARIÁVEL	58
4.9	INFORMAÇÕES FINAIS	59
4.10	ARQUIVOS GERADOS.....	60
5	CONSIDERAÇÕES FINAIS	61
5.1	LIMITAÇÕES E SUJESTÕES DE TRABALHOS FUTUROS.....	62
5.2	REPOSITÓRIO GITHUB	63
	REFERÊNCIAS BIBLIOGRÁFICAS	64

1 INTRODUÇÃO

O Brasil atualmente é um país considerado economicamente estável, porém podemos dizer que esta é uma condição relativamente nova. Até pouco tempo atrás vivíamos em uma situação de hiperinflação com preços que variavam várias vezes ao longo do mesmo dia, produtos tinham seu valor atualizado constantemente, às vezes, até enquanto estava sendo comprado. Esta situação de pouca solidez fazia com que os cidadãos brasileiros tivessem que gastar o dinheiro o mais rápido possível assim que o recebessem, pois a cada dia que passasse a variação do valor da moeda fazia com que menos coisas pudessem ser compradas.

Este cenário teve como efeito a consolidação no ilusório nacional de que manter dinheiro guardado causava prejuízo.

Outro acontecimento recente foi o pacote econômico batizado de Brasil Novo e popularmente conhecido como “Plano Color”. Este pacote trazia uma série de drásticas medidas econômicas para tentar conter a hiperinflação, a mais famosa de todas essas medidas ficou conhecida como “confisco da poupança”, por meio dela todos os investidores da caderneta de poupança teriam direito a sacar no máximo 50 mil cruzados novos de seu dinheiro investido e o restante seria devolvido em 12 meses com correção monetária e acréscimo de juros. Foi uma medida impopular que contribui muito para que o brasileiro deixasse de ver com bons olhos o fato de alguém economizar para utilizar aquele recurso em investimentos econômicos.

Como resultado desses e mais outros fatores econômicos brasileiros, a mentalidade de nossa população tende a ser, usar todo o dinheiro recebido sem guardar nada para o futuro ou deixa-lo guardado na opção mais popular, porém de pouca rentabilidade que é a poupança. Nosso povo não investe por receio de que aqueles antigos problemas do passado ressurgam nessa atual conjuntura, e as novas gerações são aconselhadas por seus antecessores a fazer a mesma coisa, tornando esse costume uma espiral eterna.

Segundo a ANBIMA (2020) apenas 38% dos brasileiros conseguiram ou quiseram poupar algum dinheiro no ano de 2019 e dentre esses, apenas 48% investiram em produtos financeiros (renda fixa, títulos públicos, ações e etc.).

Por outro lado, este mesmo estudo comprovou que 66% dos brasileiros que aplicaram seu dinheiro em algum produto financeiro em anos anteriores, reinvestiram em 2019, ou seja, constatamos aqui que o importante é dar o primeiro passo.

1.1 Objetivo do Trabalho

O objetivo deste trabalho é dar ao cidadão comum conhecimento para ser introduzido ao mundo das aplicações financeiras. O aplicativo tem o propósito de ser um caminho amigável em que um leigo percorre todas as fases necessárias de conhecimento para se tornar um investidor iniciante. Ele deve conhecer todos os fundamentos básicos das principais aplicações financeiras disponíveis no mercado brasileiro e ao mesmo tempo será aconselhado a investir de forma opcional caso se sinta seguro nestas mesmas aplicações.

O trabalho é uma forma de romper a barreira de desconfiança de um cidadão comum que quer começar a investir, mas não sabe por onde começar. É uma junção de conhecer tudo que precisa na teoria e ao mesmo tempo ter condições de aplicar na prática todos esses conceitos que aprendeu.

1.2 Estrutura do Trabalho

O trabalho está dividido em cinco capítulos. O primeiro capítulo é introdutório e apresenta a divisão do trabalho.

O segundo capítulo é um rol das tecnologias utilizadas no trabalho e também de todas as instituições financeiras que foram consultadas para a construção deste sistema, bem como as leis que foram utilizados destas instituições como regras de negócio na nossa aplicação para adequá-la a todos os requisitos legais necessários.

O terceiro capítulo contém o problema e a importância que tem o trabalho como um todo, assim como a importância do desenvolvimento dos códigos, tendo a estrutura dos códigos detalhada.

O quarto capítulo simula o usuário utilizando a aplicação, demonstrando pontos relevantes onde são aplicados os códigos mostrados no terceiro capítulo.

No quinto capítulo são feitas as considerações finais, assim como são descritos trabalhos futuros.

2 TECNOLOGIAS, TERMOS TÉCNICOS E LEIS

2.1 Python

MENEZES (2010, p.24) define Python como uma linguagem de software livre muito interessante devido a sua simplicidade e clareza. Embora seja uma linguagem simples, também é uma linguagem poderosa, e que pode ser usada para muitos fins como por exemplo administrar sistemas e desenvolver projetos de grande porte. Além de ser uma linguagem clara e objetiva que vai direto ao ponto, sem rodeios.

2.2 PyQt

PyQt é uma biblioteca de linguagem python específica para criação de aplicativos em formato desktop. Ele auxilia a criação de janelas de forma mais simples do que utilizando python puro.

2.3 Qt Designer

Qt Designer é uma ferramenta de design e construção de interfaces gráficas. O programa torna mais fácil montar código compatível com a biblioteca PyQt, pois o usuário constrói a janela graficamente e depois converte o design construído em código.

2.4 Pandas

Pandas é uma biblioteca livre sob licença BSD utilizada para análise e manutenção de dados, possui métodos para o tratamento de tabelas numéricas e outros tipos de listas.

2.5 Investidor

Segundo NETO, SANTOS E MELLO (2019, p. 6) “os investidores são credores dos emissores dos títulos e, em troca de remuneração, ou juros, emprestam seus recursos temporariamente, a um tomador.” Se dividem em três tipos: Conservador, Moderado e Arrojado

2.5.1 Investidor Conservador

O investidor conservador segundo FINANCEONE (2017, p.27) é o que preserva seus recursos, não assume riscos que comprometam seu patrimônio, não tolera perdas e falta de liquidez, geralmente é um investidor iniciante, tem objetivos de curto e médio prazo, prefere investimentos com retornos previsíveis e normalmente busca investimentos de renda fixa.

2.5.2 Investidor Moderado

O investidor moderado se diferencia do conservador, pois assume riscos um pouco maiores em troca de uma rentabilidade maior. Segundo FAYH (2020) o investidor moderado é aquele que tolera um pouco mais de risco a fim de conseguir uma rentabilidade igualmente maior. A menor liquidez e perdas controladas são aceitas, porém não se abre mão da proteção patrimonial.

2.5.3 Investidor Arrojado

O investidor arrojado é aquele que está disposto a correr qualquer tipo de risco em busca da melhor rentabilidade possível. Ele não se importa de perder todo o seu capital investido e nem se importa em ter proteção patrimonial, além de trabalhar com qualquer tipo de aplicação.

2.6 Emissores

Os emissores são organizações autorizadas pelo governo que emitem títulos com promessa de pagamento mais juros para aqueles que o adquirirem por tempo definido em documento formal.

2.7 Renda Fixa

Segundo NETO, SANTOS E MELLO (2019, p. 6) “Renda Fixa é um tipo de investimento em que rendimentos reais, nominais ou indexados às taxas flutuantes são recebidos em intervalo de tempo regulares e definidos em documentos formais.

2.7.1 Letra de Crédito do Agronegócio

Letra de Crédito de Agronegócio (LCA) é um título de crédito que pode ser emitido por financiamento público ou por empresas do ramo privado. Essas empresas tem o objetivo de financiar o ramo de agronegócio brasileiro e utilizam o dinheiro arrecadado pelos investidores para esse fim. Após o período estabelecido em contrato na adesão do título, o seu possuidor tem direito ao pagamento dos juros estipulados na data de sua criação.

2.7.2 Letra de Crédito Imobiliário

Analogamente à letra de crédito do agronegócio (LCA), a letra de crédito imobiliário (LCI) também são títulos emitidos por instituições financeiras do ramo público ou privado que financiam um ramo da economia, porém neste caso o ramo contemplado é o imobiliário. O emissor do título deve obrigatoriamente utilizar todo o dinheiro dos investidores para empréstimos ao setor de construção e depois de vencida a data estipulada em contrato deve pagar os juros ao seu possuidor.

2.7.3 Certificado de Depósito Bancário

O Certificado de Depósito Bancário (CDB) é um título de renda fixa que tem como objetivo financiar as atividades de instituições financeiras. Nada mais é, do que um empréstimo com garantia de devolução e pagamento de juros. O banco usa o dinheiro dos títulos para bancar suas operações financeiras e com parte do lucro obtido remunera o possuidor do título.

2.8 Fundo Garantidor de Crédito

Fundo Garantidor de Crédito (FGC) é um importante ator no cenário econômico brasileiro que tem como função manter a estabilidade em momentos de crise. Segundo FGC (2020) é “uma associação civil, sem fins lucrativos, com personalidade jurídica de direito privado. Muito mais do que “pagador de dívidas”, que só surge em cena em momentos dramáticos, o FGC conta com profissionais preparados para agir de maneira preventiva em todo o sistema bancário e financeiro, atuando de maneira pontual e, muitas vezes, silenciosa para garantir um funcionamento fluido e harmônico de todo o sistema. O FGC é uma barreira de proteção para o um estado de calamidade econômica. Os investimentos cobertos pelo FGC possuem proteção total até o valor

de R\$ 250.000 por CPF e por instituição financeira, e um volume global de R\$ 1.000.000 apenas para pessoa física.

2.9 Tesouro Direto

É uma maneira da união pedir dinheiro emprestado a juros menores, ou seja, ao invés de recorrer a bancos ou fundos monetários internacionais que possuem taxas de juros de empréstimos altíssimos, o governo pega emprestado com o investidor que compra títulos de dívida pública e o estado mais tarde lhe paga este empréstimo com juros. Segundo B3 (2020) o Tesouro Direto é programa do tesouro nacional o qual foi desenvolvido em parceria com o banco B3 para que se vendam títulos públicos federais para pessoas físicas. O programa visa democratizar o acesso aos títulos públicos.

O tesouro direto oferece três tipos de títulos prefixada, ligada a variação da inflação ou taxa de juros básica e Selic.

2.9.1 Tesouro Prefixado

É o título vendido com uma taxa de juros fixa que renderá pelo prazo determinado em contrato.

2.9.2 Tesouro Ligado a Variação da Inflação ou Taxa de Juros Básica

É o título vendido com uma taxa de juros que varia em função de algum indexador de mercado como por exemplo a inflação. Neste tipo de investimento o rendimento pode variar de acordo com os índices econômicos e só se poderá saber o valor na data de vencimento do título.

2.9.3 Tesouro SELIC

Segue o indexador da taxa SELIC, e só é possível saber seu valor na data da venda.

2.10 SELIC

O Sistema Especial de Liquidação e Custódia, mais conhecido como SELIC é um índice econômico que controla diversas taxas de juros cobradas pelas instituições financeiras. Segundo BTG PACUAL DIGITAL (2020) é um indicador de quanto o

governo paga de juros para as instituições financeiras que compram títulos públicos do tesouro nacional.

2.11 Inflação

É a média de aumento do custo de determinados produtos de consumo ano após ano, basicamente ele mede o aumento do custo de vida. Seu valor é definido pelo IBGE (Instituto Brasileiro de Geografia e Estatística).

2.12 Renda Variável

Segundo FINANCEONE (2017, p.23) “Diferente da Renda Fixa, nesse tipo de investimento, a remuneração ou a forma de cálculo não é conhecida no momento da aplicação. Os investimentos de Renda Variável são mais arriscados, mas permitem retornos muito maiores. São recomendados para investidores mais acostumados às oscilações do mercado, com uma certa vivência na Renda Fixa. Os riscos envolvem, inclusive, juros negativos, ou seja, o investidor pode perder parte do capital investido.”

2.13 Ações

Ações são títulos que representam parte do valor total de uma companhia, na maioria das vezes é emitido como forma de se financiar projetos futuros de uma empresa. Exemplo: uma empresa quer construir uma nova filial em outro local. Ao invés de pedir um empréstimo ela pode se dividir em várias partes e vender essas partes a investidores interessados. Com o dinheiro das vendas ela constrói sua nova filial e os investidores viram novos sócios da empresa.

2.14 Bolsa de Valores

É o local onde se negociam papéis de empresas de capital aberto (ações), além de outros valores de renda variável.

2.15 Bovespa

É a bolsa de valores de São Paulo e a mais conhecida do Brasil, lá são negociadas em sua maioria ações das maiores empresas brasileiras. Seu funcionamento é de segunda-feria a sexta-feira 10:00h até as 17:55h.

2.16 Comissão de Valores Mobiliários

Popularmente conhecida como CVM. É uma autarquia criada com a finalidade de fiscalizar, disciplinar e ajudar no desenvolvimento do mercado mobiliário brasileiro.

2.17 Associação Brasileira das Entidades dos Mercados Financeiro e de Capitais

Popularmente conhecida como ANBIMA é uma instituição criada para representar atuantes do sistema de mercado financeiro e de capitais. Surgiu de uma fusão entre a Associação Nacional das Instituições do Mercado Financeiro e a Associação Nacional dos Bancos de Investimento. Segundo BONA (2020) “ela é a entidade que busca criar procedimentos que permitam a regulação do mercado. Esses procedimentos, depois de aprovados, devem ser seguidos por todos os seus associados.

Desse modo, ela acaba tendo uma forte atuação de autorregulação porque cria normas que visam garantir transparência e segurança nos mercados financeiros de capitais. Essas normas, ao serem seguidas pelas instituições financeiras, proporcionam uma maior qualidade no mercado financeiro, a qual repercute em proteção para o investidor e segurança para as instituições.”

2.18 Código ANBIMA de Regulação e Melhores Práticas

Este código vem destrinchar e preencher algumas lacunas deixadas pela Instrução CVM Nº539, que tem como objetivo estabelecer boas práticas entre empresas do ramo financeiro e de capitais no ato de vender seus produtos a seus clientes. Ela tenta proteger os clientes contra a venda de produtos que não se adequem ao que o mesmo não deseja. Além de transformar em obrigação o ato de que as empresas prestadoras desses produtos tenham que educar seus clientes no meio do processo. Este código será o norte utilizado no nosso aplicativo para a maioria das regras de negócios impostas.

3 DESENVOLVIMENTO DA METODOLOGIA DO TRABALHO

3.1 O Problema

Para um investidor iniciante é trabalhoso unir as informações necessárias para dar início a sua carteira de investimentos se baseando no seu perfil, no valor que ele tem disponível, assim como o tempo em que pode deixar esse valor investido.

O trabalho é justificado pois agrupando e disponibilizando tais informações para o investidor iniciante, assim como gerando opções reais de investimento que estão no mercado, sendo elas personalizadas por parâmetros passados pelo próprio investidor. É possível garantir um início mais saudável para o investidor estando conscientes de que opções ele possui.

3.2 Proposta

Com o objetivo de auxiliar o investidor iniciante a dar os primeiros passos, foi desenvolvida uma aplicação que se propõe a criar uma interface simples e intuitiva ao usuário que servirá como guia de estudo para ele dar início nos seus investimentos, assim como irá gerar o perfil de investidor do usuário baseado em um questionário. Ele também poderá escolher opções de investimento em renda fixa disponíveis que sejam do seu interesse no momento, dando entrada na aplicação com o valor que ele tem disponível para investir em uma determinada opção, assim como a data limite em que deseja o retorno desse investimento, a aplicação vai avaliar essas entradas e irá gerar um arquivo do tipo planilha no formato .xlsx para o usuário contendo diversos produtos de renda fixa que correspondem aos parâmetros “valor de investimento” e “data limite de retorno” passados pelo usuário, assim como o percentual de rendimento líquido para o usuário descontado o imposto de renda regressivo para os investimentos que o possuem.

Esse arquivo será gerado se baseando em uma base de dados com opções de investimento que passou por um processo de extração, transformação e por fim a carga de seus dados que terá atualização constante, para que a requisição do usuário sempre esteja com os produtos atualizados.

Caso o usuário deseje investir em renda variável, será gerado um segundo arquivo do tipo planilha no formato .xlsx que será alimentado por uma carteira recomendada pela corretora “XP investimentos”, essa carteira recomendada será gerada apenas com o intuito de demonstrar possíveis investimentos que estão em alta

no mercado de ações para o usuário, porém será recomendado para o mesmo que estude cada uma das empresas listadas na planilha antes de tomar qualquer decisão de investimento, dado que a aplicação tem apenas como intuito ensinar novos investidores com exemplos reais do mercado e não fazer análise de ações.

3.3 Análise dos Códigos Desenvolvidos

Nesta seção serão dispostos trechos relevantes dos códigos desenvolvidos para compor a aplicação, assim como serão detalhados seus propósitos.

Todos os códigos foram desenvolvidos na linguagem Python, sendo necessário compreensão dela para pleno entendimento.

3.3.1 Processo de ETL

ETL (Extract, transform, load) ou em português, extração, transformação e carga. É o processo onde será realizada a extração dos dados diretamente da fonte, mantendo sua estrutura original. Esses dados são tratados para gerar informações úteis para a aplicação em um formato desejado. Então os dados tratados são salvos no local de destino.

3.3.1.1 Extração

Para realizar a extração de dados foi usado o processo de web scraping (raspagem web) que consiste em extrair dados de um site da web convertendo-os em informação estruturada para análise posterior. Os códigos criados para cada site são personalizados para execução no mesmo.

3.3.1.1.1 Extração CDB, LCI, LCA

Os dados referentes aos produtos de renda fixa CDB, LCI e LCA, que serão armazenados na base de dados, são extraídos no portal da corretora BTG Pactual.

Para a extração deles foi usada a biblioteca selenium que cria um acesso ao site <https://www.btgpactualdigital.com/renda-fixa/produtos> de forma automatizada, possibilitando assim ser definido via análise no HTML do site quais dados são desejados para a extração via código.

Ao encontrar o elemento desejado no HTML, esses dados serão armazenados em uma estrutura em forma de tabela para cada produto. Ou seja, serão criadas três tabelas, essas tabelas são nomeadas, dfCDB, dfLCI e dfLCA

Figura 1 - Trecho do código de extração: CDB, LCI, LCA

```

geckodriver_path = 'geckodriver.exe'

option = Options()
option.headless = True

driver = webdriver.Firefox(executable_path = geckodriver_path, options = option)

url = 'https://www.btgpactualdigital.com/renda-fixa/produtos'

driver.get(url)

time.sleep(5)

element = driver.find_element_by_xpath('/html/body/app-root/div/app-product-catalog/div/
app-product-catalog-fixed-income/app-tabs-default/
div/div[2]/div/section/div/div[2]/div[1]')

html_content = element.get_attribute('innerHTML')

htmlSoup = soup(html_content, 'html.parser')
table = htmlSoup.find(name='table')

df_full = pd.read_html(str(table))[0]

df = df_full.drop(df_full.columns[9], axis = 1)

dfCDB = df.loc[df['Produto'].str.startswith('CDB')]

dfLCI = df.loc[df['Produto'].str.startswith('LCI')]

dfLCA = df.loc[df['Produto'].str.startswith('LCA')]

```

A estrutura inicial da tabela ainda não é a desejada pois ela foi extraída da fonte em sua forma original contendo informações não desejadas para a aplicação.

Figura 2 - Exemplo da estrutura da tabela em extração: CDB, LCI, LCA

Produto	Prazo	Vencimento	Taxa	Taxa EQ. CDB	Juros	Amortização	Horário	Aplicação Mínima
CDB - PÓS-FIXADO BANCO BTG PA	1 mês	30/11/2020	106,00% d	106,00% do CDI	Somente	Somente no Vencimento	15:00	R\$ 1.000,00
CDB - PÓS-FIXADO BANCO DAYCO	1 mês	30/11/2020	102,50% d	102,50% do CDI	Somente	Somente no Vencimento	15:00	R\$ 20.000,00

3.3.1.1.2 Extração Tesouro Direto

Os dados referentes aos produtos de renda fixa do Tesouro Direto, são extraídos de forma semelhante, porém o link de acesso é utilizado é <https://www.tesourodireto.com.br/titulos/precos-e-taxas.htm>, o elemento HTML buscado é outro e essa extração gera uma tabela chamada df_tesouro.

Figura 3 -Trecho do código de extração: Tesouro Direto

```

geckodriver_path = 'geckodriver.exe'

option = Options()
option.headless = True

driver = webdriver.Firefox(executable_path = geckodriver_path, options = option)

url = 'https://www.tesourodireto.com.br/titulos/precos-e-taxas.htm'

driver.get(url)

time.sleep(5)

element_tesouro = driver.find_element_by_xpath('/html/body/main/div[1]/div[2]/div/div/div/div[1]/div/div[9]')

html_content_tesouro = element_tesouro.get_attribute('outerHTML')

htmlSoup_tesouro = soup(html_content_tesouro, 'html.parser')
table_tesouro = htmlSoup_tesouro.find(name='table')

df_full_tesouro = pd.read_html(str(table_tesouro))[0]

df_tesouro = df_full_tesouro.drop(df_full_tesouro.columns[5], axis = 1)

```

Da mesma forma que na extração o CDB, LCI e LCA, estrutura inicial da tabela df_tesouro ainda não é a desejada.

Figura 4 - Exemplo da estrutura da tabela em extração: Tesouro Direto

Título	Rentabilidade a	Investime	Preço Unitário:	Vencimento:
Tesouro Prefixado 2023	5,12%	R\$35,89	R\$897,45	01/01/2023
Tesouro Prefixado 2026	7,34%	R\$34,67	R\$693,52	01/01/2026

3.3.1.1.3 Extração carteira sugerida

Os dados referentes a carteira sugerida de renda variável são extraídos do site <https://conteudos.xpi.com.br/acoes/carteias/top-10-acoes-xp-{mes}-de-{ano}/>, a abordagem para a extração desses dados é feita de forma diferente, pois as alterações no site não se dão apenas em seu conteúdo, mas também em seu link, por isso essa extração é feita pela biblioteca Pandas, sendo necessário atualizar manualmente as variáveis {mes} e {ano}, dispostas no link, assim como fazer a verificação do seu elemento HTML no momento da extração direto no site e o adicionando a variável “outer_HTML”, assim o código irá gerar uma tabela chamada dfCarteiraXP.

Figura 5 - Trecho do código de extração: Carteira recomendada

```
mes = 'outubro'
ano = 2020
url = f'https://conteudos.xpi.com.br/acoes/carteiras/top-10-acoes-xp-{mes}-de-{ano}/'
outer_HTML = 'Elemento HTML que deve ser analisado no dia da extração'
dfCarteiraXP = pd.read_html(outer_HTML)[0]
```

A extração da carteira recomendada gera uma tabela com informações faltantes e outras desnecessárias para a aplicação, essa estrutura será tratada na fase de transformação.

Figura 6 - Exemplo da estrutura da tabela em extração: Carteira recomendada

Empresa	Ticker	Preço-alvo (R\$)
	B3SA3	6500
	BBAS3	4300

3.3.1.2 Transformação

A fase de transformação se dá por alterarmos a estrutura das tabelas criadas na fase de extração, fazendo assim com que os dados resultantes dessa transformação sejam apenas aqueles que serão úteis para a aplicação.

3.3.1.2.1 Transformação CDB, LCI, LCA

As tabelas geradas na fase de extração terão o tipo de dado na coluna “Vencimento” alterados de string para o tipo datetime que oferece uma gama maior de opções na linguagem Python, as colunas que para a aplicação são indesejadas serão excluídas, tais como: Prazo, Taxa EQ. CDB, Juros, Amortização, Horário. Uma coluna chamada Taxa é criada contendo apenas a taxa sem o seu indexador referente, sendo essa no tipo de dado float, e uma outra coluna contendo apenas o indexador é criada com o dado do tipo string.

Figura 7 - Trecho do código de transformação: CDB, LCI, LCA

```

dfCDB['Vencimento'] = pd.to_datetime(dfCDB['Vencimento'], dayfirst = True)
dfCDB = dfCDB.drop(['Prazo', 'Taxa EQ. CDB', 'Juros', 'Amortização', 'Horário'], axis =1)
dfCDB['AplicaçãoMínima'] = dfCDB['AplicaçãoMínima'].str[: -2]
dfCDB['AplicaçãoMínima'] = dfCDB['AplicaçãoMínima'].str.replace(r"^[^0-9]", "").astype(float)
dfCDB['TaxaPct'] = dfCDB['Taxa'].str.partition('%')[0]
dfCDB['TaxaPct'] = dfCDB['TaxaPct'].str.replace(',', '.').astype(float)
dfCDB['Indexador'] = dfCDB['Taxa'].str.partition('do ')[2]

dfLCA['Vencimento'] = pd.to_datetime(dfLCA['Vencimento'], dayfirst = True)
dfLCA = dfLCA.drop(['Prazo', 'Taxa EQ. CDB', 'Juros', 'Amortização', 'Horário'], axis =1)
dfLCA['AplicaçãoMínima'] = dfLCA['AplicaçãoMínima'].str[: -2]
dfLCA['AplicaçãoMínima'] = dfLCA['AplicaçãoMínima'].str.replace(r"^[^0-9]", "").astype(float)
dfLCA['TaxaPct'] = dfLCA['Taxa'].str.partition('%')[0]
dfLCA['TaxaPct'] = dfLCA['TaxaPct'].str.replace(',', '.').astype(float)
dfLCA['Indexador'] = dfLCA['Taxa'].str.partition('do ')[2]

dfLCI['Vencimento'] = pd.to_datetime(dfLCI['Vencimento'], dayfirst = True)
dfLCI = dfLCI.drop(['Prazo', 'Taxa EQ. CDB', 'Juros', 'Amortização', 'Horário'], axis =1)
dfLCI['AplicaçãoMínima'] = dfLCI['AplicaçãoMínima'].str[: -2]
dfLCI['AplicaçãoMínima'] = dfLCI['AplicaçãoMínima'].str.replace(r"^[^0-9]", "").astype(float)
dfLCI['TaxaPct'] = dfLCI['Taxa'].str.partition('%')[0]
dfLCI['TaxaPct'] = dfLCI['TaxaPct'].str.replace(',', '.').astype(float)
dfLCI['Indexador'] = dfLCI['Taxa'].str.partition('do ')[2]

df['Vencimento'] = pd.to_datetime(df['Vencimento'], dayfirst = True)
df = df.drop(['Prazo', 'Taxa EQ. CDB', 'Juros', 'Amortização', 'Horário'], axis =1)
df['AplicaçãoMínima'] = df['AplicaçãoMínima'].str[: -2]
df['AplicaçãoMínima'] = df['AplicaçãoMínima'].str.replace(r"^[^0-9]", "").astype(float)
df['TaxaPct'] = df['Taxa'].str.partition('%')[0]
df['TaxaPct'] = df['TaxaPct'].str.replace(',', '.').astype(float)
df['Indexador'] = df['Taxa'].str.partition('do ')[2]

```

Após a fase de transformação, as tabelas geradas estão no formato desejados para a aplicação e prontas para a fase de carga.

Figura 8 - Exemplo da estrutura da tabela em transformação: CDB, LCI, LCA

Produto	Vencimento	Taxa	Aplicação	TaxaPct	Indexador
CDB - PÓS-FIXADO BANCO	30/11/2020	106,00% do CDI	1000.0	106.0	CDI
CDB - PÓS-FIXADO BANCO	30/11/2020	102,50% do CDI	20000.0	102.5	CDI

3.3.1.2.2 Transformação Tesouro direto

A tabela do tesouro direto, gerada na fase de extração terá o tipo de dado na coluna “Vencimento:” alterados de string para o tipo datetime assim como as tabelas anteriores, as colunas “Preço Unitário” e “Investimento mínimo:” terão seu tipo alterado de string para float. Serão criadas duas novas colunas chamadas “RentabilidadePct” e “Indexador”, contendo o percentual de rentabilidade e o nome do indexador do produto respectivamente

Figura 9 - Trecho do código de transformação: Tesouro direto

```

df_tesouro['Vencimento:'] = pd.to_datetime(df_tesouro['Vencimento:'], dayfirst = True)
df_tesouro['Preço Unitário:'] = df_tesouro['Preço Unitário:'].str[3:]
df_tesouro['Preço Unitário:'] = df_tesouro['Preço Unitário:'].str.replace('.', '')
df_tesouro['Preço Unitário:'] = df_tesouro['Preço Unitário:'].str.replace(',', '.').astype(float)
df_tesouro['Investimento mínimo:'] = df_tesouro['Investimento mínimo:'].str[3:]
df_tesouro['Investimento mínimo:'] = df_tesouro['Investimento mínimo:'].str.replace(',', '.').astype(float)
df_tesouro['RentabilidadePct'] = np.nan
for i in range(df_tesouro['Rentabilidade anual:'].count()):
    if '+' in df_tesouro['Rentabilidade anual:'][i]:
        df_tesouro['RentabilidadePct'][i] = df_tesouro['Rentabilidade anual:'][i].partition('+')[2]
    else:
        df_tesouro['RentabilidadePct'][i] = df_tesouro['Rentabilidade anual:'][i]
df_tesouro['RentabilidadePct'] = df_tesouro['RentabilidadePct'].str.partition('%')[0]
df_tesouro['RentabilidadePct'] = df_tesouro['RentabilidadePct'].str.replace(',', '.').astype(float)
df_tesouro['Indexador'] = df_tesouro['Rentabilidade anual:'].str.partition(' ')[0]
for i in range(df_tesouro['Indexador'].count()):
    if '%' in df_tesouro['Indexador'][i]:
        df_tesouro['Indexador'][i] = np.nan
    else:
        df_tesouro['Indexador'][i] = df_tesouro['Indexador'][i]

```

A tabela contendo os dados do tesouro direto está no formato desejado para a aplicação e pronta para a fase de carga.

Figura 10 - Exemplo da estrutura da tabela em transformação: Tesouro direto

Título	Rentabilidade a	Investime	Preço Uni	Vencimento:	Rentab	Indexador
Tesouro Selic 2025	SELIC + 0,1900%	106.39	10639.38	01/03/2025	0.19	SELIC
Tesouro IPCA+ 2026	IPCA + 3,04%	56.46	2823.01	15/08/2026	3.04	IPCA

3.3.1.2.3 Transformação carteira recomendada

A tabela que contém os dados da carteira recomendada, foi extraída com informações faltantes e uma coluna indesejada. Na fase de transformação dessa tabela, será feita uma consulta no site http://www.b3.com.br/pt_br/busca/?query={ticker}, no link onde está escrito ticker, será passado o valor dos tickers que estão na tabela dfCarteiraXP. Ticker é um código usado na bolsa de valores para identificar e negociar um determinado ativo de uma empresa. Essa consulta gera uma página web com o nome da empresa referente àquele ticker. Utilizando a biblioteca BeautifulSoup, é possível extrair o nome da empresa dessa página por meio da tag HTML dela. A coluna “Empresa” é preenchida com os nomes das empresas referentes aos tickers que foram extraídos na fase de extração. Uma coluna chamada “Risco” é adicionada, ela é preenchida com a string ‘Alto’ para indicar ao investidor iniciante que existe um alto risco de investir em ações

nas quais ele não tenha estudado sobre. Essa informação será reforçada posteriormente.

Figura 11 - Trecho do código de transformação: Carteira recomendada

```
for i in range(dfCarteiraXP['Ticker'].count()):
    ticker = dfCarteiraXP['Ticker'][i]
    urlB3 = f"http://www.b3.com.br/pt_br/busca/?query={ticker}"
    soup = BeautifulSoup(requests.get(urlB3).text, "html.parser")
    dfCarteiraXP['Empresa'][i] = soup.h5.text

del dfCarteiraXP['Preço-alvo (R$)']

dfCarteiraXP['Risco'] = 'Alto'
```

Figura 12 - Exemplo da estrutura da tabela em transformação: Carteira recomendada

Empresa	Ticker	Risco
B3 S.A. - BRASIL BOLSA BALCÃO	B3SA3	Alto
BCO BRASIL S.A.	BBAS3	Alto

3.3.1.3 Carga

A fase de carga onde salvamos os dados já no formato desejado para o nosso repositório de consulta.

Como uma forma de possibilitar a consulta externa todas as tabelas citadas no processo de ETL estão sendo salvas em um repositório local que está sincronizado com um repositório aberto no Google Drive. Simulando assim uma fonte de consulta para esses dados já transformados.

As tabelas estão sendo salvas no formato .csv que significa “comma-separated-values” (valores separados por vírgulas). Isso significa que em um arquivo de texto cada coluna é separada por um delimitador, sendo a vírgula o mais comumente utilizado criando assim um arquivo estruturado que pode ser consultado e transformado em uma tabela por aplicações que possibilitem essa conversão.

Por exemplo, o arquivo .csv que representa a estrutura da tabela na Figura 8, caso seu delimitador seja definido como ponto e vírgula (;) será o seguinte texto:

Produto;Vencimento;Taxa;AplicaçãoMínima;TaxaPct;Indexador

CDB - PÓS-FIXADO BANCO BTG PACTUAL S/A;2020-11-30;106,00% do CDI;1000.0;106.0;CDI

CDB - PÓS-FIXADO BANCO DAYCOVAL S/A;2020-11-30;102,50% do CDI;20000.0;102.5;CDI

Figura 13 - Trecho do código de carga das tabelas

```
dfCDB.to_csv('CDB_Clean.csv', sep = ';', encoding = 'ansi', index = False)

dfLCA.to_csv('LCA_Clean.csv', sep = ';', encoding = 'ansi', index = False)

dfLCI.to_csv('LCI_Clean.csv', sep = ';', encoding = 'ansi', index = False)

df_tesouro.to_csv('tesouro_Clean.csv', sep = ';', encoding = 'ansi', index = False)

dfCarteiraXP.to_csv('CarteiraXP.csv', sep = ';', encoding = 'ansi', index = False)
```

3.3.2 Função download_opcoesRF

Essa função é utilizada para coletar os dados das opções de renda fixa salvas no Google Drive pelo processo de ETL. Ela faz uso da biblioteca gdown, que possibilita fazer o download de arquivos do Google Drive que estejam com seu compartilhamento disponível, apenas sendo necessário ter o link do mesmo.

Os links para cada um dos arquivos devem estar salvos na função, ela irá iterar sobre esses links fazendo o download dos arquivos referentes a eles e salvando em uma pasta predeterminada, esses arquivos são tabelas no formato .csv, nomeadas, CDB_Clean, LCA_Clean, LCI_Clean, tesouro_Clean e completo_Clean.

Figura 14 - Função download_opcoesRF

```
def download_opcoesRF():
    id_CDB = '1GXU1Uhj0hNjMFD06x4eB0FiXotdenXqC'
    id_LCA = '1k4UGTGLx_t-4B-hYnqmsIWLI4BujZTFq'
    id_LCI = '1TOfiu5xWrazGSEoqvbt156bhU2NfSF3'
    id_tesouro = '1P708G6wnhgAtxtT1he9cn6_8QUphabMw'
    id_completo = '1dewy6nZVPNsYBomfMykpwtUFFzc7GzLI'

    ids_drive = [id_CDB, id_LCA, id_LCI, id_tesouro, id_completo]
    ids_drive_nomes = ['CDB_Clean.csv', 'LCA_Clean.csv', 'LCI_Clean.csv',
                      'tesouro_Clean.csv', 'completo_Clean.csv']

    for idurl, nome in zip(ids_drive, ids_drive_nomes):
        url = f'https://drive.google.com/uc?id={idurl}'
        output = nome
        gdown.download(url, output, quiet = True)
    return
```

3.3.3 Função download_opcoesRV

Na função download_opcoesRV é feita a coleta dos dados da tabela de carteira de ações recomendada, gerada pelo trecho de código que foi detalhado no capítulo 3.2.1.2.3, nela é feito o download da tabela que está sendo compartilhada no Google Drive criando um arquivo .xlsx a partir dele.

Figura 15 - Função download_opcoesRV

```
def download_opcoesRV():
    url = 'https://drive.google.com/uc?id=1qwx-7R3fT4n2488xpg9m24uSy0dcQVFD'
    gdown.download(url, 'carteiraRV.csv', quiet = True)
    carteiraRV = pd.read_csv('carteiraRV.csv', sep = ';', encoding = 'ansi')
    writer = pd.ExcelWriter('../Relatorios/Opcoes_Adequadas_Renda_Variavel.xlsx', engine='xlsxwriter')
    carteiraRV.to_excel(writer, sheet_name='Carteira Renda Variavel', index = False)
    writer.save()
    os.remove('carteiraRV.csv')
```

3.3.4 Função ir_regressivo

A função `ir_regressivo` calcula o valor percentual que será descontado do pelo Imposto de Renda sobre o lucro de certos produtos. O CDB e o Tesouro direto possuem esse desconto, ou seja, no vencimento de um investimento desse tipo, será cobrado um valor sobre o seu lucro obtido nessa opção de investimento. Esse valor leva em consideração quanto tempo você deixou levou desde o dia em que foi feito o investimento, até o dia de retirada do mesmo.

Baseado nesse número de dias é calculado o percentual do imposto de renda de forma regressiva sendo:

- Até 180 dias – 22,5%
- De 181 até 360 dias – 20%
- De 361 dias até 720 dias – 17,5%
- A partir de 721 dias – 15%

A função recebe a data inicial onde está sendo feita a cotação, e a data final de retirada do investimento. É calculada a diferença entre essas duas datas para definir assim o número de dias do investimento.

Tendo sido calculado o número de dias, a função retornará o valor percentual do imposto de renda referente a esse número.

Figura 16 - Função ir_regressivo

```
def ir_regressivo(dataInic, dataFinal):
    dataInic = pd.to_datetime(dataInic, dayfirst = True)
    dataFinal = pd.to_datetime(dataFinal, dayfirst = True)
    dias = (dataFinal - dataInic).days
    if(dias <= 180):
        ir = 0.225
    elif(dias > 180 & dias <= 360):
        ir = 0.20
    elif(dias > 360 & dias <= 720):
        ir = 0.175
    elif(dias > 720):
        ir = 0.15
    return(ir)
```

3.3.5 Função consulta_opcoes_CDB

Essa função é utilizada para definir quais opções de CDB são adequadas aos parâmetros passados pelo usuário, sendo esses parâmetros a data máxima em que ele deseja o retorno do investimento e o valor que ele tem disponível para investir.

A função após receber esses parâmetros, busca o arquivo dfCDB_Clean que foi baixado pela função download_opcoesRF, esse arquivo retorna uma tabela para a função, na tabela serão listadas todas as opções de CDB que tenham o seu vencimento na data máxima determinada, ou inferior a ela e que possuam um investimento mínimo com um valor igual ou inferior ao determinado.

Como para o CDB é cobrado o valor de imposto de renda regressivo sobre o lucro da aplicação, é criada uma coluna referente ao rendimento líquido final da aplicação, já descontando o imposto de renda referente ao tempo determinado com o uso da função ir_regressivo.

O rendimento líquido após remoção do imposto de renda é dado por:
 (percentual de rendimento bruto) * (1 – (percentual o imposto de renda / 100))
 Exemplo: 130% * (1 – (22,5/100)) = 100,75%

Após a criação da coluna “Rendimento Líquido”, a tabela é ordenada com base no valor dessa coluna, indo do maior rendimento líquido para o menor, facilitando assim a visualização do usuário, de quais investimentos trarão um maior rendimento no fim.

São criadas também as seguintes colunas:

- Coberto pelo FGC: Que informa a existência da cobertura do investimento pelo Fundo garantidor de crédito
- Risco: Que informa o risco do tipo de investimento, para o caso de um CDB por ter a cobertura do FGC, torna esse risco baixo
- Canal de atendimento: Informa o canal de atendimento de onde foi extraído o dado do produto.

Então é salvo um arquivo no formato .csv nomeado dfOpcoesCDB contendo os produtos resultantes da análise acima.

Figura 17 - Função consulta_opcoes_CDB

```
def consulta_opcoes_CDB(dataF, valor):
    dfCDB_Clean = pd.read_csv('CDB_Clean.csv', sep = ';', encoding = 'ansi')
    dataF = pd.to_datetime(dataF, dayfirst = True)
    dfCDB_Clean['Vencimento'] = pd.to_datetime(dfCDB_Clean['Vencimento'], dayfirst = True)
    dfOpcoesCDB = dfCDB_Clean.loc[(dfCDB_Clean["Vencimento"] <= dataF) &
                                   (dfCDB_Clean["AplicaçãoMínima"] <= valor)]

    dfOpcoesCDB['TaxaComIR'] = np.nan
    dfOpcoesCDB = dfOpcoesCDB.reset_index(drop = True)
    for i in range(dfOpcoesCDB['Produto'].count()):
        dfOpcoesCDB['TaxaComIR'][i] = (dfOpcoesCDB['TaxaPct'][i] *
                                         (1 - ir_regressivo(date.today(), dfOpcoesCDB['Vencimento'][i])))
    del dfOpcoesCDB['TaxaPct']
    dfOpcoesCDB['Vencimento'] = dfOpcoesCDB['Vencimento'].dt.strftime('%d/%m/%Y')
    dfOpcoesCDB = dfOpcoesCDB.sort_values('TaxaComIR', ascending = False)
    dfOpcoesCDB = dfOpcoesCDB.reset_index(drop = True)
    dfOpcoesCDB['TaxaComIR'] = dfOpcoesCDB['TaxaComIR'].round(2).astype(str) + '%' + ' '
        + dfOpcoesCDB['Indexador'].astype(str)
    dfOpcoesCDB['TaxaComIR'] = dfOpcoesCDB['TaxaComIR'].str.replace('nan', '').astype(str)
    dfOpcoesCDB = dfOpcoesCDB.rename(columns={'Vencimento': 'Data de Resgate',
                                                'Taxa': 'Rendimento Bruto',
                                                'AplicaçãoMínima': 'Aplicação Mínima',
                                                'TaxaComIR': 'Rendimento Líquido'})

    dfOpcoesCDB['Coberto pelo FGC?'] = 'Sim'
    dfOpcoesCDB['Risco'] = 'Baixo'
    dfOpcoesCDB['Canal de Atendimento'] = 'atendimento@btgpactualdigital.com'
    dfOpcoesCDB.to_csv('dfCDBop.csv', sep = ';', encoding = 'ansi', index = False)
    return
```

Figura 18 - Exemplo da estrutura da tabela gerada pela função consulta_opcoes_CDB

Produto	Data de Resgate	Rendimento Bruto	Aplicação Mínima	Indexador	Rendimento Líquido	Coberto pelo FGC?	Risco	Canal de Atendimento
CDB - PÓS-FIXADO BANC	01/11/2021	150,00% do CDI	2000	CDI	120.0% CDI	Sim	Baixo	atendimento@btgpactualdigital.com
CDB - PÓS-FIXADO BANC	01/11/2021	114,00% do CDI	1000	CDI	91.2% CDI	Sim	Baixo	atendimento@btgpactualdigital.com

3.3.6 Funções consulta_opcoes_LCI e consulta_opcoes_LCA

O processo executado por essa função é semelhante ao da função consulta_opcoes_CDB, recebendo a data máxima em que se deseja o retorno, assim como o valor a ser investido, toda a análise é feita da mesma forma, porém não é aplicado o desconto do imposto de renda regressivo no rendimento final, pois esse tanto LCI quando LCA são isentos do pagamento desse imposto em seu lucro.

Então será feita a análise com base nas tabelas dfLCI_Clean e dfLCA_Clean para gerar respectivamente os arquivos .csv dfOpcoes_LCI e dfOpcoes_LCA.

Figura 19 - Função consulta_opcoes_LCI

```
def consulta_opcoes_LCI(dataF, valor):
    dfLCI_Clean = pd.read_csv('LCI_Clean.csv', sep = ';', encoding = 'ansi')
    dataF = pd.to_datetime(dataF, dayfirst = True)
    dfLCI_Clean['Vencimento'] = pd.to_datetime(dfLCI_Clean['Vencimento'], dayfirst = True)
    dfOpcoesLCI = dfLCI_Clean.loc[(dfLCI_Clean["Vencimento"] <= dataF) &
                                   (dfLCI_Clean["AplicaçãoMínima"] <= valor)]
    dfOpcoesLCI['Vencimento'] = dfOpcoesLCI['Vencimento'].dt.strftime('%d/%m/%Y')
    dfOpcoesLCI = dfOpcoesLCI.sort_values('TaxaPct', ascending = False)
    del dfOpcoesLCI['TaxaPct']
    dfOpcoesLCI = dfOpcoesLCI.reset_index(drop = True)
    dfOpcoesLCI = dfOpcoesLCI.rename(columns={'Vencimento': 'Data de Resgate',
                                                'Taxa': 'Rendimento Líquido',
                                                'AplicaçãoMínima': 'Aplicação Mínima'})

    dfOpcoesLCI['Coberto pelo FGC?'] = 'Sim'
    dfOpcoesLCI['Risco'] = 'Baixo'
    dfOpcoesLCI['Canal de Atendimento'] = 'atendimento@btgpactualdigital.com'
    dfOpcoesLCI.to_csv('dfLCIop.csv', sep = ';', encoding = 'ansi', index = False)
    return
```

Figura 20 - Exemplo da estrutura da tabela gerada pela função consulta_opcoes_LCI

Produto	Data de Resgate	Rendimento Líquido	Aplicação Mínima	Indexador	Coberto pelo FGC?	Risco	Canal de Atendimento
LCI - PÓS-FIXADO BANCO	28/01/2021	102,50% do CDI	10000	CDI	Sim	Baixo	atendimento@btgpactualdigital.com
LCI - PÓS-FIXADO BANCO	28/01/2021	102,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpactualdigital.com

Figura 21 - Função consulta_opcoes_LCA

```
def consulta_opcoes_LCA(dataF, valor):
    dfLCA_Clean = pd.read_csv('LCA_Clean.csv', sep = ';', encoding = 'ansi')
    dataF = pd.to_datetime(dataF, dayfirst = True)
    dfLCA_Clean['Vencimento'] = pd.to_datetime(dfLCA_Clean['Vencimento'], dayfirst = True)
    dfOpcoesLCA = dfLCA_Clean.loc[(dfLCA_Clean["Vencimento"] <= dataF) &
                                   (dfLCA_Clean["AplicaçãoMínima"] <= valor)]
    dfOpcoesLCA['Vencimento'] = dfOpcoesLCA['Vencimento'].dt.strftime('%d/%m/%Y')
    dfOpcoesLCA = dfOpcoesLCA.sort_values('TaxaPct', ascending = False)
    del dfOpcoesLCA['TaxaPct']
    dfOpcoesLCA = dfOpcoesLCA.reset_index(drop = True)
    dfOpcoesLCA = dfOpcoesLCA.rename(columns={'Vencimento': 'Data de Resgate',
                                                'Taxa': 'Rendimento Líquido',
                                                'AplicaçãoMínima': 'Aplicação Mínima'})

    dfOpcoesLCA['Coberto pelo FGC?'] = 'Sim'
    dfOpcoesLCA['Risco'] = 'Baixo'
    dfOpcoesLCA['Canal de Atendimento'] = 'atendimento@btgpactualdigital.com'
    dfOpcoesLCA.to_csv('dfLCAop.csv', sep = ';', encoding = 'ansi', index = False)
    return
```

Figura 22 - Exemplo da estrutura da tabela gerada pela função consulta_opcoes_LCA

Produto	Data de Resgate	Rendimento Líquido	Aplicação Mínima	Indexador	Coberto pelo FGC?	Risco	Canal de Atendimento
LCA - PÓS-FIXADO BANCO	28/01/2021	107,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpactualdigital.com
LCA - PÓS-FIXADO VOITER	28/01/2021	105,00% do CDI	10000	CDI	Sim	Baixo	atendimento@btgpactualdigital.com

3.3.7 Função consulta_tesouro

De forma semelhante a função consulta_CDB, aqui será calculado o valor do imposto de renda regressivo sobre a rentabilidade do produto investido, toda a análise é feita de forma semelhante, porém agora é feita na tabela tesouro_Clean. Após a análise será gerada uma nova tabela nomeada dfTesouroOp.

Figura 23 - Função consulta_opcoes_tesouro

```
def consulta_opcoes_tesouro(dataF, valor):
    dfTesouro_Clean = pd.read_csv('tesouro_Clean.csv', sep = ';', encoding = 'ansi')
    dataF = pd.to_datetime(dataF, dayfirst = True)
    dfTesouro_Clean['Vencimento'] = pd.to_datetime(dfTesouro_Clean['Vencimento'], dayfirst = True)
    dfOpcoesTesouro = dfTesouro_Clean.loc[(dfTesouro_Clean["Vencimento"] <= dataF) &
                                           (dfTesouro_Clean["Investimento mínimo"] <= valor)]

    dfOpcoesTesouro['TaxaComIR'] = np.nan
    dfOpcoesTesouro = dfOpcoesTesouro.reset_index(drop = True)
    for i in range(dfOpcoesTesouro['Título'].count()):
        dfOpcoesTesouro['TaxaComIR'][i] = (dfOpcoesTesouro['RentabilidadePct'][i] *
                                           (1 - ir_regressivo(date.today(), dfOpcoesTesouro['Vencimento:'][i])))
    del dfOpcoesTesouro['RentabilidadePct']
    dfOpcoesTesouro['Vencimento'] = dfOpcoesTesouro['Vencimento'].dt.strftime('%d/%m/%Y')
    dfOpcoesTesouro = dfOpcoesTesouro.sort_values('TaxaComIR', ascending = False)
    dfOpcoesTesouro = dfOpcoesTesouro.reset_index(drop = True)
    dfOpcoesTesouro['TaxaComIR'] = dfOpcoesTesouro['TaxaComIR'].round(2).astype(str) + '%' + ' ' + dfOpcoesTesouro['Indexador'].astype(str)
    dfOpcoesTesouro['TaxaComIR'] = dfOpcoesTesouro['TaxaComIR'].str.replace('nan', "").astype(str)
    dfOpcoesTesouro = dfOpcoesTesouro.rename(columns={'Rentabilidade anual': 'Rendimento Anual Bruto',
                                                       'Investimento mínimo': 'Investimento mínimo',
                                                       'Preço Unitário': 'Preço Unitário',
                                                       'Vencimento': 'Vencimento', 'TaxaComIR': 'Rendimento Líquido'})

    dfOpcoesTesouro['Coberto pelo FGC?'] = 'Não'
    dfOpcoesTesouro['Risco'] = 'Baixo'
    dfOpcoesTesouro['Canal de Atendimento'] = 'https://www.tesourodireto.com.br/central-de-atendimento/entre-em-contato.htm'
    dfOpcoesTesouro.to_csv('dfTesouroOp.csv', sep = ';', encoding = 'ansi', index = False)
    return
```

Figura 24 - Exemplo da estrutura da tabela gerada pela função
consulta_opcoes_tesouro

Título	Rentabilidade anual	Investimento mínimo	Preço Unitário	Vencimento	Indexador	Rendimento Líquido	Coberto pelo FGC?	Risco	Canal de Atendimento
Tesouro IPCA+ 2016	IPCA + 3,52%	40,71	4071,53	15/08/2030	IPCA	2.82% IPCA	Não	Baixo	https://www.tesourodireto.com.br/central-de-atendimento/entre-em-contato.htm
Tesouro IPCA+ 2017	IPCA + 3,04%	56,46	2823,01	15/08/2026	IPCA	2.43% IPCA	Não	Baixo	https://www.tesourodireto.com.br/central-de-atendimento/entre-em-contato.htm
Tesouro Selic 2018	SELIC + 0,1900%	106,39	10639,38	01/03/2025	SELIC	0.15% SELIC	Não	Baixo	https://www.tesourodireto.com.br/central-de-atendimento/entre-em-contato.htm

3.3.8 Função criar_resultado_RF_excel

A função consulta_tesouro pega todas as tabelas geradas de forma personalizada pelas funções consulta_CDB, consulta_LCI, consulta_LCA, consulta_tesouro e as agrupa em um arquivo no formato .xlsx nomeado Opcoes_Adequadas_Renda_Fixa. Esse arquivo vai conter todas as opções de renda

fixa que o usuário escolheu baseado na data máxima de retorno dos investimentos e o valor disponível para cada um deles.

Caso o usuário tenha escolhido um dos tipos de produto em renda fixa disponíveis, a função irá pular o tipo não escolhido, gerando apenas o arquivo .xlsx referente as opções desejadas.

Após gerar o arquivo, a função apaga todas as tabelas no formato .csv que forma usadas para a criação do arquivo .xlsx e também, apaga os arquivos que a função `download_opcoesRF` baixou no sistema do usuário.

Figura 25 - Função criar_resultado_RF_excel

```

def criar_resultado_RF_excel():
    try:
        dfCDBop = pd.read_csv('dfCDBop.csv', sep = ';', encoding = 'ansi')
    except:
        pass
    try:
        dfLCIop = pd.read_csv('dfLCIop.csv', sep = ';', encoding = 'ansi')
    except:
        pass
    try:
        dfLCAop = pd.read_csv('dfLCAop.csv', sep = ';', encoding = 'ansi')
    except:
        pass
    try:
        dfTesouroOp = pd.read_csv('dfTesouroOp.csv', sep = ';', encoding = 'ansi')
    except:
        pass

    writer = pd.ExcelWriter('Opcoes_Adequadas_Renda_Fixa.xlsx', engine='xlsxwriter')

    try:
        dfCDBop.to_excel(writer, sheet_name='CDB', index = False)
    except:
        pass

    try:
        dfLCIop.to_excel(writer, sheet_name='LCI', index = False)
    except:
        pass

    try:
        dfLCAop.to_excel(writer, sheet_name='LCA', index = False)
    except:
        pass

    try:
        dfTesouroOp.to_excel(writer, sheet_name='Tesouro', index = False)
    except:
        pass

    writer.save()

    try:
        os.remove('dfCDBop.csv')
    except:
        pass

    try:
        os.remove('dfLCIop.csv')
    except:
        pass
    try:
        os.remove('dfLCAop.csv')
    except:
        pass
    try:
        os.remove('dfTesouroOp.csv')
    except:
        pass

    os.remove('CDB_Clean.csv')
    os.remove('LCI_Clean.csv')
    os.remove('LCA_Clean.csv')
    os.remove('tesouro_Clean.csv')
    os.remove('completo_Clean.csv')

    return

```

3.3.9 Função criar_conta

A função recebe como parâmetro as respostas do cliente no formulário da tela representada na figura 29, lê o CSV que tem os clientes criados anteriormente, calcula seu perfil, utilizando o método de tendência central (mediana), onde o cliente respondeu várias perguntas com valores de 0 para conservador, 1 para conservador, 2 para moderado e 3 para agressivo e então salva em uma tabela as informações obtidas, e após isso cria um CSV na pasta Controle do executável, onde as informações do cliente podem ser acessados a qualquer momento, que é um requisito do art. 19 inciso 4.

Além disso ela também salva a data de atualização do perfil, pois segundo o Art. 19 parágrafo 7 determina que as informações de perfil do usuário devam ser feitas de no máximo 24 meses e essa informação é comparada sempre que o cliente abre a aplicação.

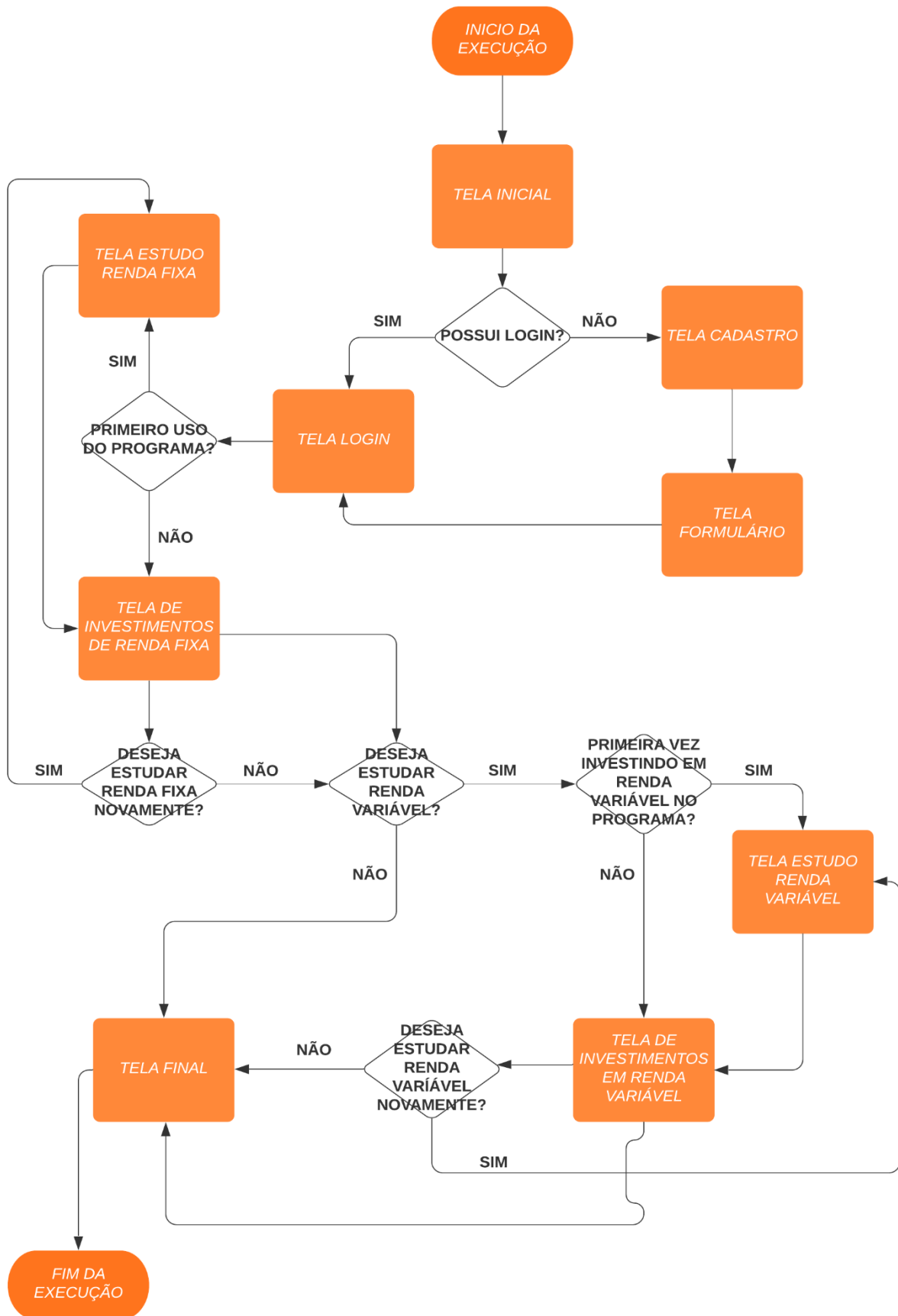
Figura 26 - Função criar_conta

```
def criar_conta(self, respostas, respostas2):
    frameclientes = pd.read_csv('../Controle/Cientes.csv', encoding='ansi', sep=";", index_col=0)
    login = list(framelogin["login"])
    senha = list(framelogin["senha"])
    respostas2.sort()
    if (respostas2[4] == 0):
        perfil = "Conservador"
    elif (respostas2[4] == 1):
        perfil = "Conservador"
    elif (respostas2[4] == 2):
        perfil = "Moderado"
    else:
        perfil = "Agressivo"
    respostas.append(login[0])
    respostas.append(senha[0])
    respostas.append(perfil)
    respostas.append(0)
    respostas.append(0)
    respostas.append(time.strftime('%m'))
    respostas.append(time.strftime('%y'))
    informacoes = [respostas]
    columns = ["Qual sua faixa etária",
               "Qual percentual do seu patrimônio está investido? (Carros, casas, ações, etc)",
               "Qual o objetivo dos seus investimentos?",
               "Como você se sentiria caso verificasse perdas em seus investimentos?",
               "Quais são as aplicações financeiras em que você tem experiência?",
               "Possui algum conhecimento sobre o mercado financeiro?",
               "Qual sua escolaridade ?",
               "Você costuma operar financeiramente com que frequência?",
               "Você espera precisar de renda extra no futuro?",
               "Qual o tempo disponível que você tem para manter seu dinheiro aplicado?",
               "Login",
               "Senha",
               "Perfil",
               "lidorf",
               "lidorv",
               "mes_criacao",
               "ano_criacao"]
    frame = pd.DataFrame(informacoes, columns=columns)
    frame.to_csv("Clientes1.csv", sep=";", encoding='ansi')
    framelogin = pd.read_csv('../Controle/Cientes1.csv', encoding='ansi', sep=";", index_col=0)
    frames = pd.concat([frameclientes, framelogin], ignore_index=True)
    frames.to_csv("Clientes.csv", sep=";", encoding='ansi')
```

3.4 Fluxograma de Execução e Modelo de Dados

Abaixo seguem representados o fluxograma da aplicação e o modelo de dados. Vale ressaltar que o modelo de dados é meramente ilustrativo, visto que todos os dados utilizados na aplicação estão salvos em arquivos CSV que ficam localizados na pasta CONTROLE da aplicação. O objetivo de incluir o modelo de dados é mostrar tacitamente a organização dos dados no programa.

3.4.1 Fluxograma de Execução



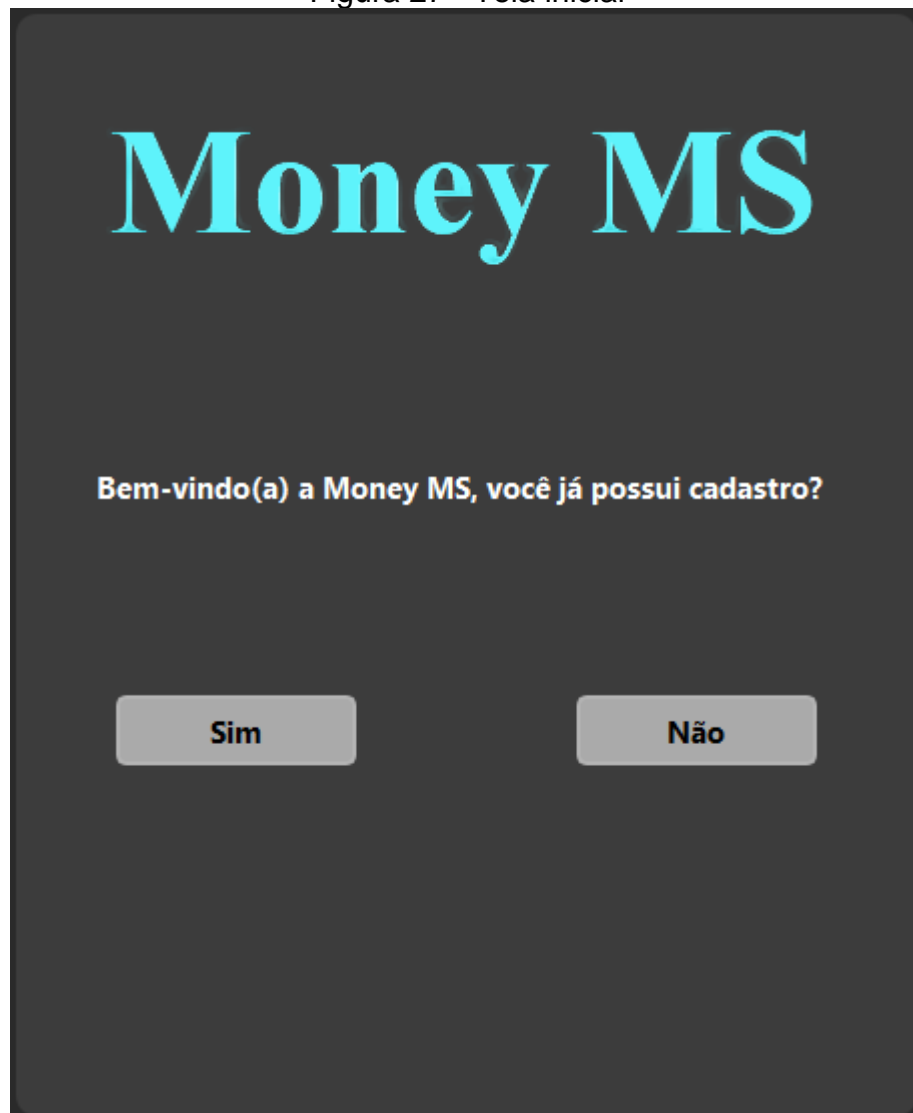
4 INTERAÇÃO DO USUÁRIO COM A APLICAÇÃO

Neste capítulo é simulada a interação de um usuário com a aplicação, fazendo uso de toda a interface visual criada para a mesma.

4.1 Tela inicial

Ao se iniciar a aplicação, o usuário será questionado na tela inicial se ele já possui seu perfil cadastrado, apresentando dois botões, um para caso o usuário já tenha feito seu cadastro em um momento anterior e o outro caso seja o primeiro acesso à aplicação. Para essa simulação será considerado um usuário efetuando seu primeiro acesso, então ele clicará no botão “Não”.


Figura 27 - Tela inicial



4.2 Cadastro

Ao clicar no botão “Não” apresentado na tela inicial, o usuário será redirecionado para a tela de cadastro, onde ele poderá cadastrar um nome de usuário e uma senha para acesso, ambos podendo conter letras e números. O objetivo principal de manter um cadastro de usuário nessa aplicação é armazenar as opções selecionadas por ele no formulário que será apresentado posteriormente, assim como salvar em seu cadastro o perfil de investidor que foi definido para ele. Seguindo assim as normas Art. 18 Parágrafo 2 do Código de Boas Práticas ANBIMA

Figura 28 - Tela de cadastro



A tela de cadastro do Money MS possui um fundo cinza escuro. No topo, o título "Money MS" está em uma fonte serifada azul clara, com "Cadastro" em uma fonte sans-serif branca logo abaixo. O formulário é composto por dois campos de entrada brancos com bordas azuis: o primeiro, rotulado "Crie seu nome de usuário", contém o texto "Usuário"; o segundo, rotulado "Crie sua senha", contém o texto "Senha". Abaixo dos campos, há um botão cinza com o texto "Cadastrar" em preto.

4.3 Formulário

O usuário irá preencher um formulário. Para cada pergunta no formulário ele poderá escolher uma única resposta. O objetivo do formulário é definir o perfil de investidor do usuário, ele foi criado baseando-se no TCC - UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL ESCOLA DE ADMINISTRAÇÃO DEPARTAMENTO DE CIÊNCIAS ADMINISTRATIVAS - Michele Bettim Bandinelli - FINANÇAS COMPORTAMENTAIS: ORIENTAÇÃO AO PERFIL DO INVESTIDOR PESSOA FÍSICA - Porto Alegre – 2010 com perguntas que são frequentemente usadas por bancos para definir o perfil de investidor de um cliente. A metodologia segundo o Art. 18 caput do Código de Boas Práticas ANBIMA é definida por critérios próprios.

Figura 29 - Tela de formulário

The image shows a web form titled "Money MS" in a large, stylized teal font. The form is set against a dark gray background and contains ten questions, each with a corresponding dropdown menu. The questions are arranged in a grid-like fashion. At the bottom center, there is a gray button labeled "Cadastrar".

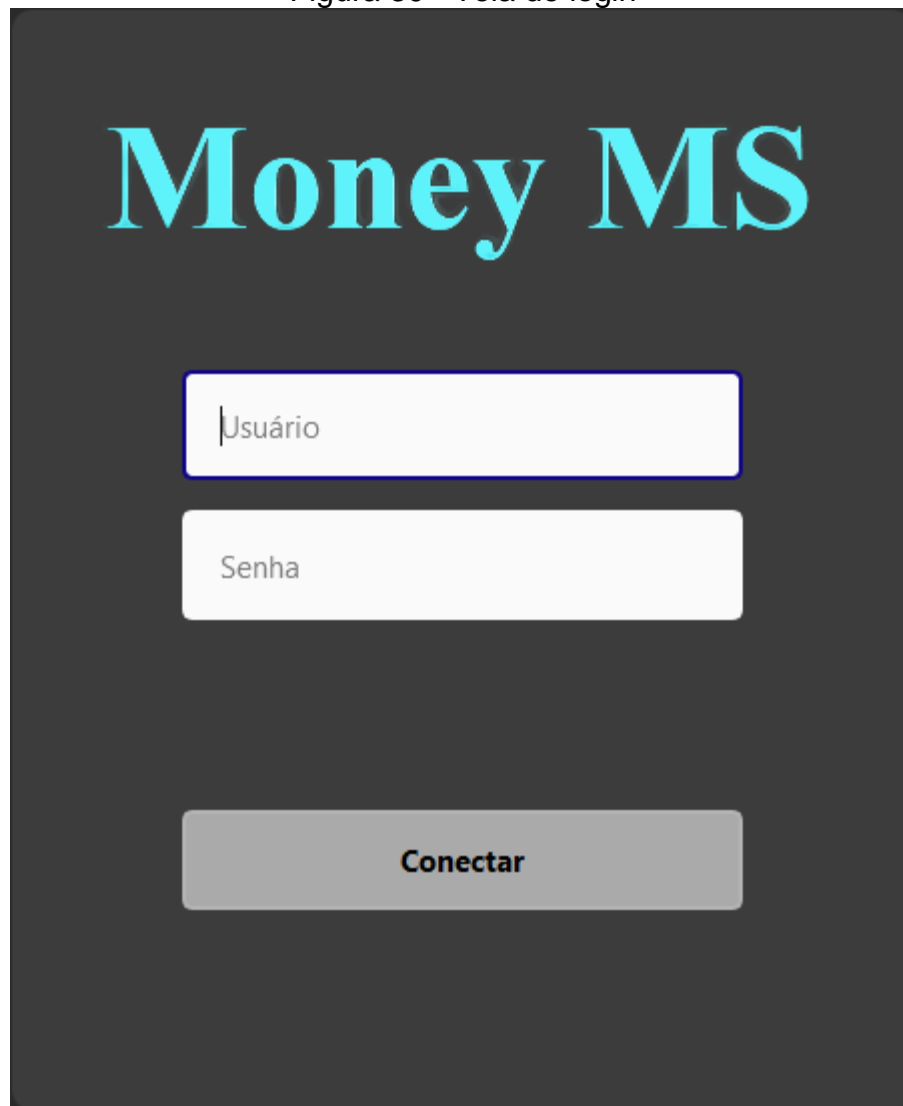
Question	Selected Answer
Qual sua faixa etária?	Abaixo de 25 anos
Quais são as aplicações financeiras em que você tem experiência?	Não possuo
Você espera precisar de renda extra no futuro?	Não. Pretendo diminuir
Qual percentual do seu patrimônio está investido? (Carros, casas, ações, etc)	Até 25%
Possui algum conhecimento sobre o mercado financeiro?	Não
Qual o tempo disponível que você tem para manter seu dinheiro aplicado?	Menos de 1 ano
Qual o objetivo dos seus investimentos?	Aposentadoria
Qual sua escolaridade?	Ensino fundamental ou inferior
Como você se sentiria caso verificasse perdas em seus investimentos?	Extremamente desanimado
Você costuma operar financeiramente com que frequência?	Pouquíssima

Cadastrar

4.4 Login

Após a definição do seu perfil de investidor, o usuário será direcionado para a tela de login, onde com seu nome de usuário e senha, definidos na Figura 28 ele terá acesso as funcionalidades da aplicação.

Figura 30 - Tela de login



The image shows a login screen for an application named "Money MS". The title "Money MS" is displayed in a large, bold, blue serif font at the top center. Below the title, there are two input fields: the first is labeled "Usuário" (User) and the second is labeled "Senha" (Password). Both fields are white with a thin blue border. Below these fields is a grey button with the text "Conectar" (Connect) in black. The entire interface is set against a dark grey background.

4.5 Estudo sobre renda fixa

Como o objetivo principal da aplicação é fornecer o início da educação financeira para o investidor iniciante, na tela de estudo sobre renda fixa, o usuário que está fazendo seu primeiro acesso deverá ler todos os tópicos listados para fazer o estudo dos principais produtos de renda fixa cadastrados. Somente após ter clicado em “Ler” para todos os produtos e ter selecionado a opção “Estou ciente”, que o usuário poderá avançar para a próxima tela.

Figura 31 - Tela de estudo sobre renda fixa



RENDA FIXA

Antes de prosseguir estude o básico sobre os principais produtos de renda fixa existentes no mercado:

Certificado de Depósito Bancário - CDB	Ler
Letra de Crédito Imobiliário - LCI	Ler
Letra de Crédito do Agronegócio - LCA	Ler
Tesouro Direto	Ler
Fundo Garantidor de Crédito - FGC	Ler
Principais Indexadores	Ler

Investir pode causar prejuízos e consequentemente emoções negativas de perda

☒ Estou Ciente

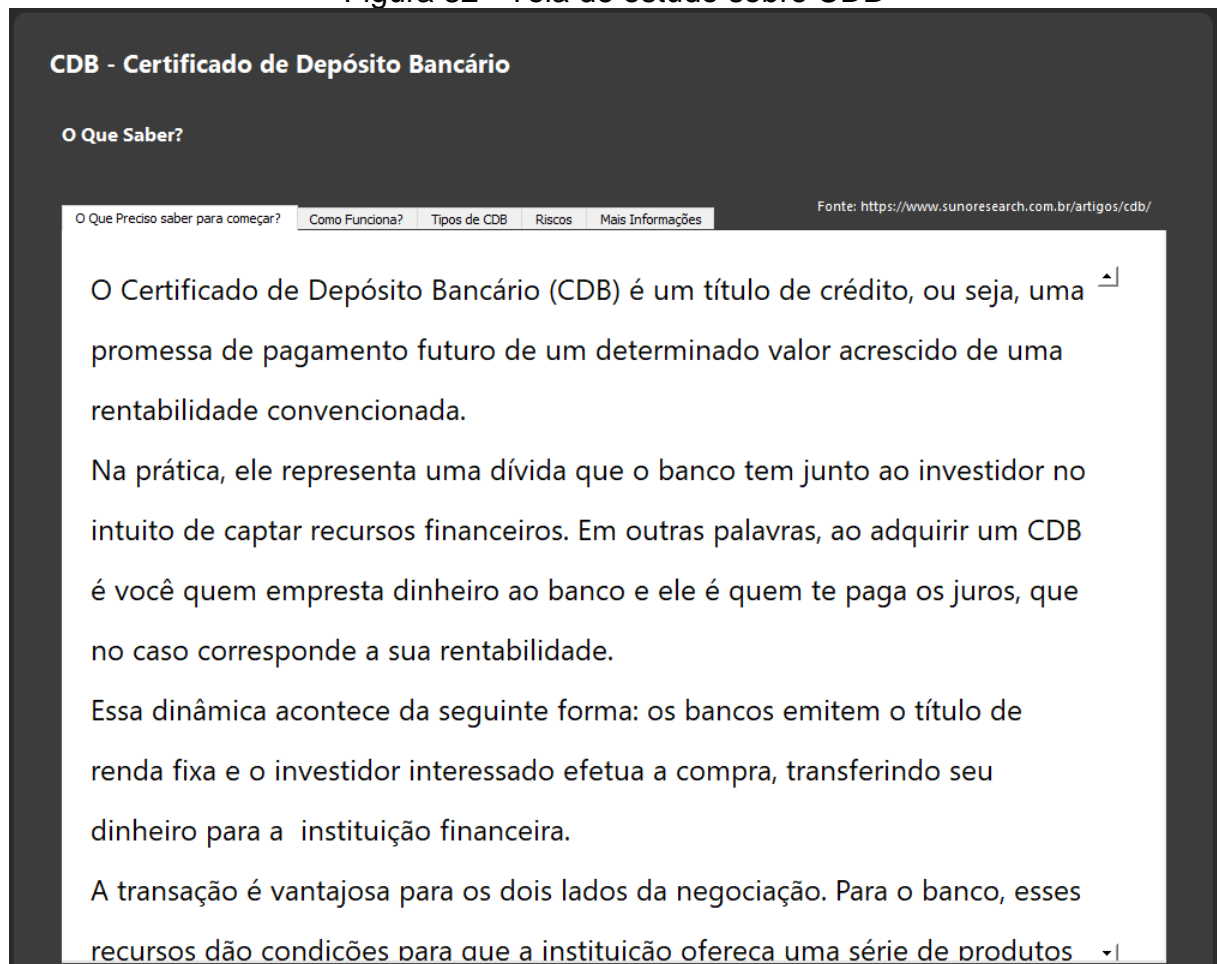
Avançar

4.5.1 Estudo sobre CDB

Quando o usuário clica no botão para ler sobre o Certificado de depósito bancário (CDB), ele é direcionado para uma tela que apresenta diversas abas com informações relevantes em forma de texto para seu estudo sobre esse produto. Essas abas são:

- O que preciso saber para começar?
- Como funciona?
- Tipos de CDB
- Riscos
- Mais informações

Figura 32 - Tela de estudo sobre CDB

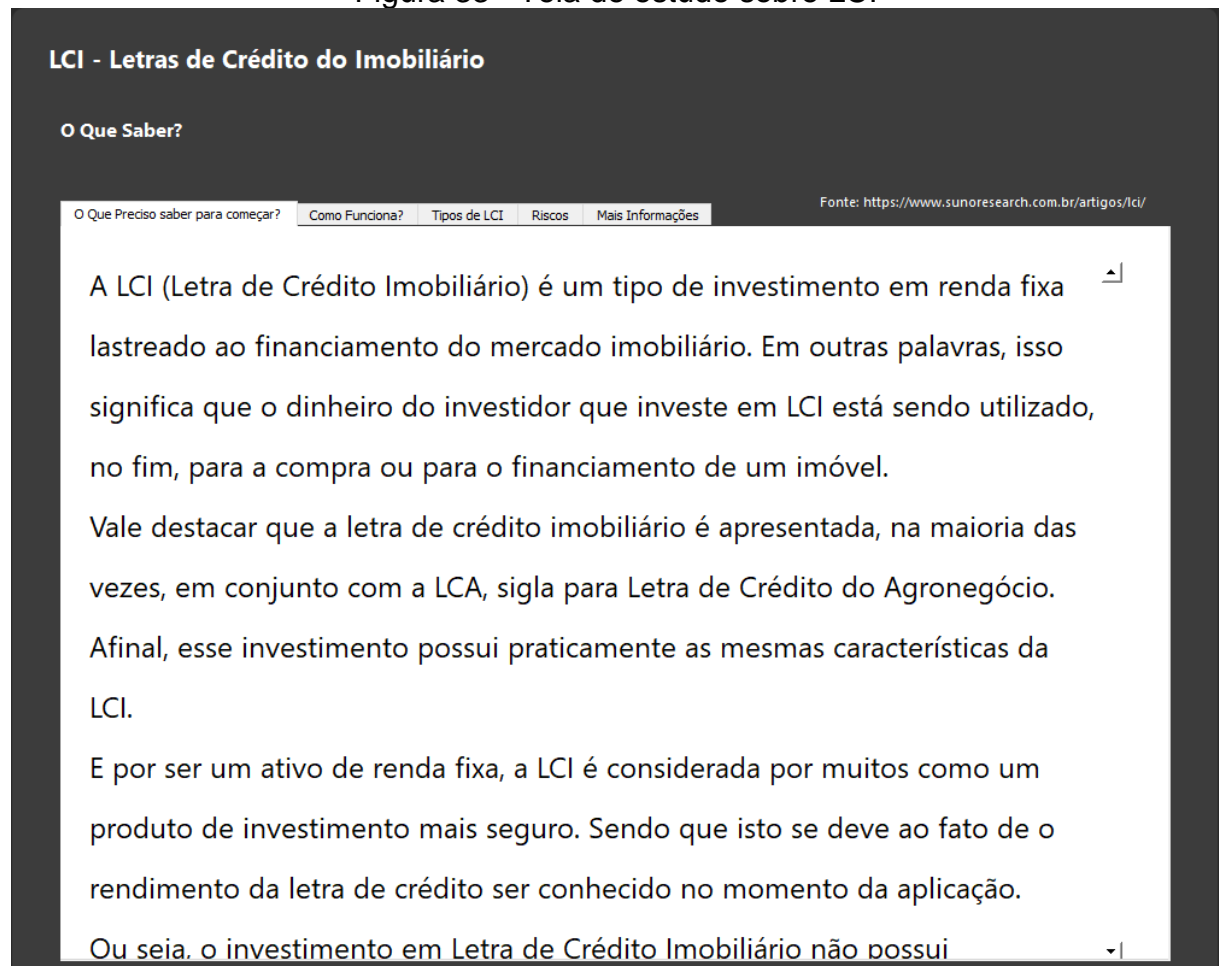


4.5.2 Estudo sobre LCI

Após fechar a tela de estudo sobre CDB, o usuário poderá clicar no botão para ler sobre letras de crédito imobiliário (LCI), a tela para qual ele será direcionado apresentará as seguintes abas para estudo:

- O que preciso saber para começar?
- Como funciona?
- Tipos de LCI
- Riscos
- Mais informações

Figura 33 - Tela de estudo sobre LCI

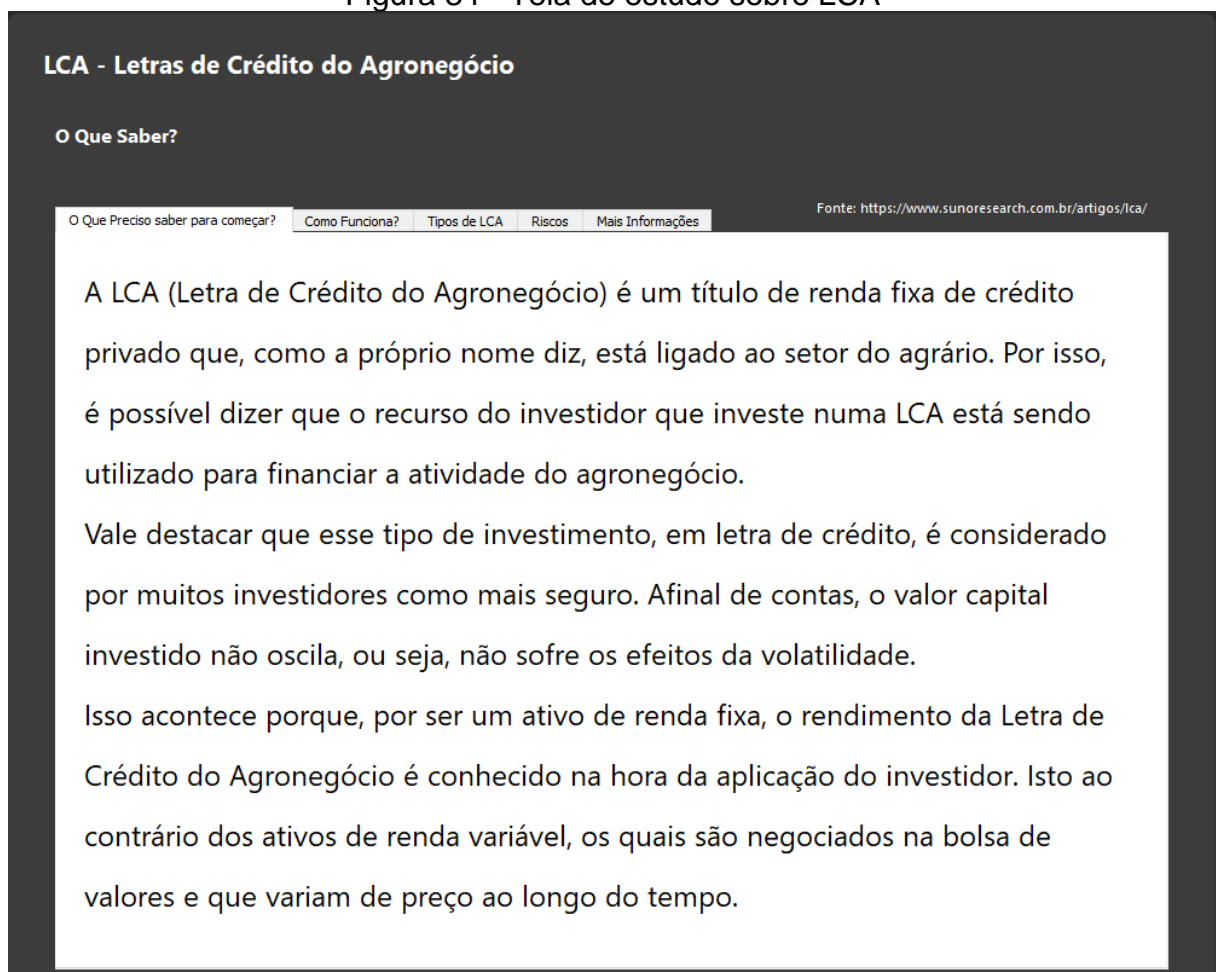


4.5.3 Estudo sobre LCA

Tendo sido fechada a tela de estudo sobre LCI, o usuário poderá clicar em ler sobre as letras de crédito do agronegócio (LCA) e será direcionado para a tela de estudo sobre LCA, as abas para estudo apresentadas nessa tela são:

- O que preciso saber para começar?
- Como funciona?
- Tipos de LCA
- Riscos
- Mais informações

Figura 34 - Tela de estudo sobre LCA

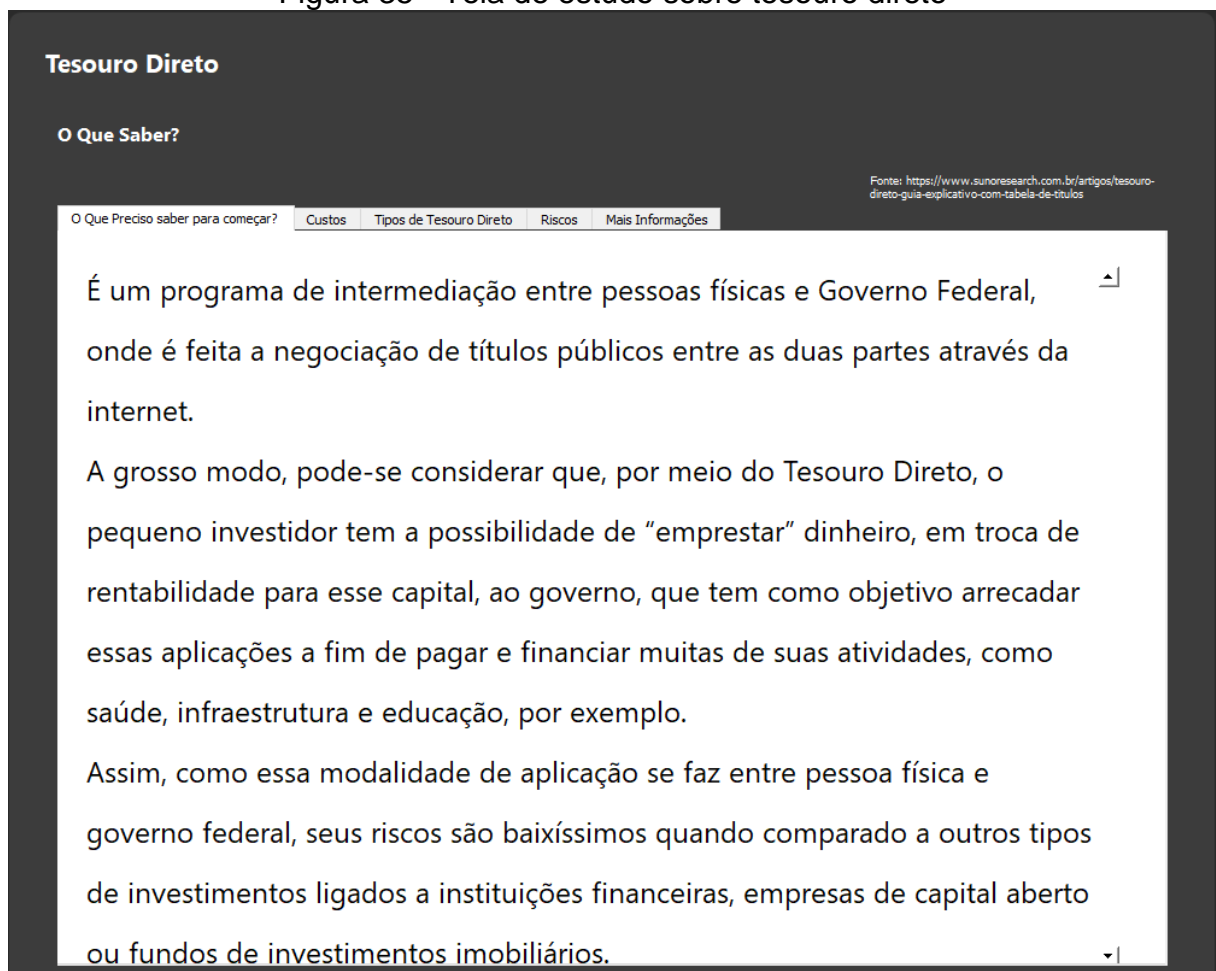


4.5.4 Estudo sobre tesouro direto

Após ter sido fechada a tela de estudo sobre LCA, o usuário poderá clicar na opção de ler sobre tesouro direto. Ao clicar, ele será direcionado para a tela de estudo sobre tesouro direto. As abas com conteúdo para estudo presentes nessa tela são:

- O que preciso saber para começar?
- Custos
- Tipos de tesouro direto
- Riscos
- Mais informações

Figura 35 - Tela de estudo sobre tesouro direto

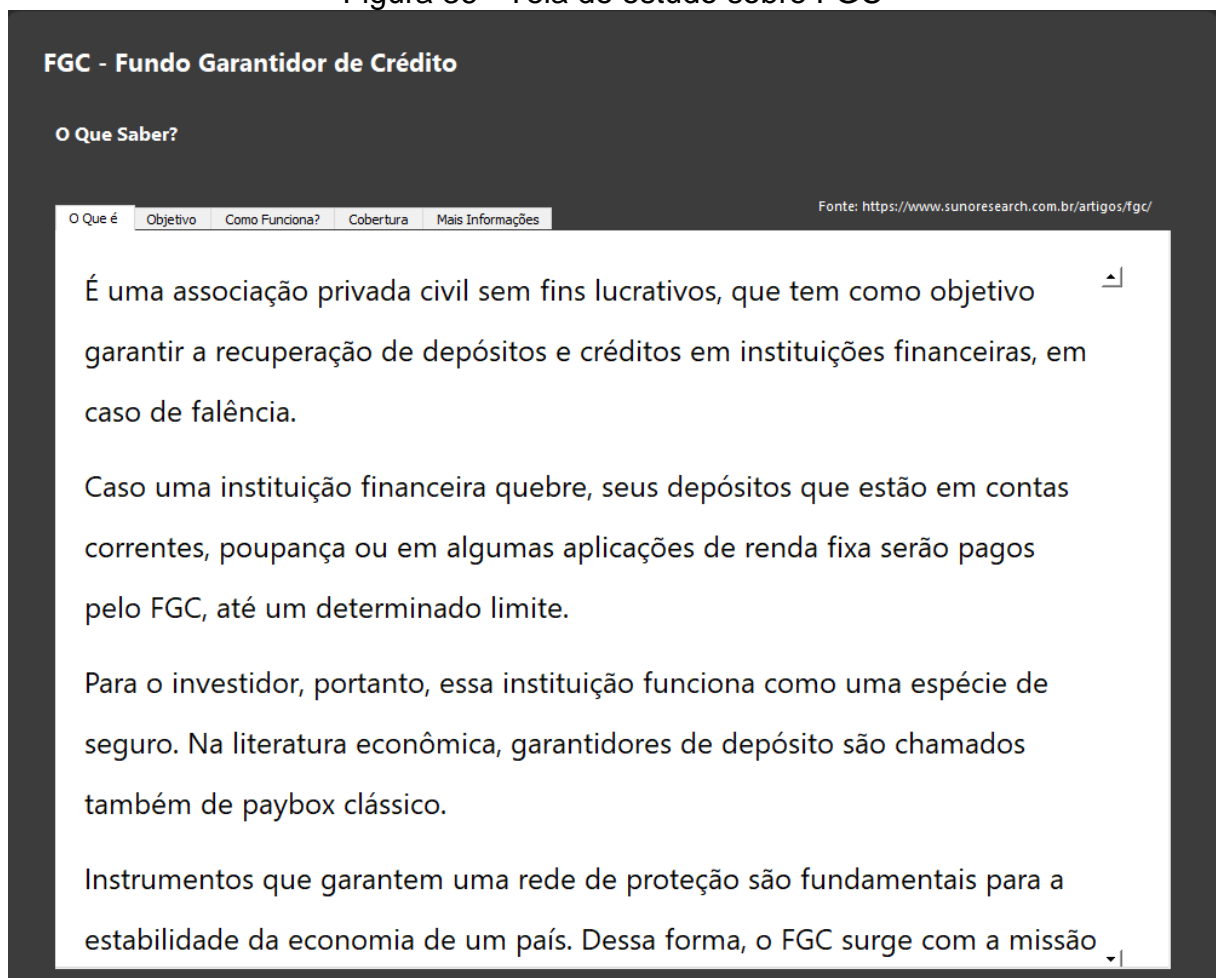


4.5.5 Estudo sobre FGC

Ao fechar a tela de estudo sobre o tesouro direto, o usuário poderá clicar em ler sobre o fundo garantidor de crédito (FGC), ao clicar, ele será direcionado para a tela de estudo sobre FGC que apresentará as seguintes abas de estudo:

- O que é?
- Objetivo
- Como funciona
- Cobertura
- Mais informações

Figura 36 - Tela de estudo sobre FGC

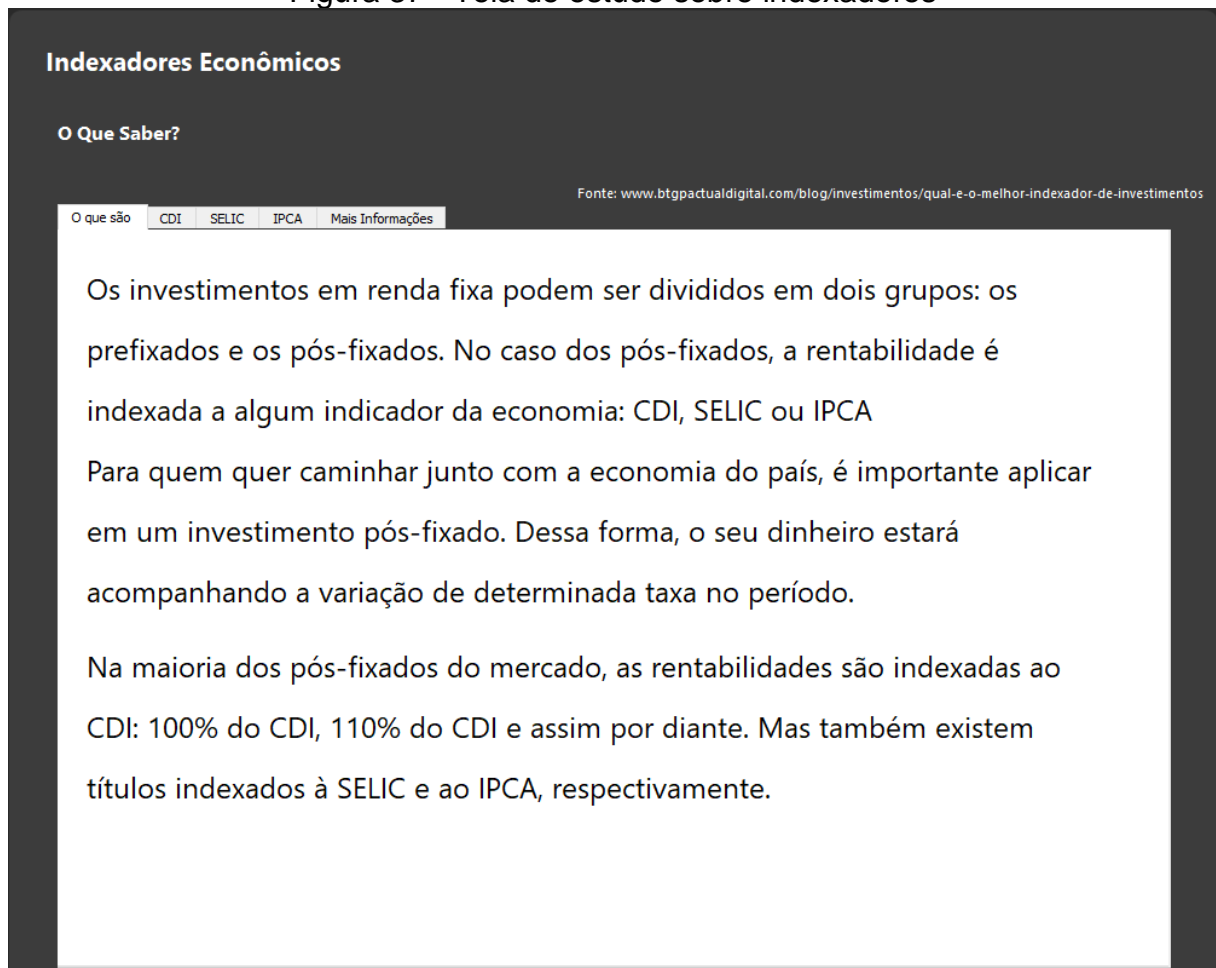


4.5.6 Estudo sobre indexadores

Tendo sido fechada a tela de estudo sobre FGC, o usuário poderá clicar em ler sobre indexadores econômicos, ao clicar ele será direcionado à tela de estudo sobre indexadores, que possui para estudo as seguintes abas:

- O que são
- CDI
- SELIC
- IPCA
- Mais informações

Figura 37 - Tela de estudo sobre indexadores



4.6 Opções de investimento em renda fixa

Após completar o estudo sobre renda fixa disponibilizado na aplicação, o usuário será direcionado para a tela de opções de investimento em renda fixa. Nessa tela será mostrado ao usuário qual é o perfil de investidor em que ele é classificado, baseado no formulário da Figura 29 e é mostrado qual o percentual de seu investimento deve ser dedicado a renda fixa, baseado em seu perfil. Este procedimento visa preencher o requisito do art. 19 parágrafo 4 alínea A do Código ANBIMA de Regulação e Melhores Práticas.

Caso o usuário tenha interesse em ver produtos reais de renda fixa que estão armazenados na base de dados da aplicação, que sejam compatíveis o valor que ele tem disponível para investimento e com a data limite de resgate desse investimento, o usuário pode escolher em quais opções ele tem interesse e selecioná-las, caso ele selecione alguma, ele deverá colocar um número inteiro representando o valor que ele quer investir naquela opção, também deverá selecionar a data limite para resgate daquela opção.

Na simulação que está sendo feita, o usuário selecionou que deseja investir em todas as opções disponíveis. Sendo, as opções, o valor disponível para investimento e a data limite as seguintes:

- CDB – R\$ 5000 – 01/09/2027
- LCA – R\$ 3500 – 15/10/2028
- LCI – R\$ 2300 – 20/11/2029
- Tesouro – R\$ 1900 – 25/12/2030

Com base das opções selecionadas pelo usuário e nos dados fornecidos pelo mesmo, serão processadas as funções referentes as opções escolhidas. Na simulação atual, será feita a leitura dos dados que estão na base da aplicação pela função descrita no capítulo 3.3.2, e em seguida será feita a análise das opções escolhidas pelo usuário passando os dados fornecidos como parâmetro, nesse caso como foram escolhidas as opções CDB, LCA, LCI e tesouro. Então as funções descritas nos capítulos 3.3.5, 3.3.6 e 3.3.7 serão processadas. Por fim será gerado o arquivo .xlsx com a função descrita no capítulo 3.3.8, nesse arquivo estarão as tabelas com os produtos compatíveis com os dados passados pelo usuário, como demonstrados nas figuras 46, 47, 48 e 49.

Após preencher esses campos o usuário poderá selecionar a opção “Deseja aprender sobre renda variável?”

Na tela existe o aviso sobre o produto não conter FGC que é uma requisição do Código de Boas Práticas ANBIMA presente em seu art. 13 em seu terceiro parágrafo

Caso essa opção não seja selecionada, ele irá ser direcionado para a tela descrita no capítulo 4.13.

Para essa simulação, será considerado que o usuário selecionou a opção “Deseja aprender sobre renda variável?”, assim, ele será direcionado para tela descrita no capítulo 4.7.

Figura 38 - Tela de opções de investimento em renda fixa

INVESTIR

Seu perfil é **Conservador**

Recomendamos 100% investidos em renda fixa
fonte: <https://blog.toroinvestimentos.com.br/perfil-de-investidor-conservador-moderado-arrojado>

Marque os investimentos em que tem interesse

Investimento	Valor	Tempo Investido
<input checked="" type="checkbox"/> CDB	5000	01/09/2027
<input checked="" type="checkbox"/> LCA	3500	15/10/2028
<input checked="" type="checkbox"/> LCI	2300	20/11/2029

O produto abaixo não possui fundo garantidor de crédito

Investimento	Valor	Tempo Investido
<input checked="" type="checkbox"/> Tesouro	1900	25/12/2030

☒ **Você deseja aprender sobre renda variável?**

Pesquisar

4.7 Estudo sobre renda variável

O usuário que desejar aprender sobre renda variável será direcionado para uma tela onde serão mostrados tópicos de estudo, para avançar da aplicação o usuário terá que ler todos os tópicos clicando no botão “Ler” que tem ao lado de cada um.

Figura 39 - Tela de estudo sobre renda variável



RENDA VARIÁVEL

Antes de prosseguir estude o básico sobre os principais fundamentos de renda variável existentes no mercado:

Como Começar a investir na Bolsa	Ler
Como Analisar uma Ação	Ler
Informações Sobre Stop	Ler
Aprofundamento no Assunto	Ler

Esta modalidade de produto não está protegida pelo Fundo Garantidor de Crédito e pode causar grandes prejuízos se investida de forma incorreta

☐ Estou Ciente

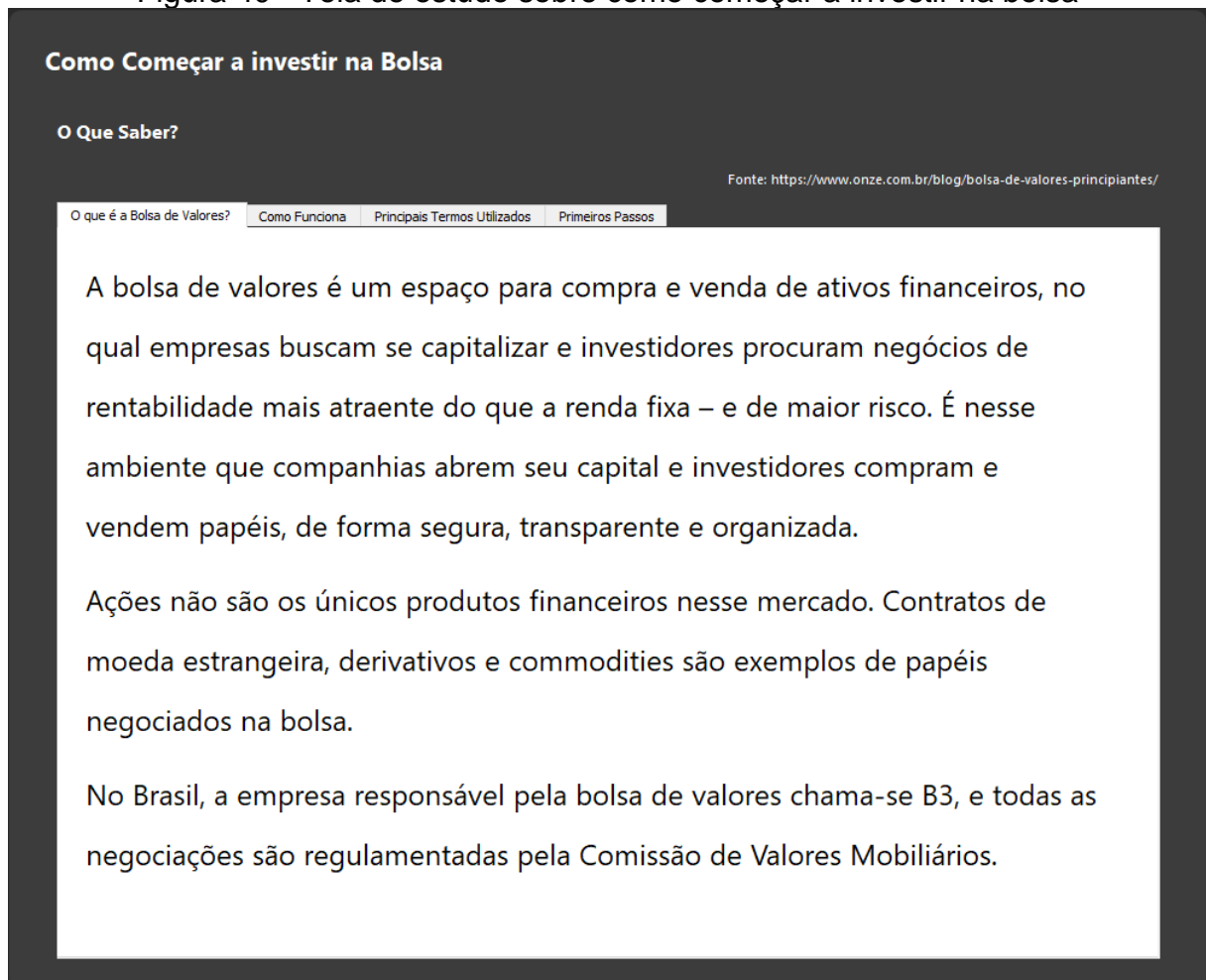
Avançar

4.7.1 Estudo sobre como começar a investir na bolsa

Ao clicar em ler sobre como investir na bolsa, o usuário será direcionado para uma tela, onde ele terá acesso as seguintes abas com conteúdo para estudo:

- O que é a bolsa de valores?
- Como funciona?
- Principais termos utilizados
- Primeiros passos

Figura 40 - Tela de estudo sobre como começar a investir na bolsa

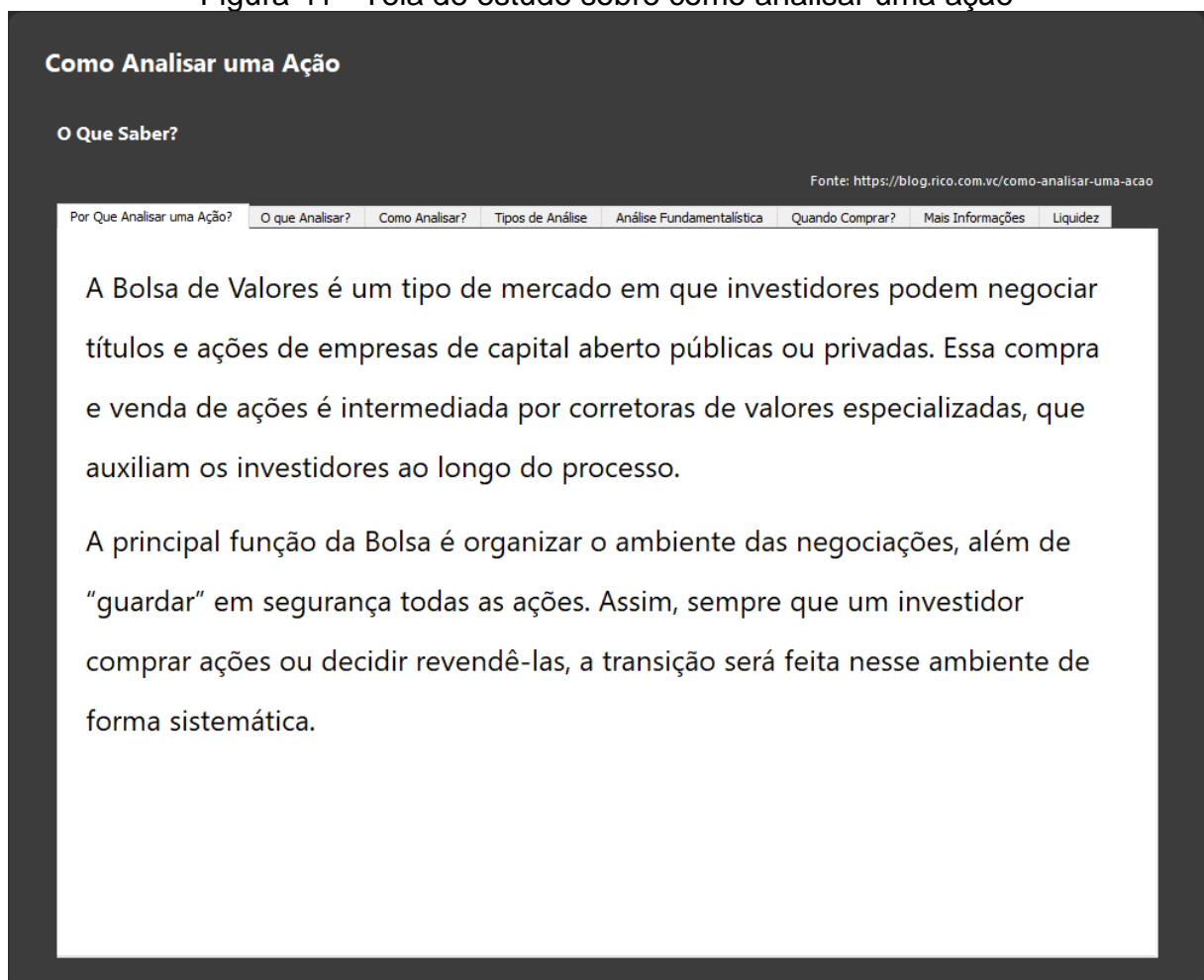


4.7.2 Estudo sobre como analisar uma ação

Após fechar a tela de estudo sobre como começar a investir na bolsa, o usuário poderá clicar em ler sobre como analisar uma ação, sendo assim direcionado para uma tela com conteúdo de estudo dividido nas seguintes abas:

- Por que analisar uma ação?
- O que analisar?
- Como analisar?
- Tipos de análise
- Análise fundamentalista
- Como comprar?
- Mais informações
- Liquidez

Figura 41 - Tela de estudo sobre como analisar uma ação

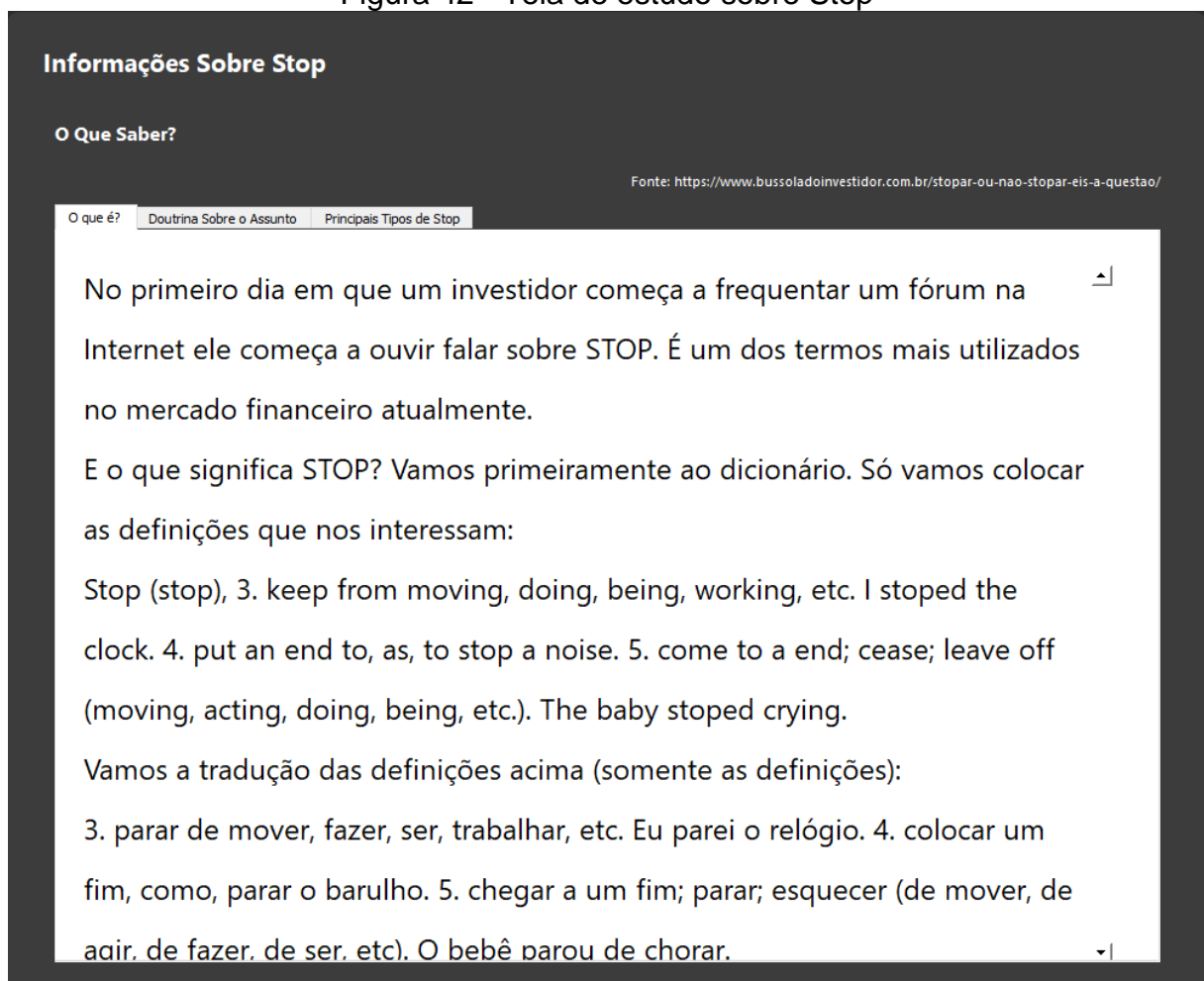


4.7.3 Estudo sobre Stop

Tendo sido finalizado o estudo sobre como analisar uma ação, após sua tela ser fechada, o usuário poderá clicar em ler sobre stop. Ao clicar, ele será direcionado para uma tela de estudo que contém as seguintes abas:

- O que é?
- Doutrina sobre o assunto
- Principais tipos de stop

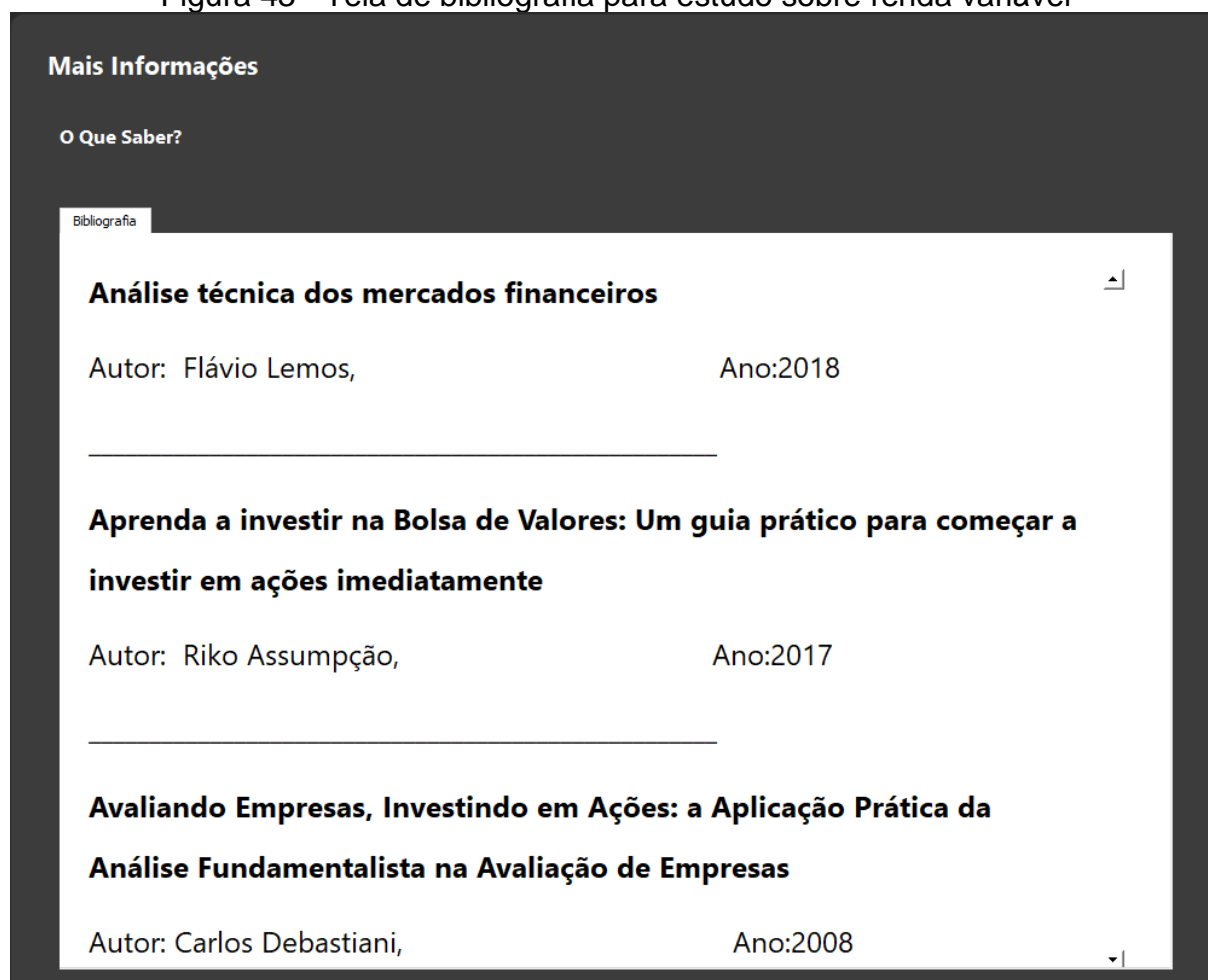
Figura 42 - Tela de estudo sobre Stop



4.7.4 Sugestão de aprofundamento sobre renda variável

Após fechar a tela de estudo sobre stop, o usuário poderá clicar em ler sobre aprofundamento do assunto, que o direcionará para uma tela com sugestões de livros para que possa se aprofundar mais no estudo de renda variável.

Figura 43 - Tela de bibliografia para estudo sobre renda variável



4.8 Avisos sobre renda variável

Após finalizar seu estudo sobre renda variável, o usuário será direcionado para uma tela de aviso sobre os riscos de se investir em renda variável, ao clicar em “concluir” será processada a função descrita no capítulo 3.3.3, gerando o arquivo .xlsx mostrado na Figura 50.

Figura 44 - Tela de avisos sobre renda variável



4.9 Informações finais

A última tela da aplicação contém algumas considerações para o usuário, como o Art. 15 inciso quinto do Código de Boas Práticas ANBIMA, assim como informa o local computador onde foram salvos e o nome dos relatórios .xlsx gerados. Ao clicar em “ok” a aplicação é encerrada.

Figura 45 - Tela de considerações finais



4.10 Arquivos gerados

Na simulação feita, foram gerados dois arquivos .xlsx, um referente as opções de renda fixa selecionadas pelo usuário, nomeado “Opcoes_Adequadas_Renda_Fixa”, nesse arquivo estão as tabelas geradas no capítulo 4.6 com as opções de CDB, LCI, LCA e tesouro direto, adequadas aos parâmetros passados pelos usuário.

Já no arquivo Opcoes_Adequadas_Renda_Variavel, está uma tabela contendo a carteira recomendada de ações pela corretora XP investimentos, para que o usuário possa fazer uso do conhecimento adquirido durante o estudo na aplicação, avaliando as ações dessa carteira recomendada.

Todos os produtos tem classificação de risco para estarem em conformidade com o Art. 9 inciso segundo do Código ANBIMA de Regulação e Melhores Práticas

Figura 46 - Tabela de opções adequadas em renda fixa: CDB

Produto	Data de Resgate	Rendimento Bruto	Aplicação Mínima	Indexador	Rendimento Líquido	Coberto pelo FGC?	Risco	Canal de Atendimento
CDB - PÓS-	01/11/2021	150,00% do CDI	2000	CDI	120.0% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/10/2025	133,00% do CDI	1000	CDI	106.4% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/10/2024	133,00% do CDI	1000	CDI	106.4% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/10/2026	130,00% do CDI	1000	CDI	104.0% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/10/2023	125,50% do CDI	5000	CDI	100.4% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	31/10/2022	120,50% do CDI	5000	CDI	96.4% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	01/11/2021	114,00% do CDI	1000	CDI	91.2% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	02/05/2022	113,00% do CDI	1000	CDI	90.4% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/10/2024	111,50% do CDI	1000	CDI	89.2% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/10/2023	111,50% do CDI	1000	CDI	89.2% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	02/05/2023	111,50% do CDI	1000	CDI	89.2% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	01/11/2021	111,50% do CDI	5000	CDI	89.2% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/10/2025	111,50% do CDI	1000	CDI	89.2% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/04/2021	113,00% do CDI	1000	CDI	87.58% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	01/03/2021	111,50% do CDI	1000	CDI	86.41% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/04/2021	108,50% do CDI	5000	CDI	84.09% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/12/2020	108,00% do CDI	1000	CDI	83.7% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	31/10/2022	104,00% do CDI	100	CDI	83.2% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	01/11/2021	103,00% do CDI	100	CDI	82.4% CDI	Sim	Baixo	atendimento@btgpact
CDB - PÓS-	30/11/2020	106,00% do CDI	1000	CDI	82.15% CDI	Sim	Baixo	atendimento@btgpact

Figura 47 - Tabela de opções adequadas em renda fixa: LCI

Produto	Data de Resgate	Rendimento Líquido	Aplicação Mínima	Indexador	Coberto pelo FGC?	Risco	Canal de Atendimento
LCI - PÓS-	30/10/2023	119,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	06/11/2023	118,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	02/05/2023	117,50% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	31/10/2022	116,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	31/10/2022	116,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	03/11/2022	115,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	02/05/2022	113,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	02/05/2022	112,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	01/11/2021	112,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	01/11/2021	110,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	03/11/2021	109,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	30/04/2021	105,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact
LCI - PÓS-	28/01/2021	102,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpact

Figura 48 - Tabela de opções adequadas em renda fixa: LCA

Produto	Data de Resgate	Rendimento Líquido	Aplicação Mínima	Indexador	Coberto pelo FGC?	Risco	Canal de Atendimento
LCA - PÓS 01/11/2021		110,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpactu
LCA - PÓS 03/11/2021		109,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpactu
LCA - PÓS 28/01/2021		107,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpactu
LCA - PÓS 30/04/2021		105,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpactu
LCA - PÓS 28/01/2021		102,00% do CDI	1000	CDI	Sim	Baixo	atendimento@btgpactu

Figura 49 - Tabela de opções adequadas em renda fixa: tesouro

Título	Rentabilidade anual	Investimento mínimo	Preço Unitário	Vencimento	Indexador	Rendimento Líquido	Coberto pelo FGC?	Risco	Canal de Atendimento
Tesouro 7,45%		34,51	690,26	01/01/2026		5.96%	Não	Baixo	https://www.tesouro
Tesouro 5,20%		35,85	896,34	01/01/2023		4.16%	Não	Baixo	https://www.tesouro
Tesouro IPCA + 3,52%		40,71	4071,53	15/08/2030	IPCA	2.82% IPCA	Não	Baixo	https://www.tesouro
Tesouro IPCA + 3,04%		56,46	2823,01	15/08/2026	IPCA	2.43% IPCA	Não	Baixo	https://www.tesouro
Tesouro SELIC + 0,1900%		106,39	10639,38	01/03/2025	SELIC	0.15% SELIC	Não	Baixo	https://www.tesouro

Figura 50 - Tabela de opções adequadas em renda variável: carteira recomendada

Empresa	Ticker	Risco
B3 S.A. - BRASIL. BOLSA. BALCÃO	B3SA3	Alto
BCO BRASIL S.A.	BBAS3	Alto
GERDAU S.A.	GGBR4	Alto
OMEGA GERAÇÃO S.A.	OMGE3	Alto
LOCAWEB SERVIÇOS DE INTERNET S.A.	LWSA3	Alto
LOJAS AMERICANAS S.A.	LAME4	Alto
MARFRIG GLOBAL FOODS S.A.	MRFG3	Alto
VALE S.A.	VALE3	Alto
VIVARA PARTICIPAÇÕES S.A.	VIVA3	Alto
VIA VAREJO S.A.	VVAR3	Alto

5 CONSIDERAÇÕES FINAIS

O novo investidor, quando se vê munido de conhecimento necessário para começar a investir ganha confiança e começa a criar interesse por mais conhecimento, é um ciclo o qual este trabalho teve o objetivo de dar início. Esperamos que com a utilização total desta ferramenta pelo usuário, ele se sinta confiante para após ter sua primeira experiência de investimento, consiga andar com as próprias pernas, investindo em outros ramos os quais não sendo tão comuns exijam mais estudo e afinho. Com o tempo e experiência o usuário se tornará mais confiante e terá oportunidades de conseguir maiores retornos financeiros indo além do que lhe foi ensinado pelo aplicativo.

5.1 Limitações e Sugestões de Trabalhos Futuros

Devido à falta de tempo hábil, este trabalho teve sua implementação em forma de um protótipo, o qual poderá futuramente ser aprimorado com a adição de diversas funcionalidades ainda não presentes no mesmo, como por exemplo:

Incluir-se novas janelas ao aplicativo com produtos financeiros mais difíceis e complexos pouco conhecidos pelo novo investidor.

Exibir os relatórios que são gerados ao término da execução do programa no formato (*extensão do Excel*) na própria janela da aplicação, tornando mais fácil sua manipulação pelo usuário. Uma ideia para esta funcionalidade está representada no protótipo abaixo:

Figura 51 – Protótipo de tela

Produto	CDB - PÓS-FIXADO BANCO BTG PACTUAL S/A
Data de Resgate	01/11/2021
Rendimento Bruto	150,00% do CDI
Aplicação Mínima	2000
Indexador	CDI
Rendimento Líquido	120.0% CDI
Coberto pelo FGC?	Sim
Risco	Baixo
Canal de Atendimento	atendimento@btgpactualdigital.com

Concluir

Outra ideia também é transformar a aplicação que se encontra em formato desktop para um formato web utilizando Django. O Django é uma ferramenta perfeita para se utilizar como frontend de aplicações Python e foi desenvolvida principalmente para ser usada nesta linguagem.

Além disso também daria mais consistência aos dados utilizados na aplicação se ao invés de serem armazenados em CSV fossem armazenados em um banco de dados que tivesse comunicação com a parte de interface gráfica da aplicação. Isso garantiria a maior consistência e maior facilidade na manipulação das informações que foram guardadas devido a utilização na aplicação.

Outro ponto é que os textos educativos presentes nas janelas de renda fixa e renda variável poderiam ser desacoplados do código Python e modularizados, fazendo com que seja mais prática a sua manipulação em caso de necessidade de atualização ou correção.

Pode-se também utilizar-se data science e machine learning para que baseados em informações anteriores de aplicações de renda fixa e de renda variável tente-se prever resultados futuros de queda ou de subida de valores em tesouro direto ou títulos de renda fixa pós fixados e até mesmo no mercado de ações. Um exemplo é utilizar a biblioteca de Python "sklearn" para previsão de padrões de queda e subida de juros, SELIC, inflação e ações para dizer ao usuário qual a melhor hora de investir e a que tem maior probabilidade de se retirar maiores retornos financeiros.

5.2 Repositório GitHub

github.com/RenatoNS/TCC

REFERÊNCIAS BIBLIOGRÁFICAS

- ANBIMA. **Raio X do investidor brasileiro**. 2020. Disponível em: <https://www.anbima.com.br/pt_br/especial/raio-x-do-investidor-2020.htm>
- NETO, José. SANTOS, José. MELLO, Eduardo. **O mercado de Renda Fixa no Brasil**. São Paulo, 2019.
- MENEZES, Nilo. **Introdução à Programação com Python**. São Paulo, 2010.
- FINANCEONE. **Guia do investidor iniciante**. 2017. Disponível em: <<https://financeone.com.br/wp-content/uploads/2017/12/E-book-InvestidorIniciante.pdf>>
- FAYH, Marcelo. **Perfil de Investidor conservador, moderado, arrojado: Investimento?** 2020. Disponível em: <<https://comoinvestir.thecap.com.br/perfil-de-investidor>>
- BTG PACTUAL DIGITAL. **TAXA SELIC: O que é, atual, mensal, anual e acumulada. Entenda tudo sobre a taxa**. 2020. Disponível em: <<https://www.btgpactualdigital.com/blog/financas/tudo-sobre-taxa-selic>>
- BRASIL. **Art.2º Código ANBIMA de Regulação e Melhores Práticas para Distribuição de Produtos de Investimento**, Brasília. de 02 de jan. de 2019.
- BRASIL. **Instrução CVM Nº539 com as alterações introduzidas pelas instruções CVM Nº 554/14, 593/17 E 604/18**, Brasília. 13 de nov. de 2013
- BONA, André. **Anbima: o que é e qual a sua importância para o mercado?** 2020. Disponível em: <<https://andrebona.com.br/anbima-o-que-e-e-qual-a-sua-importancia-para-o-mercado/>>.
- UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL ESCOLA DE ADMINISTRAÇÃO DEPARTAMENTO DE CIÊNCIAS ADMINISTRATIVAS - Michele Bettim Bandinelli - **FINANÇAS COMPORTAMENTAIS: ORIENTAÇÃO AO PERFIL DO INVESTIDOR PESSOA FÍSICA** - Porto Alegre – 2010.