



**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DO TOCANTINS
CAMPUS PARAÍSO DO TOCANTINS
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

JEFTÉ LOPES GREGORIO

PROVA DE CONCEITO PARA CONSTRUIR UM CHATBOT DE INVESTIMENTOS

**PARAÍSO DO TOCANTINS
2023**

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática ficha catalográfica fornecida pelo IFTO.

Dados Internacionais de Catalogação na Publicação (CIP)
Bibliotecas do Instituto Federal do Tocantins

G821p Gregorio, Jefté Lopes
PROVA DE CONCEITO PARA CONSTRUIR UM CHATBOT DE
INVESTIMENTOS / Jefté Lopes Gregorio. – Paraíso do Tocantins,
TO, 2023.
73 p. : il. color.

Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) – Instituto Federal de Educação, Ciência e Tecnologia do Tocantins, Campus Paraíso do Tocantins, Paraíso do Tocantins, TO, 2023.

Orientador: Me. Ivo Sócrates Moraes de Oliveira

1. Chatbot de investimentos. 2. Investidores individuais. 3. Rasa Framework. I. Sócrates Moraes de Oliveira, Ivo. II. Título.

CDD 005

A reprodução total ou parcial, de qualquer forma ou por qualquer meio, deste documento é autorizada para fins de estudo e pesquisa, desde que citada a fonte.

Elaborado pelo sistema de geração automática de ficha catalográfica do IFTO com os dados fornecidos pelo(a) autor(a).

JEFTE LOPES GREGORIO

PROVA DE CONCEITO PARA CONSTRUIR UM CHATBOT DE INVESTIMENTOS

Trabalho de Conclusão de Curso apresentado à Coordenação do Curso de Bacharelado em Sistemas de Informação, do Campus Paraíso do Tocantins do Instituto Federal do Tocantins, como exigência à obtenção do título de Bacharel(a) em Sistemas de Informação.

Aprovado em: 05/12/2023

BANCA EXAMINADORA

Me. Ivo Sócrates Moraes de Oliveira

Orientador(a)

INSTITUTO FEDERAL DO TOCANTINS - IFTO

Dr. Flávio Eliziário de Souza

INSTITUTO FEDERAL DO TOCANTINS - IFTO

Esp. Luiz Arão Araújo Carvalho

DEFENSORIA PÚBLICA DO ESTADO DO TOCANTINS



Documento assinado eletronicamente por **Ivo Socrates Moraes de Oliveira, Servidor**, em 05/12/2023, às 22:59, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Flávio Eliziário de Souza, Servidor**, em 05/12/2023, às 23:00, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Luiz Arão Araújo Carvalho, Usuário Externo**, em 06/12/2023, às 15:28, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.iftto.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **2204744** e o código CRC **501C2408**.

SUMÁRIO

| | |
|--|----|
| RESUMO | 4 |
| ABSTRACT | 5 |
| 1. INTRODUÇÃO | 6 |
| 2. PROBLEMA DE PESQUISA..... | 9 |
| 3. JUSTIFICATIVA..... | 10 |
| 4. OBJETIVOS..... | 12 |
| 4.1. OBJETIVO GERAL | 12 |
| 4.2. OBJETIVOS ESPECÍFICOS..... | 12 |
| 4.3. ORGANIZAÇÃO DO TRABALHO | 12 |
| 5. REVISÃO DE LITERATURA..... | 14 |
| 5.1. CHATBOTS..... | 14 |
| 5.1.1. <i>Chatbots baseados em Regras, IA e Híbridos</i> | 16 |
| 5.1.2. <i>Construção de chatbots</i> | 18 |
| 5.1.2.1. Dialogflow | 18 |
| 5.1.2.2. IBM Watson..... | 19 |
| 5.1.2.3. Blip..... | 19 |
| 5.1.2.4. Rasa Framework..... | 19 |
| 5.2. NLP E NLU | 22 |
| 5.3. MACHINE LEARNING | 23 |
| 5.4. CONTEINERIZAÇÃO..... | 25 |
| 5.5. INVESTIMENTOS | 26 |
| 5.5.1. <i>Renda Fixa</i> | 26 |
| 5.5.2. <i>Renda Variável</i> | 27 |
| 5.6. CURADORIA | 28 |
| 5.7. TRABALHOS RELACIONADOS..... | 30 |
| 5.7.1. <i>Real Estate Market Data Scraping and Analysis for Financial Investments</i> ... | 30 |
| 5.7.2. <i>RPA Bot Working With Stock Market Share Prices Stock Analysis & Stock Price Prediction</i> | 31 |
| 5.7.3. <i>Algorithmic Trading Bot</i> | 31 |
| 5.7.4. <i>Applications of Web Scraping in Economics and Finance</i> | 32 |

| | | |
|--------|---|----|
| 5.7.5. | <i>On a Chatbot Conducting a Virtual Dialogue in Financial Domain</i> | 32 |
| 6. | MÉTODO DE PESQUISA | 34 |
| 7. | RESULTADOS E DISCURSÕES..... | 39 |
| 7.1. | MODELAGEM DO PROJETO | 40 |
| 7.1.1. | <i>Requisitos funcionais</i> | 40 |
| 7.1.2. | <i>Requisitos não funcionais</i> | 40 |
| 7.1.3. | <i>Infraestrutura e arquitetura</i> | 41 |
| 7.2. | IMPLEMENTAÇÃO DO CHATBOT..... | 42 |
| 7.2.1. | <i>Construção da API</i> | 42 |
| 7.2.2. | <i>Construção do CronJob</i> | 43 |
| 7.2.3. | <i>Utilização do Docker</i> | 45 |
| 7.2.4. | <i>Configuração do Telegram</i> | 46 |
| 7.2.5. | <i>Configuração do Banco de Dados</i> | 47 |
| 7.2.6. | <i>Construção do Chatbot</i> | 48 |
| 7.3. | VIABILIDADE DO CHATBOT | 49 |
| 7.3.1. | <i>Saúde do modelo</i> | 50 |
| 7.3.2. | <i>Pesquisa de campo</i> | 52 |
| 7.3.3. | <i>Resultados da pesquisa de campo</i> | 53 |
| 8. | CONSIDERAÇÕES FINAIS | 57 |
| 8.1. | TRABALHOS FUTUROS..... | 58 |
| | REFERÊNCIAS | 60 |
| | APÊNDICE A – DOCKER-COMPOSE | 65 |
| | APÊNDICE B – CONECTOR CUSTOMIZADO DO TELEGRAM..... | 67 |
| | APÊNDICE C – CREDENTIALS.YML..... | 69 |
| | APÊNDICE D – SCHEMA DA API..... | 70 |

RESUMO

O presente trabalho introduz uma prova de conceito para um chatbot de investimentos, com o objetivo de auxiliar investidores individuais na tomada de decisões informadas em meio ao dinâmico cenário do mercado financeiro, especialmente quando têm pouco tempo disponível para monitorar as flutuações do mercado. Esta pesquisa destaca o papel fundamental da Inteligência Artificial Conversacional, fazendo uso de técnicas como Processamento de Linguagem Natural e Aprendizado de Máquina. Para construir o chatbot de investimentos, utilizou o Rasa Framework e o integrou com o aplicativo Telegram, permitindo o acesso em tempo real de informações financeiras dos ativos na bolsa de valor. Esta implementação seguiu a abordagem Design Science Research (DSR), alinhando-se completamente com o objetivo principal de criar uma solução pragmática para lidar com os desafios enfrentados por investidores individuais. A avaliação do desempenho do chatbot feito com os participantes resultou em feedbacks positivos, reforçando a eficácia do chatbot na entrega de informações diversificadas e atualizadas. Esses resultados enfatizam a relevância da Inteligência Artificial Conversacional na melhoria do processo de tomada de decisões em estratégias de investimento, possibilitando investidores individuais a navegar com confiança pelo complexo cenário do mercado financeiro da bolsa de valor.

Palavras-chave: 1. Chatbot de investimentos. 2. Investidores individuais. 3. Rasa Framework 4. Inteligência Artificial Conversacional. 5. Aprendizado de Máquina.

ABSTRACT

The present work introduces a proof of concept for an investment chatbot, with the aim of assisting individual investors in making informed decisions in the dynamic landscape of the financial market, especially when they have limited time to monitor market fluctuations. This research highlights the pivotal role of Conversational Artificial Intelligence, utilizing techniques such as Natural Language Processing and Machine Learning. To build the investment chatbot, the Rasa Framework was used and integrated with the Telegram application, enabling real-time access to financial information about assets in the stock market. This implementation followed the Design Science Research (DSR) approach, aligning completely with the primary goal of creating a pragmatic solution to address the challenges faced by individual investors. The performance evaluation of the chatbot with participants resulted in positive feedback, reinforcing the effectiveness of the chatbot in delivering diverse and up-to-date information. These results underscore the relevance of Conversational Artificial Intelligence in improving the decision-making process in investment strategies, enabling individual investors to navigate the complex landscape of the stock market with confidence.

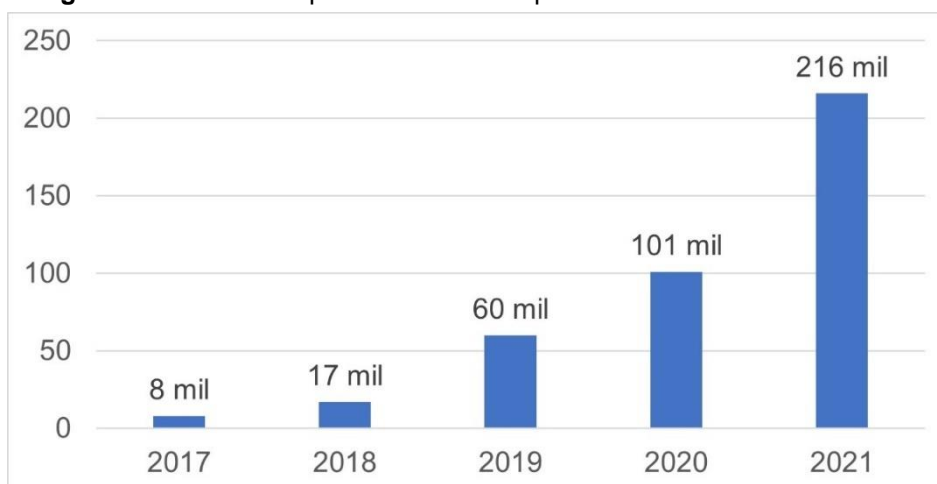
Keywords: 1. Investment chatbot 2. Individual investors 3. Rasa Framework 4. Conversational Artificial Intelligence 5. Machine Learning

1. INTRODUÇÃO

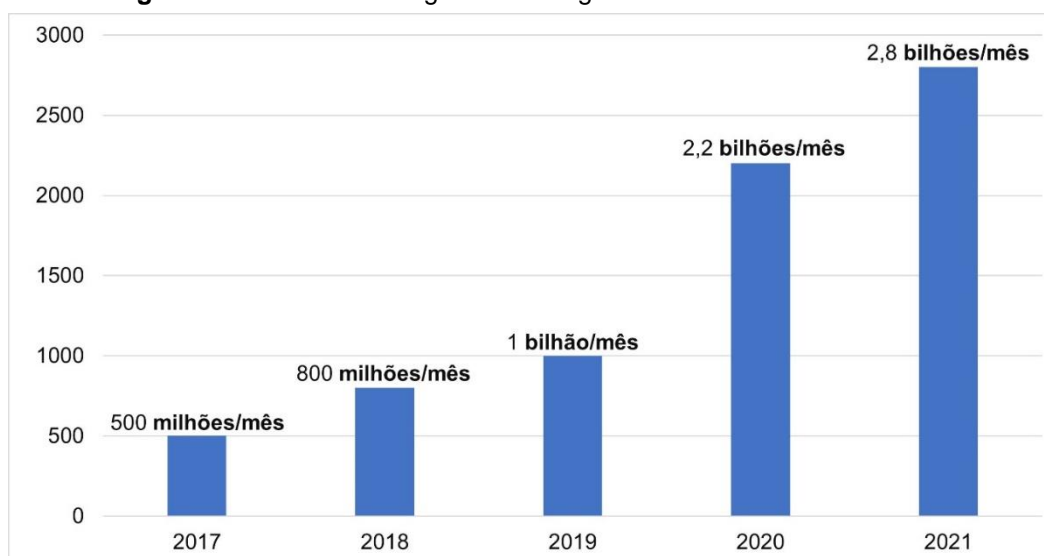
A Inteligência Artificial Conversacional pode ser compreendida como um conjunto de tecnologias que permitem a comunicação entre seres humanos e plataformas de computador. Essa comunicação é viabilizada pela combinação de ideias linguísticas e interação homem-computador, utilizando técnicas como *Natural Language Processing* (NLP - Processamento de Linguagem Natural) e *Machine Learning* (ML - Aprendizado de Máquina). Essas tecnologias são empregadas no desenvolvimento de assistentes de voz, *chatbots* ou *chatterbots* (BARROS e TEDESCO, 2016). Em outras palavras, trata-se de um programa de computador capaz de responder como se fosse uma entidade inteligente ao interagir por meio de texto ou voz. Além disso, ele é capaz de compreender uma ou mais linguagens humanas com o auxílio do NLP (VOGEL, 2017).

Os *chatbots* têm despertado grande interesse e se tornado objeto de pesquisa tanto no mercado quanto no ambiente acadêmico. Esse interesse foi impulsionado pela introdução dos *chatbots* no *Facebook Messenger* em 2016 e pela ampla utilização de aplicativos de mensagens no dia a dia da população, especialmente devido às mudanças nos hábitos das pessoas decorrentes da pandemia COVID-19. Paiva (2021) conduziu uma análise do crescimento no desenvolvimento de *chatbots* no período de 2017 a 2021, utilizando os dados obtidos na pesquisa Panorama Mobile Time, os quais são apresentados nos gráficos da Figura 1 e Figura 2.

Figura 1 – Gráfico de quantidade de *bots* produzidos no mercado brasileiro



Fonte: Adaptado (PAIVA, 2021).

Figura 2 - Gráfico do tráfego de mensagens mensal trocadas com *bots*

Fonte: Adaptado (PAIVA, 2021).

Para a criação dos gráficos apresentados nas Figuras 1 e 2, foram consideradas 96 empresas como base de análise. A Figura 1, de acordo com a pesquisa de Paiva (2021), revela que em 2019 havia 60 mil *chatbots* em operação, número que aumentou expressivamente para 101 mil em 2020, representando um crescimento de 68%. Já em 2021, houve um salto ainda maior, ultrapassando o dobro em relação ao ano anterior, com 216 mil *chatbots* produzidos. Na Figura 2, também analisada por Paiva (2021), observa-se um crescimento de 27% na quantidade de mensagens trafegadas mensalmente. Esse número passou de 2,2 bilhões em 2020 para 2,8 bilhões em 2021.

Uma das principais razões para a ampla adoção de *chatbots* em diversos tipos de negócios é sua capacidade de oferecer atendimento rápido e de alta qualidade ao cliente, reduzindo a necessidade de uma equipe extensa para fornecer esse serviço. Isso permite que a equipe direcione seu tempo e esforço para atividades estratégicas e preparatórias. Na era da informação, a qualidade do atendimento ao cliente é cada vez mais exigida, uma vez que os consumidores bem-informados estão mais conscientes de suas opções.

É comum entre os investidores individuais encontrarem dificuldades para acompanhar de perto o mercado financeiro devido à falta de tempo. A maioria desses investidores possui um emprego de tempo integral e outras responsabilidades que exigem sua atenção diária, o que dificulta a capacidade de acompanhar constantemente as flutuações do mercado.

Nesse cenário o *chatbot* se torna uma solução viável para resolver ou pelo menos mitigar os problemas que surgem com a crescente demanda por agilidade e praticidade, melhorando a eficiência nos serviços financeiros, especialmente para clientes que têm pouco tempo para acompanhar o mercado financeiro. Além disso, o *chatbot* permite a disponibilização de um serviço de atendimento 24/7, ou seja, 24 horas por dia, todos os dias da semana. Os benefícios da implantação dos *chatbots* são vastos e englobam a otimização no tempo do cliente e a economia de recursos (GADELHA, 2019). Apesar de ser um atendimento não humanizado, é possível usar técnicas de *User Experience* (UX – Experiência do Usuário) para oferecer uma experiência envolvente para o usuário.

Visando tornar as informações do mercado financeiro mais acessíveis dinamicamente entre plataformas, o presente trabalho aborda a implementação de uma *Proof of Concept* (PoC - Prova de conceito) de um *chatbot* para auxiliar os investidores individuais. Fornecendo um serviço automatizado para coletar os valores dos ativos em “tempo real”, rastrear os ativos desejados e definir um valor específico para ser notificado e por fim, pesquisar sobre assuntos de interesse sobre os ativos.

2. PROBLEMA DE PESQUISA

Para muitos investidores individuais, acompanhar de perto o mercado financeiro pode ser um desafio, principalmente devido à escassez de tempo. Esses investidores frequentemente possuem empregos em período integral e outras responsabilidades que exigem sua atenção diária, tornando difícil a tarefa de monitorar regularmente as flutuações do mercado. Essa falta de tempo e disponibilidade pode resultar em erros de tomada de decisão ou na perda de valiosas oportunidades, o que pode ter um impacto negativo no desempenho dos investimentos. É fundamental encontrar soluções eficientes que possam auxiliar esses investidores a superar esses obstáculos e maximizar o potencial de seus investimentos.

O artigo intitulado "O dilema do pequeno investidor", de autoria de Queiroz (2021), aborda a importância da educação financeira para os investidores de menor porte e como ela pode ajudá-los a enfrentar os desafios relacionados aos investimentos de forma adequada. O autor destaca a falta de tempo enfrentada pelos investidores individuais, o que dificulta a dedicação necessária ao estudo e análise do mercado financeiro, e, conseqüentemente, a escolha de investimentos com menor risco e maior rentabilidade. E sugere algumas soluções para que os investidores individuais possam superar esses desafios, incluindo a utilização de ferramentas tecnológicas, como os robôs de investimento. Essas soluções podem auxiliar os investidores a otimizar seus processos de tomada de decisão e a lidar de forma mais eficaz com as limitações de tempo.

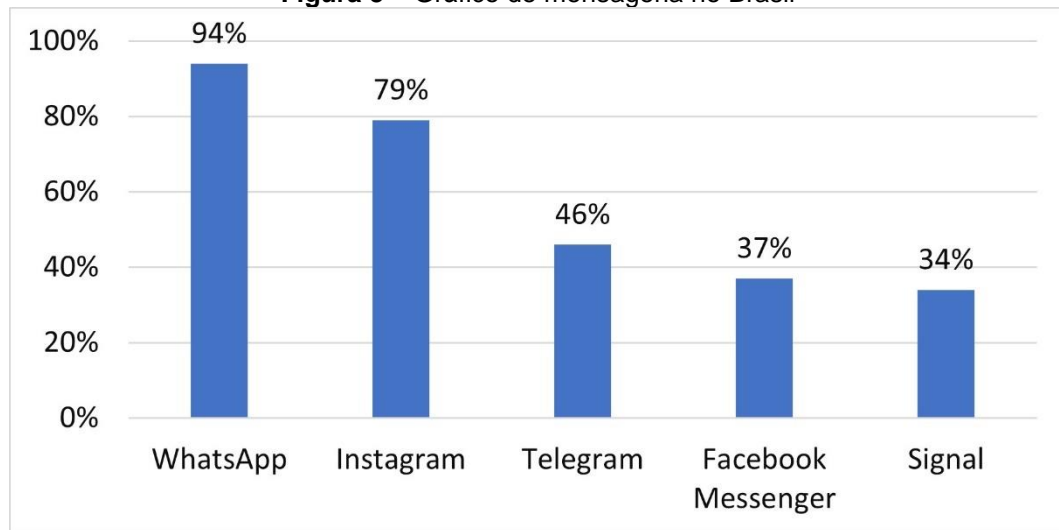
Diante desse cenário, para superar esses obstáculos, como os *chatbots* podem auxiliar os investidores individuais a tomar decisões de investimentos mais assertivas e alcançar seus objetivos financeiros de longo prazo?

3. JUSTIFICATIVA

Com a crescente demanda por agilidade e praticidade, os *chatbots* surgem como uma solução viável para aprimorar a eficiência nos serviços financeiros, especialmente para clientes com pouco tempo disponível para acompanhar o mercado financeiro. Ao oferecer respostas rápidas e precisas, os *chatbots* permitem que os clientes tenham acesso imediato a informações financeiras relevantes para suas decisões de investimento. Dessa forma, os *chatbots* podem contribuir para agilizar o processo de obtenção de informações e auxiliar os investidores na busca por melhores resultados financeiros.

Além disso, os *chatbots* podem ser programados para fornecer atualizações em tempo real sobre o mercado financeiro, mantendo os clientes constantemente informados. Com a disponibilidade 24 horas por dia, 7 dias por semana, esses assistentes virtuais podem oferecer suporte contínuo aos clientes, proporcionando acesso rápido e conveniente às informações financeiras, permitindo que os clientes se mantenham atualizados a qualquer momento.

Para esta proposta, pretende-se atender os usuários por meio da implementação do *chatbot* no *Telegram*. Canal de comunicação que, de acordo com o Paiva (2022), cresceu 15 pontos percentuais em apenas um ano, no período entre 2020 e 2021, alcançando 60% dos smartphones nacionais. Além disso, na Figura 3, é apresentado um gráfico dos aplicativos frequentemente utilizado para ler ou enviar mensagens e o *Telegram* é o terceiro aplicativo mais acessado. Além de ser *Open Source*, não cobra pelo serviço de mensageria e já possui um conector pronto para utilizar no *Rasa Open Source*, framework de código aberto para o desenvolvimento de *chatbot* (RASA, 2022). Essas características tornam o *Telegram* uma escolha promissora para a implementação do *chatbot* proposto.

Figura 3 – Gráfico de mensageria no Brasil

Fonte: Adaptado (PAIVA, 2022).

O gráfico da Figura 3 foi criado com base na porcentagem dos usuários que acessam o app no cotidiano ou quase todo dia. De acordo Paiva (2022), a pesquisa do Panorama Mobile Time/Opinion Box coletou as respostas para a seguinte pergunta: "*Pensando nos últimos meses, com que frequência você abre o WhatsApp/ Facebook Messenger/ Instagram/ Telegram/ Signal para ler ou enviar mensagens?*". Nessa pesquisa 2.081 usuários da Internet tinham o *WhatsApp* instalado, 1.495 tinham o *Facebook Messenger*, 1.734 tinham o *Instagram*, 1.255 tinham o *Telegrama* e 258 tinham o *Signal* instalado. Apesar da maior quantidade de instalações do *Facebook Messenger* em relação ao *Telegram*, a porcentagem de usuários que acessam com mais frequência o *Telegram* se destaca.

4. OBJETIVOS

4.1. Objetivo Geral

Analisar a viabilidade de implementação de um chatbot eficiente que forneça informações precisas e atualizadas sobre os ativos financeiros, proporcionando aos investidores realizar tomadas de decisões mais informadas ao comprar ou vender ações e fundos imobiliários.

4.2. Objetivos Específicos

- Realizar o levantamento de requisitos;
- Construir a API para gerenciar os ativos (CRUD);
- Modelar o sistema conversacional;
- Criar as rotinas de atualização e coleta de dados (*CronJob*);
- Construir o *chatbot* em uma infraestrutura de contêineres;
- Avaliar de maneira qualitativa e quantitativa o sistema desenvolvido.

4.3. Organização do Trabalho

O capítulo 5 trata da revisão de literatura, abordando temas como *chatbots* baseados em regras, IA ou híbridos, que estão presentes em diversos canais de comunicação do cotidiano, com uma diversidade de *frameworks* para construí-los. Ainda em revisão de literatura é explorado temas como NLP, *Natural Language Understanding* (NLU – Entendimento de Linguagem Natural) e *Machine Learning* (ML - Aprendizado de Máquina). Também é apresentado o papel fundamental da curadoria e as métricas estabelecidas e coletadas para melhoria contínua do *chatbot*. E por fim, foram citados alguns trabalhos relacionados ao problema de pesquisa ou que têm objetivos semelhantes ao proposto.

No capítulo 6 é apresentado o método de pesquisa, no qual, descreve o percurso que deverá ser trilhado para conclusão do objetivo apresentado (WAZLAWICK, 2008). Neste capítulo é abordado o foco do estudo, descrição do tipo de pesquisa, estratégia de coleta de dados e o plano de análise dos dados (IFTO, 2016). Já no capítulo 7 é apresentado os resultados e discursões, onde é abordado a geração dos artefatos provenientes da modelagem do projeto, tais como, requisitos funcionais, não funcionais, arquitetura e infraestrutura do projeto. Além de apresentar com detalhes a implementação do *chatbot* e o resultado da viabilidade da sua construção. No capítulo 8 são apresentadas as considerações finais abordando o futuro do projeto e possíveis melhorias mapeadas.

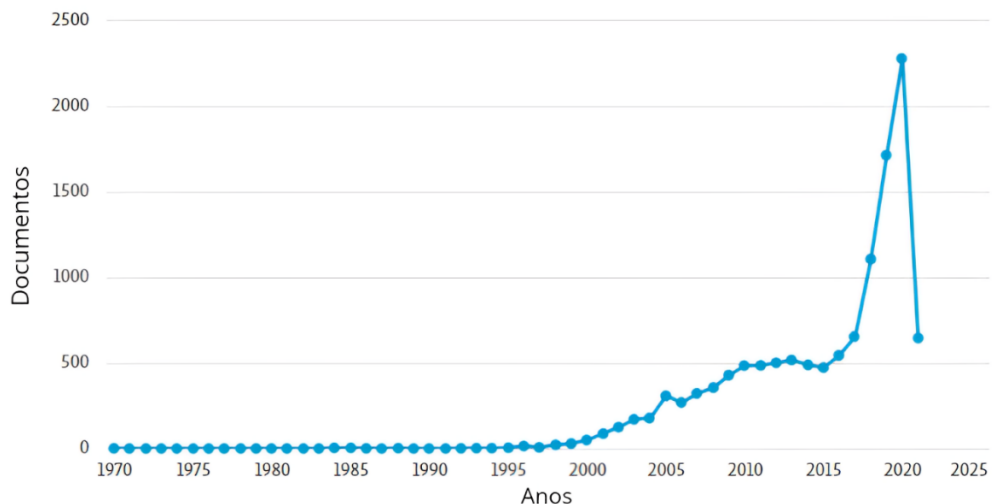
5. REVISÃO DE LITERATURA

5.1. Chatbots

O primeiro *chatbot* a simular a interação humana se chamava ELIZA e foi criado na década de 60, pelo cientista da computação e pesquisador do *Massachusetts Institute of Technology* (MIT - Instituto de Tecnologia de Massachusetts), Joseph Weizenbaum (COMARELLA e CAFÉ, 2008). A partir de então, surgiu diversos projetos e estudos que implementavam e melhoravam a interação dos *chatbots* e *voicebots*. Dentre eles é possível citar (CORREIA, 2018): *Chatbot Parry*, desenvolvido em 1971; *Chatbot Jabberwacky* desenvolvido em 1988; *Chatbot* A.L.I.C.E. desenvolvido em 1995; *Chatbot Smarterchild* desenvolvido em 2001; Assistente Virtual Siri desenvolvido em 2011; Assistente Virtual *Google Now* desenvolvido em 2012; Assistente Virtual Alexa desenvolvido em 2014; *Chatbot* Cortana desenvolvido em 2014; *Chatbots* do *Messenger* desenvolvido em 2016; *Chatbot* Tay desenvolvido em 2016; *Chatbot* IBM Watson desenvolvido em 2017.

O gráfico da Figura 4 demonstra os resultados de uma pesquisa feita nos artigos publicados no Scopus, em que é analisado os resultados de buscas no Scopus no período de 1970 a 2021 para as palavras-chave “**chatbot**”, “agentes conversacionais” ou “sistema de conversação” (CALDARINI; JAF; MCGARRY, 2022).

Figura 4 - Resultados de pesquisa no Scopus, de 1970 a 2021 para as palavras-chave “chatbot” ou “agentes conversacionais” ou “sistema de conversação”.



Fonte: Adaptado (CALDARINI; JAF; MCGARRY, 2022).

No gráfico da Figura 4 aconteceu um crescimento rápido a partir do ano de 2016 em relação aos anos anteriores, na quantidade de pesquisas sobre os temas que envolviam *chatbots*. Esse "Boom" foi decorrente da utilização de *chatbots* no *Facebook Messenger* (DALE, 2016). Mark Zuckerberg em abril de 2016, anunciou oficialmente que estava abrindo sua plataforma *Messenger* em versão beta para permitir a entrada de *chatbots* no aplicativo (ROSENBERG, 2016). Dessa forma, diversas pessoas começaram a interagir com *chatbots* dentro de uma das maiores redes sociais do mercado naquele momento (DALE, 2016).

Vale ressaltar que em dezembro de 2016 foi o ano em que o *Rasa Framework* lançou a sua primeira versão oficial (KONG e WANG, 2021). O fato do Rasa ser uma estrutura de aprendizado de máquina de código aberto, proporcionou diversas contribuições da comunidade para o *framework*, e isso acarretou a melhoria contínua da ferramenta e notoriedade rápida dentro do mercado de inteligência artificial e construção de *chatbots*.

Em 2021, houve uma queda significativa nos resultados de pesquisa do Scopus. No mesmo ano o relatório da consultoria Gartner, intitulado "*The 4 Trends That Prevail on the Gartner Hype Cycle for AI, 2021*" destaca que os *chatbots* estão na fase chamada de "vale da desilusão" (Gartner Hype Cycle, 2021). Nesta fase, as expectativas elevadas em relação à tecnologia não estão sendo atendidas, à medida que os desafios, limitações e problemas práticos se tornam mais evidentes. Esses fatores podem ter acarretado o declínio no entusiasmo e no interesse, ao mesmo tempo em que aumentava a conscientização sobre os riscos associados aos *chatbots*, como a disseminação de desinformação e a propensão à discriminação. Como resultado, essa fase é marcada por uma visão menos favorável da tecnologia, acompanhada de desapontamento e ceticismo.

O tempo que uma tecnologia permanece no "vale da desilusão" no *Gartner Hype Cycle* pode variar amplamente e não segue um padrão fixo. A duração desse estágio depende de diversos fatores, como a complexidade da tecnologia, a rapidez com que os desafios e obstáculos são superados e como o mercado reage. Algumas tecnologias podem passar por essa fase relativamente rápido, enquanto outras podem levar mais tempo para amadurecer e superar os problemas iniciais. Em média, pode variar de alguns anos a uma década ou mais (STEINERT e LEIFER, 2010).

A abordagem *Gartner Hype Cycle* oferece uma perspectiva sobre a trajetória de amadurecimento de uma tecnologia ao longo do tempo, retratando a jornada de expectativas e assimilação de uma inovação tecnológica. Este modelo identifica cinco fases essenciais: (1) "Início da Inovação", (2) "Pico das Expectativas Inflacionadas", (3) "Profundezas da Desilusão", (4) "Declive do Esclarecimento" e (5) "Planalto da Produtividade". O *Hype Cycle* ilustra de forma eloquente como as primeiras expectativas em relação a uma tecnologia podem ser amplificadas, seguidas por um período de desilusão, até finalmente convergirem para uma fase de adoção que é estável e produtiva. É uma ferramenta valiosa para empresas, auxiliando na avaliação do estágio de maturidade de uma tecnologia e na tomada de decisões embasadas sobre investimentos e adoção (SHI e HERNIMAN, 2023).

5.1.1. Chatbots baseados em Regras, IA e Híbridos

O desenvolvimento de diferentes tipos de *chatbots* atendem a diferentes contextos, se ajustando a necessidade do caso de uso. Os *“chatbot são descritos como sistemas capazes de conversar com usuários de maneira natural. Oferecem auxílio ao usuário em uma interação homem-máquina. Possuem capacidade de examinar e até mesmo influenciar o comportamento do seu usuário, perguntando e respondendo às suas perguntas”* (ABDUL-KADER e WOODS, 2015 *apud* JÚNIOR, 2018, p.03). A utilização de inteligência artificial não se aplica a todos os *chatbots*. Dessa forma os *chatbots* podem ser classificados em grupos, que diferencia na forma de se comunicarem com os usuários, sendo eles:

- a) Chatbots estruturados – conhecidos como baseados em regras, funcionam mediante ações e comandos previamente pensados. Todas as suas “habilidades” usam como referência árvores de decisões e/ou fluxos de navegação. O seu funcionamento se dá através de comandos específicos ou palavras-chaves. Caso o *chatbot* não esteja configurado para uma determinada ação ou mensagem, ele não saberá como agir — são mais limitados.
- b) Chatbots baseados em inteligência artificial - têm a habilidade de compreender o que o usuário deseja expressar tomando como referência o que foi escrito ou perguntado. Não estão limitados a regras pré-estabelecidas e conseguem entender a linguagem natural dos usuários com base nas variações possíveis e o modelo de inteligência artificial pode ser continuamente aprimorado de acordo com os inputs dos usuários.
- c) Chatbots híbridos - tipo de *chatbot* mais utilizado, é a fusão dos dois modelos anteriores e possuem parte do fluxo de forma estruturada e outros pontos com o uso da inteligência artificial. Esta categoria é a mais indicada porque, na sua grande maioria, existem assuntos que são

facilmente solucionados com a aplicação de um fluxo estruturado — sendo necessária a aplicação da inteligência artificial em alguns pontos específicos (ALVES, 2019 *apud* FEITOZA, 2021, p.41)

Os *chatbots* baseados em regras são *chatbots* desprovidos de sistema cognitivo, como motor de NLP. Normalmente o fluxo de conversa com o usuário é guiado pelo *chatbots* por meio de botões, limitando o fluxo de conversa e a personalização (MAGALHÃES e CASTRO, 2019). Dessa forma os *chatbots* baseados em regras podem responder com mais confiança, entretanto, não tem flexibilidade para responder mensagens que estão fora da base de conhecimento.

A capacidade de aprender e compreender a linguagem natural é a principal característica dos *chatbots* baseados em inteligência artificial, e o seu objetivo é simular uma experiência de usuário quase humana com os clientes (KAR e HALDAR, 2016). Esses *chatbots* permitem que os usuários tenham uma conversa fluida sem muitos cliques. Visto que, usam tecnologias de NLP para entender a intenção por trás da pergunta e resolver o problema do cliente sem qualquer assistência humana (KERLYL; HALL; BULL, 2007). Entretanto, os *chatbots* de IA costumam passar por um processo de aprendizado, que torna seu processo de implementação complexo, e algumas vezes mais demorado.

Os *chatbots* híbridos combinam os benefícios de ambos os grupos de *chatbots* já citados, ou seja, podem ser utilizados para resolver problemas ou questionários simples, além de, fornecer o poder computacional cognitivo para entender o contexto de interações mais complexos e auxiliar ou resolver o problema do usuário (FEITOZA, 2021). A implementação não é tão longa e complicada quanto como a IA, pois usa estruturas e respostas pré-definidas. Além disso, como é alimentado por IA, o *chatbot* é melhorado constantemente para entender a intenção do usuário e ser mais assertivo nas respostas.

Os *chatbots* possuem dois comportamentos fundamentais, que é o entendimento da mensagem do usuário e a escolha da resposta com base em sua compreensão (GADELHA, 2019). Dessa forma, ao receber a mensagem deve ser feita uma análise para interpretar a mensagem do usuário. Por meio de *regex*, palavras-chaves ou padrões pré-definidos (LACERDA, 2019). Após a classificação da intenção, o *chatbot* retorna uma resposta de acordo com a maior confiança encontrada. Nesta atividade, também é viável desenvolver métodos para que o *chatbot* compreenda o contexto atual da conversa para entregar a melhor resposta possível (LACERDA, 2019).

5.1.2. Construção de *chatbots*

Como abordado no tópico anterior, os *chatbots* são classificados no que tange à forma de se comunicarem com os usuários. E no mercado, existe uma grande variedade de *frameworks*, ferramentas e abordagens que auxiliam no desenvolvimento e manutenção dos *chatbots*.

5.1.2.1. *Dialogflow*

O *DialogFlow* (2022) é uma plataforma de processamento de linguagem natural de propriedade do *Google*, que pode ser utilizada para construção de interfaces conversacionais, utilizados em aplicativos de conversação, como *chatbots* e *voicebots*. De acordo com o *Dialogflow* (2022) a plataforma disponibiliza dois serviços para construção de agentes conversacionais:

- a. Dialogflow CX: Fornece um tipo de agente avançado adequado para agentes grandes ou muito complexos.
- b. Dialogflow ES: Fornece o tipo de agente padrão adequado para agentes pequenos e simples.

5.1.2.2. IBM Watson

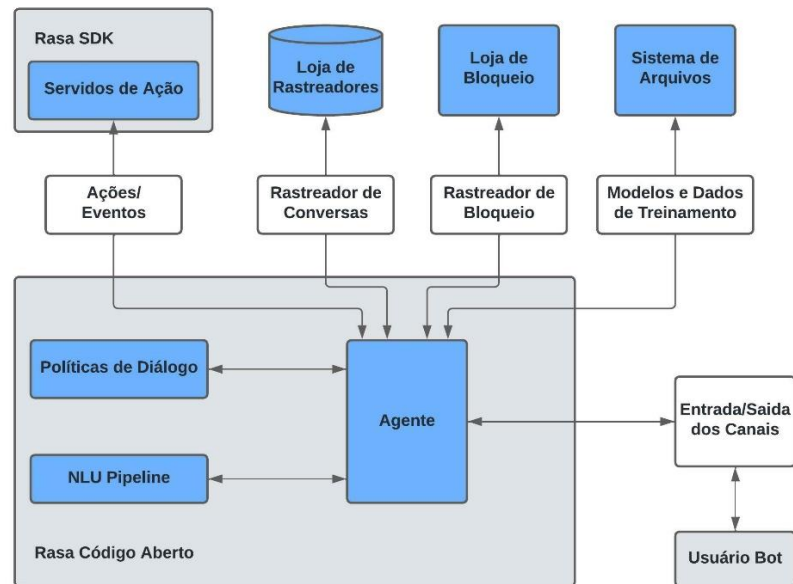
Com o lançamento oficial em 2011, o *IBM Watson* é um serviço baseado em nuvem que pode ser acessado de qualquer lugar. Disponibiliza recursos para o processamento de linguagem natural, possibilitando a detecção do contexto da conversa em mensagens de texto estruturadas e não estruturadas, além de gerenciar as respostas do usuário (NUNES, 2021).

5.1.2.3. Blip

Blip é uma plataforma da *Take* com foco na construção e gerenciamento de aplicações conversacionais. O *Blip* fornece várias ferramentas e uma interface intuitiva para construção e manutenção dos *chatbots*, juntamente com integração nativa de diversos canais de comunicação. Tais como, *WhatsApp*, *Messenger*, *Telegram*, entre outros. Entretanto, o *BLiP* não possui seu próprio serviço de NLP, e depende de provedores de inteligência artificial externos, tais como: *Dialogflow*, *IBM Watson*, *LUIS*, *RASA NLU*, entre outros (BLIP, 2022).

5.1.2.4. Rasa Framework

O *Rasa Open Source* é um *framework* de código aberto para o desenvolvimento de *chatbot*. Possui dois componentes principais: O *Rasa NLU* é responsável pela categorização da intenção e extração de entidade do conjunto de dados de treinamento e o segundo componente é o *Dialog Management Model* (DMM - Modelo de gerenciamento de diálogo), conhecido como, *Rasa Core*, sua arquitetura padrão utiliza o aprendizado profundo baseado em Rede Neural chamado *Transformers* para criar modelos de diálogo mais precisos e eficientes, que de acordo com as intenções identificadas e as entidades extraídas, constrói modelos de resposta probabilísticos (MESHRAM *et al.*, 2021).

Figura 5 - Visão geral da arquitetura Rasa

Fonte: Adaptado (Rasa Framework, 2022)

Na Figura 5 os dois componentes principais do *Rasa Open Source* é o NLU Pipeline e a Política de Diálogo. Como já abordado, o modelo NLU é construído por um pipeline treinado que processa a entrada do usuário. Já a Política de Diálogo decide a próxima ação com base nos dados extraídos durante a conversa, ou seja, realiza o gerenciamento do diálogo (RASA FRAMEWORK, 2022).

A arquitetura abrange diversos serviços essenciais para o pleno funcionamento do chatbot. Incluído nesses serviços está o servidor de ação Rasa, responsável por executar ações personalizadas, como requisições a APIs e consultas em bancos de dados, entre outras funcionalidades. Além disso, integra o rastreador de conversa, responsável por armazenar o histórico de mensagens do chatbot, e o rastreador de bloqueio, utilizado para assegurar a exclusividade das conversas. Por fim, o sistema de arquivos desempenha um papel crucial ao controlar o modelo de aprendizado de máquina desenvolvido durante o treinamento do chatbot, garantindo o carregamento adequado do modelo durante a inicialização do servidor Rasa (RASA FRAMEWORK, 2022).

Em um fluxo de funcionamento simples de conversa com *chatbot* usando a arquitetura do *Rasa Open Source*, o NLU avalia e entende o contexto quando o *chatbot* recebe entrada do usuário. Com base na situação, o *Rasa Core* determina o que deve ser feito e todos os eventos são armazenados no banco de dados do rastreador de conversa. Caso seja necessário o serviço de ações personalizadas, é realizado a chamada de API para o servidor de ação, e o rastreador de bloqueio vai garantir que o *Rasa Open Source* processe as mensagens na sequência correta (RASA FRAMEWORK, 2022).

Além do *Rasa Open Source*, é possível utilizar o *Rasa X* como interface gráfica para auxiliar na manutenção do *chatbot*. A utilização do *Rasa X* visa diminuir a curva de aprendizagem de novos usuários com Desenvolvimento Orientado à Conversa utilizado para agilizar o processo de ajustes e melhorias do chatbot (RASA X, 2022). Na Figura 6 é representado o Quadrante Mágico para Plataformas de IA Conversacional Corporativa, publicado por Magnus Revang, Anthony Mullen e Bern Elliot na Gartner, em 24 de janeiro de 2022 (GARTNER, 2022).

Figura 6 - Quadrante Mágico para Plataformas de IA Conversacional Corporativa



Fonte: Adaptado (GARTNER, 2022)

De acordo com o quadrante mágico do *Gartner* na Figura 6, o Rasa está localizado no quadrante, competidores de nicho, ou seja, são empresas que têm realizações excepcionais em um grupo exclusivo, mas são limitadas em sua capacidade de funcionar em um contexto mais amplo. Vale ressaltar que, o *Rasa* é um *Framework* de código aberto, possibilitando a rápida evolução do *framework* no mercado devido a contribuição da comunidade e a flexibilidade durante o desenvolvimento para adequar o *framework* de acordo com caso de uso.

É importante destacar que, o Rasa não se encontra no Quadrante Mágico para Plataformas de IA Conversacional Corporativa de 2023 por não atender a todos os critérios de inclusão e exclusão. Entretanto, a exclusão do Rasa do Quadrante Mágico não é uma indicação de sua falta de capacidade ou qualidade, mas sim uma reflexão dos critérios específicos e requisitos restritos estabelecidos pela entidade avaliadora.

5.2. NLP e NLU

Graças aos avanços no aprendizado profundo e ao poder que ele trouxe para avaliar o texto em um nível semântico, com descobertas como a incorporação de palavras, o NLP e NLU tiveram sucesso espetacular em tarefas linguísticas e semânticas nos últimos anos (MIKOLOV *et al.*, 2013), visto que, as metodologias anteriores eram principalmente restritas à análise estatística de texto.

"Parte do otimismo em torno do Processamento de Linguagem Natural veio dos escritos de Noam Chomsky, que nos anos 50 propôs a teoria sobre Estruturas Sintáticas, que era uma teoria formal da estrutura da linguagem humana. A teoria dele também tentava fornecer uma estrutura para o conhecimento humano, baseada no conhecimento sobre linguagem" (COPPIN, 2010, p.10).

O NLP tem como objetivo transformar a entrada não estruturada em um formato de dados estruturados para extrair informações, reconhecer voz e texto para desenvolver respostas. Há uma variedade de algoritmos de NLP disponíveis, cada um com uma abordagem distinta para diferentes tipos de problemas linguísticos. *“Por exemplo, as redes neurais recorrentes ajudam a gerar a sequência apropriada de texto. As cadeias de Markov ocultas tendem a ser usadas para marcação de parte da fala. N-grams, um modelo de linguagem simples, no qual, atribui probabilidades a frases para prever a precisão de uma resposta”* (KAVLAKOGLU, 2020).

O NLU determina uma ontologia pertinente referente a organização de dados, no qual, define como palavras e frases estão relacionadas. Enquanto as pessoas conseguem analisar e distinguir contextos parecidos, de forma intuitiva em um bate-papo rotineiro, uma máquina precisa combinar esses contextos para analisar e determinar o significado de cada conversa. Ou seja, o NLU não envolve apenas o entendimento da organização das palavras, mas também tem o discernimento do assunto e do contexto (NORVIG, 2013).

5.3. Machine Learning

A IA é um ramo relativamente novo dentro da ciência e da engenharia, o termo foi usado pela primeira vez em 1956, logo após a segunda guerra mundial (NORVIG, 2013). Pesquisadores neste assunto ainda estão debatendo o conceito de inteligência artificial. A compreensão de algo artificial não é complexo, visto que, está relacionado a tudo que é criado pelo homem. De acordo com o dicionário de Cambridge a definição de artificial é:

Feito por pessoas, muitas vezes como uma cópia de algo natural (CAMBRIDGE DICTIONARY, 2022, Tradução nossa).

A complexidade está no entendimento do conceito de inteligência e a definição de algo artificialmente inteligente. O dicionário Michaelis (2022) define inteligência como sendo a “*Faculdade de entender, pensar, raciocinar e interpretar; entendimento, intelecto, percepção, quengo*”. Além disso, existem concepções distintas sobre inteligência, para a Filosofia: “*Princípio espiritual e abstrato considerado a fonte de toda a intelectualidade*” (MICHAELIS, 2022), Psicologia: “*Capacidade de resolver situações novas com rapidez e êxito, adaptando-se a elas por meio do conhecimento adquirido*” (MICHAELIS, 2022).

Apesar das diversas definições sobre inteligência artificial, Lima, Pinheiro e Santos (2016 *apud* SCHUNK, 2020) de forma geral e clara, afirmam que a IA pode ser entendida como uma série de atos que seriam considerados inteligentes se conduzidos por uma pessoa. Dessa forma, é possível compreender melhor que a IA é uma área da computação que se preocupa em ensinar as máquinas a pensar e agir em determinado contexto de forma inteligente.

Computing Machinery and Intelligence (Máquinas Computacionais e Inteligência), foi escrito por Alan Turing em 1950, no qual, apresentou um teste hipotético com o objetivo de determinar a capacidade de um sistema de computador se comportar de maneira inteligente (TURING, 1950). De acordo com Norvig (2013), o computador precisaria ter as seguintes capacidades para ter maiores chances de passar no teste de Turing:

- a. Processamento de linguagem natural para permitir que ele se comunique com sucesso em um idioma natural;
- b. Representação de conhecimento para armazenar o que sabe ou ouve;
- c. Raciocínio automatizado para usar as informações armazenadas com a finalidade de responder a perguntas e tirar novas conclusões;
- d. Aprendizado de máquina para se adaptar a novas circunstâncias e para detectar e extrapolar padrões.

5.4. Containerização

Os contêineres ganharam notoriedade tanto no mercado quanto no ambiente acadêmico, a partir de 2013, com o surgimento do *Docker*. É válido ressaltar que no início dos anos 2000, surgiram as primeiras soluções baseadas nas características dos contêineres com os projetos de *FreeBSD Jails* (2000), *Linux VServer* (2001), *Solaris* (2004), *OpenVZ Linux Containers* (2005), entre vários outros. Entretanto, a chamada de sistema *chroot*, que modifica o diretório raiz de um processo e seus descendentes para uma nova posição nos arquivos do sistema, foi introduzida em 1979 durante o desenvolvimento do *Unix V7*, iniciando o isolamento de processos no *UNIX* (FREEBSD, 2020) (GOMES, 2018) (MCILROY, 1979). Isso marcou o início do isolamento do processo, que segregou o acesso ao arquivo de cada processo.

De acordo com o *RedHat* (2021), a containerização é definida como o empacotamento do *software*, incluindo componentes como bibliotecas e outras dependências, a fim de criar ambientes segregados para operar na nuvem, na infraestrutura tradicional ou até mesmo no *desktop*. Os contêineres fazem isso por meio de uma espécie de virtualização do sistema operacional (SO) para o gerenciamento da quantidade de CPU, memória e o armazenamento disponível para ser acessível (IBM CLOUD EDUCATION, 2021).

Implantar e orquestrar contêineres passam a ser tarefas difíceis com o aumento em escala de contêineres (FREIRE, 2020), acarretando conflitos nas portas de comunicação e dificuldade para gerenciar a execução dos contêineres. Com a orquestração de contêineres é possível resolver grande parte dos problemas que englobam o gerenciamento dos contêineres em alta escala, visto que, é possível ter maior controle das configurações dos contêineres para proporcionar isolamento adequado e atualizações das aplicações em escala (REDHAT, 2019).

Embora várias abordagens estejam surgindo para orquestração de contêineres, o *Kubernetes* é o mais indicado para aplicações de estruturas robustas (RODRIGUES, 2019). O *Kubernetes* é um sistema de código aberto para orquestração de contêineres com frequentes melhorias feitas por meio da contribuição da comunidade. Bastante utilizado para determinar os recursos necessários para manter a infraestrutura funcional e suportar as solicitações dos usuários, garantindo que as aplicações sejam atualizadas sem tempo de inatividade (KUBERNETES, 2021).

5.5. Investimentos

O Brasil viu o surgimento de suas primeiras bolsas de valores em 1890. No início dos anos 1990, a abertura do mercado de capitais para investidores estrangeiros trouxe otimismo aos profissionais do setor. Desde então, o mercado de capitais tem desempenhado um papel crucial no crescimento econômico do país, resultando em maior dinamismo e participação global (JACOMASSI, 2022).

A abertura do mercado de capitais trouxe consigo uma série de resultados favoráveis que tiveram um impacto significativo no desenvolvimento econômico do país. Dentre esses resultados, destacam-se a expansão dos negócios, o fortalecimento e maior profissionalização do mercado, o crescimento no número de corretoras, o aumento das emissões de valores mobiliários, entre outros. Essas expectativas foram concretizadas ao longo do tempo, com notáveis avanços após a estabilização da moeda brasileira em 1994 (JACOMASSI, 2022).

Vale ressaltar que, existem diversas opções de investimento disponíveis, cada um com suas características particulares, que podem ser categorizados em dois grandes grupos: os de renda fixa e os de renda variável (VASCONCELOS e NEGRES, 2022).

5.5.1. Renda Fixa

Renda fixa é um tipo de investimento em que a remuneração é conhecida antecipadamente. É caracterizada por uma taxa de juros fixa ou indexada, proporcionando estabilidade e segurança ao investidor. Alguns exemplos são:

- **Tesouro Direto:** A iniciativa do governo brasileiro permite o investimento direto em títulos públicos federais sem intermediários. É uma opção segura e acessível para investidores individuais, contribuindo para a democratização do acesso ao mercado financeiro (PESENTE, 2019).

- **Certificado de Depósito Bancário (CDB):** É um título emitido por instituições financeiras para captar recursos. É uma forma de investimento em que o investidor recebe juros sobre o valor emprestado ao banco. O CDB pode ser pré-fixado ou pós-fixado (NETO, 2014).
- **Letra de Crédito Imobiliário (LCI) e Letra de Crédito do Agronegócio (LCA):** são títulos emitidos por instituições financeiras com o objetivo de financiar o setor imobiliário e o agronegócio, respectivamente. Ambos são investimentos em que os investidores emprestam dinheiro para a instituição e recebe juros sobre o valor investido, geralmente isentos de imposto de renda para pessoas físicas (NUBANK, 2023).

5.5.2. Renda Variável

Renda variável é uma forma de investimento em que a rentabilidade não é fixa e pode variar de acordo com as condições do mercado. Investir em renda variável, como ações e fundos de investimento, implica em assumir maior risco, mas também oferece a oportunidade de obter retornos mais significativos. Alguns exemplos são:

- **Ações:** São títulos que representam a propriedade de uma parte de uma empresa. Ao comprar ações, os investidores se tornam acionistas e têm direito a participar nos lucros e decisões da empresa. As ações são negociadas em bolsas de valores, como a B3 (Brasil, Bolsa, Balcão), onde os investidores podem comprar e vender esses títulos (PESENTE, 2019).
- **Fundos de investimento:** Os fundos de investimento constituem uma forma coletiva de aplicação de capital, unindo diversos investidores em uma única estrutura. Sob a gestão de profissionais especializados, esses fundos diversificam seus investimentos em vários ativos, como ações, títulos e imóveis, seguindo estratégias específicas. A participação dos investidores se dá por meio da aquisição de cotas do fundo, o que lhes confere proporção nos ganhos obtidos. Essa abordagem coletiva oferece vantagens como a diversificação de riscos e a gestão especializada, tornando-se uma opção atrativa para investidores (NETO, 2014).

- **Fundos Imobiliários:** Os investidores adquirem cotas do fundo, tornando-se proprietários proporcionais dos imóveis, recebendo rendimentos como aluguéis. Esses fundos oferecem acesso ao mercado imobiliário de forma acessível, com diversificação e liquidez (SCOLESE et al., 2015).
- **ETFs:** *Exchange Traded Funds* são fundos de investimento negociados em bolsa que têm como principal objetivo acompanhar o desempenho de um índice de mercado específico. Esses fundos são compostos por uma diversidade de ativos, tais como ações e títulos, proporcionando aos investidores uma abordagem altamente diversificada e eficiente para investir em diferentes setores e segmentos do mercado (NUBANK, 2023).

5.6. Curadoria

Quando aplicadas a *chatbots*, as metodologias tradicionais de avaliação de usabilidade e experiência do usuário em softwares e sistemas online, podem não ser tão confiáveis. Apesar das diferentes ferramentas disponíveis para a criação de um *chatbot*, a complexidade de desenvolver um, principalmente os *chatbots* híbridos, ainda é grande. É difícil avaliar e comparar diferentes sistemas de *chatbot* em termos de eficiência, eficácia, cumprimento de metas e satisfação do usuário (MAROENGSI et al., 2019).

Conversation-Driven Development (CDD) é uma abordagem de melhores práticas para o desenvolvimento de *chatbots* (RASA, 2022). Este procedimento envolve avaliar a interação do usuário e incorporar o assistente de IA a partir dos resultados e dados coletados. De acordo com Rasa (2022) o CDD inclui os seguintes processos não lineares, que reforça a necessidade de revisar as ações anteriores repetidamente à medida que desenvolve e melhora o *chatbot*:

1. Compartilhe o assistente com os usuários o mais rápido possível.
2. Revise as conversas regularmente.
3. Anote mensagens e utilize como dados de treinamento NLU.
4. Teste os comportamentos esperados do assistente.
5. Acompanhe as falhas do assistente e meça o desempenho ao longo do tempo.
6. Corrija como o assistente lida com conversas malsucedidas.

O CDD é comparável à ideia de Rosenberg (2006) de "aprender pelo uso", no qual, afirma que o engajamento com o usuário final é o início do processo de aprendizado de uma nova tecnologia, visto que, existem alguns cenários de interação homem-máquina que se tornam aparentes somente após uso intensivo e de longo prazo. Dessa forma, o aprendizado pelo uso é bastante utilizado durante a curadoria de *chatbots*. Como os consumidores se envolvem com o *chatbot* regularmente, é mais fácil identificar ocorrências imprevistas ou novos requisitos que devem ser incorporados ao *chatbot* para garantir a qualidade com respostas mais assertivas (REINA e CRUZ, 2020).

Saber quais métricas devem ser estudadas antes de construir o *chatbot* é tão crucial quanto o trabalho de curadoria. Visto que, entender o objetivo do *chatbot*, auxilia na condução do processo de curadoria. Realizar consultas sobre a causa que acarretou a necessidade de um *chatbot* e identificação de frustrações do cliente podem auxiliar na definição das métricas que sinalizarão, ou não, o sucesso do projeto durante o exame das métricas definidas (UBOTS, 2021).

De acordo com Morgan e Rego (2006), os *feedbacks* são frequentemente usados para avaliar o comportamento das métricas que afetam a performance dos serviços e produtos. A satisfação do cliente pode ser conceituada de diversas formas. Alguns autores estabelecem uma sólida relação entre a satisfação do cliente com os produtos ou excelência na prestação de serviços (ANDERSON et al., 1994). A satisfação do cliente também pode ser expressa como a distância entre as suas expectativas e a performance real (HUTCHESON e MOUTINHO, 1998).

Communication Service Assessment Tool (CSAT), também conhecido como *Customer Satisfaction Score*, é uma ferramenta importante para as empresas, pois permite que elas avaliem a satisfação dos clientes com os serviços prestados. Através de pesquisas, o CSAT ajuda as empresas a identificar oportunidades de melhorias em seus processos, serviços e produtos, de modo a aumentar a satisfação dos clientes (SMIRNOVA, 2020).

5.7. Trabalhos Relacionados

5.7.1. *Real Estate Market Data Scraping and Analysis for Financial Investments*

O projeto "*Real Estate Market Data Scraping and Analysis for Financial Investments*" apresenta a coleta e análise de dados do mercado imobiliário para investimentos financeiros. No estudo, é discutido a importância das informações do mercado imobiliário para os investidores. Essas informações incluem tendências de mercado, a probabilidade de valorização imobiliária e uma avaliação de risco do investimento (SANTOS, 2018).

O trabalho apresenta diversas formas de coletar dados, como a utilização de APIs, técnicas de *web scraping* e a mineração de dados. Além disso, aborda as técnicas analíticas aplicadas para identificar as variáveis que influenciam o valor dos imóveis e para prever como o mercado imobiliário se comportará no futuro, incluindo métodos como a regressão linear, as árvores de decisão e as redes neurais (SANTOS, 2018).

Além disso, o artigo examina as desvantagens associadas à análise de dados do mercado imobiliário, tais como a falta de padrões de dados, a qualidade inconsistente dos dados e o impacto imprevisível de fatores externos sobre o mercado. Por fim, o artigo apresenta recomendações para pesquisas futuras, que englobam a melhoria da qualidade dos dados, a adoção de métodos de análise de dados mais sofisticados e a integração de informações do mercado imobiliário com outras fontes de informações, a fim de obter uma análise mais minuciosa do setor financeiro (SANTOS, 2018).

5.7.2. *RPA Bot Working With Stock Market Share Prices | Stock Analysis & Stock Price Prediction*

O projeto intitulado "*RPA Bot Working With Stock Market Share Price | Stock Analysis & Stock Price Prediction*" apresenta a criação de um robô programado em RPA (Automação de Processos Robóticos), que tem por finalidade coletar, examinar e prever preços de ações no mercado de ações. O robô foi desenvolvido utilizando técnicas de extração de dados e aprendizado de máquina, sendo que obtém os dados de preços e os dados históricos por meio da API do *Yahoo Finance* (YUSUF; KUMAR; KRISHNAN, 2022).

O estudo apresenta a efetividade do robô na coleta e interpretação de dados do mercado de ações, bem como seus resultados nas previsões de preços. O autor enfatiza a relevância do emprego da tecnologia de automação para acelerar o processo de tomada de decisão no mercado de ações e chama a atenção para a possibilidade de estender e adaptar o robô para outras áreas financeiras e de investimentos (YUSUF; KUMAR; KRISHNAN, 2022).

5.7.3. *Algorithmic Trading Bot*

O emprego de algoritmos para automatizar a tomada de decisões de investimento é investigado no artigo intitulado "*Algorithmic Trading Bot*". O texto apresenta um modelo de bot de negociação que examina informações de mercado em tempo real e executa transações no nome do usuário utilizando algoritmos avançados (MATHUR, 2021).

O autor, além de abordar os obstáculos no desenvolvimento de *bots* de negociação, como a proteção da confidencialidade e privacidade das informações financeiras dos usuários, explora a relevância dos algoritmos para gerar julgamentos financeiros precisos e eficazes. O artigo também exemplifica como os *bots* de negociação podem ser utilizados em diversos cenários financeiros, como a negociação de ações e criptomoedas (MATHUR, 2021).

O artigo também discute os próximos desenvolvimentos na aplicação de algoritmos no setor financeiro, como a integração com outras tecnologias, como aprendizado de máquina e inteligência artificial, e a criação de algoritmos cada vez mais complexos para aumentar a precisão e eficácia das negociações (MATHUR, 2021).

5.7.4. *Applications of Web Scraping in Economics and Finance*

O trabalho "*Applications of Web Scraping in Economics and Finance*" investiga como o método de *web scraping* pode ser aplicado nas áreas de finanças e economia. O objetivo principal deste estudo é demonstrar como a coleta automatizada de dados pode ser utilizada para obter informações relevantes para diversas finalidades, como detecção de fraudes e previsões de preços de ações (ŚPIEWANOWSKI; TALAVERA; VI, 2022).

Um dos exemplos discutidos no artigo é como o uso de dados coletados automaticamente pode levar a previsões mais precisas para o mercado imobiliário. Outro exemplo é a identificação de fraudes em transações financeiras através da técnica de *web scraping*. É importante destacar que os autores ressaltam a necessidade de considerar a qualidade dos dados coletados e realizar análises abrangentes antes de tomar decisões financeiras com base nessas informações. Além disso, questões éticas e legais relacionadas à prática de *web scraping* também são discutidas no estudo (ŚPIEWANOWSKI; TALAVERA; VI, 2022).

5.7.5. *On a Chatbot Conducting a Virtual Dialogue in Financial Domain*

O artigo "*On a Chatbot Conducting a Virtual Dialogue in Financial Domain*" explora o uso de *chatbots* para realizar diálogos virtuais no setor financeiro. A fim de fornecer informações aos usuários sobre tópicos financeiros, como investimentos, bancos e finanças pessoais. O modelo proposto utiliza técnicas de processamento de linguagem natural e aprendizado de máquina (GALITSKY; ILVOVSKY, 2019).

O artigo também discute os desafios enfrentados na criação de *chatbots* financeiros, como a complexidade do setor financeiro e a necessidade de garantir a segurança e privacidade dos dados financeiros dos usuários. Além disso, o relatório examina as perspectivas para o uso de *chatbots* no setor financeiro no futuro, como sua crescente sofisticação e integração com outras tecnologias, incluindo *blockchain* e inteligência artificial (GALITSKY; ILVOVSKY, 2019).

6. MÉTODO DE PESQUISA

O presente trabalho visa implementar uma PoC (Prova de Conceito) com o objetivo de analisar a viabilidade de implementar um *chatbot* eficiente que fornece informações precisas e atualizadas sobre os ativos financeiros, proporcionando tomadas de decisões mais informadas ao comprar ou vender ativos. Para isso, a abordagem de *Design Science Research* (DSR) é empregada. Essa metodologia foi escolhida porque se concentra na criação de soluções práticas e inovadoras para problemas complexos, como o desenvolvimento do *chatbot*.

Design Science Research (DSR) tem como objetivo a criação e avaliação de artefatos destinados a solucionar desafios do mundo real, com o propósito de enriquecer o acervo de conhecimentos de um determinado campo. Os artefatos referem-se a entidades tangíveis ou intangíveis que demonstram aplicabilidade na resolução de problemas ou no aprimoramento da compreensão de um fenômeno (DRESCH; LACERDA; JUNIOR, 2015).

Embora a prova de conceito não seja uma garantia, ela possibilita uma maior tranquilidade para modelar o design e lançamento de um novo serviço ou produto, visto que, utiliza uma estrutura de desenvolvimento iterativo, metodologia amplamente utilizada no desenvolvimento de software. Esse método enfatiza a construção gradual do *software* por meio de ciclos iterativos, em que cada iteração inclui etapas de coleta de requisitos, projeto, implementação, testes e avaliação.

Como o nome indica, a Prova de Conceito é fundamentada e construída por meio de um conjunto de testes (provas) para coletar o *feedback* das partes interessadas e usuários, sobre a solução proposta.

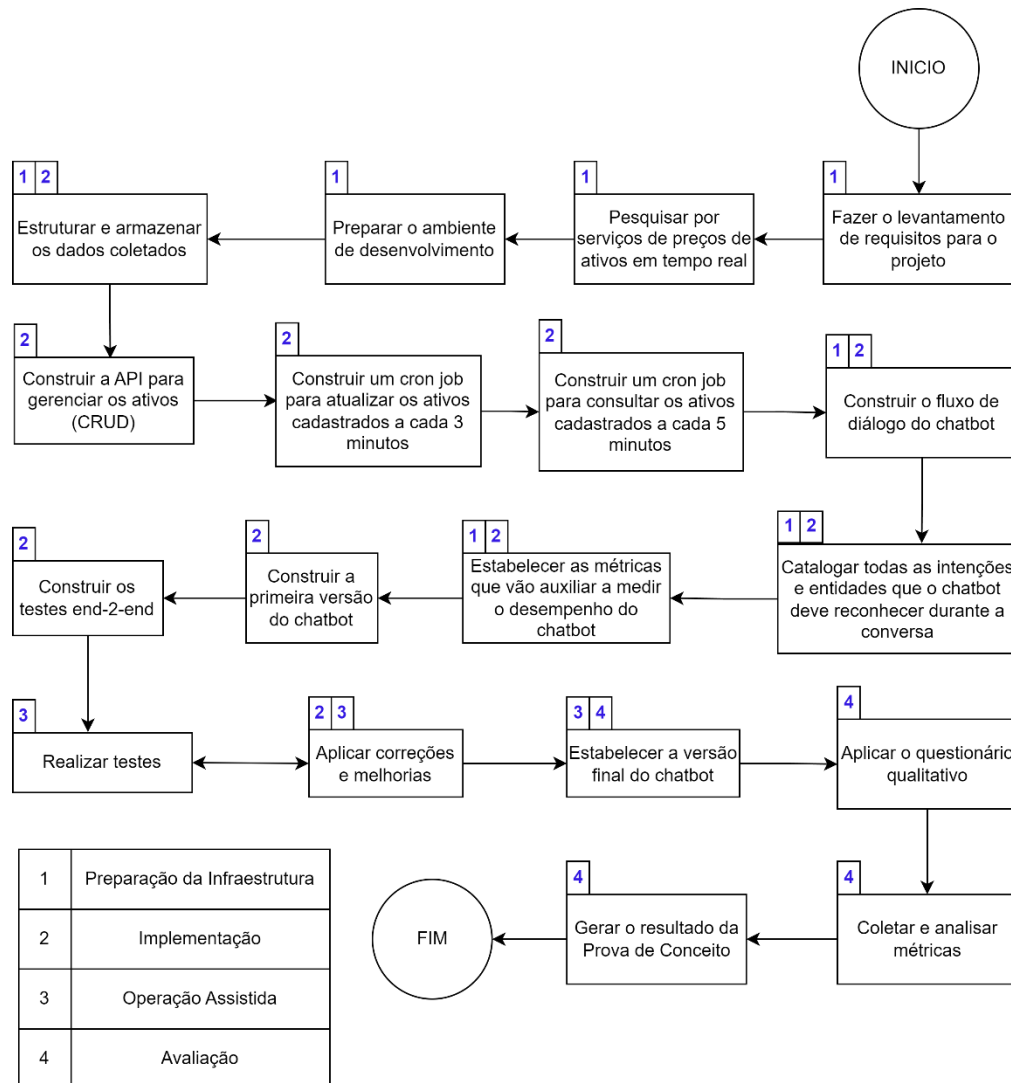
De acordo com Missao e Batista Jr. (2003), a Prova de Conceito deve ser desenvolvida seguindo as seguintes etapas:

- **Preparação da Infraestrutura:** Nesta fase, as atividades e itens mais importantes são selecionados como pilar para a Prova de Conceito. O foco está em testar, avaliar e iniciar o desenvolvimento da implementação completa após uma implementação rápida e exequível em menor proporção.

- Implementação: O objetivo desta fase é colocar os sistemas em funcionamento para que os problemas iniciais do software possam ser reconhecidos e corrigidos por meio de testes diretos com as partes interessadas e/ou usuários-chave.
- Operação Assistida: Aqui, os usuários essenciais empregarão metodologias de controle de qualidade para realizar testes mais extensos. Assim é garantido a qualidade e correções de potenciais falhas no desenvolvimento.
- Avaliação: Nesta etapa, o desempenho da solução é avaliado para identificar falhas e evidenciar melhorias para formalizar todo o processo de implementação de PoC para avaliar a factibilidade da implementação de um projeto mínimo viável.

A Figura 7 representa o fluxo completo de desenvolvimento previsto para a conclusão da PoC, contemplando todas as etapas necessárias para construção de uma PoC, segundo Missao e Batista Jr. (2003). Desde a preparação até a avaliação dos resultados obtidos com a finalização da PoC.

Figura 7 – Modelo de Projeto



Fonte: Elaboração própria, 2022

No começo do fluxo da Figura 7, evidência a necessidade inicial do projeto de fazer os levantamentos dos requisitos funcionais e não funcionais. Além disso, para coletar em tempo real os valores de um ativo na bolsa de valores, rastrear o ativo para notificar quando a ação está em um valor específico e buscar notícias, foi necessário entender as necessidades do usuário e o que ele espera do *chatbot*. Para isso deverá foi realizado o levantamento dos requisitos do usuário.

Foi necessária uma etapa voltada para estudos e pesquisas para fazer o levantamento das soluções disponíveis para coletar em tempo real o valor dos ativos. Logo em seguida foi realizado a configuração de todo o ambiente onde a PoC foi construída e testada. Vale ressaltar que, não foi configurado um ambiente em cloud para hospedar a PoC, visto que, os recursos computacionais que o rasa utiliza geram custos mensais para disponibilização dos serviços

Após a coleta e análise dos dados, foi essencial proceder com a sua estruturação e armazenamento em uma base de dados. Esse processo viabilizou o acesso aos dados pelo Rasa por meio de uma API, que foi desenvolvida com o propósito de consultar e armazenar informações relacionadas aos valores dos ativos e aos dados dos usuários. Para essa finalidade, optou-se por utilizar o MongoDB como sistema de gerenciamento de banco de dados.

Foi implementado um *CronJob* que atualiza os valores na base de dados a cada 3 minutos e monitora as cotações das ações a cada 5 minutos. Essa abordagem evita a necessidade de realizar múltiplas consultas HTTP à API para obter informações sobre o mesmo ativo, em vez disso, permite a consulta direta à base de dados para acessar os valores dos ativos atualizados. Essa otimização aprimora o desempenho da solução, tornando-a facilmente escalável e resiliente.

Antes de começar a implementação do *chatbot* foi projetado todo fluxo conversacional do usuário, no qual, é construído todo fluxo de conversa do *chatbot* com o usuário. Nessa etapa foi necessário catalogar todas as intenções e entidades que o *chatbot* deve reconhecer durante a conversa, assim como a análise e criação dos exemplos para o treinamento do *chatbot*. Visto que, para que o *chatbot* consiga responder adequadamente é preciso ter um conjunto de intenções e entidades bem estruturadas, para gerar o modelo de classificação utilizando o NLU pipeline do Rasa. As métricas utilizadas para validação da performance do chatbot, envolve o *F1-Score*, *Precision* e *Recall*, obtidas após o treino e testes dos modelos gerados.

Após estabelecer as métricas essenciais, foi iniciado a implementação da primeira versão do *chatbot* e a construção dos testes *end-2-end* para realizar os testes iniciais e avaliar como o *chatbot* se comporta em situações reais e identificar problemas antes que o *chatbot* seja implantado. Vale ressaltar que, durante o desenvolvimento do *chatbot* foi utilizado a prática de *Conversation-Driven Development* (CDD).

E por fim, foi elaborado um questionário para uma avaliação qualitativa do *chatbot* como todo. O questionário vai ser enviado para os *stackholders*, ou seja, o grupo de pessoas que vão estar envolvidos direto ou indiretamente durante o desenvolvimento e/ou testes do *chatbot*. O resultado do questionário vai auxiliar a avaliar a PoC, para chegar à conclusão se é uma implementação viável, que, de acordo com Missao e Batista Jr. (2003) é a última etapa do desenvolvimento da PoC.

7. RESULTADOS E DISCURSÕES

No presente capítulo, serão apresentados os resultados decorrentes da análise da implementação do *chatbot*, cujo objetivo é analisar a viabilidade de implementar um chatbot eficiente que fornece informações precisas e atualizadas sobre os ativos financeiros, proporcionando aos investidores individuais meios para tomar decisões mais embasadas ao comprar ou vender ativos, a partir do acesso facilitado a dados relevantes no mercado de ativos.

Nesta seção, os dados coletados serão analisados minuciosamente, a fim de avaliar o desempenho do *chatbot* proposto, bem como seu potencial impacto no contexto do mercado financeiro. Através dessa avaliação criteriosa, busca-se contribuir para o aprimoramento das estratégias de investimento, promovendo uma abordagem mais informada e sustentada por tecnologias de inteligência artificial.

Vale ressaltar, que todos os objetivos específicos foram contemplados nos tópicos posteriores, evidenciando a relevância de cada objetivo para conclusão da POC com sucesso. São abordados os artefatos resultantes da modelagem do projeto, no qual é apresentado os requisitos funcionais, que descrevem as principais funcionalidades e interações esperadas do *chatbot*, além dos requisitos não funcionais, que englobam critérios de desempenho, segurança e usabilidade. Esses elementos são fundamentais para compreender a estrutura e os fundamentos do *chatbot* em questão.

Também será discutida a arquitetura adotada e a API criada, detalhando os componentes e fluxos de informação do sistema, assim como a infraestrutura necessária para a implementação e operação do *chatbot*. Essa análise abrangente permitirá uma compreensão mais detalhada do projeto e sua viabilidade no contexto de investimentos financeiros.

7.1. Modelagem do projeto

7.1.1. Requisitos funcionais

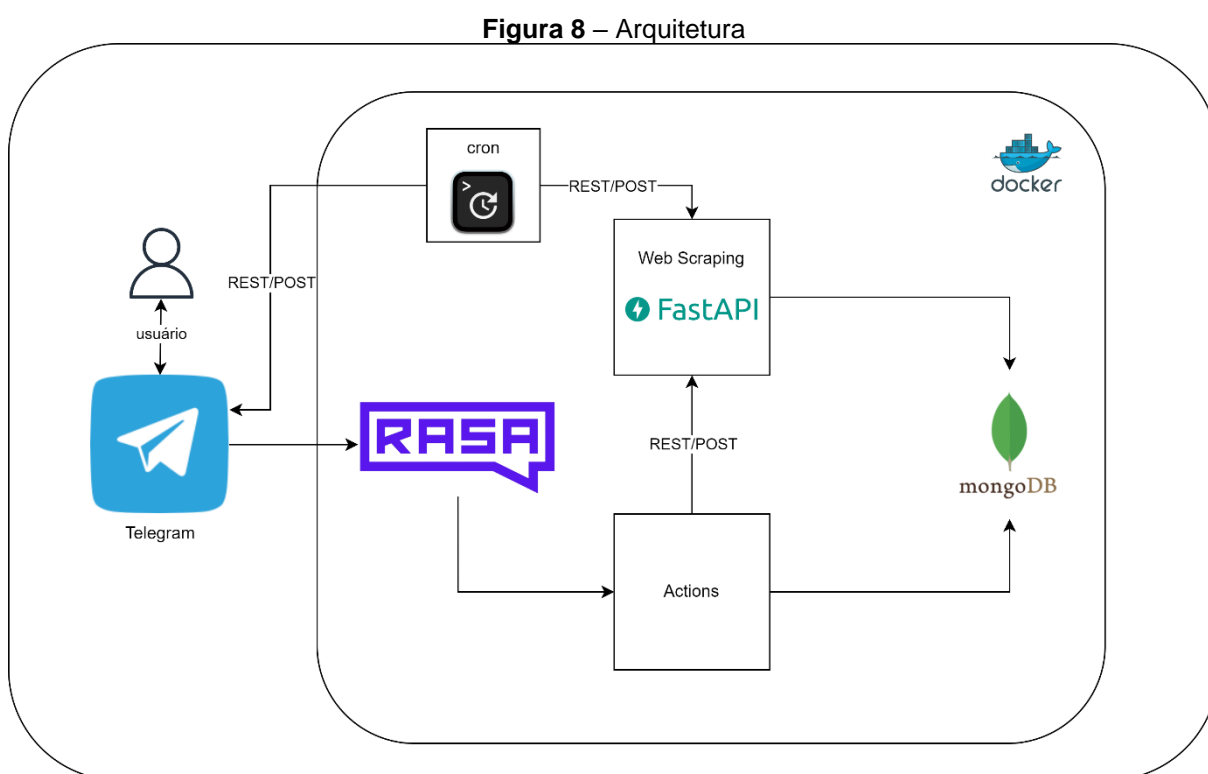
1. O *chatbot* deve permitir que o usuário rastreie a ação desejada e defina um valor específico para ser notificado.
2. O *chatbot* deve ser capaz de enviar notificações ao usuário quando o valor de uma ação atingir o preço definido pelo usuário.
3. O *chatbot* deve permitir que o usuário cancele o rastreamento de uma ação a qualquer momento.
4. O *chatbot* deve permitir que o usuário rastreie múltiplas ações simultaneamente.
5. O *chatbot* deve ser capaz de exibir as informações coletadas sobre as ações em tempo real.
6. O *chatbot* deve ser fácil de usar e ter uma interface amigável ao usuário.
7. O *chatbot* deve buscar informações atualizadas sobre notícias de ativos.

7.1.2. Requisitos não funcionais

1. O *chatbot* deve ter alta disponibilidade e desempenho para que as informações sejam atualizadas em tempo real.
2. O *chatbot* deve ser confiável, garantindo que as informações apresentadas ao usuário sejam precisas e atualizadas.
3. O *chatbot* deve ser compatível com o Telegram.
4. O *chatbot* deve ser capaz de armazenar e gerenciar informações dos usuários de forma segura, como as ações rastreadas e as notificações enviadas.

7.1.3. Infraestrutura e arquitetura

A infraestrutura é responsável por fornecer os recursos computacionais, a conectividade e o ambiente necessário para que o software seja implantado, executado e mantido em funcionamento (SCHMITT e DAMASCO, 2012). Para esse projeto foi utilizado a plataforma de virtualização de contêineres *Docker*, que simplifica o processo de desenvolvimento, implantação e execução de aplicativos em uma máquina *Linux*, *ubuntu* 18.04, com acesso à Internet. Na Figura 8 é apresentado a arquitetura composta por diferentes componentes de software que desempenham funções específicas no sistema.



Fonte: Elaboração própria, 2023

Para construção da arquitetura foi utilizado o *Docker* para gerenciar contêineres, imagens e outros recursos relacionados, apresentados na Figura 8. A arquitetura foi elaborada para escalabilidade, flexibilidade, modularidade e reutilização do sistema, bem como a sua aderência aos requisitos não funcionais, como desempenho, segurança e manutenibilidade. Para isso foi feito um estudo para selecionar as ferramentas e serviços mais eficientes disponíveis para construir o *chatbot*. Na seção 7.2 vai ser abordado cada uma delas com mais detalhes.

7.2. Implementação do *chatbot*

Ao implementar o *chatbot*, foram realizadas configurações de ambiente no *Linux* com *Docker*, incluindo o desenvolvimento de funcionalidades que envolveram, a criação de rotinas no sistema operacional *Linux*, integração com APIs e *Web Scraping*. Além disso, foi configurado o banco de dados, o canal do Telegram e por fim, o levantamento de dados para treinar o NLU (Natural Language Understanding) e a política de resposta do *chatbot*.

Essas etapas foram fundamentais para garantir a interação eficaz entre o *chatbot* e os usuários, proporcionando uma experiência de conversação fluida e coerente.

7.2.1. Construção da API

Foi desenvolvida uma API utilizando o *FastAPI*, com o propósito de coletar informações sobre os valores das ações. Essa API consome os dados provenientes da API do *Yahoo Finance* e os armazena em uma base de dados com estrutura chave-valor utilizando o *MongoDB*.

O *FastAPI* foi escolhido devido ao seu destaque notável em termos de eficiência no desempenho, principalmente devido à sua capacidade de processamento de solicitações assíncronas. Isso torna o *FastAPI* uma escolha superior em comparação ao *Flask*, que não possui suporte a solicitações assíncronas (TURING, 2023).

No contexto deste projeto, o desempenho em relação ao volume de solicitações não é um parâmetro crítico. No entanto, ao implantar o *chatbot* em um ambiente corporativo, é crucial contar com uma estrutura assíncrona de comunicação. Essa abordagem é fundamental para assegurar a capacidade de resposta e a escalabilidade adequada do *chatbot*, independentemente do número de solicitações recebidas. Ao adotar uma comunicação assíncrona, evitam-se bloqueios e atrasos significativos, resultando em um *chatbot* mais robusto e resiliente. Isso fortalece a performance e a disponibilidade do *chatbot*, garantindo uma experiência fluida e responsiva para os usuários.

Para a implementação de um ambiente de produção é essencial utilizar serviços que conseguem suportar altas demandas de requisições. Para isso existe algumas soluções no mercado que possibilita a utilização de serviços para obter informações financeiras em tempo real para projetos financeiros. Dentre as soluções, a Market Data Cedro se destaca por ser um serviço consolidado no mercado com dados precisos e confiáveis, APIs e integrações, dados históricos e em tempo real, suporte e personalização (MARKET DATA CEDRO, 2023).

7.2.2. Construção do CronJob

Para executar o *CronJob* foi criado uma imagem *Docker* utilizando a distribuição *Linux Alpine* devido ao seu tamanho reduzido e eficiência. O *Alpine* foi projetado com o objetivo de ser pequeno e seguro, oferecendo um ambiente de execução otimizado para contêineres (ALPINE LINUX, 2023). O *crontab* foi utilizado para que algumas requisições do *chatbot* fossem executadas automaticamente de segunda-feira a sexta-feira no período de 10:00 até 18:00 horas, horário de funcionamento para negociação de ativos.

A coleta dos dados ocorre em dois momentos distintos por meio da utilização do *CronJob*: a cada intervalo de 3 e de 5 minutos. No primeiro momento, dentro do ciclo de 3 minutos, ocorre a atualização de todos os ativos previamente registrados na base de dados, buscando assim o valor mais atualizado no mercado de ativos. No segundo momento, dentro do ciclo de 5 minutos, é realizada uma verificação para identificar se algum ativo está dentro da faixa de preço estabelecida pelo usuário, para que o usuário receba as notificações dos valores dos ativos.

O comportamento abordado acima ocorre por meio das regras descritas no arquivo *crontab.development*.

```
* /5 10-18 * * 1-5 /usr/scheduler/api.py 2>&1 | tee -a /var/log/cron.log
* /3 10-18 * * 1-5 /usr/scheduler/update_actives.py 2>&1 | tee -a /var/log/cron.log
```

1. **`* /5 10-18 * * 1-5 /usr/scheduler/api.py 2>&1 | tee -a /var/log/cron.log`**

- Este *CronJob* será executado a cada 5 minutos, dentro das horas de 10 a 18 (ou seja, das 10:00 às 18:59).

- Os campos `"*/5"` e `"10-18"` correspondem à frequência e às horas permitidas, respectivamente.
- O comando a ser executado é `"/usr/scheduler/api.py"`.
- `"2>&1"` redireciona os erros de saída (`stderr`) para a saída padrão (`stdout`).
- `"| tee -a /var/log/cron.log"` redireciona a saída padrão para o comando `"tee"` que, por sua vez, grava essa saída no arquivo `"/var/log/cron.log"` e a exibe no console.
- O uso do `"tee"` permite registrar a saída do *CronJob* em um arquivo de log para fins de registro ou depuração.

2. `*/3 10-18 * * 1-5 /usr/scheduler/update_actives.py 2>&1 | tee -a /var/log/cron.log`

- Este *CronJob* será executado a cada 3 minutos, dentro das horas de 10 a 18, de segunda a sexta-feira.
- Os campos `"*/3"` e `"10-18"` correspondem à frequência e às horas permitidas, respectivamente.
- O comando a ser executado é `"/usr/scheduler/update_actives.py"`.
- Novamente, `"2>&1"` redireciona os erros de saída para a saída padrão.
- `"| tee -a /var/log/cron.log"` redireciona a saída padrão para o comando `"tee"`, que grava a saída no arquivo `"/var/log/cron.log"` e a exibe no console.

Em ambos os casos, esses *CronJobs* agendam a execução dos arquivos `"api.py"` e `"update_actives.py"` em intervalos regulares, dentro de um período específico de horas e em dias úteis (de segunda a sexta-feira). Os logs das execuções são registrados no arquivo `"/var/log/cron.log"` para análise posterior.

7.2.3. Utilização do Docker

Todos os serviços estão organizados em uma arquitetura utilizando o *Docker* para simplificar o ciclo de vida do desenvolvimento, visto que, o *Docker* permite a separação dos aplicativos de suas infraestruturas, possibilitando a entrega rápida da aplicação (DOCKER, 2023). Sua arquitetura cliente-servidor se encarrega de coordenar as operações, possibilitando a orquestração de contêineres, imagens e recursos associados.

Para garantir o funcionamento correto do *chatbot*, foram desenvolvidas várias imagens, cada uma com uma finalidade específica. Essa abordagem de construção e utilização de várias imagens permitiu uma divisão clara de responsabilidades e a criação de um ambiente modular e escalável por meio dos contêineres. Para facilitar a criação e a administração de uma aplicação que requer a execução de múltiplos containers foi utilizado o *docker-compose* apresentado no Apêndice A.

Para a construção da API, optou-se pela utilização de uma imagem do *framework FastAPI*, que oferece uma maneira eficiente e escalável de criar interfaces de programação de aplicativos. A estrutura de armazenamento *NoSQL* escolhida para o projeto foi a imagem do *MongoDB*, devido à sua flexibilidade e capacidade de lidar com grandes volumes de dados.

Para a implementação da lógica do *chatbot*, foram empregados o servidor *Rasa* e o *Rasa SDK*. O *Rasa* é uma plataforma de código aberto que permite a criação de *chatbots* inteligentes, capazes de compreender e responder a interações em linguagem natural. Já o *Rasa SDK* é uma biblioteca que facilita o desenvolvimento de ações personalizadas para o *chatbot*, permitindo a integração com sistemas externos e a execução de tarefas específicas.

Além disso, uma imagem da distribuição *Linux Alpine* foi utilizada para a criação das rotinas do *CronJob*. O *Alpine* é conhecido por sua leveza e eficiência, tornando-o uma escolha adequada para ambientes de contêineres. Através das rotinas do *CronJob*, foi possível agendar e automatizar determinadas tarefas, garantindo o bom funcionamento do *chatbot*.

7.2.4. Configuração do Telegram

O *Telegram* foi o canal de comunicação adotado para interagir com o *chatbot*, a escolha baseou-se em diversos fatores vantajosos: o *Telegram* é um *software Open Source*, ou seja, não cobra pelo serviço de mensageria e já oferece um conector pré-configurado para integração com o *Rasa Open Source*. O *Rasa* é o *framework* de desenvolvimento de *chatbot* de código aberto usado no projeto.

Embora o *Rasa* tenha um conector pré-configurado, foi necessário personalizá-lo para viabilizar o envio de mensagens em negrito e permitir o direcionamento de mensagens conforme o fluxo de respostas. A inclusão da formatação em negrito revelou-se essencial para destacar informações importantes, ressaltar os pontos-chaves e enfatizar aspectos relevantes nas respostas. Essa abordagem proporcionou maior clareza e objetividade nas mensagens. O código do conector personalizado encontra-se detalhado no Apêndice B.

Para utilizar o *Chatbot Rasa* no *Telegram*, é necessário realizar a configuração inicial através do *BotFather*. O *BotFather* é fundamental nesse processo, uma vez que é por meio dele que é possível criar um nome exclusivo para o *chatbot* dentro do *Telegram*. Essa etapa é essencial para permitir que os usuários do *Telegram* possam enviar mensagens diretamente ao *chatbot*. Além disso, durante a configuração no *BotFather*, um token exclusivo é gerado para disponibilizar o *chatbot* no *Telegram*.

No *Rasa*, é necessário configurar as credenciais no arquivo *credentials.yml*, conforme apresentado no código do Apêndice C. Nesse arquivo, é necessário adicionar o token gerado pelo *BotFather* e o URL exclusivo gerado pelo *NGROK* para permitir o acesso público à porta local 5005 do servidor *Rasa*.

7.2.5. Configuração do Banco de Dados

Durante a implementação da API, o *PostgreSQL* foi escolhido como o sistema de gerenciamento de banco de dados para armazenar tanto os dados dos ativos coletados quanto as informações dos usuários, apresentado no Apêndice D. No entanto, após análise mais aprofundada dos dados armazenados, foi determinado que um banco de dados *NoSQL* seria mais adequado para atender às necessidades do sistema.

No âmbito deste projeto, optou-se por empregar a estrutura de armazenamento *NoSQL* devido as suas características distintivas, que propiciam uma adaptação flexível a uma variedade de tipos e estruturas de dados. Destaca-se, especialmente, a capacidade de escalabilidade horizontal, fundamental para enfrentar grandes volumes de informações de maneira eficiente. Em contrapartida, os bancos de dados relacionais demonstram habilidade na gestão de dados estruturados, semiestruturados e, em certa medida, não estruturados, embora apresentem limitações significativas no que tange à escalabilidade. (KHAN *et al.*, 2023).

Vale ressaltar que, para a implementação do *Rasa* em ambientes corporativos, é altamente recomendado que as conversas sejam persistidas em um banco de dados, visando facilitar o monitoramento, a auditoria e a sustentação do projeto. Nesse sentido, o *MongoDB* se destaca como uma opção utilizada pela comunidade *Rasa* para armazenamento das conversas. Optar por um banco de dados com uma estrutura de armazenamento semelhante reduz significativamente a complexidade do processo de desenvolvimento.

7.2.6. Construção do Chatbot

Para a construção do *chatbot*, adotou-se a arquitetura ilustrada na Figura 8, mencionado na seção 7.1.3. Além disso, foram desenvolvidos artefatos essenciais, incluindo os esquemas de dados apresentados no Apêndice D e os fluxos conversacionais elaborados para categorizar as intenções e respostas do chatbot. Essa abordagem permitiu a criação de uma base de dados para o treinamento do NLU (*Natural Language Understanding*) e das políticas de resposta do *chatbot*. A avaliação da performance do modelo é um procedimento essencial para verificar se o *chatbot* está compreendendo as intenções do usuário e respondendo de forma precisa e os detalhes sobre essa avaliação podem ser encontrados na seção 7.3.1 deste trabalho, que aborda a saúde do modelo.

As principais tarefas realizadas pelo *chatbot* incluem:

1. Integração com API para obter preços em tempo real: Foi implementada uma funcionalidade que permite ao *chatbot* pesquisar e recuperar o preço atualizado de ativos financeiros por meio de uma chamada à API.
2. Rastreamento personalizado de até 5 ativos: O usuário tem a capacidade de definir até 5 ativos específicos que deseja rastrear. O *chatbot* é capaz de acompanhar esses ativos selecionados e fornecer seus valores.
3. Gerenciamento dos ativos do usuário: O *chatbot* oferece a opção para que o usuário exclua ativos da lista de rastreamento. Isso permite uma maior personalização e controle sobre quais ativos estão sendo monitorados.
4. Armazenamento eficiente em uma coleção exclusiva: Para evitar sobrecarregar a API com requisições frequentes, todos os ativos selecionados pelo usuário são salvos em uma coleção específica na base de dados. Dessa forma, as informações são atualizadas em tempo real e outros usuários podem consultar a base de dados em vez de fazer chamadas repetidas à API.
5. Explicação do funcionamento do *chatbot*: Caso o usuário não esteja familiarizado com o funcionamento do *chatbot*, foi implementada uma opção de pedido de ajuda que fornece explicações detalhadas por meio de um fluxo de resposta dedicado.

6. Atualização periódica de preços e notificação aos usuários: A cada 3 minutos, o *chatbot* atualiza os preços dos ativos na base de dados. Além disso, a cada 5 minutos, verifica se o preço está acima ou abaixo do valor informado pelo usuário, possibilitando o envio de notificações relevantes.
7. Gerenciamento de notificações: Os usuários têm a opção de solicitar a interrupção do envio de notificações pelo *chatbot*, garantindo flexibilidade e controle sobre o recebimento de informações.
8. Pesquisar notícias sobre ativos: O *chatbot* permite que o usuário pesquise informações sobre ativos e, em resposta, pode retornar até 3 resultados que estão em alta, relacionados à pesquisa realizada.

Essas tarefas oferecem aos usuários uma experiência financeira mais eficiente, personalizada e informada, ajudando os usuários a tomar decisões embasadas e acompanhar o mercado de investimentos com maior facilidade.

Para explorar a sequência completa de interações conversacionais, é essencial visitar o seguinte link no repositório GitHub: https://github.com/JefteLG/chatbot_finance/tree/main/conversation_flow. Nesse contexto, será possível obter uma visão abrangente das interações planejadas para o chatbot, proporcionando uma compreensão detalhada e abrangente de seu funcionamento.

7.3. Viabilidade do chatbot

Com o objetivo de finalizar a PoC, é imprescindível realizar uma análise criteriosa da viabilidade do *chatbot*. Nesta etapa, o desempenho da solução é avaliado para identificar falhas e evidenciar melhorias para formalizar todo o processo de implementação da PoC. Essa avaliação se faz necessária para verificar se a construção da solução é factível.

7.3.1. Saúde do modelo

A análise da saúde do modelo *Rasa* é essencial para criar um *chatbot* eficiente. Isso envolve avaliar a precisão das respostas, a coerência nas interações e a capacidade de lidar com casos complexos. Ao identificar lacunas e áreas de melhoria, pode-se realizar ajustes no *chatbot*, melhorando sua compreensão e respostas.

Além disso, a análise ajuda a otimizar o fluxo de diálogo, oferecer uma experiência fluente aos usuários e aprimorar o treinamento contínuo do modelo. Em suma, a análise da saúde do modelo *Rasa* permite corrigir problemas, otimizar o *chatbot* e garantir respostas assertivas, resultando em uma experiência de conversação mais satisfatória.

Para validação, serão utilizadas métricas como *F1-Score*, *Precision* e *Recall*, que serão obtidas após o treinamento e testes dos modelos gerados. Após realizar ajustes nos dados de treinamento, foi obtido um modelo capaz de compreender intenções e gerar respostas. Na Figura 9, são demonstrados os valores de *F1-Score*, *Precision* e *Recall* para o NLU, enquanto na Figura 10, é obtido os mesmos valores para a política de resposta do modelo gerado.

Figura 9 – Tabela com as métricas das intenções

| | Intenção | Precisão | Recall | F1 Score | Número de exemplos |
|---|-----------------------|----------|--------|----------|--------------------|
| ● | gerenciar_notificacao | 100.0% | 100.0% | 100.0% | 17 |
| ● | ajuda | 100.0% | 100.0% | 100.0% | 11 |
| ● | afirmar | 100.0% | 100.0% | 100.0% | 15 |
| ● | valor_ativo | 100.0% | 100.0% | 100.0% | 19 |
| ● | negar | 100.0% | 100.0% | 100.0% | 15 |
| ● | saudar | 100.0% | 100.0% | 100.0% | 18 |
| ● | noticia | 100.0% | 100.0% | 100.0% | 10 |
| ● | rastrear_ativos | 100.0% | 100.0% | 100.0% | 18 |
| ● | menu | 100.0% | 100.0% | 100.0% | 8 |
| ● | micro avg | 100.0% | 100.0% | 100.0% | 131 |

Fonte: Elaboração própria, 2023

Figura 10 – Tabela com as métricas das respostas do *bot*

| | Resposta | Precisão | Recall | F1 Score | Número de ocorrências |
|---|----------------------------|----------|--------|----------|-----------------------|
| ● | action_add_ativo | 100.0% | 100.0% | 100.0% | 2 |
| ● | utter_menu | 100.0% | 100.0% | 100.0% | 5 |
| ● | action_greet_init | 100.0% | 100.0% | 100.0% | 3 |
| ● | action_active_activate | 100.0% | 100.0% | 100.0% | 1 |
| ● | action_news | 100.0% | 100.0% | 100.0% | 1 |
| ● | utter_ajuda_2 | 100.0% | 100.0% | 100.0% | 1 |
| ● | utter_activate_negative | 100.0% | 100.0% | 100.0% | 1 |
| ● | action_reset_slots | 100.0% | 100.0% | 100.0% | 17 |
| ● | action_fim_remover_ativo | 100.0% | 100.0% | 100.0% | 1 |
| ● | utter_ajuda_1 | 100.0% | 100.0% | 100.0% | 1 |
| ● | action_manage_notification | 100.0% | 100.0% | 100.0% | 2 |
| ● | utter_news | 100.0% | 100.0% | 100.0% | 3 |
| ● | action_active | 100.0% | 100.0% | 100.0% | 2 |
| ● | utter_menu_ativos | 100.0% | 100.0% | 100.0% | 11 |
| ● | action_remover_ativo | 100.0% | 100.0% | 100.0% | 4 |
| ● | action_listen | 100.0% | 100.0% | 100.0% | 55 |
| ● | utter_active | 100.0% | 100.0% | 100.0% | 5 |
| ● | utter_active_not_activate | 100.0% | 100.0% | 100.0% | 1 |
| ● | action_fim_add_active | 100.0% | 100.0% | 100.0% | 1 |
| ● | action_activate_affirm | 100.0% | 100.0% | 100.0% | 1 |

Fonte: Elaboração própria, 2023

Ao analisar a Figura 9 e 10, é possível constatar que a saúde do modelo apresenta um ótimo desempenho. As métricas do *F1-Score*, *Precision* e *Recall* demonstram resultados altos e consistentes, o que indica uma boa capacidade de compreensão das intenções e respostas.

Para a elaboração dos dados de teste para criar o teste *end-2-end*, no qual, simula uma conversa entre o usuário e o *chatbot*, foi reservada uma porção dos conjuntos de dados que seriam utilizados no processo de treinamento. A seleção desses dados de teste é fundamental para verificar a generalização do entendimento e o comportamento do modelo diante de situações não previamente observadas.

Esses indicadores são fundamentais para avaliar o desempenho do *chatbot* Rasa. Portanto, com base nessas informações, pode-se afirmar que o modelo está obtendo bons resultados para ser construído um MVP. No entanto, é importante continuar monitorando regularmente o desempenho do *chatbot* para garantir que ele mantenha sua precisão e qualidade ao longo do tempo de desenvolvimento e novos cenários de testes.

É importante destacar que o modelo de avaliação da saúde do *chatbot* abrange diversas métricas, além das mencionadas anteriormente. Outras métricas utilizadas é a matriz de confusão e o histograma que apresenta a distribuição de confiança das previsões de intenções. Essas informações podem ser encontradas no link: https://github.com/JefteLG/chatbot_finance/blob/main/README.md.

7.3.2. Pesquisa de campo

A pesquisa foi feita em algumas etapas no mês de julho e agosto de 2023, no qual, grupos de pessoas interessadas em investimentos financeiros foram convidadas para testar o chatbot. Foi um total de 4 ciclos de testes, com grupos de 3, 5, 8 e 6 pessoas respectivamente para coletar os *insights* necessários para construir um chatbot que atenda às necessidades desse público.

A fim de avaliar a satisfação dos usuários, foi adotada uma escala de 1 a 5, comumente utilizada no CSAT, na qual os participantes classificaram suas experiências de acordo com diferentes níveis de satisfação. A escala incluiu as seguintes categorias: "Muito insatisfeito", "Insatisfeito", "Neutro", "Satisfeito", "Muito satisfeito". Com base nessas respostas, é possível encontrar pontos de melhorias no sistema.

A pesquisa foi conduzida por meio da aplicação de um questionário composto por três campos distintos para ser preenchidos. Campo 1: "Com base na sua experiência com o *chatbot*, por favor, selecione a opção que melhor descreve o seu nível de satisfação em relação à utilização da aplicação: 1 - Muito insatisfeito, 2 - Insatisfeito, 3 - Neutro, 4 - Satisfeito, 5 - Muito satisfeito". Campo 2: "Você utilizaria esta aplicação como uma ferramenta para otimizar suas transações de compra e venda de ativos?". Campo 3: "Caso queira deixar algum comentário adicional, por favor, utilize este espaço".

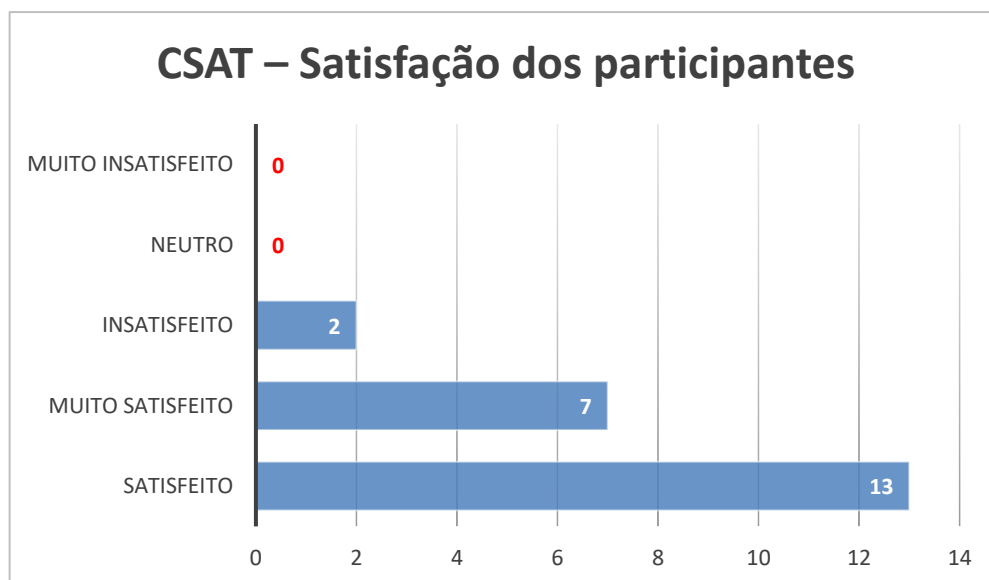
Foram identificados os que indicaram satisfação (notas 4 e 5), os que demonstraram insatisfação (notas 1 e 2) e os neutros (nota 3). No próximo tópico desta pesquisa (7.3.3), estão disponíveis as respostas ao questionário. A Figura 11 ilustra o campo 1, a Figura 12 representa o campo 2, já o campo 3 é apresentado na Tabela 1.

7.3.3. Resultados da pesquisa de campo

Após uma análise da saúde do modelo apresentando na seção 7.3.1, bem como dos resultados obtidos nos testes *end-2-end* e nos testes com interações de pessoas, ficou evidente o desempenho satisfatório do *chatbot* ao fornecer respostas precisas e concisas em relação aos valores dos ativos e às pesquisas de notícias sobre o mercado financeiro.

Foi feita uma pesquisa para avaliar a satisfação dos usuários que interagiram com o *chatbot* usando o CSAT, assim como explicado na sessão 7.3.2. O resultado da satisfação dos usuários é demonstrado na Figura 11.

Figura 11 – CSAT – Satisfação dos participantes



Fonte: Elaboração própria, 2023

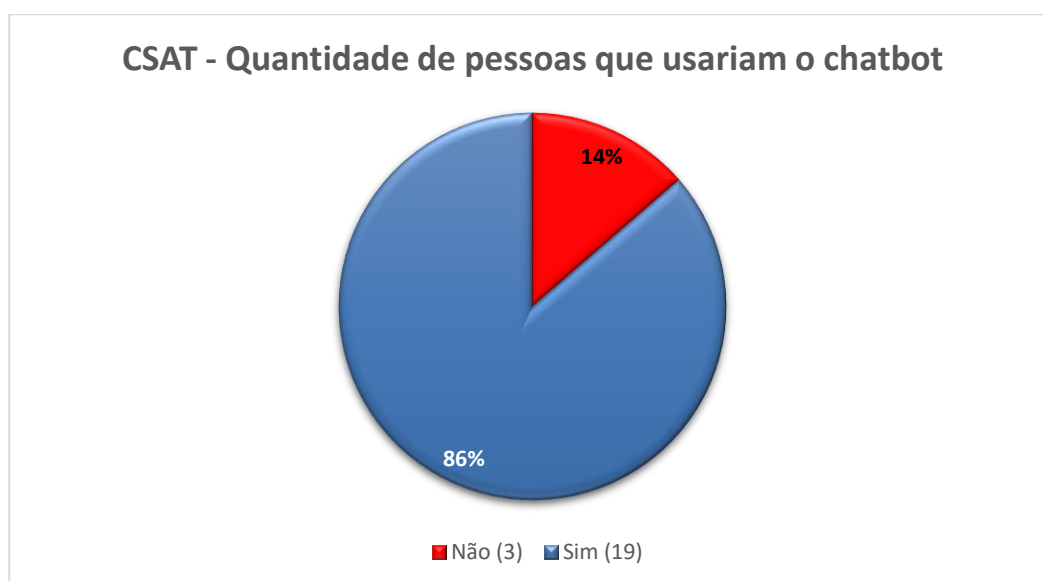
A análise do gráfico na Figura 11 revela que a maioria dos usuários estão satisfeitos com o serviço oferecido. Dos 22 participantes que responderam a pesquisa, nenhum indicou estar "muito insatisfeito" ou "neutro". Apenas duas pessoas afirmaram estar "insatisfeita", enquanto sete responderam estar "muito satisfeitos" e treze declararam estar "satisfeitos".

É importante destacar que as duas pessoas que mencionaram estar insatisfeita relataram que o *chatbot* não apresentava os valores de criptomoedas e não possibilitava a compra e venda de ativos. Esses *feedbacks* oferecem uma oportunidade valiosa para aprimorar o serviço, considerando a crescente importância desse tipo de informação no contexto atual.

Esses resultados demonstram uma tendência positiva de satisfação entre os usuários, refletindo a eficácia e a qualidade do serviço proporcionado pelo *chatbot*. A alta porcentagem de respostas "satisfeitos" e "muito satisfeitos" é um indicativo promissor de que o *chatbot* está atendendo de forma satisfatória às expectativas dos usuários e oferecendo uma experiência positiva de interação.

Além desta pesquisa, também foi aplicado um questionário adicional com o propósito de verificar a intenção do uso por parte dos participantes que testaram o *chatbot* apresentado na Figura 12. Neste questionário, buscou-se compreender se eles considerariam utilizar o *chatbot* em suas interações futuras.

Figura 12 – CSAT - Quantidade de pessoas que usariam o chatbot



Fonte: Elaboração própria, 2023

Na Figura 12, são apresentados os resultados referentes ao interesse dos participantes em utilizar o chatbot. Neste contexto, observou-se que três participantes não demonstraram interesse em utilizar o chatbot devido às limitações existentes relacionadas à compra e venda de ativos e tipos de negociações, que não englobam criptomoedas. No entanto, é notável que dezenove participantes afirmaram que estariam dispostos a utilizar o chatbot para auxiliar em suas transações de ativos, mesmo diante das limitações mencionadas.

É relevante destacar que, entre os três participantes, um deles já estava familiarizado com uma ferramenta de gestão de compra e venda de ativos conhecida como *Smarttbot*. A *Smarttbot* é uma empresa que oferece serviços relacionados à negociação automatizada de ativos financeiros, com ênfase especial no mercado de ações e criptomoedas. Sua plataforma proporciona automação de estratégias de investimento, possibilitando que *traders* e investidores automatizem suas operações, estabeleçam critérios de entrada e saída, e conduzam negociações de forma mais eficaz e eficiente (NEW TRADING, 2023).

Adicionalmente, foi solicitado que os participantes fornecessem quaisquer comentários relacionados a melhorias e sugestões de novas funcionalidades que poderiam ser incorporadas ao chatbot, como evidenciado na Tabela 1.

Tabela 1 – CSAT - Comentários de melhoria

| Comentários de melhoria |
|---|
| Rastrear mais ativos |
| Além de rastrear o preço é interessante comprar ativos também |
| Rastrear criptomoedas |
| Sugestões ou recomendações de compra |
| Interagir por meio de comandos de voz e não apenas chat |
| Adicionar o relatório diário referente ao dia anterior |
| Notificações com oportunidade de investimento alertando os riscos |
| Suporte multilíngue |
| Integrar com uma corretora para possibilitar a compra de ativos |
| simulação de investimento |
| Recomendações personalizadas com base no perfil do investidor |
| Acessibilidade |
| Notificar quando o valor do ativo estiver próximo do preço alvo |
| Lembrete para olhar o valor de ações |
| Mostrar todos os ativos, para saber os que estão cadastrados |
| Possibilitar a compra e venda de moedas - FOREX |

Fonte: Elaboração própria, 2023

Com a inclusão desse questionário apresentado na Tabela 1 foi possível obter *insights* valiosos sobre a aceitação e a percepção dos usuários em relação ao *chatbot*. Com base em suas respostas, é possível compreender melhor como o *chatbot* está sendo percebido por seu público-alvo e identificar oportunidades para aprimoramentos que atendam às expectativas e necessidades dos usuários.

8. Considerações Finais

De acordo com os resultados da avaliação de viabilidade, o *chatbot* demonstrou desempenho satisfatório ao fornecer respostas precisas e concisas às perguntas relacionadas aos valores dos ativos e às pesquisas de notícias sobre o mercado financeiro.

Ao responder de forma assertiva, o *chatbot* demonstrou um alto nível de precisão na obtenção dos dados financeiros e foi capaz de fornecer informações relevantes e atualizadas aos usuários. Essa habilidade é fundamental para garantir a confiança dos usuários no serviço prestado pelo *chatbot*, especialmente em contextos financeiros onde a precisão das informações é crucial para a tomada de decisões.

Além disso, a capacidade do *chatbot* em ser conciso nas respostas também é um aspecto positivo. Respostas diretas e bem formuladas tornam a interação com o *chatbot* mais eficiente e agradável para os usuários, evitando informações desnecessárias e fornecendo apenas os dados essenciais solicitados.

A partir desses resultados positivos, é possível constatar que o *chatbot* está no caminho certo para se tornar uma solução valiosa e comercializável na área de finanças. A precisão e a concisão das respostas aumentam a satisfação dos usuários e aumentam a probabilidade de que o *chatbot* seja adotado no mercado.

Em resumo, os resultados positivos obtidos na avaliação de viabilidade mostram que o *chatbot* está no caminho certo para se tornar uma ferramenta bem-sucedida e confiável para fornecer informações sobre valores de ativos e notícias financeiras. Com o contínuo aprimoramento, é possível impulsionar ainda mais o potencial do *chatbot* e torná-lo uma solução comercialmente viável e valiosa para os usuários.

8.1. Trabalhos Futuros

Para criação de um *chatbot* comercializável, é necessário percorrer um caminho cuidadoso que abrange diversos aspectos críticos para o sucesso do projeto. Além de oferecer uma experiência de interação fluída e eficiente com os usuários, o desenvolvimento de um *chatbot* comercial exige atenção especial a questões legais, éticas, técnicas para sustentação e resiliência.

Em primeiro lugar, a coleta de dados é um ponto crucial a ser abordado. Ao utilizar técnicas como *web scraping* para obter informações de *websites*, é imprescindível garantir que essa prática esteja em total conformidade com as leis de proteção de dados e privacidade vigentes no local em que os dados estão sendo extraídos. Nesse sentido, o desenvolvimento deve ser diligente ao criar políticas de privacidade claras e acessíveis, além de adotar práticas de segurança robustas para proteger os dados coletados e garantir a confiança dos usuários.

Outro aspecto-chave é a obtenção dos dados em si. Em vez de recorrer somente à *web scraping*, é altamente recomendado buscar APIs que forneçam os dados necessários de forma legal e autorizada. APIs são interfaces que permitem a comunicação segura e eficiente com sistemas externos, garantindo acesso a dados atualizados e evitando problemas legais relacionados ao *scraping* não autorizado.

No cenário dos *chatbots* financeiros, especialmente aqueles que oferecem informações sobre valores de ativos, a escolha de APIs confiáveis para coletar dados em alta escala e tempo real é um fator crítico. Isso assegura que o *chatbot* possa fornecer informações precisas e atualizadas aos usuários, conferindo maior credibilidade ao serviço oferecido.

Após alguns estudos, um serviço de referência no mercado para suprir essa necessidade é a API da Cedro, disponível em <https://www.marketdatacloud.com.br/sobre-nos-market-data-cloud/>. A API da Cedro oferece uma solução robusta para a coleta de valores de ativos financeiros em tempo real, com alta disponibilidade e confiabilidade (MARKET DATA CEDRO, 2023).

Ao utilizar a API da Cedro, o *chatbot* pode obter acesso a um amplo conjunto de dados financeiros, incluindo cotações de ações, índices, moedas e outras informações relevantes do mercado. A atualização em tempo real dos dados proporciona aos usuários uma experiência mais enriquecedora e confiável, permitindo que eles tomem decisões informadas sobre seus investimentos e estratégias financeiras.

Além disso, A API da Cedro é projetada para atender às necessidades de *chatbots* financeiros de alta demanda, garantindo eficiência na entrega dos dados e possibilitando escalabilidade conforme o serviço se torna mais popular.

Durante a realização da Prova de Conceito (PoC), uma arquitetura foi desenvolvida com o uso do *Docker*, visando simplificar o ciclo de vida do desenvolvimento. Entretanto, para a implantação em um ambiente real, está previsto o uso do *Amazon Elastic Kubernetes Service* (EKS).

A escolha de utilizar o *Docker* na PoC foi uma decisão estratégica para facilitar o desenvolvimento e testes iniciais, uma vez que o *Docker* permite empacotar a aplicação e suas dependências em um contêiner, garantindo portabilidade e consistência entre ambientes de desenvolvimento e produção.

No entanto, ao avançar para a fase de implantação em ambiente de produção, a adoção do *Amazon EKS* se apresenta como uma solução robusta e escalável. O *Amazon EKS* é um serviço gerenciado pela AWS que facilita a orquestração de contêineres com o *Kubernetes*, permitindo gerenciar, escalar e atualizar os contêineres de forma eficiente e confiável (AWS, 2023).

A utilização do *Amazon EKS* traz diversos benefícios, como a redução do esforço necessário para administrar o *Kubernetes*, além de oferecer alta disponibilidade e escalabilidade automática, proporcionando uma infraestrutura sólida e resiliente para o *chatbot* em um ambiente real de produção.

REFERÊNCIAS

- ALPINE LINUX. 2023. Disponível em: <<https://www.alpinelinux.org/about/>>. Acesso em: 27 jun. 2023.
- Amazon, AWS. 2023. Disponível em: <<https://aws.amazon.com/pt/kubernetes/>>. Acesso em: 21 jul. 2023.
- ANDERSON, Eugene et al. Customer satisfaction, market share, and profitability: Findings from Sweden. *Journal of marketing*, v. 58, n. 3, p. 53-66, 1994.
- ARTIFICIAL. In CAMBRIDGE DICTIONARY. Dicionário inglês. 2022. Disponível em <<https://dictionary.cambridge.org/pt/dicionario/ingles/artificial>>. Acesso em: 09 mai. 2022.
- BARROS, Flavia; TEDESCO, Patricia. Agentes Inteligentes Conversacionais: Conceitos Básicos e Desenvolvimento. In 35º JAI – Jornada de Atualização em informática, pp. 169-218. 2016. Disponível em: <<https://doi.org/10.5753/sbc.6>>. Acesso em: 19 fev. 2022.
- BLIPDOCS. BLIP. 2022. Disponível em: <<https://docs.blip.ai/>>. Acesso em: 02 abr. 2022.
- CALDARINI, Guendalina; JAF, Sardar; MCGARRY, Kenneth. A literature survey of recent advances in chatbots. *Information*, v. 13, n. 1, p. 41, 2022. Disponível em: <<https://www.mdpi.com/2078-2489/13/1/41>>. Acesso em: 24 set. 2023.
- COMARELLA, Rafaela; CAFÉ, Lígia. CHATTERBOT: conceito, características, tipologia e construção. *Informação & Sociedade: Estudos*, [S. l.], v. 18, n. 2, 2008. Disponível em: <<https://periodicos.ufpb.br/ojs/index.php/ies/article/view/1758>>. Acesso em: 15 mai. 2022.
- COPPIN, Ben. Inteligência Artificial. LTC: Grupo GEN, 2010.
- CORREIA, Mafalda. A evolução dos chatbots e os seus benefícios para marcas e consumidores. 2019. Dissertação - (Mestrado em Ciências da Comunicação) – Universidade Católica Portuguesa. Disponível em: <<http://hdl.handle.net/10400.14/27105>>. Acesso em: 03 mai. 2022.
- DALE, Robert. The return of the chatbots. *Natural Language Engineering*, v. 22, n. 5, p. 811-817, Cambridge University. 13 set. 2016. Disponível em: <<https://doi.org/10.1017/S1351324916000243>>. Acesso em: 14 abr. 2022.
- DIALOGFLOW. Google Cloud. 2022. Documentação Dialogflow. Disponível em: <<https://cloud.google.com/dialogflow/docs>>. Acesso em: 30 mar. 2022.
- Docker Docs, DOCKER. 2023. Disponível em: <<https://docs.docker.com/get-started/overview/>>. Acesso em: 14 mai. 2023.
- DRESCH, Aline; LACERDA, Daniel; JÚNIOR, José. Design science research: método de pesquisa para avanço da ciência e tecnologia. Grupo A, 2015. E-book. ISBN 9788582605530. Disponível em: <<https://integrada.minhabiblioteca.com.br/#/books/9788582605530/>>. Acesso em: 04 out. 2023.
- EQUIPE UBOTS. Ubots. Métricas de chatbot – Saiba quais são e como analisar. 13 jan. 2021. Disponível em: <<https://ubots.com.br/blog/metricas-de-chatbot/>>. Acesso em: 29 mar 2022.
- FEITOZA, Felipe. Cecílio: um chatbot para automação do atendimento aos usuários em Instituições Federais de Ensino Superior. 2021. Dissertação (Pós-graduação em Gestão nas Organizações Aprendentes para obtenção de título de Mestre) – Universidade Federal da Paraíba (PPGOA/UFPB). Disponível em: <<https://repositorio.ufpb.br/jspui/handle/123456789/21943>>. Acesso em: 08 mar. 2022.
- FREEBSD. FreeBSD. Chroot. 29 set 2020 Disponível em: <<https://www.freebsd.org/cgi/man.cgi?query=chroot&sektion=2#end>>. Acesso: 28 mai. 2022.

FREIRE, João. Orquestração de Containers Usando Kubernetes e Docker Swarm. 2021. Dissertação (Mestre em Engenharia Informática). Universidade Beira Interior.

GADELHA, Igor. O uso de chatbots no atendimento de clientes de revenda por catálogo. 2019. Dissertação (Mestrado em Computação Aplicada) - Núcleo de Desenvolvimento Amazônico em Engenharia, Universidade Federal do Pará, Tucuruí, 2019. Disponível em: <<http://repositorio.ufpa.br:8080/jspui/handle/2011/12417>>. Acesso em: 30 mar. 2022.

GALITSKY Boris; ILVOVSKY, Dmitry. On a Chatbot Conducting a Virtual Dialogue in Financial Domain. Proceedings of the First Workshop on Financial Technology and Natural Language Processing, p. 99 – 101. Macao, China. 2019. Disponível em: <https://aclanthology.org/W19-5517>. Acesso em: 28 jan. 2023.

GARTNER HYPE CYCLE. Gartner. 2021. Disponível em: <<https://www.gartner.com/en/articles/the-4-trends-that-prevail-on-the-gartner-hype-cycle-for-ai-2021>>. Acesso em: 23 set. 2023.

Gartner. Gartner Magic Quadrant for Enterprise Conversational AI Platforms 24 jan 2022. Disponível em: <<https://www.gartner.com/en/documents/4010683>>. Acesso em: 03 mar. 2022.

GOMES, Jorge et al. Enabling rootless Linux Containers in multi-user environments: the u docker tool. 01 nov 2018. In Computer Physics Communications, v. 232, p. 84-97.

HUTCHESON, Graeme; MOUTINHO, Luiz. Measuring preferred store satisfaction using consumer choice criteria as a mediating factor. Journal of marketing Management, v. 14, n. 7, p. 705-720, 1998.

IBM CLOUD EDUCATION. IBM. 23 jun 2021. Containers. Disponível em: <<https://www.ibm.com/br-pt/cloud/learn/containers>>. Acesso: 28 mai. 2022.

IFTO. Regulamento da organização didático-pedagógica dos cursos de graduação presenciais do IFTO. 2016. Instituto Federal do Tocantins, Palmas – Tocantins, 4 de nov. de 2016. Disponível em: <<http://www.ifto.edu.br/ifto/colegiados/consup/documentos-aprovados/regulamentos/regulamentos-cursos-graduacao/regulamento-da-organizacao-didatico-pedagogica-dos-cursos-de-graduacao-do-ifto.pdf>>. Acesso em: 17 mar. 2022.

INTELIGÊNCIA. In MICHAELIS, Dicionário Brasileiro da Língua Portuguesa, Cia. Artificial. Melhoramentos, 2022. Disponível em: <<http://michaelis.uol.com.br/modernoportugues/busca/portugues-brasileiro/artificial>>. Acesso em: 06 mai. 2022.

JACOMASSI, Guilherme; OLIVEIRA, Edenis. Taxa Selic e Investidores (Pessoa Física) em Renda Variável: Estudo Com Dados da B3. Revista Foco, [S. l.], v. 15, n. 2, p. e352, 2022. Disponível em: <<https://ojs.focopublicacoes.com.br/foco/article/view/352>>. Acesso em: 10 jun. 2023.

JÚNIOR, Sérgio. Um chatbot para responder FAQs. 2018. Artigo de Periódico (Graduação em Sistemas de Informação) - Universidade Federal da Paraíba. Disponível em: <<https://si.dcx.ufpb.br/wp-content/uploads/2018/07/TCC-Sergio-Ricardo-2017.2.pdf>>. Acesso em: 02 mar. 2022.

KAR, Rohan; HALDAR, Rishin. Applying chatbots to the internet of things: Opportunities and architectural elements. arXiv preprint arXiv:1611.03799, 2016.

KAVLAKOGLU, Eda. IBM. 2020. NLP vs. NLU vs. NLG: the differences between three natural language processing concepts. Disponível em: <<https://www.ibm.com/blogs/watson/2020/11/nlp-vs-nlu-vs-nlg-the-differences-between-three-natural-language-processing-concepts/>>. Acesso em: 05 mai 2022.

KERLYL, Alice; HALL, Phil; BULL, Susan. Bringing chatbots into education: Towards natural language negotiation of open learner models. In: International Conference on Innovative Techniques and Applications of Artificial Intelligence. Springer, London, 2006. p. 179-192.

KHAN, W.; Kumar, T.; Zhang, C.; Raj, K.; Roy, A.M.; Luo, B. SQL and NoSQL Database Software Architecture Performance Analysis and Assessments—A Systematic Literature Review. *Big Data Cogn. Comput.* 2023, 7, 97. <https://doi.org/10.3390/bdcc7020097>.

KONG, Xiaoquan; WANG, Guan. *Conversational AI with Rasa: Build, Test, and Deploy AI-powered, Enterprise-grade Virtual Assistants and Chatbots*. Packt Publishing. 2021.

KUBERNETES. Kubernetes. 29 ago 2021. O que é Kubernetes? Disponível em: <https://kubernetes.io/pt-br/docs/concepts/overview/what-is-kubernetes>>. Acesso: 28 abr. 2022.

LACERDA, Arthur. Rasa-ptbr-boilerplate: FLOSS project that enables brazilian portuguese chatbot development by non-experts. 2019. 49 f., il. Monografia (Graduação em Engenharia de Software) - Universidade de Brasília, Brasília. Disponível em: <https://bdm.unb.br/handle/10483/23059>>. Acesso em: 22 mar 2022.

MAGALHÃES, Marcela; CASTRO, Rayssa. Avaliação do uso de Chatbots por parte das empresas como meio de atendimento ao consumidor. *Especialização Em Big Data Estratégico (Web Intelligence & Data Analytics)*. Centro de Referência em Inteligência Empresarial (CRIE) – Universidade Federal do Rio de Janeiro. 2019.

Market data cloud, MARKET DATA CEDRO. Disponível em: <https://www.marketdatacloud.com.br/>>. Acesso em: 26 jun. 2023.

MAROENGSI, Wari et al. A survey on evaluation methods for chatbots. In: *Proceedings of the 2019 7th International Conference on Information and Education Technology (ICIET 2019)*. Association for Computing Machinery, New York, NY, USA. 29 mar 2019. p. 111-119. Disponível em: <https://doi.org/10.1145/3323771.3323824>>. Acesso em: 23 abr. 2022.

MATHUR, Medha et al. Algorithmic Trading Bot. *ITM Web Conf. Volume 40, International Conference on Automation, Computing and Communication 2021*. Disponível em: <https://doi.org/10.1051/itmconf/20214003041>>. Acesso em: 20 jan. 2023.

MCILROY, Doug; KERNIGHAN, Brian. *UNIX™ Time-Sharing System: Unix Programmer's Manual*. eds, Seventh Edition, Vol. I, AT&T Bell Laboratories: Murray Hill, New Jersey, January 1979.

MESHRAM, Siddhant et al. College Enquiry Chatbot using Rasa Framework. 2021. In *Asian Conference on Innovation in Technology (ASIANCON)*, 2021. pp. 1-8. IEEE.

MIKOLOV, Tomas et al. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'13)*. Curran Associates Inc., Red Hook, NY, USA, pp. 3111–3119.

MISSAO, Carlos; BATISTA, Edgard. 2003. Desenvolvimento de uma metodologia de negócios para sistemas de informação relacionados com o gerenciamento da cadeia de suprimento. In *XXIII Encontro Nacional de Engenharia de Produção - ENEGEP - Ouro Preto, MG, Brasil*. 2003.

MORGAN, Neil; REGO, Lopo. The value of different customer satisfaction and loyalty metrics in predicting business performance. In *Marketing science*, v. 25, n. 5, p. 426-439, 2006. Disponível em: <https://doi.org/10.1287/mksc.1050.0180>>. Acesso em: 27 abr. 2022.

NETO, Alexandre. *Finanças Corporativas e Valor*. 7. Ed. São Paulo: Atlas, 2014.

NEW TRADING. 2023. Disponível em: <https://www.mynewtrading.com.br/sobre-smarttbot/>>. Acesso em: 07 out. 2023.

NORVIG, Peter; RUSSEL, Stuart. *Inteligência Artificial*. LTC: Grupo GEN, 2013.

NUBANK. 2023. Disponível em: <https://blog.nubank.com.br/tipos-de-investimentos/>>. Acesso em: 10 jun. 2023.

NUNES, Marcos. MOODLEBOT: um chatbot baseado no IBM Watson para o Moodle. 2021. 67 f. Monografia (Graduação em Sistemas de Informação) - Instituto de Ciências Exatas e Aplicadas, Universidade Federal de Ouro Preto, João Monlevade. Disponível em: <<https://monografias.ufop.br/handle/35400000/3626>>. Acesso em: 26 mar. 2022.

PAIVA, Fernando. Mapa do Ecossistema Brasileiro de Bots. Pesquisa Panorama Mobile Time. fev. 2021. [S.l.]. Disponível em <<https://www.mobiletime.com.br/pesquisas/mapa-do-ecossistema-brasileiro-de-bots-2021/>>. Acesso em: 13 mar. 2022.

PAIVA, Fernando. Mensageria no Brasil. Panorama Mobile Time/Opinion Box. fev. 2022. [S.l.]. Disponível em <<https://www.mobiletime.com.br/pesquisas/mensageria-no-brasil-fevereiro-de-2022/>>. Acesso em: 06 mar. 2022.

PESENTE, Ronaldo. Mercados Financeiros. Universidade Federal da Bahia. Livro Digital. 2019. Disponível em: <<http://educapes.capes.gov.br/handle/capes/553591>>. Acesso em: 25 set. 2023.

QUEIROZ, Rodrigo de Campos. O dilema do pequeno investidor. [Dissertação de mestrado]. Escola de Administração de Empresas de São Paulo da Fundação Getúlio Vargas, São Paulo, 2021. Disponível em: <https://hdl.handle.net/10438/30740>. Acesso em: 30 jan. 2023.

RASA FRAMEWORK. Rasa X. 2022. Disponível em: <<https://rasa.com/docs/rasa-x/>>. Acesso em: 08 mar. 2022.

RASA FRAMEWORK. Rasa. 2022. Disponível em: <<https://rasa.com/docs/rasa/>>. Acesso em: 05 mar. 2022.

RASA. Rasa. 2022. Conversation-Driven Development. Disponível em: <<https://rasa.com/docs/rasa/conversation-driven-development/>>. Acesso em: 02 mai. 2022.

REDHAT. RedHat. 2 dez 2019. O que é orquestração de containers? Disponível em: <<https://www.redhat.com/pt-br/topics/containers/what-is-container-orchestration>>. Acesso em: 28 mai. 2022.

REDHAT. RedHat. 8 abr 2021. O que é containerização? Disponível em: <<https://www.redhat.com/pt-br/topics/cloud-native-apps/what-is-containerization>>. Acesso em: 28 mai. 2022.

REINA, Darío; CRUZ, Clarice. Curadoria de chatbots: conceptualização, estratégias e indicadores de desempenho. RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação, n. 40, p. 1-14, 2020.

RODRIGUES, Guilherme. Criação de um Ambiente utilizando Kubernetes como Orquestrador de Contêineres. 24f. 2019. Artigo – (Graduação em Engenharia de Software) UNICESUMAR – Centro Universitário de Maringá. 2 dez 2019. Disponível em: <<http://rdu.unicesumar.edu.br/handle/123456789/5357>>. Acesso em: 12 mar 2022.

ROSENBERG, Nathan. Por dentro da caixa preta: tecnologia e economia. Inside the black box: technology and economics. Tradutor (José Emílio Maiorino). Ed. Unicamp, 2006, p. 185-213. 85-268-0742-. Coleção Clássicos da Inovação.

ROSENBERG, Seth. Facebook. Como construir bots para o Messenger. 12 abr. 2016. Disponível em: <<https://developers.facebook.com/blog/post/2016/04/12/bots-for-messenger/>>. Acesso em: 30 abr. 2022.

SANTOS, João. Real Estate Market Data Scraping and Analysis for Financial Investments. [Dissertação de mestrado]. Faculdade de Engenharia da Universidade do Porto. Porto, 2018. Disponível em: <https://hdl.handle.net/10216/116510>. Acesso em: 15 jan. 2023.

SCHMITT, Jonathan; DAMASCO, Kleber Leonardo. Desenvolvimento de um serviço utilizando as modalidades IAAS e PAAS para cloud computing: um estudo de caso. 2012. Disponível em: <https://repositorio.animaeducacao.com.br/handle/ANIMA/11025>. Acesso em: 20 jun. 2023.

SCHUNK, Leandro. O Uso de Inteligência Artificial por Meio de Chatbots no Processo de Atendimento ao Cliente: Um Estudo Sobre Seus Benefícios. 2020. Dissertação (Mestre em Administração de Empresas) - Fundação Getúlio Vargas, Escola de Administração de Empresas de São Paulo. Disponível em: <<https://hdl.handle.net/10438/29943>>. Acesso em: 06 mai. 2022.

SCOLESE, Daniel et al. Análise de Estilo de Fundos Imobiliários no Brasil. Revista de Contabilidade e Organizações. Journal of Accounting and Organizations. USP - Universidade de São Paulo. Vol. 9, nº 23, jan./abr. 2015.

SHI, Yuwei; HERNIMAN, John. The role of expectation in innovation evolution: Exploring hype cycles. Technovation, v. 119, p. 102459, 2023.

SMIRNOVA, Polina. Customer satisfaction as a tool for service development: Case: Fish & Fun Holiday Village. LAB University of Applied Sciences 2020. Disponível em: <<https://www.theseus.fi/handle/10024/354606>>. Acesso em: 26 set. 2023.

STEINERT, Martin; LEIFER, Larry. Scrutinizing Gartner's hype cycle approach. In: Picmet 2010 technology management for global economic growth. IEEE, 2010. p. 1-13.

TURING, Alan. Computing Machinery And Intelligence. In Mind, Volume LIX, Issue 236, 01 out. 1950, p. 433–460. Disponível em: <<https://doi.org/10.1093/mind/LIX.236.433>>. Acesso em: 14 abr. 2022.

TURING. Python FastAPI vs Flask: uma comparação detalhada. Disponível em: <<https://www.turing.com/kb/fastapi-vs-flask-a-detailed-comparison#asynchronous-tasks>>. Acesso em: 21 jun. 2023.

VASCONCELOS, Jose; NEGRES, Paulo. Renda Variável: Contratos Futuros Como Forma de Alavancagem Financeira a Partir do Mini-Índice Ibovespa. Revista Novos Desafios, v. 2, n. 1, p. 33-49, 2022.

VOGEL, Jorin. Chatbots: Development and Applications. Tese (Graduação em International Media and Computing) - HTW Berlin, University of Applied Sciences. Berlin, 2017.

WAZLAWICK, Raul. Metodologia de Pesquisa para Ciência da Computação. 2. ed. Rio de Janeiro: Elsevier, 2008. p. 159.

YUSUF, Ahamed; KUMAR. Lineesh; KRISHNAN. Navaneetha. RPA Bot Working With Stock Market Share Prices | Stock Analysis & Stock Price Prediction. International Journal of Emerging Technologies and Innovative Research. Volume 9, Issue 3. Coimbatore, Tamil Nadu, India. IJ Publication. 2022. Disponível em: <https://www.jetir.org/view?paper=JETIR2203414>. Acesso em: 15 jan. 2023.

ŚPIEWANOWSKI, Piotr; TALAVERA, Oleksandr; VI, Linh. Applications of Web Scraping in Economics and Finance. Oxford Research Encyclopedia of Economics and Finance. 2022. Disponível em: <https://doi.org/10.1093/acrefore/9780190625979.013.652>. Acesso em: 20 jan. 2023.

APÊNDICE A – DOCKER-COMPOSE

```
version: "3"

## Todos os serviços necessários para utilizar o chatbot

services:
  rasa:
    container_name: rasa
    build: ./rasa
    ports:
      - 5005:5005
    env_file: ./rasa/.env
    environment:
      - ENV=local
    depends_on:
      - actions
    volumes:
      - ./rasa:/app/

  actions:
    container_name: actions
    build:
      context: ./actions
    ports:
      - 5055:5055
    env_file: ./actions/.env
    environment:
      - ENV=local
    volumes:
      - ./actions:/app/actions/

  web_scraping:
    container_name: web_scraping
    build: ./web_scraping
    ports:
      - 80:80
    env_file: ./web_scraping/.env
    environment:
      - ENV=local
    depends_on:
      - mongodb
    volumes:
      - ./web_scraping:/code/
```

```
mongodb:
  container_name: mongodb
  image: mongo:6.0
  ports:
    - 27017:27017
  volumes:
    - ./mongo_backup:/data/db
  environment:
    - MONGO_INITDB_ROOT_USERNAME=user
    - MONGO_INITDB_ROOT_PASSWORD=tst123
    - MONGO_INITDB_DATABASE=bot_invest

cron:
  container_name: cron
  build: ./cron
  depends_on:
    - rasa
  env_file: ./cron/.env
  environment:
    - ENV=local
  volumes:
    - ./cron:/app/

networks:
  bot_invest:
    driver: bridge
```

APÊNDICE B – Conector Customizado do Telegram

```
import asyncio
import logging
from typing import Optional, Text, Any, Dict

from aiogram import Bot
from rasa.core.channels import TelegramInput, InputChannel
from rasa.core.channels.telegram import TelegramOutput
from aiogram.utils.exceptions import TelegramAPIError
from rasa.shared.exceptions import RasaException

logger = logging.getLogger(__name__)

class CustomTelegramOutput(TelegramOutput):

    def __init__(self, access_token: Optional[Text], parse_mode:
Optional[Text]) -> None:
        Bot.__init__(self, access_token, parse_mode=parse_mode)

class CustomTelegramInput(TelegramInput):

    @classmethod
    def from_credentials(cls, credentials: Optional[Dict[Text, Any]]) ->
InputChannel:
        if not credentials:
            cls.raise_missing_credentials_exception()

        return cls(
            credentials.get("access_token"),
            credentials.get("verify"),
            credentials.get("webhook_url"),
            credentials.get("parse_mode"),
        )

    def __init__(
        self,
        access_token: Optional[Text],
        verify: Optional[Text],
        webhook_url: Optional[Text],
        parse_mode: Optional[Text],
        debug_mode: bool = True,
    ) -> None:
        super().__init__(access_token, verify, webhook_url, debug_mode)
        self.parse_mode = parse_mode
```

```
def get_output_channel(self) -> CustomTelegramOutput:
    """Loads the telegram channel."""
    channel = CustomTelegramOutput(self.access_token, self.parse_mode)

    try:
        asyncio.run(channel.set_webhook(url=self.webhook_url))
    except TelegramAPIError as error:
        raise RasaException(
            "Failed to set channel webhook: " + str(error)
        ) from error

    return channel
```

APÊNDICE C – CREDENTIALS.YML

```
channel.custom_telegram.CustomTelegramInput:  
  access_token: ${TOKEN_TELEGRAM}  
  verify: invest_bot  
  webhook_url: ${NGROK_URL}/webhooks/telegram/webhook  
  parse_mode: Markdown
```


APÊNDICE D – SCHEMA DA API

```
schema_example_Users = {
  "id_user": {
    "notificacao_rastreio": bool,
    "notificacao_relatorio": bool,
    "rastreio": {
      "IRDM11" : {
        "valor" : float,
        "status" : string
      },
      "PETR4" : {
        "valor" : float,
        "status" : string
      }
    }
  }
}

schema_example_Actives = {
  "actives": {
    "IRDM11" : float,
    "PETR4" : float,
  }
}
```