

Bruna Heloisa Feitosa Veloso
Orientador: Juliana Oliveira de Carvalho

Plataforma Inteligente para Auxílio na Escolha de Tecnologias para Desenvolvimento de Chatbots

Picos - PI
4 de março de 2016

Bruna Heloisa Feitosa Veloso
Orientador: Juliana Oliveira de Carvalho

Plataforma Inteligente para Auxílio na Escolha de Tecnologias para Desenvolvimento de Chatbots

Trabalho de conclusão de curso de graduação
apresentado ao Departamento de Sistemas de
Informação da Universidade Federal do Piauí
como parte dos requisitos para obtenção do
grau de Bacharel em Sistema de Informação

Universidade Federal do Piauí
Campus Senador Helvídio Nunes de Barros
Bacharelado em Sistemas de Informação

Picos - PI
4 de março de 2016

FICHA CATALOGRÁFICA
Serviço de Processamento Técnico da Universidade Federal do Piauí
Biblioteca José Albano de Macêdo

V443p Veloso, Bruna Heloisa Feitosa.

Plataforma inteligente para auxílio na escolha de tecnologias para desenvolvimento de chatbots / Bruna Heloisa Feitosa Veloso – 2025.
50 f.

1 Arquivo em PDF.

Indexado no catálogo *online* da biblioteca José Albano de Macêdo-CSHNB
Aberto a pesquisadores, com restrições da Biblioteca.

Trabalho de Conclusão de Curso (Graduação) – Universidade Federal do Piauí, Curso de Bacharelado em Sistemas de Informação, Picos, 2025.
“Orientador: Juliana Oliveira de Carvalho”.

1. Assistente virtual – chatbots. 2. Tecnologia – tomada de decisão. 3. Sistemas informacionais. I. Veloso, Bruna Heloisa Feitosa. II. Carvalho, Juliana Oliveira de. III. Título.

CDD 005.7

Elaborada por Maria Letícia Cristina Alcântara Gomes
Bibliotecária CRB nº 03/1835

**PLATAFORMA INTELIGENTE PARA AUXÍLIO NA ESCOLHA DE TECNOLOGIAS PARA
DESENVOLVIMENTO DE CHATBOTS**

BRUNA HELOÍSA FEITOSA VELOSO

Monografia aprovada como exigência parcial para obtenção do grau de Bacharel em Sistemas
de Informação.

Data de Aprovação

Picos – PI, 27 de junho de 2025

Documento assinado digitalmente
 **JULIANA OLIVEIRA DE CARVALHO**
Data: 01/07/2025 19:07:20-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dra. Juliana Oliveira de Carvalho

Documento assinado digitalmente
 **ALCILENE DALÍLIA DE SOUSA**
Data: 01/07/2025 15:06:58-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dra. Alcilene Dalília de Sousa

Documento assinado digitalmente
 **FRANK CESAR LOPES VERAS**
Data: 01/07/2025 18:58:00-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Frank César Vêras Lópes

Agradecimentos

Agradeço primeiramente a Deus, por ter sido meu alicerce em cada momento dessa caminhada. Foi Ele quem me sustentou quando pensei em desistir, quem me deu força quando tudo parecia difícil demais, e quem me guiou mesmo quando o caminho era incerto.

Às minhas avós, Isabel Dantas (in memoriam) e Raimunda da Cruz (in memoriam), uma professora e uma merendeira, que mesmo em caminhos diferentes na educação sempre caminharam lado a lado no amor, no cuidado e, acima de tudo, na fé de que eu podia chegar até aqui. Foram suas palavras e gestos que plantaram as sementes do meu esforço, foram meu abrigo, meu impulso e meu alicerce. Duas mulheres que me mostraram que o conhecimento é um trono que ninguém pode tomar, e que vale a pena persistir, mesmo quando o caminho é difícil. Vocês foram minha primeira sala de aula, meu primeiro lar e o amor mais constante da minha vida. Este trabalho carrega em si a força de suas histórias, o reflexo da esperança que sempre depositaram em mim.

Ao meu irmão, Alyson Junior, que desde pequeno me olhava com admiração. Quando me escolheu para entrar com você na sua festinha do ABC, eu entendi que ser sua irmã era um dos maiores papéis da minha vida. Sempre fui a “irmã inteligente”, aquela em quem você confiava... mas foi você quem me deu forças para seguir. Cada escolha que fiz, cada madrugada que enfrentei, cada lágrima que derramei, foi também por você. Sonhei com um futuro melhor pra nós dois, e é por você que continuo.

À Valderlândia Macêdo, Ianny Karine e Karen Ingrid, que estiveram comigo por tantos anos, em tantos momentos, com tanto afeto. O tipo de amizade que atravessa fases e permanece firme, mesmo quando tudo muda.

À Nara Andrade, que caminhou comigo nesse trecho final, tornando os dias mais leves e os desafios mais possíveis. Sua parceria foi respiro e lembrança de que partilhar o fardo o torna mais fácil de carregar.

À minha orientadora, Juliana, por ter sido farol nesse percurso. Sua orientação sensível e ao mesmo tempo firme me ajudou a enxergar possibilidades quando as ideias ainda eram rascunhos. Obrigada por acreditar no meu projeto, pelas sugestões cuidadosas e por me mostrar que caminhar com responsabilidade e criatividade é o que transforma uma ideia em algo real.

Também aos meus gatos, companheiros silenciosos das noites de estudo e fontes constantes de conforto e alegria, não seria nada sem vocês.

Por fim, agradeço a mim mesma por ter seguido até aqui, com coragem, dedicação e resiliência. Este trabalho representa não apenas um resultado acadêmico, mas a soma de muitos esforços e o reflexo das pessoas que me acompanharam ao longo desse percurso.

Alan Turing

"I propose to consider the question. Can machines think?"

Resumo

O uso de chatbots tem crescido exponencialmente nos últimos anos, devido à essa popularização muitas empresas e instituições de diversos setores, como atendimento ao cliente, educação, saúde e comércio eletrônico, tem optado por integrarem plataformas de agentes conversacionais ao seu sistema. Esse crescimento está ligado à evolução das tecnologias usadas para processamento da linguagem natural e inteligência artificial que tornam esses agentes cada vez mais eficientes e acessíveis. Apesar do aumento no uso dessas soluções, o processo de desenvolvimentos dessas ferramentas ainda é um grande desafio para muitas pessoas e empresas principalmente aquelas que não possuem conhecimento aprofundado na área ou formação técnica. Uma das principais dificuldades encontradas por esses usuários é a escolha da tecnologia mais adequada para o desenvolvimento, ou seja, escolher o que mais funciona para as suas necessidades, uma vez que existem dezenas de plataformas disponíveis cada uma com requisitos e características distintas.

Diante desse cenário, este trabalho propõe o desenvolvimento da plataforma Neko, uma ferramenta inteligente que busca centralizar informações voltada para o auxílio na escolha da tecnologia para criação de chatbots. A proposta é oferecer uma interface acessível e intuitiva, onde o usuário poderá informar características sobre o seu projeto, como nível de conhecimento técnico, objetivo da aplicação deseja, orçamento estabelecido e outras necessidades. A partir desses dados a plataforma realiza uma análise e recomenda a tecnologia mais adequada com base naquele contexto específico. A Neko busca atender principalmente usuários iniciantes ou advindos de outras áreas não técnicas, democratizando o acesso ao desenvolvimento de chatbots, facilitando esse processo e promovendo a tomada de decisões mais assertiva no início dos projetos que garante um maior sucesso nesses projetos baseados em agentes conversacionais.

Palavras-chave: Chatbots, Tecnologias, Assistente virtual, Recomendação, Tomada de decisão.

Abstract

The use of chatbots has grown exponentially in recent years. As a result, companies and institutions from various sectors such as customer service, education, healthcare, and e-commerce have increasingly adopted conversational agents in their systems. This growth is driven by the evolution of natural language processing and artificial intelligence technologies, which have made these tools more efficient and accessible. However, despite their rising popularity, the development of chatbot solutions remains a significant challenge, especially for individuals and organizations without technical training or expertise. One of the main difficulties lies in choosing the most suitable technology among the numerous available platforms, each with its own requirements and features.

In response to this scenario, this work proposes the development of Neko, an intelligent platform designed to assist in selecting the appropriate technology for chatbot development. The platform offers an accessible and user friendly interface, allowing users to input information about their project such as technical knowledge level, application goals, budget, and specific needs. Based on this data, Neko analyzes the context and recommends the most suitable technology. The platform is aimed primarily at beginners or individuals from non-technical backgrounds, democratizing access to chatbot development, simplifying the process, and promoting more assertive decision-making at the start of projects, thereby increasing the chances of success in conversational agent implementations.

Lista de ilustrações

Figura 1 – Histórico de Evolução dos Chatbots	18
Figura 2 – Metodologia do Desenvolvimento	30
Figura 3 – Tela de Início da plataforma Neko	37
Figura 4 – Tela de primeira interação da plataforma Neko	38
Figura 5 – Tela de revisão das informações da plataforma Neko	39
Figura 6 – Tela final da plataforma Neko	40
Figura 7 – Distribuição de gênero dos participantes	43
Figura 8 – Nível de familiaridade dos usuários com desenvolvimento de chatbots	44
Figura 9 – Avaliação média dos atributos da plataforma	45

Lista de tabelas

Tabela 1 – Comparativo entre plataformas para criação de <i>chatbots</i>	22
Tabela 2 – Comparação entre trabalhos relacionados sobre desenvolvimento de <i>chatbots</i>	27

Lista de abreviaturas e siglas

API	Application Programming Interface (Interface de Programação de Aplicações)
AI	Artificial Intelligence (Inteligência Artificial)
AWS	Amazon Web Services
BERT	Bidirectional Encoder Representations from Transformers
DB	Database (Banco de Dados)
GPT	Generative Pre-trained Transformer
GUI	Graphical User Interface (Interface Gráfica do Usuário)
HTTP	Hypertext Transfer Protocol
IA	Inteligência Artificial
JSON	JavaScript Object Notation
LUIS	Language Understanding Intelligent Service
NLP/PLN	Natural Language Processing / Processamento de Linguagem Natural
No-code	Desenvolvimento sem código
PLN	Processamento de Linguagem Natural
SQL	Structured Query Language (Linguagem de Consulta Estruturada)
TTS	Text-to-Speech (Texto para Fala)
UI	User Interface (Interface do Usuário)
URL	Uniform Resource Locator
UX	User Experience (Experiência do Usuário)

Sumário

1	Introdução	13
1.1	Objetivos Gerais e Específicos	14
1.2	Justificativa	14
2	Referencial Teórico	17
2.1	A evolução e impacto dos <i>chatbots</i> na tecnologia	17
2.2	Tecnologias aplicadas no desenvolvimento de <i>chatbots</i>	19
2.3	Critérios para Seleção de Plataformas de Desenvolvimento de <i>Chatbots</i>	22
2.3.1	Objetivo do Projeto	23
2.3.2	Facilidade de Uso	23
2.3.3	Custo	23
2.3.4	Suporte à Inteligência Artificial	24
2.3.5	Integrações Disponíveis	24
2.3.6	Comunidade e Documentação	24
2.3.7	Escalabilidade	25
2.3.8	Segurança e Privacidade	25
3	Trabalhos Relacionados	26
4	Metodologia	30
4.1	Planejamento e Levantamento de Requisitos	30
4.1.1	Requisitos Funcionais	31
4.1.2	Requisitos Não Funcionais	32
4.2	Tecnologias e Ferramentas Utilizadas	32
4.2.1	Front-end	33
4.2.2	Back-end	33
4.2.3	Banco de Dados	33
4.2.4	Processamento de Linguagem Natural (PLN)	33
4.2.5	Ferramentas de Desenvolvimento e Gerenciamento	34
4.3	Desenvolvimento do Módulo de Recomendação (back-end)	34
4.4	Desenvolvimento da Interface	36
4.4.1	Interface e Experiência do Usuário	36
4.5	Iterações e Melhorias	40
4.6	Testes e Validação	41
5	Resultados	43

5.0.1	Discussão Sobre os Resultados e Trabalhos Futuros	46
5.1	Conclusão	47
Referências	48

1 Introdução

Nas últimas décadas, vivenciamos o surgimento de uma tecnologia revolucionária que transformou a maneira como nos comunicamos, possibilitando a comunicação instantânea por meio de *e-mails*, mensagens, *videochamadas* e redes sociais. Por meio da *internet*, é possível nos conectarmos com pessoas em todo o mundo, independentemente da distância geográfica. Mas, além da comunicação, essa tecnologia também modificou o nosso acesso à informação, oferecendo uma maior quantidade de dados em um tempo quase mínimo (Juliana Theodoro, 2025).

Com o avanço das Tecnologias da Informação e Comunicação (TICs), observa-se também a transformação da forma como as pessoas interagem com sistemas computacionais. Nesse contexto, programas que simulam interações humanas por meio da linguagem natural, chamados de *chatbots*, vêm ganhando destaque como soluções para o atendimento automatizado, em experiências personalizadas para os usuários e na otimização de processos organizacionais.

Os *chatbots* são apontados como o futuro da comunicação moderna, pois vêm transformando a forma como empresas e organizações se comunicam com seus usuários. Essa ferramenta está entre as soluções mais utilizadas e vem se expandindo cada vez mais nos últimos anos. Isso ocorre porque ela torna ainda mais fácil o atendimento ao cliente, a automação de processos e a assistência técnica ao usuário. Além disso, os *chatbots* são máquinas capazes de aprender e processar grandes quantidades de dados, tornando-se uma fonte valiosa de informações de maneira simples e rápida, que pode alimentar relatórios e auxiliar na tomada de decisões pelas empresas, agregando ainda mais à base de conhecimento (Diógenes, 2025).

Com o avanço da tecnologia, a chegada da Inteligência Artificial (IA) e o surgimento do *processamento de linguagem natural* (*Natural Language Processing* – NLP), essas ferramentas têm-se tornado ainda mais completas, proporcionando interações cada vez mais naturais e personalizadas.

Dessa forma, a crescente busca por soluções conversacionais, com o objetivo de agregar valor a projetos e empresas, impulsionou o surgimento de diversas tecnologias e *frameworks* voltados ao desenvolvimento de *chatbots*, como o *Dialogflow*, *Rasa*, *Botpress*, *Amazon Lex*, entre outros. No entanto, essa variedade de ferramentas disponíveis, com suas abordagens técnicas e requisitos diversos, tem provocado um cenário desafiador, principalmente para usuários iniciantes ou com pouca familiaridade com técnicas de desenvolvimento de *software*.

Esse contexto evidencia um problema recorrente nas fases iniciais dos projetos de *chatbots*, que é a dificuldade em escolher a tecnologia mais apropriada, considerando critérios como facilidade de uso, curva de aprendizado, integração com outras plataformas

e custo.

Empresas e desenvolvedores enfrentam a ausência de comparações claras e a variedade de critérios que devem ser analisados durante esse processo, o que dificulta a escolha da ferramenta mais adequada para seus projetos, conforme as necessidades específicas da solução idealizada. Essa escolha torna-se um ponto-chave para o desenvolvimento: ao optar por uma tecnologia inadequada, podem surgir diversos problemas, como maior tempo de entrega, custos elevados, limitações na escalabilidade do *chatbot* e, em alguns casos, até mesmo a inviabilidade do projeto.

Dessa maneira, o presente estudo propõe o desenvolvimento da ferramenta *Neko*, uma plataforma concebida para auxiliar os usuários no processo de tomada de decisão quanto à escolha de tecnologias para a criação de seus *chatbots*. A ferramenta funciona a partir das necessidades de cada usuário, coletadas por meio de uma sequência de perguntas e avaliadas conforme alguns critérios, entre eles: facilidade de uso, custo, integrações e suporte à inteligência artificial. A partir disso, é realizada uma análise comparativa entre diferentes tecnologias, para que o sistema recomende a mais adequada ao perfil e às necessidades apresentadas pelo usuário.

1.1 Objetivos Gerais e Específicos

O objetivo geral da pesquisa fundamenta-se no desenvolvimento de uma plataforma para auxiliar usuários na escolha da melhor tecnologia para o desenvolvimento de *chatbots* próprios. Por meio dessa ferramenta, deverá ser possível interagir e buscar auxílio para a tomada de decisões de maneira facilitada. Esse trabalho tem como objetivos específicos:

1. Realizar um levantamento das principais tecnologias e *frameworks* utilizados no desenvolvimento de *chatbots*, identificando suas funcionalidades, linguagens de programação, pontos fortes, limitações, nível de suporte e possibilidades de integração;
2. Estabelecer um conjunto de critérios de comparação entre as tecnologias mapeadas, considerando aspectos como facilidade de uso, curva de aprendizado, custo, escalabilidade, integração com outras ferramentas e suporte à inteligência artificial;
3. Projetar e implementar a plataforma *Neko*, um assistente virtual funcional, capaz de coletar informações dos usuários, aplicar os critérios definidos e recomendar, de forma automatizada e personalizada, a tecnologia mais adequada para cada cenário apresentado.

1.2 Justificativa

O mercado de *chatbots* tem apresentado um crescimento exponencial nos últimos anos, impulsionado pela popularização de agentes conversacionais e pela crescente demanda por soluções automatizáveis em diversas áreas. Estima-se que o valor global de mercado tenha alcançado aproximadamente US\$ 6,3 bilhões em 2023, comparado ao ano de 2022,

que registrou US\$ 5,1 bilhões. Além disso, projeções indicam que esse número poderá chegar a 27,3 bilhões de dólares até 2030 (CHUDLEIGH, 2024). Esse salto evidencia, além do avanço comercial, o contínuo investimento em inovação tecnológica para aplicações baseadas em inteligência artificial e linguagem natural.

Esse avanço no mercado também foi acompanhado pelo surgimento de dezenas de plataformas e *frameworks* voltados ao desenvolvimento e construção de *chatbots*, como o *Dialogflow*, da *Google Cloud*, que permite o desenvolvimento de plataformas e agentes conversacionais utilizando recursos avançados de Processamento de Linguagem Natural (PLN) para *chatbots* ou aplicações de voz; o *Rasa*, que é um *framework* de código aberto que permite criar *chatbots* e assistentes de IA com maior controle e personalização; o *Botpress*, plataforma que descomplica a criação de agentes inteligentes com uma interface voltada para o navegador e apresenta um método de construção por arrastar e soltar, tornando-o acessível para programadores e não programadores; o *IBM Watson*, que utiliza Processamento de Linguagem Natural e aprendizado de máquina para analisar grandes volumes de dados com aplicações em diversas áreas, desde saúde até finanças; e o *Amazon Lex*, que oferece o mesmo serviço de Inteligência Artificial (IA) da Alexa, da AWS, permitindo a criação de interfaces para conversação em aplicações, podendo ser usado com voz e texto. Embora essa variedade de soluções traga mais possibilidades ao desenvolvimento, ela também impõe desafios no momento da seleção da ferramenta mais adequada para cada projeto.

Na prática, observa-se que muitos projetos de *chatbots* são iniciados sem uma análise aprofundada da plataforma que será utilizada, o que pode comprometer o sistema e até mesmo inviabilizar o projeto. A seleção de uma plataforma inadequada pode gerar custos elevados, dificuldades durante a integração do projeto, limitações de manutenção e incompatibilidade com os objetivos definidos. Esse cenário evidencia a lacuna causada pela ausência de ferramentas que centralizem essas informações e que possam auxiliar os usuários na etapa inicial de seleção tecnológica, com base em critérios personalizados, objetivos do projeto e perfil do usuário.

Diante dessa problemática, torna-se evidente, em diferentes contextos, a necessidade de uma solução que auxilie desenvolvedores e empresas na escolha da tecnologia mais adequada para o desenvolvimento de *chatbots*. No âmbito acadêmico, a proposta se insere nas áreas de engenharia de *software*, computação aplicada e sistemas inteligentes, buscando contribuir com uma abordagem inovadora para o problema de seleção tecnológica, ainda pouco explorado em trabalhos relacionados ao desenvolvimento de *chatbots*. No âmbito tecnológico, a plataforma se destaca como um avanço na democratização do desenvolvimento, ao ter como objetivo fornecer suporte a usuários não especializados, incentivando a adoção consciente e eficiente dessas tecnologias em diversas áreas. Em caráter social, *Neko* pode fomentar a disseminação de informações de diversas áreas de maneira prática, promovendo o empoderamento de profissionais da educação, saúde e empreendedorismo

que desejam utilizar *chatbots* em suas práticas, mas enfrentam barreiras técnicas para iniciá-los.

Diante do exposto, este trabalho destaca-se por oferecer uma ferramenta prática, inovadora e com impacto relevante, tanto para o desenvolvimento de *chatbots* quanto para o aumento do acesso a essas tecnologias. A proposta do *Neko* vai além de ser um instrumento de recomendação: busca também contribuir para o aperfeiçoamento das interações entre empresas e seus clientes, promovendo inclusão digital e maior autonomia do usuário, além de permitir que este racionalize melhor todo o processo de desenvolvimento, colaborando com a construção de sistemas mais eficientes, acessíveis e adequados às demandas reais de seus criadores.

2 Referencial Teórico

Esta seção compreende os conteúdos fundamentais para haver um melhor entendimento deste estudo. Sendo dividida da seguinte maneira: 3.1 A evolução e impacto dos *chatbots* na tecnologia, 3.2 Tecnologias aplicadas no desenvolvimento de *chatbots*, 3.3 Critérios para Seleção de Plataformas de Desenvolvimento de *Chatbots*.

2.1 A evolução e impacto dos *chatbots* na tecnologia

Os *chatbots* são programas de computador projetados para simular conversas com usuários por meio de interfaces de texto ou voz. Também conhecidos como agentes conversacionais, têm como principal objetivo permitir que os usuários se comuniquem com sistemas computacionais como se estivessem interagindo com outro ser humano, de forma intuitiva, com interações mais naturais, rápidas e eficazes (JURAFSKY; MARTIN, 2020b). Essas plataformas têm se tornado uma das principais inovações tecnológicas dos últimos anos e vêm ganhando crescente relevância no cenário tecnológico, revolucionando a interação entre usuários e sistemas computacionais.

O termo *chatbot* surgiu apenas em 1990, mas sua origem não é recente. A concepção de *chatbots* está ligada ao desenvolvimento da Inteligência Artificial (IA) e à busca por sistemas computacionais capazes de simular o comportamento humano. Essa ideia teve origem com Alan Turing, matemático britânico considerado o pai da computação moderna, que, em seu artigo *Computing Machinery and Intelligence* (1950), propôs o modelo do Teste de Turing, que consistia em avaliar se uma máquina conseguiria se passar por um humano. Foi com essa proposta que nasceram os fundamentos para os sistemas conversacionais desenvolvidos nas décadas seguintes (TURING, 1950).

Os *chatbots* apareceram na década de 1960, com o desenvolvimento do *ELIZA*, um dos primeiros programas a simular uma conversa humana. O *ELIZA*, criado por Joseph Weizenbaum no MIT - *Massachusetts Institute of Technology*, utilizava métodos simples de avaliação de padrões para representar um psicoterapeuta. Ele foi um dos primeiros sistemas aptos a realizar essa simulação por meio de palavras-chave e regras de substituição (WEIZENBAUM, 1966). Embora atualmente seja bastante limitado, esse projeto foi um marco inicial para a compreensão do Teste de Turing e para o desenvolvimento de tecnologias mais aprimoradas que conhecemos hoje em dia.

Na década de 1970, o sistema *PARRY*, desenvolvido por Kenneth Colby, procurou simular um paciente com esquizofrenia, incorporando regras mais complexas de diálogo. Mais adiante, com o crescimento da capacidade computacional e o surgimento de bancos de dados mais robustos, novos sistemas começaram a explorar abordagens baseadas em recuperação de informações e aprendizado supervisionado. Contudo, até o início dos anos

2000, os *chatbots* ainda apresentavam comportamento fortemente restrito a padrões pré-programados, com capacidade limitada de manter contexto ou compreender a intenção do usuário (KLOPFENSTEIN et al., 2017).

A partir da década de 2010, com a ascensão do *big data* e o fortalecimento de tecnologias como *machine learning*, *deep learning* e modelos pré-treinados de linguagem natural, como o *BERT*, *GPT* e *T5*, foi possível acompanhar o desenvolvimento de agentes cada vez mais capazes de interpretar o que o usuário deseja, manter o contexto da conversa e oferecer respostas contextualizadas e embasadas (BROWN et al., 2020). Dessa forma, observa-se que o *chatbot* deixou de ser uma ferramenta limitada a pequenas interações ou simulações e está cada vez mais evoluindo para um sistema adaptável e inteligente, muitas vezes integrado a plataformas maiores de atendimento automatizado e assistentes virtuais.

Além disso, os *chatbots* têm sido amplamente adotados em diversas áreas, como comércio, atendimento ao cliente, educação, saúde, bancos e serviços públicos, proporcionando redução de custos operacionais, disponibilidade contínua e respostas em tempo real (KLOPFENSTEIN et al., 2017). Com o surgimento de assistentes pessoais como *Siri* (Apple), *Alexa* (Amazon), *Google Assistant* e *Cortana* (Microsoft), a tecnologia de agentes conversacionais foi incorporada ao dia a dia de milhares de pessoas. Esses assistentes vão além da simples conversação, pois são capazes de controlar dispositivos, agendar compromissos, responder perguntas e realizar buscas.



Figura 1 – Histórico de Evolução dos Chatbots

Fonte: Veloso, Bruna (2025)

A Figura 1 ilustra a linha do tempo com os principais marcos da evolução dos *chatbots*, desde os primeiros sistemas desenvolvidos nas décadas de 1960 e 1970, como o *ELIZA* e o *PARRY*, até os modelos atuais baseados em inteligência artificial. Essa representação

visual contribui para compreender o progresso e os avanços significativos dessa tecnologia ao longo das últimas décadas.

No entanto, o desenvolvimento de *chatbots* eficientes ainda representa um desafio, principalmente no que se refere à compreensão da linguagem natural em seus diversos níveis e também à manutenção do contexto conversacional ao longo de múltiplas interações. Questões como ambiguidade, polissemia, uso de gírias, linguagem informal e variações culturais tornam o processo de comunicação máquina-humano bastante complexo (JURAFSKY; MARTIN, 2020b).

Outro ponto que merece destaque é o papel desempenhado pelos *chatbots* na transformação digital das organizações. Segundo um estudo da *Juniper Research*, estima-se que os *chatbots* economizarão às empresas mais de US\$ 8 bilhões por ano até 2025, reduzindo significativamente a necessidade de interação humana para tarefas repetitivas. Além disso, um relatório da *Grand View Research* calcula que o mercado global de *chatbots* alcançará um valor de US\$ 27 bilhões até 2030, impulsionado pelo crescimento da adoção dessas ferramentas em diversos setores.

Exemplificativamente, bancos como o Banco do Brasil e o Bradesco empregam *chatbots* para o atendimento ao cliente e suporte financeiro. No setor de *e-commerce*, a Amazon e a Magazine Luiza implementaram assistentes virtuais para apoiar seus consumidores durante as compras. Já no ramo da saúde, hospitais e operadoras como a Unimed vêm investindo em *chatbots* para triagem e esclarecimento de dúvidas médicas. Ao permitir uma comunicação direta e personalizada com os usuários, eles se tornaram parte fundamental das estratégias de *customer experience*, oferecendo atendimento 24 horas por dia e todos os dias da semana, reduzindo o tempo de resposta e possibilitando a coleta automatizada de dados. Dessa forma, tornaram-se peças-chave no relacionamento com o cliente (ADAMOPOULOU; MOUSSIADES, 2020).

Com base nisso, é possível concluir que a trajetória dos *chatbots* vai além de avanços apenas tecnológicos e organizacionais, trata-se também de uma mudança no paradigma da interação entre humanos e máquinas, aproximando-se cada vez mais do ideal visionado por Turing, no qual a inteligência de uma máquina é percebida por sua capacidade de manter conversas funcionais e coerentes com seres humanos. Assim, os *chatbots* deixam de ser meramente instrumentos de automação e passam a representar também uma estratégia de experiência do usuário, agregando valor ao serviço prestado e à imagem das organizações que os utilizam.

2.2 Tecnologias aplicadas no desenvolvimento de *chatbots*

A implementação de ferramentas de *chatbots* demanda o arranjo de diferentes métodos e tecnologias para certificar um comportamento satisfatório. Esse processo tem sido viabilizado com o surgimento de *frameworks* e bibliotecas que trazem interfaces para criação

e implantação desses agentes com níveis distintos de complexidade.

As plataformas de desenvolvimento podem ser divididas em dois grupos predominantes: plataformas que são embasadas em regras e as plataformas que possuem apoio na inteligência artificial e no processamento de linguagem natural. As baseadas em regras funcionam por meio de uma lógica condicional com fluxos de decisão, enquanto as segundas integram modelos de *PLN*, essenciais para que o *chatbot* interprete e compreenda as intenções e contextos do usuário a partir das entradas, tornando essa interação mais natural e fluida (ADAMOPOULOU; MOUSSIADES, 2020).

Entre as principais ferramentas disponíveis atualmente, destacam-se: *Dialogflow*, *Microsoft Bot Framework*, *Rasa*, *IBM Watson Assistant*, *Amazon Lex*, *Botpress* e *ManyChat*. Abaixo, discutiremos um pouco mais sobre cada uma delas, identificando suas principais características.

1. ***Dialogflow*** – A plataforma desenvolvida pelo Google tornou-se uma das mais populares no desenvolvimento de *bots*, por oferecer uma interface intuitiva e recursos avançados como integração de *PLN*, suporte nativo a diversos idiomas, integração com o *Google Assistant* e também com plataformas sociais como *Telegram*, *Facebook Messenger* e *WhatsApp*. Ela ainda conta com um grande diferencial: o uso do modelo *BERT* para compreensão de intenções e entidades, o que agrega alta precisão na interpretação das mensagens (Google Cloud, 2024).
2. ***Microsoft Bot Framework*** – Desenvolvida pela Microsoft, essa ferramenta é voltada para desenvolvedores com mais conhecimento técnico. Oferece ampla flexibilidade e conta com integrações com serviços do *Azure*, como o *Language Understanding Intelligent Service (LUIS)*. Essa plataforma se destaca ainda por oferecer suporte à implementação de *chatbots* multicanais, seu poder direcionado a aplicações corporativas e também pela capacidade de criar fluxos conversacionais complexos, com controle extremamente detalhado sobre o código (Microsoft, 2023).
3. ***Rasa*** – A *Rasa* é uma plataforma de código aberto que oferece maior controle sobre o desenvolvimento e a implementação de *bots* com inteligência artificial, pois permite a personalização de modelos de *PLN*, integração com *APIs* externas, bancos de dados e sistemas internos. Seu principal diferencial está na privacidade dos dados (pois tudo é executado localmente), na flexibilidade para criação e na comunidade de desenvolvedores extremamente ativa. Contudo, exige um conhecimento mais avançado de programação, especificamente na linguagem *Python*, além de conceitos de aprendizado de máquina.
4. ***IBM Watson Assistant*** – O *Watson Assistant* é uma ferramenta que combina recursos de *PLN* com integração ao ecossistema da *IBM Cloud*, permitindo que o desenvolvedor crie *bots* com fluxos condicionais, compreensão de intenções e conexão

com sistemas corporativos internos. Seu uso é bastante comum em grandes organizações, especialmente nos setores financeiro e de saúde, onde se exigem maiores níveis de segurança e privacidade. Sua interface é intuitiva e acessível, e ele também possui interfaces visuais para a construção dos diálogos, embora, em projetos de larga escala, seu custo possa ser elevado (IBM, 2023).

5. **Amazon Lex** – Trata-se de uma plataforma de *chatbots* da *Amazon Web Services (AWS)*, com recursos de *PLN* baseados na mesma tecnologia utilizada pela *Alexa*. Essa plataforma permite a criação de *chatbots* com reconhecimento de fala e texto, sendo especialmente interessante para aplicações que exigem comandos de voz. Também há integração com outros serviços da *AWS*, como o *Lambda* e o *DynamoDB*, o que a torna ainda mais robusta para soluções em nuvem, embora seja necessário maior familiaridade com o sistema da *Amazon* (SERVICES, 2025).
6. **Botpress** – O *Botpress* é voltado para desenvolvedores, atuando como uma alternativa de código aberto que permite a construção de *bots* com uma interface gráfica moderna e com suporte à tecnologia de *PLN* personalizada. Sua principal característica está na forma como combina o melhor de dois mundos: interface acessível e recursos técnicos mais avançados. Ele também possui a possibilidade de ser hospedado localmente e adaptado a regras de negócio específicas, tornando-o ideal para soluções corporativas com exigências particulares (BOTPRESS, 2025).
7. **Plataformas no-code** – Além das plataformas mais conhecidas, como vimos acima, há ainda um ecossistema de plataformas como *ManyChat*, *Tars*, *Chatfuel* e *Landbot*, que oferecem soluções de *bots* baseadas em fluxos de diálogo e atuam com maior ênfase em marketing, vendas e atendimento ao cliente. Por serem *no-code*, não exigem conhecimento técnico aprofundado e, por isso, são voltadas para usuários que não dominam programação, mas desejam criar agentes simples com funcionalidade prática. Embora não ofereçam tantos recursos robustos, são amplamente utilizadas por pequenos negócios e campanhas promocionais.

De acordo com a Tabela 1, é possível observar um comparativo entre as plataformas mais utilizadas para a criação de *chatbots*. Cada coluna da tabela representa aspectos distintos que influenciam diretamente na escolha da tecnologia: a coluna “IA/*PLN*” indica o nível de suporte à inteligência artificial e ao processamento de linguagem natural; “Código Aberto” mostra se a plataforma é de código livre ou proprietário; “Facilidade” diz respeito ao nível de complexidade no uso da ferramenta; “APIs” representa a capacidade de integração com outras aplicações; e, por fim, “Ideal para” sugere os principais contextos de uso da ferramenta.

Tabela 1 – Comparativo entre plataformas para criação de *chatbots*

Plataforma	IA/PLN	Código Aberto	Facilidade	APIs	Ideal para
<i>Dialogflow</i>	Alta	Não	Alta	Sim	Projetos com IA rápida
<i>Microsoft Bot</i>	Alta	Parcial	Média	Sim	Soluções corporativas
<i>Rasa</i>	Alta	Sim	Baixa	Sim	Projetos altamente personalizados
<i>IBM Watson</i>	Alta	Não	Alta	Sim	Grandes empresas
<i>Amazon Lex</i>	Alta	Não	Média	Sim	Aplicações com voz
<i>Botpress</i>	Média	Sim	Média	Sim	Controle local e personalização
<i>ManyChat</i>	Baixa	Não	Alta	Limitada	Marketing e pequenos negócios

Ao comparar os dados da Tabela 1, percebe-se que ferramentas como *Dialogflow*, *Microsoft Bot Framework* e *IBM Watson Assistant* apresentam alto nível de suporte à IA/PLN e facilidade de uso, sendo recomendadas para aplicações empresariais e soluções rápidas. Por outro lado, plataformas como *Rasa* e *Botpress*, embora exijam mais conhecimento técnico, oferecem maior controle, personalização e privacidade dos dados. Já plataformas como *ManyChat*, voltadas ao público não técnico, focam em aplicações simples e rápidas voltadas ao marketing e vendas, mesmo com recursos limitados de inteligência artificial.

2.3 Critérios para Seleção de Plataformas de Desenvolvimento de *Chatbots*

Como visto anteriormente, a seleção de uma plataforma para o desenvolvimento de *chatbots* é uma das principais etapas para garantir o sucesso de um projeto. Essa escolha deve considerar, além dos aspectos técnicos, os objetivos de negócio, a experiência da equipe e o que se espera com o funcionamento da ferramenta para os usuários finais. Conforme destaca [McTear, Callejas e Griol \(2016\)](#), uma escolha inadequada de plataforma pode acarretar limitações técnicas, aumento de custos, retrabalho e comprometimento da experiência do usuário.

Portanto, para garantir maior acerto nas decisões, alguns critérios têm sido apontados como essenciais, tanto na literatura quanto no setor técnico. Nesta seção, serão apresentados e discutidos os principais critérios de avaliação que serão utilizados pela plataforma *Neko* como base para a recomendação das tecnologias adequadas aos usuários.

2.3.1 Objetivo do Projeto

Segundo Radziwill e Benton (2017), a falta de alinhamento pode levar ao desperdício de recursos ou ao excesso de funcionalidades irrelevantes, dificultando o desenvolvimento e a manutenção do sistema. Ou seja, a qualidade identificada em um sistema inteligente está diretamente ligada à forma como suas funcionalidades se adequam aos objetivos específicos do projeto.

Todas as plataformas estudadas anteriormente possuem características que determinam em quais situações elas se tornam mais adequadas. Por isso, é de suma importância compreender o objetivo central do projeto para que, com maior clareza, seja possível identificar qual será a mais apropriada para sua finalidade.

Por exemplo, plataformas como o *Dialogflow* são eficazes para interfaces de atendimento automatizado com integração a sistemas mais complexos, enquanto soluções como o *ManyChat* são voltadas principalmente para automações de marketing. Já o uso de APIs mais simples, como a do *Telegram*, pode ser ideal para projetos com escopo restrito, que necessitam de mais controle do código ou para fins educacionais, como o desenvolvimento de protótipos ou assistentes mais personalizadas.

2.3.2 Facilidade de Uso

Esse critério diz respeito à usabilidade da plataforma, ou seja, está relacionado à curva de aprendizado necessária para sua utilização. Como argumenta Kvale e Brinkmann (2018), a usabilidade de uma ferramenta está diretamente ligada à sua eficácia e adoção, pois plataformas que exigem um nível mais baixo de configuração inicial ou que contam com tutoriais e exemplos práticos tornam o desenvolvimento menos propenso a erros e também mais prático e rápido.

Plataformas com interfaces gráficas mais intuitivas e com grande suporte visual de conversa tendem a ser mais acessíveis para iniciantes ou grupos com menos experiência em programação, como é o caso do *Botpress* e do *ManyChat*. Por isso, esse critério torna-se decisivo para a escolha da plataforma que será utilizada.

2.3.3 Custo

O aspecto financeiro é um dos fatores determinantes, principalmente para projetos educacionais, de pequenas empresas ou de pessoas autônomas que estão começando agora. Algumas das plataformas incluem planos gratuitos com recursos limitados, enquanto outras optam por um modelo de cobrança baseado no número de mensagens ou usuários ativos. Tendo em vista que a presença de custos inesperados durante o ciclo de desenvolvimento do *chatbot* pode comprometer sua escalabilidade e continuidade.

Por isso, é de suma importância considerar o orçamento disponível e a previsão de crescimento da plataforma. Além disso, a avaliação do custo-benefício e do modelo de precificação pode evitar impactos financeiros negativos no futuro.

2.3.4 Suporte à Inteligência Artificial

Vimos que plataformas como o *Dialogflow*, *IBM Watson* e *Rasa* fazem uso de técnicas avançadas de *PLN* para identificar intenções, tornando possível criar agentes mais naturais e elaborados. Esse suporte a mecanismos de inteligência artificial é um dos diferenciais para aplicações que demandam interações mais complexas, maior contexto de conversa e que podem necessitar de adaptação do *bot* ao comportamento do usuário. [Jurafsky e Martin \(2020a\)](#) destaca que o *PLN* se tornou a base para a construção de agentes conversacionais, e sua utilização pode aprimorar ainda mais a experiência do usuário.

2.3.5 Integrações Disponíveis

Conforme destacado por [Chatterjee, Mazumdar e Das \(2021\)](#), a integração eficaz com ferramentas externas pode ampliar o potencial dos *bots* e também contribuir para a automação de processos organizacionais. Essa capacidade de integração com outros sistemas e serviços é fundamental para garantir a interoperabilidade com redes sociais, bancos de dados e *APIs* externas. Atualmente, há muitas soluções que oferecem conectores prontos para ferramentas como *Telegram*, *Facebook Messenger* e *WhatsApp*, por exemplo.

Em muitos casos, a escolha da plataforma pode ser limitada por essa necessidade de conexão com determinados canais de comunicação ou serviços. Assim, a análise das integrações suportadas deve ser realizada de acordo com as necessidades técnicas do projeto.

2.3.6 Comunidade e Documentação

De acordo com [Raymond \(1999\)](#), em projetos de *software* livre, “muitas mãos tornam o trabalho leve”, o que reflete a importância do apoio comunitário na evolução e confiabilidade das plataformas. Ou seja, a existência de uma comunidade ativa e de uma documentação clara e atualizada exerce grande influência sobre a curva de aprendizado, a resolução de problemas durante o desenvolvimento e a capacidade de personalizar o *chatbot*.

Rasa e *Botpress*, plataformas *open source*, contam com comunidades extremamente engajadas, fóruns ativos para discussão e resolução de dúvidas, e com contribuições contínuas da comunidade global. Além disso, possuem documentação completa e oferta de exemplos práticos que ajudam a reduzir a dependência de suporte técnico especializado.

2.3.7 Escalabilidade

Essa característica está diretamente ligada à capacidade de uma plataforma suportar o crescimento do *chatbot*. Isso pode ser medido em termos de usuários simultâneos, complexidade do fluxo de diálogo, armazenamento de dados e também desempenho geral da aplicação.

Plataformas como o *Dialogflow* e o *Amazon Lex* são baseadas em nuvem e geralmente oferecem melhores soluções para escalabilidade automática. Enquanto isso, soluções locais exigem infraestrutura própria e maior planejamento.

Clements, Kazman e Klein (2012) argumenta que arquiteturas escaláveis são essenciais para aplicações de longo prazo e, por isso, considerar o potencial de crescimento do projeto é fundamental para evitar limitações futuras.

2.3.8 Segurança e Privacidade

A segurança da informação e a privacidade dos dados dos usuários são aspectos críticos, especialmente em projetos que lidam com dados sensíveis ou que estejam sujeitos à regulamentação de legislações como a *LGPD* (Lei Geral de Proteção de Dados) no Brasil e o *GDPR* (*General Data Protection Regulation*) na Europa. Por isso, é necessário que plataformas e aplicações adotem medidas de segurança robustas, como criptografia, controles de acesso e anonimização de dados.

O uso de soluções que não fornecem garantias mínimas de proteção pode resultar em riscos legais e prejuízos à reputação, expondo usuários e organizações. A confiança do usuário é diretamente proporcional à transparência e à segurança da tecnologia utilizada.

Com base em Stallings (2017), a segurança deve ser considerada desde a concepção do sistema, sendo um requisito não funcional essencial em qualquer aplicação moderna, incluindo agentes conversacionais.

Com base nas características vistas até aqui, é possível perceber que a escolha adequada de uma tecnologia de desenvolvimento não deve ser pautada unicamente na popularidade ou preferência pessoal, sendo necessária uma análise crítica desses critérios, que leve em conta fatores técnicos, estratégicos e operacionais. Cada critério abordado contribui de forma significativa para a construção de um modelo de recomendação mais preciso, considerando viabilidade, eficiência e continuidade da solução proposta.

A plataforma *Neko*, ao incorporar tais considerações e aspectos em seu processo de recomendação, visa oferecer uma base orientadora significativa para a tomada de decisão técnica, contribuindo assim para um desenvolvimento sustentável, bem-sucedido e alinhado às necessidades específicas do usuário.

3 Trabalhos Relacionados

Nesta seção, são apresentados e analisados alguns trabalhos relacionados ao desenvolvimento e à aplicação de *chatbots* em diferentes contextos. O objetivo principal da análise dos estudos é identificar diferentes abordagens que buscam facilitar a recomendação ou criação de plataformas ligadas à construção de *chatbots*. Com isso, buscamos compreender as propostas e limitações dessas soluções. Com base nisso, torna-se possível fundamentar o presente trabalho, destacando suas contribuições e também justificando a relevância encontrada para o desenvolvimento da ferramenta *Neko*. Seleccionamos estudos que abordam desde *frameworks* no desenvolvimento dos agentes até plataformas de recomendação e avaliação comparativa entre tecnologias voltadas para esse objetivo, e foi realizada a comparação quanto a aspectos como tipo de abordagem, critérios utilizados para avaliação, tecnologias analisadas e também o nível de personalização oferecido, conforme ilustrado na Tabela 2.

Tabela 2 – Comparação entre trabalhos relacionados sobre desenvolvimento de *chatbots*

Trabalho	Ferramentas analisadas	Abordagem	Foco principal	Metodologia
Rai e Bagde (2024)	Microsoft Bot, ChatGPT, Watson, Dialogflow, Lex, RASA, Wit.ai, Many-Chat	Comparativa (frameworks vs. plataformas)	Evolução histórica, capacidades de <i>NLP</i>	Revisão e comparação técnica
Sharma e Joshi (2020)	RASA (NLU, Core, DIET)	Estudo aprofundado de uma ferramenta	Implementação técnica, customizações, uso real	Testes práticos em <i>IDE</i> com banco de dados/ <i>API</i>
Dagkoulis e Moussiades (2022)	Dialogflow, Lex, Watson, Microsoft Bot	Comparativa de plataformas <i>CDP</i>	Inovação e eficiência no uso de <i>CDPs</i>	Atribuição de notas com base em critérios definidos
Vidya, Vernekar e Kelkar (2023)	Microsoft Bot Framework, RASA	Estudo de caso comparativo	Recomendação baseada em experiência prática	Desenvolvimento de dois <i>chatbots</i> idênticos
Abdellatif, Badran, Costa e Shihab (2021)	IBM Watson, RASA, Dialogflow, Microsoft LUIS	Avaliação de plataformas <i>NLU</i>	Desempenho em tarefas de Engenharia de <i>Software</i>	Avaliação com dois <i>datasets</i> reais
Veloso (2025)	Dialogflow, RASA, Botpress, Amazon Lex, Watson	Plataforma de recomendação interativa	Auxílio personalizado na escolha de tecnologia para <i>chatbots</i>	Coleta de dados via formulário + análise automatizada

Dentre os trabalhos identificados, destaca-se inicialmente o estudo de Rai e Bagde (2024), que realizou uma avaliação comparativa entre os *frameworks* disponíveis para o desenvolvimento de *bots* e também entre as plataformas. Foram avaliadas as ferramentas Microsoft Bot Framework, ChatGPT, IBM Watson Assistant, Google Dialogflow, Amazon Lex, RASA NLU e Wit.ai. Durante o trabalho, os autores apresentaram um panorama completo sobre o processo evolutivo dos *chatbots*, tornando possível classificá-los de acordo com os métodos usados na interação e também pelos níveis de inteligência. O principal ponto destacado no estudo é que os *frameworks* oferecem maior flexibilidade e integração com tecnologias avançadas, como o Processamento de Linguagem Natural (*PLN*), sendo ideais para projetos que necessitam de maior personalização e autonomia no fluxo conversacional. Já as plataformas de *chatbots*, como ManyChat, Chatfuel, Flow XO e Botsify, seriam mais limitadas quanto ao uso de tecnologias avançadas de *PLN*, embora mais simples de utilizar e com métodos de implantação rápidos, voltadas para usuários com menor domínio técnico.

Posteriormente, em um estudo direcionado especialmente ao *framework* Rasa, [Sharma e Joshi \(2020\)](#) realiza uma análise mais aprofundada sobre sua estrutura e funcionalidades. Esse trabalho enfatiza a arquitetura modular do Rasa, composta por dois principais componentes: o Rasa NLU, responsável pela interpretação da linguagem, e o Rasa Core, destinado à gestão do diálogo. Durante o estudo, ficam evidentes outras características do Rasa, como sua capacidade de integração com *APIs* e bancos de dados, suporte ao aprendizado por reforço com o modelo *DIET*, e a possibilidade de personalização do fluxo de mensagens por meio de arquivos *sockets*, permitindo a coleta de metadados do usuário (como IP e porta de conexão). Também foram realizados testes práticos utilizando o *IDE PyCharm* em ambiente *Windows*, o que reforça as ideias abordadas no estudo.

Outro trabalho relevante para este estudo é o de [Dagkoulis e Moussiades \(2022\)](#), que apresenta uma avaliação comparativa entre diferentes plataformas de desenvolvimento de *chatbots*, chamadas de *Chatbot Development Platforms (CDPs)*, oferecidas por grandes empresas como Google, Amazon, Microsoft e IBM. A metodologia empregada abrange a definição dos critérios utilizados para a comparação, como capacidade de integração, usabilidade, suporte a *PLN*, adaptabilidade a domínios específicos e frequência de atualizações. A avaliação é realizada com base em um sistema de pontuação atribuído conforme o desempenho em testes práticos e requisitos presentes em projetos reais. Os resultados indicam que não existe uma solução universalmente superior, e a escolha da plataforma mais adequada depende do contexto da aplicação e dos recursos disponíveis para seu desenvolvimento.

Em seguida, os autores [Kelkar et al. \(2023\)](#) propõem uma abordagem mais experimental, com um estudo comparativo entre os *frameworks* Microsoft Bot Framework e RASA. Para esse propósito, foram construídos dois *chatbots* com funcionalidades idênticas, permitindo uma observação mais direta entre as ferramentas, principalmente em termos de desenvolvimento, documentação, suporte à personalização e desempenho na interação com os usuários finais. O estudo demonstra que ambas as tecnologias apresentam vantagens distintas: o Rasa exibiu maior facilidade para ajustes finos no fluxo conversacional, enquanto o Microsoft Bot Framework se destacou na integração com o ecossistema *Azure* e com sistemas de conhecimento da Microsoft.

Por fim, temos o trabalho de [Shihab et al. \(2021\)](#), que foca sua análise nas plataformas de *Natural Language Understanding (NLU)* no contexto da Engenharia de *Software*. Durante o estudo, são avaliadas as ferramentas IBM Watson, Google Dialogflow, RASA e Microsoft LUIS, considerando três critérios principais: acurácia na classificação de intenções (*intents*), estabilidade dos escores de confiança e eficiência na extração de entidades. Os testes foram realizados com dois conjuntos de dados reais: um voltado à interação com repositórios de *software* e outro com perguntas extraídas do *Stack Overflow*. Com base nos resultados, o IBM Watson é apontado como o sistema mais equilibrado, com *F1-score* superior a 84 na classificação de *intents*. No entanto, o RASA apresentou os escores de

confiança mais consistentes, com mediana superior a 0,91, enquanto o Microsoft LUIS se destacou na extração de entidades.

Dessa forma, destaca-se também o presente trabalho, que propõe a criação de uma plataforma interativa chamada *Neko*, voltada à recomendação automatizada de tecnologias para o desenvolvimento de *chatbots*. Diferentemente dos estudos anteriores, que se concentram na análise técnica ou comparativa entre ferramentas específicas, este projeto tem como foco central a personalização da escolha tecnológica, considerando as necessidades, objetivos e o nível de conhecimento técnico de cada usuário. A metodologia adotada envolve o mapeamento de diversas plataformas e *frameworks* amplamente utilizados no mercado, como Dialogflow, Rasa, Botpress, Amazon Lex e IBM Watson, acompanhado da definição de critérios objetivos de avaliação, tais como facilidade de uso, custo, integração com outras ferramentas, suporte à inteligência artificial e escalabilidade. A partir disso, foi desenvolvida uma interface funcional que coleta dados por meio de perguntas direcionadas ao usuário e aplica uma lógica de recomendação baseada nos critérios estabelecidos. Com isso, a ferramenta busca contribuir diretamente para a democratização do acesso ao desenvolvimento de *chatbots*, oferecendo suporte à tomada de decisão de forma simplificada, acessível e personalizada, principalmente para iniciantes e usuários com pouca ou nenhuma familiaridade técnica.

4 Metodologia

De acordo com a Figura 2, a metodologia adotada neste trabalho foi estruturada em etapas que orientam desde a definição dos requisitos até a avaliação final da solução desenvolvida. Cada etapa representa uma fase essencial do processo de construção do *chatbot Neko*, buscando garantir clareza e organização durante o desenvolvimento.

A figura apresenta, de forma visual, o fluxo seguido no projeto: inicia-se com a pesquisa e o levantamento de requisitos, seguido da seleção das tecnologias mais adequadas, conforme discutido no capítulo anterior, e da modelagem da solução. Em seguida, ocorre a implementação do *chatbot*, utilizando os recursos definidos previamente e, por fim, a fase de testes e ajustes, que visa validar o desempenho da aplicação e assegurar que os objetivos definidos foram atendidos.

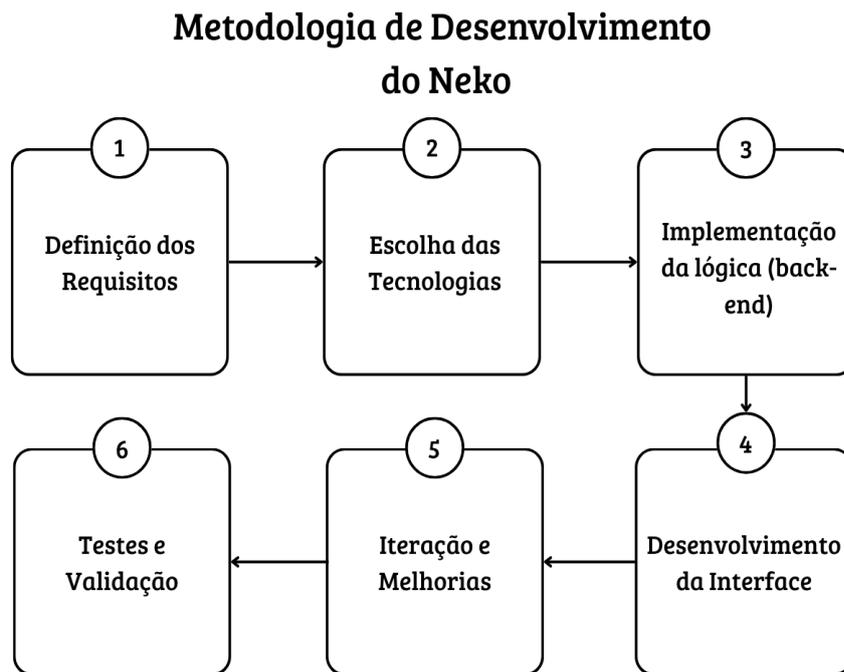


Figura 2 – Metodologia do Desenvolvimento

Fonte: Veloso, Bruna (2025)

4.1 Planejamento e Levantamento de Requisitos

Nessa primeira etapa, tem-se como objetivo o planejamento inicial do projeto e a definição dos requisitos funcionais e não funcionais da aplicação, que veremos com mais detalhes a seguir. Esse levantamento foi realizado após uma revisão bibliográfica e uma análise das plataformas existentes disponíveis para o desenvolvimento de *chatbots*.

Além dessa coleta inicial sobre as plataformas, foi realizada também uma pesquisa por meio de algumas ferramentas de recomendação que existem no mercado e que são muito utilizadas por iniciantes na tecnologia, como o *GitHub Copilot* e o *Stack Overflow Trends*. Foi possível observar, por exemplo, que muitos usuários têm maior dificuldade em selecionar ferramentas que se adequem ao seu contexto, seja por falta de uma orientação clara, sobrecarga de informações ou interfaces muito complexas. Outro ponto relevante foi a ausência de plataformas que consigam unir simplicidade de uso com automatização da personalização.

Com base nessa pesquisa, também foi possível identificar de forma mais clara o conjunto de critérios prioritários para os usuários que estão em dúvida entre ferramentas de tecnologia, como a curva de aprendizado, suporte da comunidade, compatibilidade para integração com outros serviços, custo, entre outros. Reunimos esses critérios como atributos que nortearam nossa modelagem para o sistema de recomendação usado na plataforma e permitiram a definição dos requisitos funcionais e não funcionais.

4.1.1 Requisitos Funcionais

Os requisitos funcionais são as funcionalidades que serão oferecidas pelo sistema, ou seja, descrevem o que o sistema deve realizar para atender às necessidades do usuário, permitindo que a experiência de uso seja eficiente e coerente com o objetivo principal do projeto. Os principais requisitos funcionais definidos foram:

- **RF01 - Entrada de preferências:** o sistema deve permitir que o usuário forneça as características do projeto e o contexto no qual ele será utilizado;
- **RF02 - Sistema de pontuação e ranqueamento:** o sistema deve processar os critérios informados pelo usuário, compará-los com os dados das tecnologias cadastradas e, a partir disso, calcular uma pontuação para ranqueá-las conforme a adequação ao perfil do usuário;
- **RF03 - Apresentação das recomendações:** o sistema deve exibir ao usuário uma lista personalizada de tecnologias recomendadas, com suas respectivas pontuações e informações adicionais (descrição, área de uso, nível de dificuldade e links úteis);
- **RF04 - Navegação entre etapas:** o sistema deve permitir que o usuário navegue de forma fluida pelas etapas de coleta de dados, visualização dos resultados e exploração das recomendações, tornando possível que ele edite os critérios informados anteriormente;
- **RF05 - Cadastro e atualização de tecnologias:** o sistema deve permitir, internamente, a inclusão, modificação e exclusão das tecnologias disponíveis no banco de dados da aplicação;

4.1.2 Requisitos Não Funcionais

Enquanto os requisitos funcionais definem as ações do sistema, como visto anteriormente, os requisitos não funcionais tratam da qualidade do sistema, ou seja, estão relacionados ao desempenho, segurança, usabilidade e confiabilidade da aplicação. Para o desenvolvimento da ferramenta *Neko*, foram definidos os seguintes requisitos não funcionais:

- **RNF01 – Usabilidade e simplicidade:** é necessário que a interface do sistema seja intuitiva, clara e de fácil navegação, mesmo para usuários leigos e com pouco ou nenhum conhecimento técnico;
- **RNF02 – Acessibilidade:** a aplicação deve ser responsiva e acessível em diferentes dispositivos, como desktop, tablet e smartphome, seguindo boas práticas de acessibilidade visual e de navegação;
- **RNF03 – Desempenho:** é indispensável que o tempo de resposta para o carregamento de páginas e a apresentação das recomendações seja otimizado, com foco na experiência positiva do usuário;
- **RNF04 – Escalabilidade:** a estrutura do sistema precisa permitir a inclusão de novas tecnologias, critérios de recomendação e melhorias futuras, sem comprometer seu funcionamento ou tempo de processamento;
- **RNF05 – Modularidade e organização do código:** é fundamental que o sistema seja desenvolvido de forma modular, com separação clara de responsabilidades, a fim de facilitar a manutenção e futuras expansões do projeto;
- **RNF06 – Integridade dos dados:** os dados inseridos e processados pelo sistema devem ser armazenados de forma consistente e segura;

4.2 Tecnologias e Ferramentas Utilizadas

A escolha das ferramentas e tecnologias utilizadas durante o desenvolvimento do *Neko* foi definida com base na simplicidade, acessibilidade e viabilidade da implementação, considerando o objetivo do projeto, o contexto e os recursos disponíveis. A preferência foi por aplicar soluções com baixo custo computacional, ampla documentação e uma curva de aprendizado compatível com o tempo disponível para o projeto, mas sem comprometer sua eficiência e escalabilidade no futuro.

4.2.1 Front-end

O *front-end* refere-se à parte do site ou aplicação com a qual o usuário interage diretamente, ou seja, engloba a interface gráfica do projeto e inclui elementos como menus, botões, imagens, entre outros. Essa camada foi implementada utilizando HTML (*HyperText Markup Language* — Linguagem de Marcação de Hipertexto), CSS (*Cascading Style Sheets* — Folhas de Estilo em Cascata) e *JavaScript*. Essa seleção permitiu um maior controle sobre a estrutura e o estilo da aplicação, possibilitando a customização dos componentes e garantindo a compatibilidade com a maioria dos navegadores utilizados atualmente, mesmo em dispositivos mais antigos ou com recursos limitados.

A organização dessa primeira parte foi estruturada de forma articulada, separando as responsabilidades entre marcação (HTML), estilização (CSS) e lógica (*JavaScript*). Optou-se pelo desenvolvimento do design responsivo manualmente, para garantir uma boa visualização em diferentes tamanhos de tela, com atenção à usabilidade e acessibilidade.

4.2.2 Back-end

Essa segunda camada diz respeito à parte da aplicação à qual o usuário não tem acesso e que é responsável por conduzir a lógica, os dados e as funcionalidades que ficam por trás da interface gráfica, garantindo o funcionamento das aplicações. Nessa camada de servidor, foi utilizado o *Node.js* em conjunto com o *framework Express*. Essa escolha se deu por conta de sua leveza, simplicidade e compatibilidade com o *JavaScript*. Ao optarmos por essa combinação, foi possível construir rotas HTTP para o envio e recebimento de dados, gerenciar as requisições feitas pelo *front-end* e controlar a lógica de recomendação.

A API desenvolvida em *Express* opera como um intermediário entre o *front-end* e o banco de dados, sendo também responsável por processar as entradas dos usuários, aplicar os critérios de recomendação e retornar as respostas de forma estruturada.

4.2.3 Banco de Dados

Para o banco de dados, decidiu-se utilizar o *Google Sheets* por meio de integração via API. Essa abordagem possibilitou uma solução mais prática, com fácil manutenção e baixo custo, além de facilitar a visualização, edição e atualização dos dados utilizados na plataforma, como o registro de cada tecnologia, os critérios avaliativos e os pesos atribuídos a cada item.

4.2.4 Processamento de Linguagem Natural (PLN)

Pensando em futuras melhorias e no enriquecimento da ferramenta por meio de uma melhor interpretação das preferências dos usuários, foram utilizadas também bibliotecas simples de *JavaScript* para PLN, que permitiram a identificação inicial de palavras-chave,

sinônimos e padrões de linguagem. Apesar de o projeto ainda não utilizar modelos de aprendizado profundo, essa camada de análise contribuiu para o estudo e a experimentação de formas de tornar as respostas mais relevantes e adequadas às entradas fornecidas pelo usuário.

4.2.5 Ferramentas de Desenvolvimento e Gerenciamento

Além das tecnologias vistas até aqui, durante todo o ciclo de desenvolvimento foram adotadas também algumas ferramentas de apoio para garantir maior organização e produtividade. São elas:

- **Visual Studio Code:** editor de código-fonte gratuito, utilizado como ambiente de desenvolvimento principal. É bastante conhecido por suas extensões, que permitem adaptar a plataforma a diversas necessidades, como, neste caso, o suporte a *JavaScript* e *Node.js*;
- **Git e GitHub:** essas duas ferramentas, em conjunto, permitem o controle de versão e o armazenamento do código-fonte. O *Git* é um sistema que rastreia alterações no projeto, enquanto o *GitHub* é a plataforma que hospeda os repositórios que utilizam *Git*;
- **Trello:** utilizado para o gerenciamento de tarefas e o planejamento das etapas do projeto, permitindo uma divisão clara entre entregas e prazos;
- **Google Forms:** aplicado na etapa de coleta de dados dos usuários, facilitando também a obtenção de *feedbacks*.

A partir disso, essas escolhas permitiram o desenvolvimento de uma aplicação acessível e adaptável. O uso de *JavaScript* tanto no *front-end* quanto no *back-end* garantiu consistência na lógica e maior integração entre as camadas da aplicação, enquanto a integração com o *Google Sheets* possibilitou o controle inicial dos dados com flexibilidade para futuras expansões ou migrações, caso haja uma evolução da aplicação para uma arquitetura mais robusta posteriormente.

Além disso, todas as tecnologias utilizadas possuem documentação atualizada, comunidades ativas com diversos tutoriais e baixo custo de implementação, o que tornou o projeto sustentável e viável em um contexto acadêmico, especialmente para iniciativas que enfrentam limitações semelhantes.

4.3 Desenvolvimento do Módulo de Recomendação (back-end)

Como visto anteriormente, a lógica utilizada para o funcionamento da plataforma Neko foi implementada utilizando *JavaScript* com *Node.js*, integrado ao *framework Express*, que

permitiu a criação das rotas e também o gerenciamento das requisições *HTTP*. O principal objetivo dessa camada foi garantir que o processamento das preferências informadas pelo usuário estivesse correto e que a geração das recomendações tecnológicas personalizadas também estivesse de acordo com os critérios previamente definidos durante o levantamento de requisitos.

Para isso, a arquitetura do *back-end* foi definida de forma modular, a fim de tornar o código mais claro e facilitar a manutenção. Dessa forma, cada funcionalidade foi organizada em arquivos separados, buscando permitir a escalabilidade da solução sem comprometer a compreensão do código. As primeiras requisições são provenientes do *front-end* e são enviadas via método *POST*, sendo posteriormente recebidas e tratadas pelo servidor *Express*, que extrai os dados enviados pelo usuário e os processa por meio da função principal de recomendação. Nessa função, é aplicada uma lógica baseada em um sistema de pontuação, no qual cada tecnologia presente no banco de dados é avaliada conforme os critérios informados.

Utilizou-se o *Google Sheets*, acessado por meio de sua *API* oficial, como banco de dados. Essa escolha mostrou-se adequada para a fase inicial do projeto, pois proporcionou uma solução leve, gratuita e de fácil integração, eliminando a necessidade de configurar um sistema de banco de dados relacional ou *NoSQL*. Os dados armazenados incluem informações sobre cada tecnologia analisada, como nome, descrição, área de aplicação e alguns requisitos técnicos. A cada requisição, a aplicação realiza uma chamada assíncrona a essa planilha, obtendo os registros atualizados e executando a filtragem e o ranqueamento das tecnologias com base na lógica descrita.

Visando tornar a recomendação mais significativa, foi incluído um sistema de pontuação ponderada, que utiliza pesos distintos entre os critérios, conforme a importância atribuída durante as entrevistas e a revisão da literatura. Por exemplo, o critério “nível de familiaridade” tem maior peso para usuários iniciantes, enquanto “compatibilidade com outras tecnologias” é mais relevante para quem já possui algum conhecimento técnico. Em seguida, a pontuação final de cada ferramenta é convertida em um *ranking*, que é retornado ao *front-end* em formato *JSON*, contendo todas as recomendações acompanhadas de breves justificativas para apoiar a tomada de decisão do usuário.

Além disso, no que diz respeito à segurança, embora a aplicação seja de pequeno porte e não faça o tratamento direto de dados sensíveis, foram aplicadas práticas básicas de proteção, como a sanitização de entradas para evitar injeções e a verificação dos tipos de dados recebidos. Essa estrutura também foi planejada para permitir futuras melhorias, como a integração com autenticação de usuários ou a migração para bancos de dados mais robustos, conforme o crescimento da plataforma.

4.4 Desenvolvimento da Interface

Para o desenvolvimento da interface da plataforma *Neko*, priorizou-se três características: simplicidade, acessibilidade e usabilidade. Por isso, foram utilizadas tecnologias *web* fundamentais, com o objetivo de garantir leveza, compatibilidade entre navegadores e maior facilidade de manutenção.

Seguindo os princípios do Design Centrado no Usuário (DCU), a construção da interface buscou criar uma experiência intuitiva e funcional. Com base nisso, o layout foi planejado para ser claro e direto, com uma navegação simples e elementos dispostos de forma lógica. As cores, tipografias e ícones foram padronizados para manter a consistência visual em todas as telas, promovendo maior familiaridade para o usuário. Também foram incorporadas práticas básicas de acessibilidade, como alto contraste entre elementos, fontes legíveis e suporte à navegação via teclado, garantindo o uso por um público mais amplo.

A estrutura da interface foi dividida em duas seções principais: a página inicial e a página onde está disponível o chat para interação com o *bot*. A página inicial apresenta o propósito do sistema, seus objetivos, instruções de uso e um botão que direciona para a interação com o *Neko*. Já a página de interação foi criada para que o usuário informe suas preferências e necessidades. Para isso, o *JavaScript* foi utilizado para validar e processar as informações do formulário em tempo real, promovendo uma interação mais dinâmica. Ao final da interação, e ainda na mesma tela, o *bot* exibe as recomendações tecnológicas, apresentando os nomes das ferramentas sugeridas, a pontuação de compatibilidade e pequenas justificativas que contextualizam essa escolha, além de *links* úteis para auxiliar o usuário nas próximas etapas de desenvolvimento.

Além das funcionalidades principais, também foram incluídas orientações durante a interação, que explicam brevemente termos técnicos, contribuindo para uma maior inclusão e autonomia de usuários que ainda não possuem tanto conhecimento na área.

Dessa forma, fica evidente que a adoção de tecnologias *web* nativas, sem o uso de *frameworks*, foi uma decisão estratégica. Essa abordagem permitiu maior controle sobre a interface e reduziu a complexidade do sistema, mantendo-se fiel à proposta de acessibilidade e facilidade de compreensão do código. Ao privilegiar soluções desenvolvidas de forma simples, a plataforma reforça seu compromisso com a democratização do acesso à informação tecnológica e coloca a experiência do usuário no centro do processo de desenvolvimento.

4.4.1 Interface e Experiência do Usuário

Com o objetivo de oferecer uma experiência simples, fluida e intuitiva, a plataforma *Neko* foi desenvolvida com foco na usabilidade e na personalização do processo de recomendação. A interface prioriza a comunicação direta com o usuário por meio de um *chatbot*

interativo, possibilitando uma navegação orientada e acolhedora, mesmo para aqueles com pouca familiaridade com tecnologias.

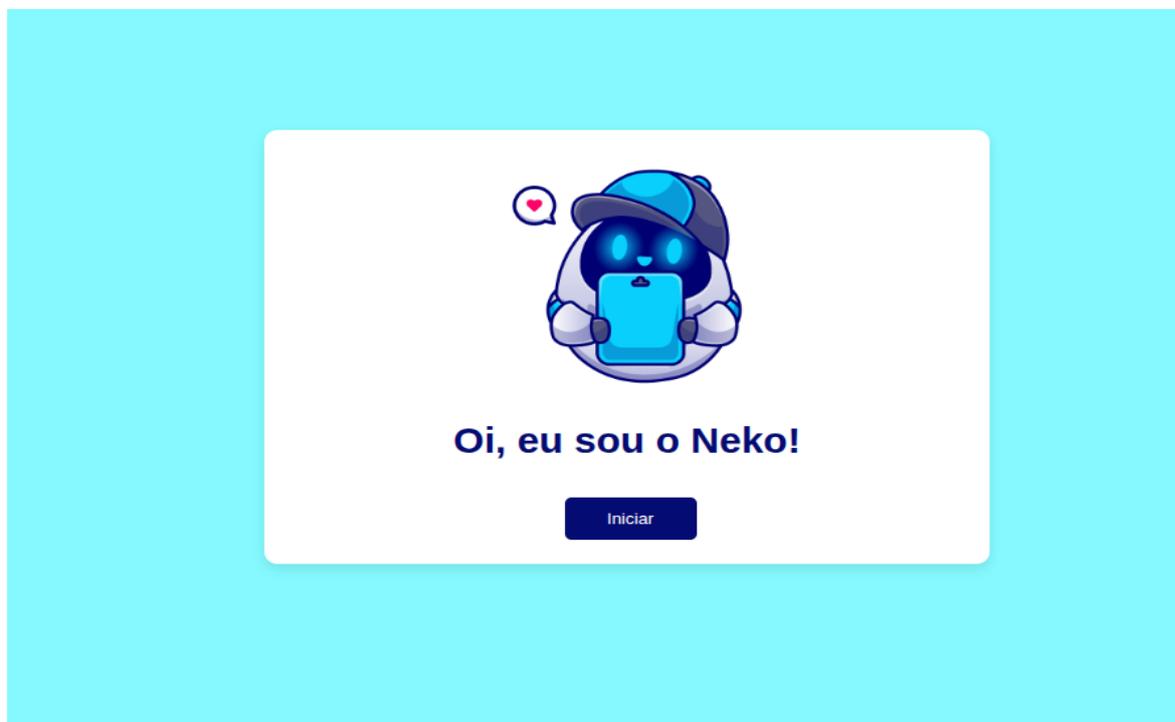


Figura 3 – Tela de Início da plataforma Neko

Fonte: Veloso, Bruna (2025)

A primeira interface, vista na Figura 3, apresentada ao usuário, é a tela de início, onde aparece apenas o mascote da plataforma acompanhado da saudação “Oi, eu sou o Neko”, além de um botão “Iniciar”. Essa tela tem como objetivo transmitir uma primeira impressão amigável e estabelecer uma conexão leve e informal com o usuário, incentivando a continuidade da navegação.

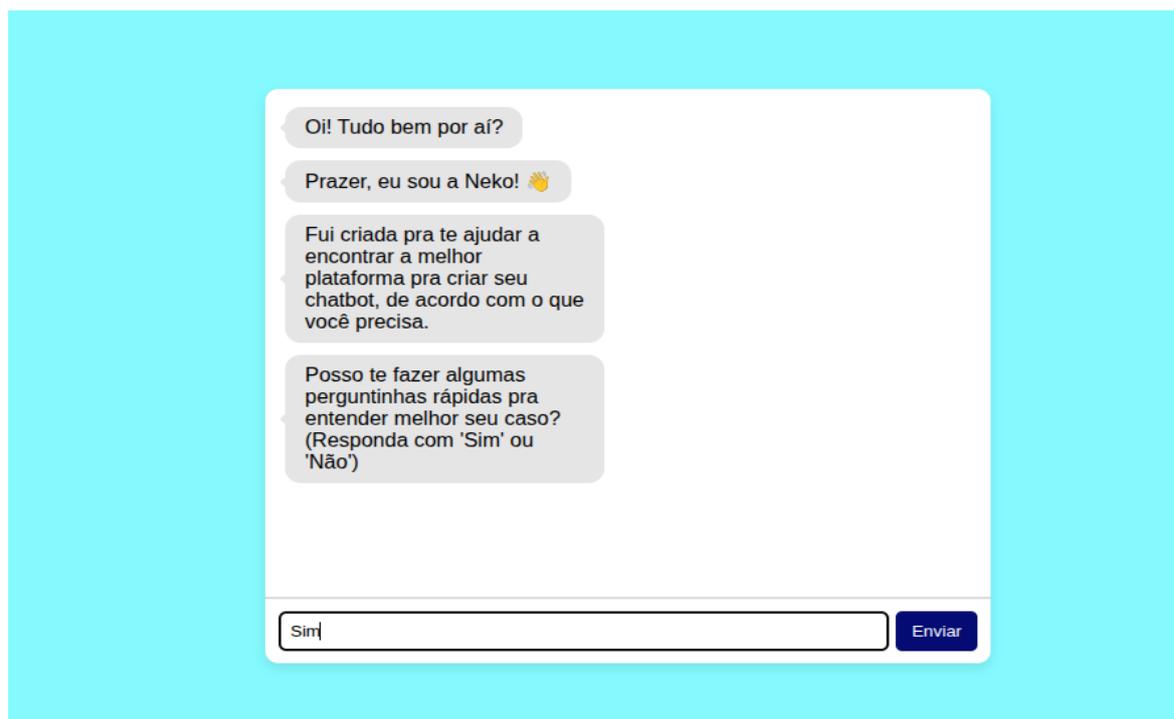


Figura 4 – Tela de primeira interação da plataforma Neko

Fonte: Veloso, Bruna (2025)

Em seguida, ao clicar em “Iniciar”, o usuário é direcionado para a tela de primeira interação (Figura 4), onde o assistente virtual Neko se apresenta formalmente, explica brevemente a proposta da plataforma e solicita autorização para dar início à sequência de perguntas. Essa etapa tem como função contextualizar o usuário e prepará-lo para o processo de coleta de informações, reforçando que as recomendações geradas serão personalizadas conforme as respostas fornecidas.

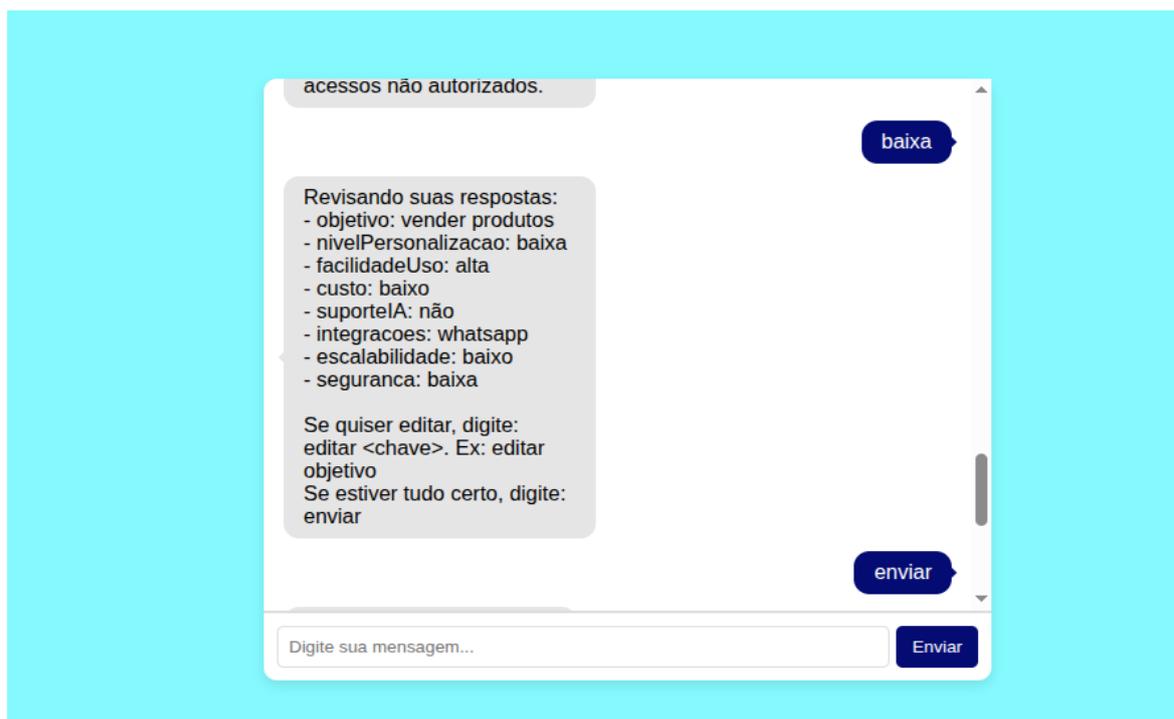


Figura 5 – Tela de revisão das informações da plataforma Neko

Fonte: Veloso, Bruna (2025)

Na sequência, a plataforma apresenta um conjunto de perguntas objetivas relacionadas às preferências, necessidades e limitações técnicas do usuário. Após o término da coleta de dados, é exibida a tela de revisão de respostas (Figura 5), onde o usuário tem a oportunidade de revisar e editar os dados fornecidos antes de enviá-los definitivamente. Essa funcionalidade foi pensada para garantir maior controle e autonomia ao usuário, permitindo correções e ajustes com facilidade.

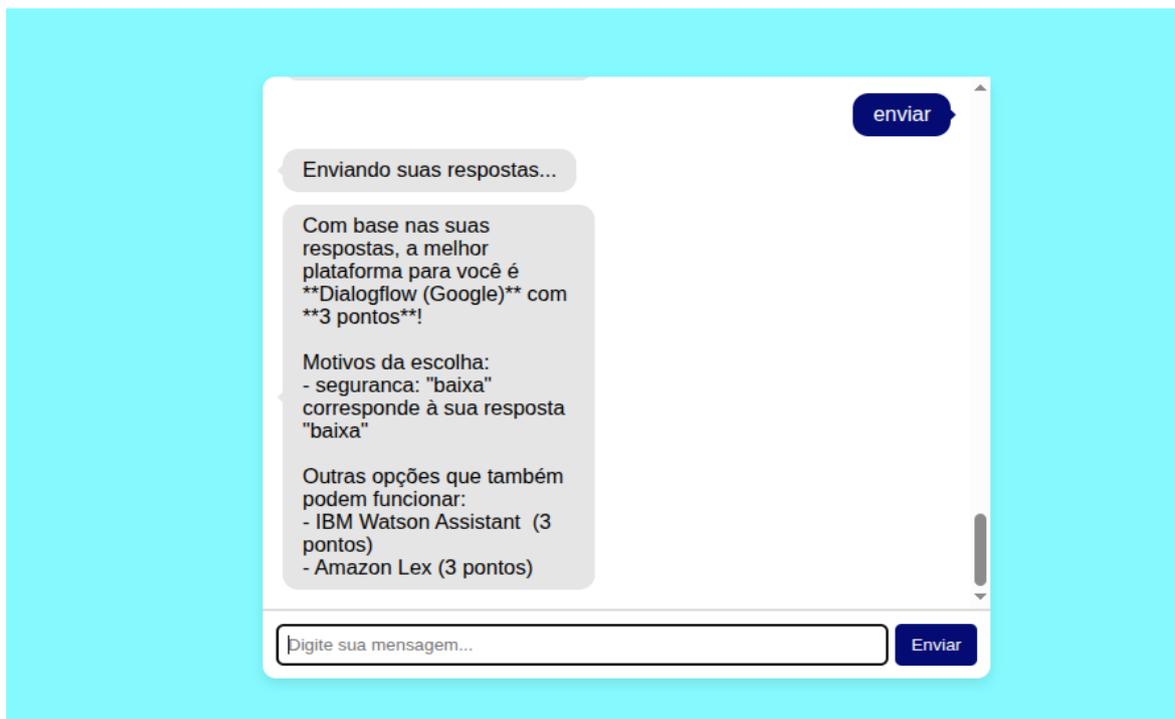


Figura 6 – Tela final da plataforma Neko

Fonte: Veloso, Bruna (2025)

Por fim, ao confirmar o envio das respostas, a plataforma exibe a tela de processamento e recomendação (Figura 6), na qual o sistema apresenta a sugestão principal de tecnologia para o desenvolvimento do *chatbot*, acompanhada de uma justificativa para a escolha, além de outras opções secundárias que também se adequam ao perfil identificado. Essa entrega final visa oferecer não apenas uma solução imediata, mas também ampliar o conhecimento do usuário sobre as alternativas disponíveis, promovendo uma escolha mais consciente e fundamentada.

4.5 Iterações e Melhorias

Nesta etapa, buscou-se o aprimoramento da plataforma *Neko*, visando melhorias na eficiência, usabilidade e maior alinhamento com os perfis dos usuários. Após a finalização da primeira versão funcional da ferramenta, adotou-se uma abordagem incremental e centrada no usuário para garantir a evolução contínua da plataforma. Essa abordagem baseou-se em ciclos curtos de desenvolvimento, testes e análise de métricas de uso.

Os testes foram realizados mediante a liberação da plataforma para um grupo inicial composto por estudantes e profissionais iniciantes na área de tecnologia. Por meio da utilização da ferramenta, aplicação de formulários *online* de forma anônima e sessões observadas de uso, foram coletados retornos sobre diversos aspectos: experiência de uso,

clareza da comunicação, relevância das perguntas, desempenho da plataforma e opiniões sobre as recomendações oferecidas.

A partir das respostas, foi possível identificar pontos fortes do sistema, como a simplicidade de navegação e a linguagem acessível. No entanto, também foram reveladas algumas limitações, como a ocorrência de empates entre as plataformas recomendadas, ausência de justificativas claras para as recomendações e a presença de termos técnicos que não eram explicados ao usuário.

Diante disso, revisou-se a lógica utilizada para a recomendação. A função de pontuação passou a considerar pesos distintos atribuídos pelo próprio usuário. Por exemplo, em projetos que exigem maior segurança e pouca personalização, o usuário pode atribuir um peso maior ao critério de segurança. Além disso, foram adicionados novos atributos para comparação, como a disponibilidade de documentação oficial e a existência de comunidades e suporte ativos.

No *front-end*, realizaram-se melhorias voltadas à experiência do usuário, como uma página principal com informações sobre a plataforma, mensagens informativas durante o carregamento, e uma apresentação mais estruturada dos resultados, destacando os principais critérios que influenciaram cada recomendação. Esse processo buscou tornar o sistema mais transparente, além de educar o usuário sobre os fatores que orientam cada sugestão, promovendo maior confiança na plataforma.

Durante esse processo de organização e acompanhamento, utilizaram-se ferramentas como o *Trello*, que possibilitou a visualização eficiente das tarefas em andamento, o controle de versões e reuniões, e a priorização com base em critérios técnicos e nas necessidades dos usuários. Com essa metodologia ágil de iteração, foi possível manter o foco na entrega contínua de valor.

Essa etapa reforçou o compromisso da ferramenta com a melhoria contínua e com o princípio de colocar o usuário no centro do desenvolvimento. Cada ciclo de ajuste aproximou o sistema de seu objetivo principal: oferecer uma solução acessível, coesa e orientada à tomada de decisão por parte de quem busca orientação tecnológica. Ao unir simplicidade de uso e inteligência na recomendação, a plataforma firma-se como uma proposta criativa e moderna para o ecossistema de inteligência artificial e apoio ao desenvolvimento de software.

4.6 Testes e Validação

Por fim, a etapa de testes e validação do funcionamento final foi essencial, considerando que a proposta da ferramenta *Neko* está diretamente ligada à recomendação personalizada de tecnologias. Tornou-se necessário garantir, além da estabilidade do sistema, a relevância dos resultados entregues ao usuário final. Para isso, adotou-se uma abordagem que combinou testes funcionais, testes de usabilidade e validação das recomendações geradas.

Inicialmente, foram realizados testes funcionais para assegurar que todos os fluxos principais do sistema — como o envio das respostas, o processamento e a geração das recomendações — estivessem operando corretamente e de forma coesa. Esses testes seguiram a perspectiva de caixa-preta, que verifica o comportamento do sistema a partir das entradas e saídas esperadas, sem considerar sua lógica interna. Foram definidos testes com diferentes perfis de usuários simulados, variando critérios como nível de experiência, áreas de interesse e preferências tecnológicas. Essa validação envolveu a verificação do comportamento das rotas da *API*, a integridade da base de dados utilizada, a coerência na pontuação e o retorno dos resultados. Durante esse processo, nenhuma falha crítica foi identificada, embora tenham sido observados ajustes necessários em casos de empates na pontuação e na categorização de tecnologias genéricas.

Para complementar os testes funcionais, também foram realizados testes de usabilidade com usuários reais. Para esse fim, organizou-se uma sessão guiada de uso da plataforma, acompanhada da aplicação de um formulário de avaliação. Participaram 32 voluntários de diferentes idades e áreas, que foram convidados a utilizar a plataforma e, em seguida, relatar suas observações quanto à clareza das instruções, facilidade de uso, desempenho, estética da interface e utilidade prática. No geral, as respostas indicaram um alto índice de satisfação, com destaque para a simplicidade da interface e a facilidade de navegação. No entanto, também foram apontadas oportunidades de melhoria, especialmente na personalização dos resultados, na necessidade de descrições mais detalhadas sobre as tecnologias recomendadas e na forma como a plataforma se comunica com o usuário.

Por fim, acompanharam-se métricas operacionais, como o tempo médio de resposta do sistema e a taxa de conclusão do fluxo de uso. De forma geral, os resultados obtidos na fase de testes reais permitiram confirmar que a ferramenta desenvolvida oferece uma base sólida e funcional, com potencial para aplicação prática e expansão. Além disso, demonstraram que a proposta atende a uma demanda real e possui espaço significativo para continuidade e aperfeiçoamento.

5 Resultados

Nesta seção de resultados, analisam-se as respostas obtidas por meio do formulário aplicado aos voluntários após os testes, sendo a maior parte composta por mulheres estudantes ou profissionais da área de tecnologia. O objetivo principal desta etapa foi compreender a percepção real dos usuários quanto à funcionalidade, usabilidade e efetividade do sistema de recomendação, bem como identificar padrões de aceitação e oportunidades de melhorias futuras.

A amostra foi composta por 32 participantes, com idades entre 18 e 34 anos, sendo 68,5% do gênero feminino, como mostra a Figura 7. Sobre o nível de familiaridade com ferramentas de desenvolvimento de *chatbots*, 46,9% classificaram-se como tendo nenhuma familiaridade, 30,4% como básica, 18,7% como intermediária e apenas 4% como avançada, conforme ilustrado na Figura 8. Esses dados iniciais validam o público-alvo da ferramenta e reforçam o contexto adotado no projeto, que busca focar em pessoas em fase inicial de inserção na área da tecnologia.

Distribuição de Gênero dos Participantes

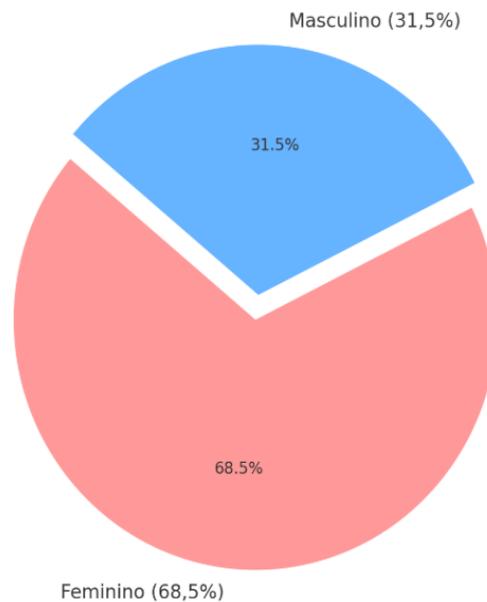


Figura 7 – Distribuição de gênero dos participantes

Fonte: Veloso, Bruna (2025)

Nível de Familiaridade com Desenvolvimento de Chatbots

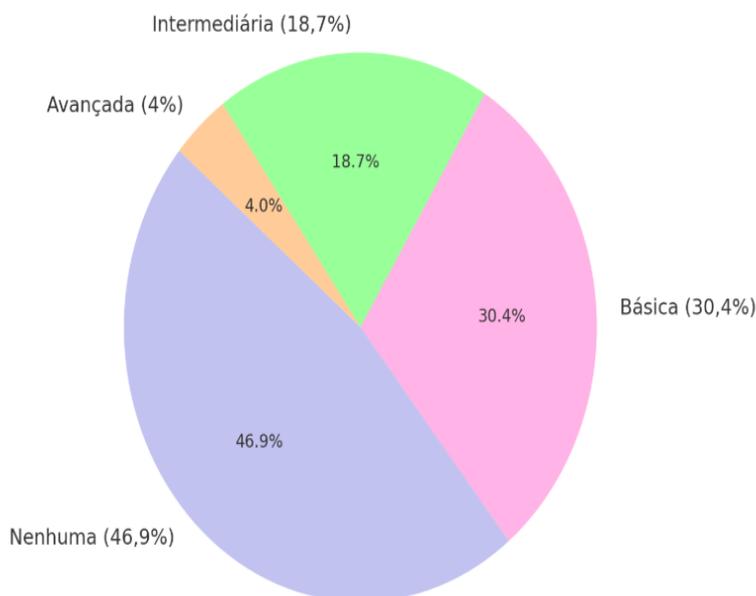


Figura 8 – Nível de familiaridade dos usuários com desenvolvimento de chatbots

Fonte: Veloso, Bruna (2025)

Em relação à avaliação da interface e da usabilidade do sistema, os resultados, em geral, foram positivos, como é possível observar na Figura 9. O atributo “clareza das perguntas” recebeu média de 4,5 (em uma escala de 1 a 5), e a “facilidade de uso” foi avaliada com média de 4,7. Esses dados mostram que a plataforma conseguiu atingir um de seus objetivos principais: ser intuitiva e acessível, mesmo para pessoas com pouca experiência em tecnologia ou desenvolvimento de *software*.

Com relação à utilidade das recomendações, a avaliação também foi positiva, com média de 4,3. Ainda que algumas participantes tenham indicado o desejo de recomendações mais diversas ou adaptadas a nichos específicos, a maioria considerou que as sugestões estavam alinhadas com as características informadas durante a interação. A sensação de personalização foi mencionada por 71,8% dos usuários como “presente” ou “muito presente”, o que valida a forma como o modelo de pontuação adotado na lógica do *back-end* opera.

Quanto ao desempenho da aplicação, a característica “velocidade de carregamento das páginas” recebeu média de 4,5, e a estabilidade do sistema foi mantida durante toda a aplicação dos testes, sem relatos de falhas ou erros. A interface visual da plataforma também foi destacada positivamente, com média de 4,4 na avaliação da aparência da ferramenta. Esse *feedback* demonstra que elementos como mensagens de carregamento, organização das perguntas e retorno estruturado das recomendações contribuíram para uma experiência mais fluida e confiável para o usuário.

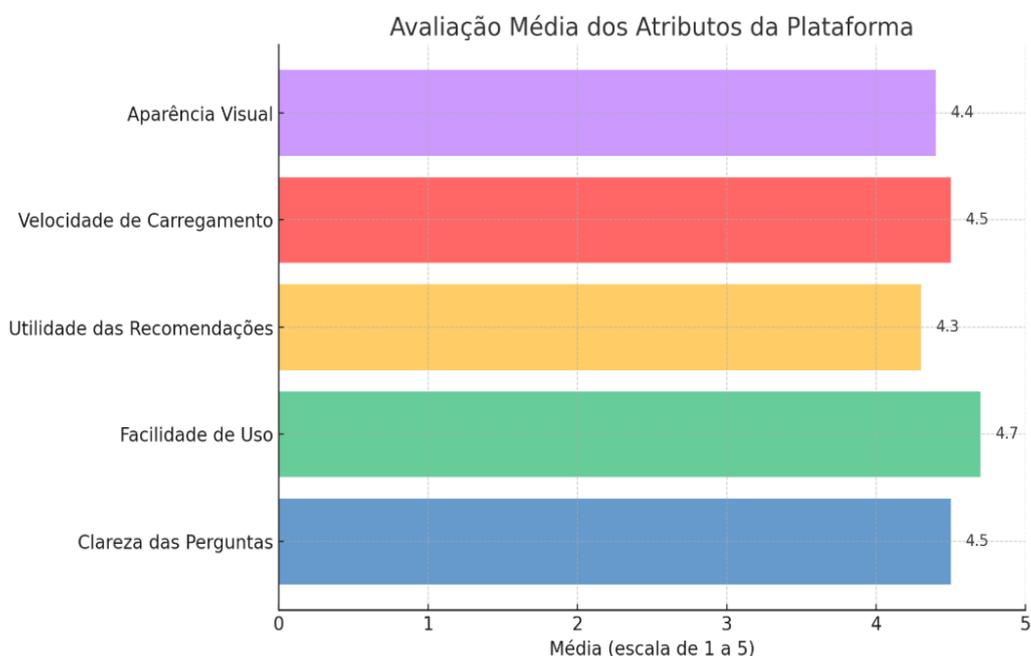


Figura 9 – Avaliação média dos atributos da plataforma

Fonte: Veloso, Bruna (2025)

Além das seções de múltipla escolha, o formulário incluiu uma seção aberta para comentários. Nela, foram destacados alguns diferenciais da plataforma, como a simplicidade das perguntas, a linguagem e explicações acessíveis, o incentivo a quem está começando e o *design* agradável e limpo. Também surgiram sugestões, como a inclusão de exemplos práticos de aplicações com as tecnologias recomendadas e a possibilidade de tornar os resultados mais personalizados, com atributos adicionais (como tempo de aprendizado ou foco em projetos pessoais versus profissionais).

A intenção de uso futuro da plataforma atingiu a média de 84,3%, demonstrando ainda mais o potencial da solução como ferramenta de apoio à tomada de decisão — especialmente em contextos de ensino, suporte profissional e capacitação de iniciantes interdisciplinares na área de tecnologia. Esse índice de aceitação reforça a aplicabilidade da solução no mundo real, especialmente se forem mantidos os ciclos contínuos de melhorias e desenvolvimento, com possibilidade de integração com algoritmos de aprendizado de máquina.

Em resumo, os resultados alcançados reforçam que a ferramenta *Neko* cumpre seu objetivo principal de ser uma solução acessível, funcional e personalizada para recomendação de ferramentas de desenvolvimento. Os dados coletados não apenas validam as decisões tomadas durante as etapas de planejamento e implementação, como também indicam direções para futuras expansões e melhorias.

5.0.1 Discussão Sobre os Resultados e Trabalhos Futuros

A plataforma Neko representa uma contribuição relevante ao propor uma solução acessível voltada a iniciantes que enfrentam dificuldades na escolha de tecnologias para o desenvolvimento de *chatbots*. A proposta auxilia no processo de tomada de decisão técnica por meio de recomendações personalizadas, baseadas em critérios como custo, facilidade de uso, escalabilidade e possibilidades de integração. Com isso, a ferramenta não apenas orienta tecnicamente os usuários, como também promove a democratização do acesso à informação tecnológica, podendo ser utilizada como um recurso complementar em contextos de ensino e capacitação profissional.

Outro ponto de destaque é o modo como a plataforma aproxima a linguagem acadêmica da realidade prática do mercado, contribuindo para uma experiência mais autônoma, educativa e alinhada às necessidades reais dos usuários. Ao considerar perfis diversos, especialmente de pessoas que estão iniciando na área de tecnologia, a plataforma reforça sua proposta de inclusão digital e formação orientada à prática.

Apesar dos resultados positivos alcançados, o projeto ainda apresenta algumas limitações que merecem ser consideradas. A primeira refere-se à base de dados reduzida, uma vez que o número de tecnologias cadastradas ainda é limitado, o que pode restringir a diversidade e abrangência das recomendações geradas. Além disso, a lógica atual de recomendação é simplificada, baseada em critérios fixos e pesos definidos estaticamente, o que pode comprometer a personalização em cenários mais complexos. Outro ponto é a ausência de funcionalidades como autenticação de usuários e armazenamento de histórico, o que dificulta o acompanhamento contínuo da experiência do usuário ao longo do tempo.

Diante dessas limitações, algumas direções para trabalhos futuros já foram identificadas. Entre elas, destaca-se a possibilidade de incorporar algoritmos de inteligência artificial com o intuito de refinar o processo de recomendação, permitindo análises mais aprofundadas a partir de perfis e comportamentos anteriores. Também se planeja expandir a base de dados, tanto em quantidade de tecnologias quanto na diversidade dos critérios avaliados, possibilitando recomendações mais robustas e alinhadas a diferentes contextos de uso.

Além disso, está prevista a criação de uma área personalizada para o usuário, com funcionalidades como login, histórico de interações e armazenamento de sugestões salvas. Essa funcionalidade poderá ser complementada pela geração de relatórios personalizados, fornecendo ao usuário um guia prático para orientar os próximos passos em seus projetos. Por fim, considera-se a reformulação da interface para uma experiência totalmente conversacional, baseada em *chatbots* com técnicas de Processamento de Linguagem Natural (PLN), a fim de tornar a navegação mais fluida, natural e interativa.

Essas perspectivas futuras apontam para o contínuo aprimoramento da plataforma, reforçando seu potencial como uma ferramenta educacional e profissional alinhada com os desafios atuais da inclusão digital e da escolha consciente de tecnologias no cenário da inteligência artificial.

5.1 Conclusão

O presente trabalho teve como objetivo principal o desenvolvimento da ferramenta *Neko*, pensada para oferecer uma plataforma voltada ao apoio na escolha de tecnologias para o desenvolvimento de *chatbots*, especialmente para pessoas ou grupos que não possuem formação ou atuação na área de tecnologia ou que estão em estágio inicial. Por meio de uma metodologia estruturada em levantamento de requisitos, curadoria das tecnologias utilizadas atualmente no mercado e implementação focada na usabilidade e experiência do usuário, tornou-se possível a construção de uma solução funcional e eficiente.

A abordagem adotada ao longo do projeto demonstrou-se eficaz para a criação de um sistema com simplicidade de uso e boa performance, além de viabilizar a implementação de uma lógica de recomendação baseada em critérios relevantes para o contexto dos usuários. Isso foi possível por meio da aplicação de formulários para coleta de *feedback* e testes com usuários, os quais proporcionaram dados importantes para a avaliação do projeto e o aprimoramento contínuo da plataforma, permitindo ajustes na lógica de pontuação e melhorias na interface.

Ao analisarmos os resultados obtidos, é possível perceber um alto índice de aceitação e satisfação por parte das pessoas que participaram dos testes, principalmente em relação à clareza das perguntas, à sensação de personalização nas recomendações e à facilidade de navegação na plataforma. Esses elementos comprovam a viabilidade da proposta da ferramenta e a consistência da execução do projeto, evidenciando que é possível unir soluções digitais de apoio à aprendizagem tecnológica que sejam simples, mas também relevantes para o contexto.

Além disso, o uso de tecnologias acessíveis como *HTML*, *CSS*, *JavaScript*, *Node.js*, *Express* e a integração com o *Google Sheets* via *API* reforça a viabilidade técnica da construção de aplicações úteis mesmo com recursos limitados. A escolha das tecnologias, aliada à organização do projeto por meio de ferramentas como *Trello*, *GitHub* e *Google Forms*, contribuiu para uma gestão eficiente e para o desenvolvimento de um produto final com boa relação entre custo, tempo e impacto.

Em conclusão, a plataforma *Neko* cumpre seu papel como uma solução orientada por dados e validada por usuárias reais. Os aprendizados obtidos durante todo o processo abrem espaço para desdobramentos futuros, como a integração com sistemas educacionais, a ampliação do escopo de recomendações e o uso de técnicas mais avançadas de personalização, como aprendizado de máquina. Dessa forma, este trabalho contribui para a discussão sobre acessibilidade tecnológica, apoio à formação de novos profissionais e desenvolvimento de sistemas de recomendação aplicados a contextos educacionais e profissionais.

Referências

- ADAMOPOULOU, E.; MOUSSIADES, L. Chatbots: History, technology, and applications. *Machine Learning with Applications*, v. 2, p. 100006, 2020. Citado 2 vezes nas páginas 19 e 20.
- BOTPRESS. *Botpress Documentation*. 2025. Disponível em: <https://botpress.com/docs/>. Acesso em: 10 mai. 2025. Citado na página 21.
- BROWN, T. B. et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, v. 33, 2020. Disponível em: <<https://arxiv.org/abs/2005.14165>>. Citado na página 18.
- CHATTERJEE, S.; MAZUMDAR, R.; DAS, D. Chatbot: Architecture, design, and future applications. In: *Proceedings of the 2nd International Conference on Advanced Computational and Communication Paradigms (ICACCP)*. [S.l.]: IEEE, 2021. p. 1–6. Citado na página 24.
- CHUDLEIGH, S. *Principais estatísticas do Chatbot para 2025: Percepções, crescimento do mercado, tendências*. [S.l.], 2024. Citado na página 15.
- CLEMENTS, P. C.; KAZMAN, R.; KLEIN, M. H. *Evaluating Software Architectures: Methods and Case Studies*. [S.l.]: Addison-Wesley, 2012. ISBN 978-0321154958. Citado na página 25.
- DAGKOULIS, N.; MOUSSIADES, L. A comparative evaluation of chatbot development platforms. In: *Proceedings of the 13th International Conference on Information, Intelligence, Systems and Applications (IISA)*. [S.l.]: IEEE, 2022. p. 1–6. Citado na página 28.
- Diógenes. *Chatbots: futuro da comunicação digital*. 2025. Disponível em: <<https://robbu.global/chatbots-futuro-da-comunicacao-digital/>>. Citado na página 13.
- Google Cloud. *Dialogflow documentation*. 2024. Acesso em 2024. Disponível em: <<https://cloud.google.com/dialogflow>>. Citado na página 20.
- Juliana Theodoro. *Meios de comunicação: o que são e evolução (dos antigos aos atuais)*. 2025. Disponível em: <<https://www.significados.com.br/meios-de-comunicacao/>>. Citado na página 13.
- JURAFSKY, D.; MARTIN, J. H. *Speech and Language Processing*. 3. ed. [S.l.]: Prentice Hall, 2020. Manuscript draft, available online at <https://web.stanford.edu/~jurafsky/slp3/>. Citado na página 24.
- JURAFSKY, D.; MARTIN, J. H. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. 3. ed. [s.n.], 2020. Draft version. Disponível em: <<https://web.stanford.edu/~jurafsky/slp3/>>. Citado 2 vezes nas páginas 17 e 19.

- KELKAR, A. et al. Comparative study on chatbot frameworks. In: *2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS)*. [S.l.]: IEEE, 2023. p. 1766–1772. Citado na página 28.
- KLOPFENSTEIN, L. C. T. et al. The rise of bots: A survey of conversational interfaces, patterns, and paradigms. *Proceedings of the 2017 Conference on Designing Interactive Systems*, p. 555–565, 2017. Citado na página 18.
- KVALE, S.; BRINKMANN, S. *InterViews: Learning the Craft of Qualitative Research Interviewing*. 3. ed. [S.l.]: SAGE Publications, 2018. Citado na página 23.
- MCTEAR, M. F.; CALLEJAS, Z.; GRIOL, D. *The Conversational Interface: Talking to Smart Devices*. [S.l.]: Springer, 2016. ISBN 978-3-319-32966-4. Citado na página 22.
- Microsoft. *Bot Framework Documentation*. 2023. Acesso em 2023. Disponível em: <<https://learn.microsoft.com/en-us/azure/bot-service/bot-builder-overview>>. Citado na página 20.
- RADZIWILL, N. M.; BENTON, M. C. Evaluating quality of chatbots and intelligent conversational agents. *arXiv preprint arXiv:1704.04579*, 2017. Citado na página 23.
- RAI, J. H.; BAGDE, P. O. Building chatbots: A guide to frameworks and platforms. In: *AIP Conference Proceedings*. [S.l.]: AIP Publishing, 2024. v. 3180, p. 020012. Citado na página 27.
- RAYMOND, E. S. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. [S.l.]: O'Reilly Media, 1999. ISBN 978-0596001087. Citado na página 24.
- SERVICES, A. W. *Amazon Lex Documentation*. 2025. Disponível em: <https://docs.aws.amazon.com/lex/>. Acesso em: 10 mai. 2025. Citado na página 21.
- SHARMA, R.; JOSHI, M. An analytical study and review of open source chatbot framework, rasa. *International Research Journal of Engineering and Technology (IRJET)*, v. 7, n. 8, p. 3779–3783, 2020. Disponível em: <<https://www.irjet.net/archives-V7/i8/IRJET-V7I8752.pdf>>. Citado na página 28.
- SHIHAB, E. et al. A comparison of natural language understanding platforms for chatbots in software engineering. *Empirical Software Engineering*, Springer, v. 26, n. 5, p. 1–30, 2021. Citado na página 28.
- STALLINGS, W. *Cryptography and Network Security: Principles and Practice*. 7. ed. [S.l.]: Pearson, 2017. ISBN 978-0134444284. Citado na página 25.
- TURING, A. M. Computing machinery and intelligence. *Mind*, LIX, n. 236, p. 433–460, 1950. Citado na página 17.
- WEIZENBAUM, J. Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, v. 9, n. 1, p. 36–45, 1966. Citado na página 17.



TERMO DE AUTORIZAÇÃO PARA PUBLICAÇÃO DIGITAL NA BIBLIOTECA “JOSÉ ALBANO DE MACEDO”

Identificação do Tipo de Documento

- Tese
- Dissertação
- Monografia
- Artigo

Eu, **Bruna Heloisa Feitosa Veloso**, autorizo com base na Lei Federal nº 9.610 de 19 de Fevereiro de 1998 e na Lei nº 10.973 de 02 de dezembro de 2004, a biblioteca da Universidade Federal do Piauí a divulgar, gratuitamente, sem ressarcimento de direitos autorais, o texto integral da publicação: Plataforma Inteligente para Auxílio na Escolha de Tecnologias para Desenvolvimento de Chatbots, de minha autoria, em formato PDF, para fins de leitura e/ou impressão, pela internet a título de divulgação da produção científica gerada pela Universidade.

Picos-PI, 07 de Julho de 2025.

Documento assinado digitalmente
gov.br BRUNA HELOISA FEITOSA VELOSO
Data: 07/07/2025 22:30:05-0300
Verifique em <https://validar.iti.gov.br>

Assinatura