

Assistente de Atendimento aos Usuários do Sistema de Certificados do IFBA Campus Vitória da Conquista

Edvaldo Moreira Lemos Júnior¹, Pablo Freire Matos²

¹Discente de BSI, ²Docente IFBA

Instituto Federal de Educação, Ciência e Tecnologia da Bahia (IFBA)
Av. Sérgio Vieira de Mello, 3.150, Zabelê, 45.078-900 - Vitória da Conquista - BA - Brasil

{emljunior@yahoo.com, pablofmatos@gmail.com}

Abstract. *This paper proposes the development of a Chatbot to assist users of the Certificate System at IFBA - Campus Vitória da Conquista. The growing demand for digital services and the need to enhance user interaction highlight the importance of implementing this solution. The chatbot was developed using artificial intelligence, machine learning, and natural language processing techniques to provide efficient and automated support. This study includes a theoretical review of the technologies involved and discusses potential enhancements for future work. The central hypothesis is that integrating the chatbot into the system will significantly improve user experience and streamline administrative processes.*

Resumo. *Este artigo propõe o desenvolvimento de um Chatbot para auxiliar os usuários do Sistema de Certificados do IFBA Campus Vitória da Conquista. A crescente demanda por serviços digitais e a necessidade de melhorar a interação com os usuários justificam a implementação dessa solução. O Chatbot foi desenvolvido utilizando técnicas de inteligência artificial, aprendizado de máquina e processamento de linguagem natural. O projeto apresenta uma revisão teórica sobre as tecnologias envolvidas, além da possibilidade de melhorias por meio de trabalhos futuros. A hipótese central é que a utilização do Chatbot melhore significativamente a interação entre os usuários e os administradores do sistema.*

1. Introdução

A crescente demanda por serviços nos diversos setores têm feito as empresas modernas se reinventarem constantemente, principalmente para o atendimento aos seus clientes que, por sua vez, exigem respostas rápidas, eficazes e disponibilidade de atendimento no seu tempo. Para melhorar essa interação com os usuários de seus sistemas, o uso da tecnologia é cada vez mais empregada no dia a dia não apenas nas grandes corporações, mas, também, em instituições diversas de tamanhos e objetivos variados.

Atualmente a tecnologia mais procurada para uma interação automatizada com usuários de serviços é o uso de assistentes digitais ou robôs de atendimento (*Chatbots*). Esses robôs são algoritmos de computador programados para dialogar com usuários e de acordo com Miklosik, Evans e Qureshi (2021), é “um software que aceita linguagem natural como entrada e gera linguagem natural como saída, engajando-se em uma conversa”.

Segundo o Mapa do Ecossistema Brasileiro de Bots [Paiva 2021], a adoção desse tipo de tecnologia no Brasil vem crescendo expressivamente para diversas finalidades: 65% atendimento ao cliente, 8% apoio ao *back-office*, 7% cobrança, 6% vendas, 3% *marketing* e 11% para outras finalidades. Os *Chatbots* podem desempenhar um papel indispensável na transformação digital de diversas áreas do negócio, trazendo benefícios não apenas para o atendimento ao cliente, mas também para estratégias de *marketing* [Miklosik, Evans e Qureshi 2021]. No mundo, segundo dados de Shukairy (2020), 67% dos consumidores usaram *Chatbot* para suporte ao cliente em 2019 gerando uma economia de aproximadamente 30% no custo de suporte ao cliente. Ainda de acordo com esta pesquisa, 40% dos consumidores não se importam se um *Chatbot* ou um ser humano os ajudem, desde que recebam o suporte necessário.

Diante dos benefícios proporcionados pela tecnologia de assistentes virtuais, este projeto propõe o desenvolvimento de um *Chatbot* para atender os usuários do Sistema de Certificados do IFBA, *Campus Vitória da Conquista* [Ramos *et al.* 2018]. O *Chatbot* será integrado ao sistema já existente o qual possui uma comunidade ativa e crescente – no início de 2025, registrava mais de 3.000 usuários em 74 eventos, com aproximadamente 9.000 certificados emitidos. Esse cenário ressalta a necessidade de um assistente virtual para otimizar o atendimento e melhorar a experiência dos usuários.

Por se tratar de uma pesquisa que sugere uma solução para um problema específico, segundo Gil (2010), esta é classificada como pesquisa aplicada. Também, por ser realizada em ambiente controlado, pode-se classificá-la quanto à sua estratégia de coleta de dados como pesquisa de laboratório, segundo os conceitos definidos por Appolinário (2012).

Para a elaboração deste estudo, foram coletadas informações específicas sobre *Chatbots*, suas aplicações, a tecnologia mais adequada a ser empregada e estudos correlatos. Em seguida, o *Chatbot* foi desenvolvido. Na etapa de desenvolvimento, além do levantamento teórico, utilizou-se como estudo de caso o Sistema de Certificados [Ramos *et al.* 2018], sendo realizado em duas camadas: *back-end* e *front-end*.

O restante deste artigo está organizado da seguinte forma: Na Seção 2 é sintetizado os conceitos fundamentais das principais tecnologias para a compreensão deste trabalho. Na Seção 3 são apresentados os trabalhos correlatos semelhantes ao proposto nesta pesquisa. Na Seção 4 é demonstrado o desenvolvimento do trabalho e as ferramentas utilizadas. Por fim, na Seção 5, são apresentadas as considerações finais.

2. Fundamentação Teórica

Nesta seção é apresentada a fundamentação teórica necessária para a compreensão do desenvolvimento do trabalho. Na Subseção 2.1 é apresentado o conceito de inteligência artificial, seguido pelo aprendizado de máquina na Subseção 2.2 e pelo processamento de linguagem natural na Subseção 2.3. Na Subseção 2.4 é explicado sobre o conceito de *Chatbot*. Por fim, na Subseção 2.5 é feito um comparativo entre *Chatterbot* e *Chatbot*.

2.1. Inteligência Artificial

Ao longo do tempo a inteligência artificial foi descrita de forma diferente [Grewal 2014], variando de acordo às aplicações mecânicas. Segundo Sage (1990), a Inteligência Artificial (IA) concentra-se no desenvolvimento de paradigmas ou algoritmos que requeram máquinas para realizar tarefas cognitivas e que sejam capazes de armazenar

conhecimento, aplicar o conhecimento armazenado e adquirir novo conhecimento através da experiência.

Os avanços da IA tiveram início nos anos 1950 com a criação dos computadores eletrônicos propostos por John Von Neumann. Alan Turing já previa que essas máquinas poderiam interagir com humanos. Até a década de 1980, a pesquisa em IA seguia dois paradigmas: a IA forte, que buscava executar qualquer tarefa intelectual humana [Jones 2021], e a IA fraca, que não considerava a capacidade de aprendizado das máquinas [Costa 2020]. Ainda nos anos 1980, surgiu o conceito de Aprendizado de Máquina, ampliando o campo da IA, que também engloba o Processamento de Linguagem Natural.

2.2. Aprendizado de Máquina

O Aprendizado de Máquina (AM) está presente em muitas operações computacionais do cotidiano, seja numa simples consulta em sites de busca de Internet ou mesmo em *softwares* específicos para diagnóstico de doenças. Ao utilizar as técnicas apropriadas de AM, corporações podem prever mudanças significativas no seu modelo de negócio e agir antecipadamente para obter melhores resultados. O AM é uma forma de IA que permite que um sistema aprenda com os dados, ao invés de programação explícita [Hurwitz e Kirsch 2018].

Há diferentes técnicas que podem ser empregadas na aprendizagem de máquina que são importantes para melhorar a precisão dos modelos preditivos. As principais são: aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço. Na aprendizagem supervisionada o algoritmo recebe um conjunto de treinamento, composto por uma tabela de dados contendo atributos e suas respectivas classes. Esse treinamento permite que o modelo aprenda padrões e, posteriormente, classifique corretamente novos dados com base nesses atributos. Na aprendizagem não supervisionada, o algoritmo recebe os dados sem rótulos, agrupando-os em grupos (*clusters*) pela similaridade dos atributos. Em seguida, analisa os grupos e adiciona rótulos aos dados. A aprendizagem por reforço difere dos dois anteriores porque o algoritmo não é treinado com um conjunto de dados predefinido, mas sim aprende por tentativa e erro, ajustando suas ações com base em recompensas e penalidades.

Para Hurwitz e Kirsch (2018), algoritmos são conjuntos de instruções que orientam um computador na interação, manipulação e transformação de dados. A inexistência de um algoritmo universal capaz de resolver todos os problemas torna essencial a escolha do método mais adequado para alcançar os resultados desejados. Embora essa escolha possa implicar um custo computacional significativo, os benefícios podem justificar o investimento.

Alguns tipos de algoritmos usados em aprendizagem de máquinas são: Bayesiano: Muito utilizado na categorização de textos baseados na frequência de palavras, além de não necessitar de uma grande quantidade de dados de teste para obter resultados classificados precisos. Agrupamento (*clustering*): É do tipo não supervisionado porque não há rótulo nos dados. O Algoritmo *k-Nearest Neighbors* se destaca por agrupar novos dados classificando-os por similaridade mais próxima. Árvore de decisão: Esse tipo de algoritmo ilustra os resultados de uma decisão por meio de ramificações, em que cada nó corresponde a um resultado possível. Os algoritmos de árvore de decisão são capazes de fazer previsões de forma rápida e precisa.

2.3. Processamento de Linguagem Natural

O Processamento de Linguagem Natural (PLN) é a principal forma que os computadores usam para interpretar textos e fala. Vespoor e Cohen (2013) definem que PLN é a análise de dados linguísticos, mais comumente na forma de dados textuais, objetivando a construção de uma representação do texto que acrescente estrutura à linguagem natural não estruturada. Portanto, PLN é de fundamental importância no uso em aplicações de tradução e interpretação de textos, busca de informações em documentos e na interface homem-máquina, mais especificamente no uso em *Chatbot*.

Os aspectos da pesquisa em PLN envolvem estruturas essenciais da linguagem humana, como o som (fonologia), a estrutura (morfologia e sintaxe) e o significado (semântica e pragmática). Além disso, a tokenização desempenha um papel importante na segmentação de palavras, delimitando caracteres e estabelecendo seus limites. A Figura 1 ilustra o processo de análise do PLN.

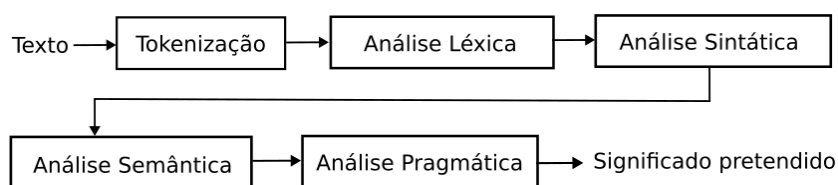


Figura 1: Análise do PLN. Fonte: Adaptado de Dale (2010).

2.4. Chatbot

Em 1966, Joseph Weizenbaum (1923-2008) desenvolveu um programa de processamento de linguagem natural chamado ELIZA [Weizenbaum 1966]. Segundo Khan e Das (2018), "ELIZA examinava palavras-chave de entrada do usuário e acionava regras de transformação de saída". Dessa forma, além de estabelecer um padrão de comunicação, o programa também serviu de inspiração para o desenvolvimento de outras aplicações de interação com computadores.

Como ilustrado na Figura 2, alguns *Chatbots* são baseados em regras estáticas (a), enquanto outros utilizam Inteligência Artificial (a e b). Além disso, há modelos híbridos que combinam ambas as abordagens (c) [Oliveira 2018].

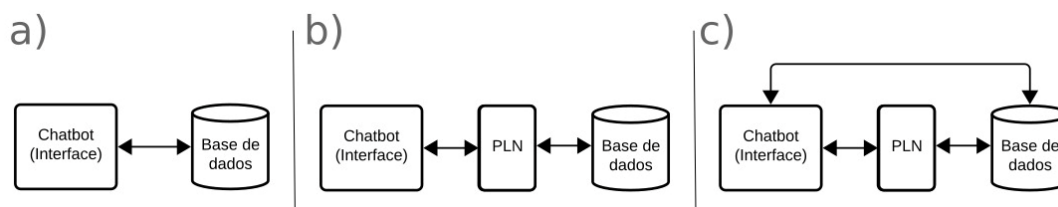


Figura 2. Modelos de Chatbots. Fonte: Próprio Autor (2025).

Chatbots baseados em regras estáticas operam com um conjunto predefinido de perguntas e respostas, seguindo uma estrutura de árvore de decisão. Nessa abordagem, a interação do usuário direciona a conversa para respostas específicas armazenadas em uma base de dados, garantindo maior previsibilidade e controle. Esse modelo é particularmente adequado para aplicações simples e atendimento a perguntas frequentes. *Chatbots* implementados com IA utilizam PLN para compreender melhor as interações e fornecer respostas mais naturais, acessando a base de dados de forma mais inteligente,

aprendendo com novas conversas e aprimorando suas respostas ao longo do tempo. Entretanto, requer um maior recurso computacional e treinamento de dados. Já os *Chatbots* híbridos combinam regras estáticas com inteligência artificial, permitindo que o sistema tenha estrutura e previsibilidade, mas também flexibilidade e aprendizado contínuo. Eles operam tanto por meio de fluxos predefinidos quanto por técnicas de PLN para entender melhor as perguntas dos usuários e fornecer respostas mais precisas. Atualmente, através dos *Chatbots* é possível um contato primário com IA que é cada vez mais implementada em sistemas de interação [Spanhol 2017].

Novas necessidades de interação com o computador têm feito com que desenvolvedores de tecnologia, em especial a de comunicação e interação com máquinas, aperfeiçoem e criem novas bibliotecas específicas a fim de melhorar o desenvolvimento de *Chatbots*. Um exemplo desse tipo de biblioteca é a *Chatterbot* [Navlani 2024], que é projetada especificamente para gerar *Chatbots*. Ela usa uma seleção de algoritmos de aprendizado de máquina para fabricar respostas variadas aos usuários de acordo com suas solicitações, facilitando o desenvolvimento.

Na Figura 3 é ilustrada no diagrama do processo de criação de um *Chatbot*. Nesse processo, o *Chatbot* seleciona a melhor resposta entre diferentes mecanismos de correspondência, garantindo maior precisão nas interações. Para isso, ele avalia diversas opções e escolhe aquela com maior grau de confiança antes de responder ao usuário.

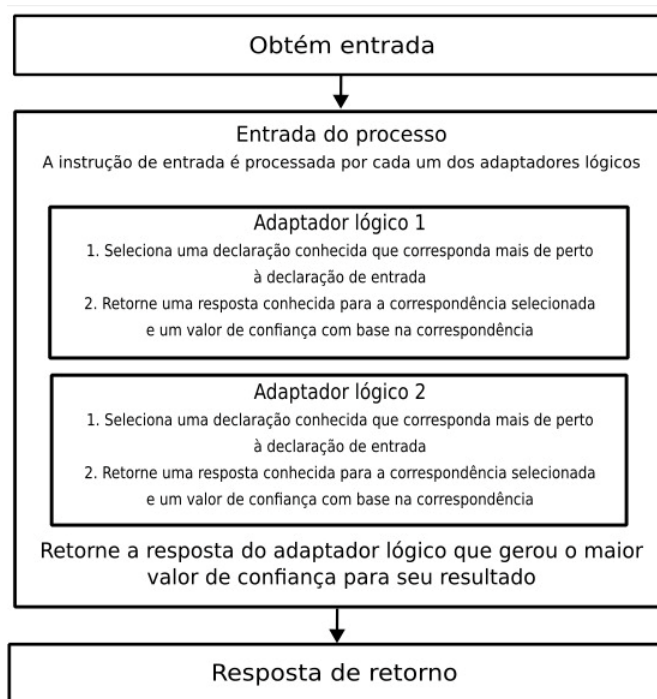


Figura 3: Diagrama do Fluxo do Processo de Criação de um Chatbot. Fonte: Navlani (2024).

Os mecanismos de correspondência são os métodos ou algoritmos utilizados pelos adaptadores lógicos para identificar a melhor correspondência entre a entrada do usuário e as respostas disponíveis. Já os adaptadores lógicos são componentes do *Chatbot* responsáveis por processar a entrada do usuário, aplicar um mecanismo de correspondência e retornar uma resposta acompanhada de um valor de confiança. Um adaptador lógico pode operar com base na similaridade semântica, em regras pré-definidas ou em técnicas de aprendizado de máquina.

Adamopoulou e Moussiades (2020) classificam os *Chatbots* em diferentes tipos: quanto ao domínio do conhecimento, o serviço prestado, os objetivos, o processamento de entrada e o método de geração de resposta, o auxílio humano e o método de construção. Essa classificação se torna importante no projeto de um *Chatbot*, principalmente para classificar o modelo gerador de respostas que pode considerar o nível de ajuda humana predefinido.

2.5. Comparação entre Chatterbot e Chatbot

O *Chatterbot* é uma opção útil quando se deseja um *Chatbot* inteligente, capaz de aprender com as interações. Por outro lado, um *Chatbot* baseado em fluxo é mais adequado para atendimentos estruturados, como no suporte técnico. A Tabela 1 mostra um comparativo das principais características entre um *Chatterbot* e um *Chatbot*.

Ao obedecer a um fluxo definido, um *Chatbot* pode parecer desvantajoso quando comparado com um *Chatterbot*, porém, a escolha entre um ou outro depende do objetivo do sistema e das necessidades do usuário. Cada abordagem possui vantagens e desvantagens, sendo mais adequada para diferentes tipos de aplicações. Portanto, a escolha entre *Chatterbot* e *Chatbot* baseado em fluxo depende do contexto da aplicação.

Se a necessidade for um *Chatbot* dinâmico, capaz de aprender com o tempo e responder a perguntas variadas, o *Chatterbot* é mais vantajoso. Se o objetivo for um atendimento estruturado e previsível, sem risco de respostas inadequadas, o *Chatbot* baseado em fluxo é a melhor opção. Há ainda a possibilidade de combinar ambas as abordagens, formando uma aplicação híbrida, utilizando um *Chatbot* de fluxo para perguntas estruturadas e o *Chatterbot* para perguntas abertas.

Tabela 1. Comparação entre Chatterbot e Chatbot.

Característica	<i>Chatbot</i> Tradicional (Fluxo)	<i>Chatterbot</i> (PLN)
Aprende sozinho?	Não	Sim
Responde perguntas abertas?	Não, apenas perguntas programadas	Sim
Treinamento necessário?	Não, apenas segue um roteiro	Sim, precisa de exemplos
Adaptação a novas perguntas?	Não, só responde perguntas predefinidas	Sim, aprende novas frases

A decisão de utilizar, neste trabalho, um *Chatbot* baseado em PLN foi motivada pela necessidade de oferecer uma experiência mais dinâmica e eficiente aos usuários. Conforme demonstrado na Tabela 1, o *Chatterbot* apresenta vantagens significativas em relação ao *Chatbot* tradicional baseado em fluxo.

3. Trabalhos Correlatos

Com o crescimento das novas tecnologias e plataformas dedicadas à criação de *Chatbots*, o uso desses sistemas tem aumentado significativamente. Embora sejam aplicados para resolver diversos problemas, sua maior demanda está no suporte a aplicações e no atendimento a usuários, especialmente em empresas que atuam no *e-commerce*. Ghidini e Mattos (2018) propuseram o desenvolvimento de uma ferramenta de atendimento automático aos clientes que possa ser inserida no portal de empresas. Ao usar uma base de conhecimento de terceiros, como a plataforma Arisa Nest, a solução proporcionaria maior agilidade nas respostas às dúvidas dos clientes.

Maciel (2019) desenvolveu um robô homônimo para responder perguntas acadêmicas na Universidade Federal do Ceará. Segundo o autor, o robô foi bem aceito pela comunidade, embora exija melhorias contínuas. Já Almeida Júnior (2017) demonstrou que o uso de *Chatbots* vai além das interações comerciais e do suporte a produtos e serviços, abrangendo outras áreas de aplicação. Em seu estudo, o autor desenvolveu um robô denominado Beck, baseado na terapia cognitivo-comportamental, com o objetivo de auxiliar adolescentes com depressão. Os resultados apresentados indicam que o *Chatbot* obteve um desempenho satisfatório.

A Tabela 2 apresenta uma comparação entre o *Chatbot* deste trabalho e os estudos analisados. Os trabalhos analisados demonstraram que os *Chatbots* podem ser aplicados em diversos contextos, como atendimento ao cliente [Ghidini e Mattos 2018] e saúde mental [Almeida Júnior 2017]. O diferencial da solução proposta está na possibilidade de personalização e integração com bancos de dados, tornando-a adaptável a diferentes áreas de aplicação.

Além disso, enquanto alguns *Chatbots* analisados, como o de Ghidini e Mattos (2018), dependem de plataformas externas, a solução proposta possui controle total sobre sua base de conhecimento. A análise dos trabalhos correlatos permitiu contextualizar o *Chatbot* proposto em relação a outras soluções existentes, destacando suas similaridades e diferenças em termos de aplicação, tecnologia e propósito.

Tabela 2. Comparação entre o Chatbot proposto e os Trabalhos Correlatos.

Critério	Almeida Júnior (2017)	Ghidini e Mattos (2018)	Maciel (2019)	Este Trabalho (2025)
Objetivo	Apoio psicológico	Atendimento automatizado	Assistente acadêmico	Atendimento automatizado
Tipo de Chatbot	Regras	Regras	Regras	Híbrido (Regras+PLN)
Base de Conhecimento Própria?	✓ Sim	✗ Não	✓ Sim	✓ Sim
Aprendizado com Interações?	✗ Não	✗ Não	✗ Não	✓ Sim
Foco no Atendimento Direto?	✗ Não	✓ Sim	✓ Sim	✓ Sim
Uso de IA para Respostas?	✗ Não	✗ Não	✗ Não	✓ Sim
Possui Avaliação de Usuários?	✓ Sim	✗ Não	✓ Sim	✗ Não

4. IFBot

Para desenvolver este trabalho, inicialmente, foi feita uma análise da necessidade de um assistente de atendimento virtual para o Sistema de Certificados do IFBA *Campus* Vitória da Conquista, disponível em <http://certificados.conquista.ifba.edu.br/>. Essa análise foi conduzida em conjunto com o coordenador do projeto, que, ao observar o aumento significativo no número de eventos e participantes, sugeriu a implementação de um *Chatbot* (Gráfico 1).

O objetivo desse assistente virtual é otimizar o suporte aos usuários do sistema, automatizando as respostas às solicitações mais frequentes, que incluem: (i) A verificação do cadastro do participante de um evento na base de dados; (ii) A possibilidade de solicitação de alteração de algum dado cadastrado, por exemplo, o e-mail; e (iii) Verificar se possui certificados gerados para algum evento. Por questões de segurança, nem o usuário participante nem o *Chatbot* realizam alterações diretas na base de dados. Caso

seja necessário, o assistente coletará as informações requeridas, incluindo um documento oficial com foto e enviará um e-mail ao administrador do sistema para verificação e possível contato com o participante.

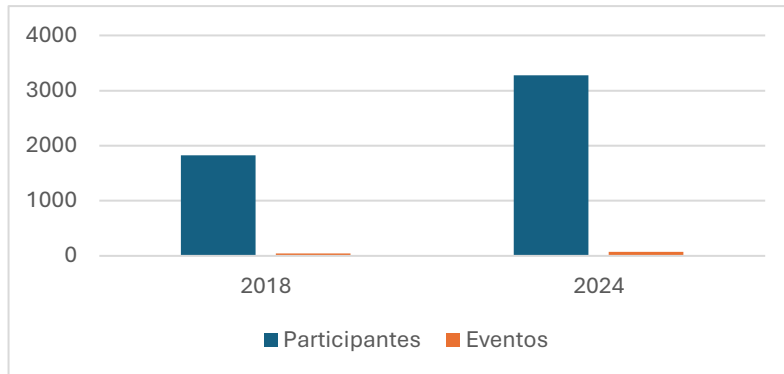


Gráfico 1: Eventos e Participantes de 2018 a 2024. Fonte: Próprio Autor (2025).

A Subseção 4.1 apresenta os casos de uso da aplicação. Em seguida, a Subseção 4.2 trata da análise do fluxo das funções preestabelecidas, enquanto a Subseção 4.3 aborda a arquitetura do sistema. A Subseção 4.4 descreve as tecnologias utilizadas, e a Subseção 4.5 apresenta parte do código de integração entre *back-end* e *front-end*. Por fim, a Subseção 4.6 exibe a interface do *Chatbot* desenvolvido.

4.1. Casos de Uso

Os casos de usos são representações gráficas das funcionalidades de um sistema sob a perspectiva do usuário, descrevendo interações entre as funcionalidades e o usuário (ator). É uma ferramenta de extrema importância para o entendimento das funcionalidades do sistema, garantindo os objetivos de cada ator de forma clara e simples.

A Figura 4 apresenta o diagrama de caso de uso do *Chatbot*, delineando o ator usuário, suas respectivas funções e interações com o sistema. O ator usuário representa qualquer indivíduo que interaja com o *Chatbot*, podendo realizar solicitações relacionadas a cadastros, regularização de dados incompletos ou ausência de certificados de eventos.

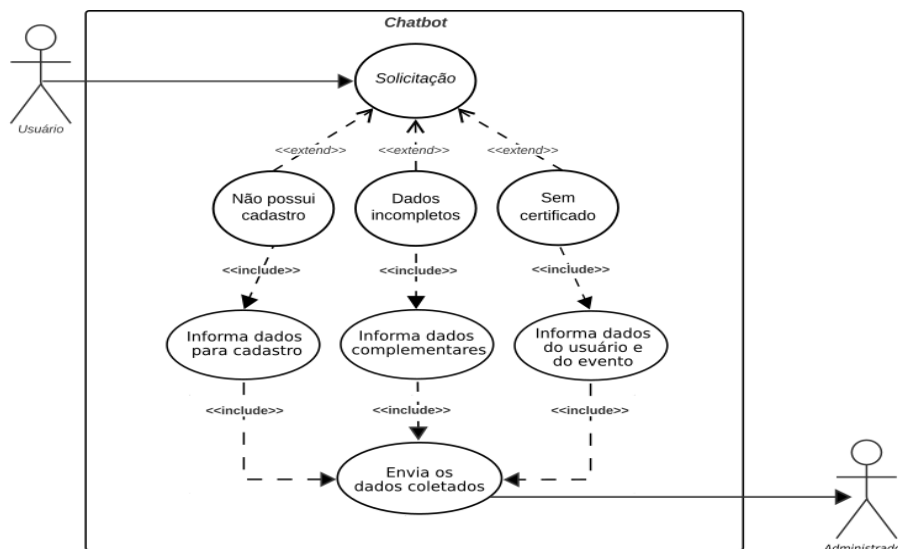


Figura 4: Casos de Uso. Fonte: Próprio Autor (2025).

Além disso, o usuário também fornece informações ao *Chatbot* quando solicitado, estabelecendo uma comunicação bidirecional. Após o fornecimento das informações, os dados são enviados/encaminhados ao administrador do sistema, responsável por analisar, validar e, se necessário, realizar as devidas alterações conforme as solicitações dos usuários. Dessa forma, o administrador atua como um recurso complementar para demandas que exigem intervenção humana.

4.2. Análise dos Fluxos

Nesta subseção será feita uma análise das três opções disponíveis para consulta direta na tela inicial do *Chatbot*, cada uma caracteriza um caso de uso específico. Os casos de uso apresentados na Figura 4 demonstram como as funções predefinidas do *Chatbot* atuam em diferentes cenários de atendimento ao usuário. O fluxo de cada opção segue uma lógica estruturada que permite a verificação, a validação e o envio das informações coletadas para o administrador do sistema analisar e, se necessário, corrigi-las, garantindo um atendimento automatizado.

A função da primeira opção tem a responsabilidade de solicitar informações ao usuário e armazená-las para que o administrador possa verificar, na base de dados, possíveis erros no cadastro. Ela valida se o CPF e a data de nascimento estão registrados no banco de dados. Caso os dados não sejam encontrados, o *Chatbot* solicita informações complementares e as registra, seguindo o fluxo de execução apresentado no Fluxo 1.

Fluxo 1: Execução da Opção 1.

1. O *Chatbot* solicita ao usuário que informe seu CPF;
2. O CPF é verificado no banco de dados:
 - Se o CPF existir, o *Chatbot* solicita a data de nascimento e verifica a correspondência;
 - Caso contrário, o *Chatbot* informa ao usuário e solicita o preenchimento manual dos dados.
3. Caso o CPF e a data de nascimento sejam encontrados, o *Chatbot* confirma os dados e finaliza o processo;
4. Se o CPF e a data de nascimento não forem encontrados, o *Chatbot* solicita os seguintes campos: Nome completo, telefone e e-mail;
5. Após a coleta dos dados, o *Chatbot* exibe um resumo das informações e habilita a opção de envio desses dados por e-mail.

O segundo caso de uso é a validação e atualização de cadastro. Esta função permite que o *Chatbot* valide dados cadastrais e, se necessário, colete dados do usuário para atualização junto ao administrador do sistema. O fluxo inicia com a verificação do CPF e segue com a confirmação da data de nascimento. O Fluxo 2 exibe a execução dessa escolha.

Fluxo 2: Execução da Opção 2.

1. O *Chatbot* solicita o CPF do usuário;
2. O *Chatbot* consulta o CPF no banco de dados e encerra a sessão caso não o encontre;
3. Se o CPF for encontrado, o *Chatbot* solicita a data de nascimento:
 - Se a data de nascimento corresponder ao CPF, o *Chatbot* confirma os dados e finaliza o processo;
 - Caso contrário, o *Chatbot* pergunta ao usuário se deseja atualizar o cadastro.
4. Caso o usuário opte por atualizar os dados, o *Chatbot* solicita: data de nascimento, nome completo, RG, telefone e e-mail;
5. Por fim, o *Chatbot* exibe o novo cadastro e habilita a opção de envio dos dados coletados por e-mail.

Por fim, a opção 3, permite que o *Chatbot* consulte eventos nos quais o usuário participou. Se nenhuma participação for encontrada, são coletados dados do usuário e enviados para o administrador do sistema para análise. O Fluxo 3 mostra a execução desta função.

Fluxo 3: Execução da Opção 3.

1. O *Chatbot* solicita ao usuário que informe o CPF e a data de nascimento;
2. O *Chatbot* verifica no banco de dados se os dados correspondem a um usuário cadastrado:
 - Se não encontrar o usuário, informa ao usuário e encerra a sessão.
3. Se o usuário for encontrado, o *Chatbot* busca eventos cadastrados na tabela de participações:
 - Se houver eventos, o *Chatbot* exibe a lista de eventos com data de início, data de fim e nome do evento;
 - Caso contrário, o *Chatbot* solicita os dados de um novo evento.
4. Caso o usuário precise informar um novo evento, o *Chatbot* solicita os seguintes dados: Nome e ano do evento, telefone de contato e e-mail;
5. Em seguida, o *Chatbot* exibe um resumo das informações e habilita a opção de envio dos dados coletados por e-mail.

4.3. Arquitetura do Sistema

A Figura 5 demonstra a arquitetura da aplicação, dividida em módulos que interagem entre si à medida que sua respectiva função é requisitada. O módulo de interface (*front-end*) é a parte visível para o usuário e responsável por mostrar as funcionalidades. Através dele é possível interagir com o sistema, fazer perguntas, ver as respostas e anexar documentos. O *back-end* controla as interações do *Chatbot*, lidando com as três opções predefinidas e distintas de perguntas ao usuário, bem como as perguntas aleatórias, verifica informações no banco de dados e, ao final, permite o envio dos dados coletados para o e-mail do administrador. Os módulos *front-end* e *back-end* foram desenvolvidos utilizando, respectivamente, JavaScript/HTML e Python.

O módulo de PLN é responsável por aprender a partir das interações, armazenando pares de perguntas e respostas por meio da biblioteca *Chatterbot*. Essa biblioteca realiza buscas no banco de dados de conhecimento e gera respostas com base em novas interações. Já o módulo de Banco de Dados armazena informações relevantes sobre os eventos, incluindo dados dos usuários, certificados e outros registros.

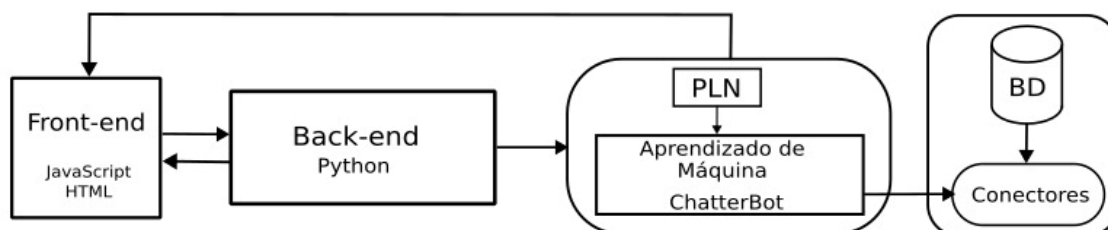


Figura 5. Arquitetura da Aplicação. Fonte: Próprio Autor (2025).

4.4. Tecnologias Utilizadas

Depois de pesquisar sobre as tecnologias para criação de *Chatbot* disponíveis no mercado, foram escolhidas as linguagens Python, JavaScript e HTML, bem como as bibliotecas *Chatterbot* e Flask para gerenciar as interações com o *Chatbot*. O motivo de escolha dessas linguagens se deu por serem de fácil acesso, amplamente conhecidas e testadas e

de uso gratuito, o que permite desenvolver e implementar sistemas a um custo relativamente baixo.

Python é a linguagem utilizada no *back-end*, a qual é responsável pela lógica, processamento dos dados e conexão com o banco de dados para em seguida retornar uma resposta. JavaScript é usada no *front-end* e responsável pela interatividade da página, modificando elementos do HTML sem a necessidade de recarregar uma página, capturando a mensagem do usuário e enviando para o *back-end*. HTML, *HyperText Markup Language*, também usada no *front-end*, define a estrutura das páginas web permitindo a organização de elementos na tela.

Essas ferramentas trabalhando juntas permitem um fluxo de interação coeso para um sistema que necessita de respostas estruturadas e precisas como um *Chatbot*.

4.5. Integração entre Back-end e Front-end

No *back-end* há uma rota para executar o arquivo HTML logo que inicia a aplicação. O HTML, por sua vez, também tem um script para chamar o JavaScript. A partir dessas instâncias é possível uma interação contínua entre o *back-end* e o *front-end*.

No *front-end*, o script HTML define a estrutura da interface do *Chatbot*, incluindo um botão para abrir/fechar o *chat*, uma área de exibição de mensagens, um campo de entrada de texto para o usuário digitar mensagens e botões adicionais com funcionalidades como anexar arquivos e enviar e-mail. Na Figura 6 é possível visualizar o código da função de interação entre o *back-end* e o *front-end*.

```
25     sendMessage(message = null) {
26         const userInput = message || this.userInput.value;
27         this.addMessage(userInput, 'user');
28         this.userInput.value = '';
29         fetch('/chat', {
30             method: 'POST',
31             headers: {
32                 'Content-Type': 'application/json'
33             },
34             body: JSON.stringify({ message: userInput, user_id: this.userId })
35         })
36         .then(response => response.json())
37         .then(data => {
38             if (typeof data === 'string') {
39                 this.addMessage(data, 'bot');
40             } else if (data && data.response) {
41                 this.addMessage(data.response, 'bot', true);
42                 if (data.show_upload_buttons !== undefined) {
43                     this.showUploadButtons(data.show_upload_buttons);
44                 }
45             } else {
46                 console.error('Formato de resposta desconhecido:', data);
47             }
48         })
49         .catch(error => {
50             console.error('Erro ao enviar mensagem:', error);
51         });
52     }
```

Figura 6: Função de interagir com *back-end* e *front-end*. Fonte: Próprio Autor (2025).

O código implementa a função `sendMessage()` (Linha 25), que envia mensagens do usuário para um servidor via requisição POST e exibe a resposta do *Chatbot* na interface. A função captura a mensagem do usuário, a exibe na interface e limpa o campo de entrada (Linhas 26 a 28). Em seguida, envia a mensagem e o ID do usuário em formato JSON para a rota `/chat` (Linhas 29 a 35). A resposta do servidor é convertida para JSON (Linha 36) e processada: se for uma *string*, é exibida como mensagem do *Chatbot* (Linhas 37 a 39); se for um objeto JSON com a propriedade `response`, a mensagem é exibida e botões de *upload* são verificados (Linhas 40 a 44). Caso o formato da resposta seja inválido (Linhas 45 a 47) ou ocorram erros na requisição (Linhas 49 a 51), esses são registrados no console para depuração.

4.6. Tela do Chatbot

Para otimizar o uso da tecnologia empregada no desenvolvimento do *Chatbot*, especialmente na camada de *front-end* acessada por navegadores web, é essencial compreender que essa camada constitui o principal ponto de interação entre o usuário e o sistema, caracterizando-se como uma Interface Homem-Máquina (IHM). As IHMs desempenham um papel crucial ao mediar a comunicação entre humanos e computadores, oferecendo uma representação visual clara e intuitiva na troca de informações. Nesse contexto, foi desenvolvida uma interface que alia funcionalidade e usabilidade, permitindo que os usuários explorem os recursos disponibilizados pelo sistema. A Figura 7 ilustra a tela de interação do *Chatbot*. O código-fonte do projeto está disponível no repositório: <https://github.com/IFBot-certificados>.



Figura 7. Tela principal do Chatbot. Fonte: Próprio Autor (2025).

5. Considerações Finais

Este trabalho propôs e desenvolveu um *Chatbot* para auxiliar os usuários do Sistema de Emissão e Validação de Certificados do IFBA *Campus* Vitória da Conquista. Mesmo que, ainda não disponibilizado para os usuários finais, o *Chatbot* é uma solução promissora para melhorar a interação entre os usuários e o Sistema de Certificados. A capacidade de responder às demandas dos usuários, aliada à possibilidade de aprendizado contínuo, faz do *Chatbot* uma ferramenta importante para a instituição. Além disso, a implementação do *Chatbot* pode reduzir a carga de trabalho dos administradores do sistema, permitindo que eles se concentrem em tarefas mais estratégicas.

No entanto, é importante destacar que o *Chatbot* está em fase de desenvolvimento e pode ser aprimorado em diversos aspectos. Futuros trabalhos podem incluir a expansão das funcionalidades do *Chatbot*, como a integração com outras tecnologias, tais como, a melhoria da precisão das respostas por meio de técnicas avançadas de PLN ou implementando IA Generativa com uso de LLA (*Large Language Model*) e a realização de testes com usuários para avaliar a usabilidade e a eficácia do sistema. É também possível integração com plataformas de *chat* populares como WhatsApp e Telegram.

Referências

- Adamopoulou, E. e Moussiades, L. (2020). An Overview of Chatbot Technology. In: Maglogiannis, I., Iliadis, L., Pimenidis, E. (eds) *Artificial Intelligence Applications and Innovations*, v. 584, p. 373–383.
- Almeida Junior, O. A. (2017) *Beck: Um chatbot baseado na terapia cognitivo-comportamental para apoiar adolescentes com depressão*. Dissertação (Ciência da Computação), UFPE, Recife/PE.
- Appolinário, F. (2012) *Metodologia da Ciência: filosofia e prática da pesquisa*. 2ª ed. sp: Cengage Learning.
- Costa, S. R. (2020) *A Contribuição da Inteligência Artificial na Celeridade dos Trabalhos Repetitivos no Sistema Jurídico*. Dissertação (Mídia e Tecnologia), Universidade Estadual Paulista, Bauru/SP.
- Dale, R. (2010) “Classical approaches to natural language processing”. In: Handbook of NLP, 2ª ed. Editor / I, Nitin, D, Fred J. Boca Raton, Florida: CRC Press, Taylor & Francis Group, p. 3-7.
- Ghidini, I. e Mattos, W. W. (2018) *Desenvolvimento e Aplicação de um Chatbot para Auxiliar Atendimento ao Cliente*. Trabalho de Conclusão do Curso (Sistemas de Informação), UNISUL, Palhoça/SC.
- Gil, A. C. (2010) *Como Elaborar Projetos de Pesquisa*. Atlas, São Paulo.
- Grewal, D. S. (2014) “A Critical Conceptual Analysis of Definitions of Artificial Intelligence as Applicable to Computer Engineering”. *IOSR Journal of Computer Engineering*, v. 16, i. 2, p. 09-13.
- Hurwitz, J. e Kirsch, D. (2018) *Machine Learning for Dummies*. Jorh Wiley & Sons, NJ.
- Jones, M. T. (2021) “A beginner's guide to artificial intelligence and machine learning”, <https://developer.ibm.com/articles/cc-beginner-guide-machine-learning-ai-cognitive>, [acessado em 21/02/2025].
- Khan, R. e Das, A. (2018) *Build Better Chatbots: A complete guide to getting started with Chatbots*. Apress, NY.
- Maciel, H. B. (2019) *Ferramentas e Criação de Chatbot: Maciel o robô acadêmico*. Trabalho de Conclusão do Curso (Engenharia de Software), Universidade Federal do Ceará, Russas/CE.
- Miklosik, A, Evans, N. e Qureshi, A. (2021) “The Use of Chatbots in Digital Business Transformation: A Systematic Literature Review”. *IEEE Access*. v. 9, p. 106530-106539.
- Navlani, A. (2024) “Building a Chatbot using Chatterbot in Python: Tutorials”, <https://www.datacamp.com/community/tutorials/building-a-chatbot-using-chatterbot>, [acessado em 13/02/2025].
- Oliveira, F. A. (2018) *Estudo sobre Chatbots: Desenvolvendo uma solução para controle de filas*. Trabalho de Conclusão do Curso (Análise e Desenvolvimento de Sistemas), Fundação Educacional do Município de Assis (FEMA), Assis/SP.
- Paiva, F. (2021) “Mapa do Ecossistema Brasileiro de Bots”, <https://www.mobiletime.com.br/pesquisas/mapa-do-ecossistema-brasileiro-de-bots-2021/>, [acessado em 30/01/2025].
- Ramos, L. L, Silva, J. P, Sobreira, A. D. e Matos, P. F. (2018) Sistema Web e Open Source de Gerenciamento de Emissão e Validação de Certificados nos Institutos Federais. In: *XII CONNEPI - Congresso Norte Nordeste de Pesquisa e Inovação*, Recife, PE. P. 1-10.
- Sage, A. P. (1990) *Concise Encyclopedia of Information Processing in Systems and Organizations*. New York: Pergamon
- Shukairy, A. (2020) “Chatbots in Customer Service: Statistics and Trends”, <https://www.invespro.com/blog/chatbots-customer-service/>, [acessado em 13/02/ 2025].
- Spanhol, T. S. (2017) *Um estudo sobre a interação entre usuários e chatterbots*. Trabalho de Conclusão do Curso (Tecnologias da Informação e Comunicação). UFSC, Araranguá/SC.
- Verspoor, K. e Cohen, K. B. (2013) “Natural Language Processing”. In: Dubitzky, W., Wolkenhauer, O., Cho, KH., Yokota, H. (eds) *Encyclopedia of Systems Biology*. Springer, New York, NY.
- Weizenbaum, J. (1966) “*ELIZA: A computer program for the study of natural language communication between man and machine*”. *Communications of the ACM*, v. 9, i. 1, p 36–45.

FICHA CATALOGRÁFICA ELABORADA PELO SISTEMA DE BIBLIOTECAS DO
IFBA, COM OS DADOS FORNECIDOS PELO(A) AUTOR(A)

L557a Lemos Júnior, Edvaldo Moreira

Assistente de atendimento aos usuários do sistema de certificados do IFBA Campus Vitória da Conquista/ Edvaldo Moreira Lemos Júnior; orientador Prof. Pablo Freire Matos - - Vitória da Conquista: IFBA, 2025.

13 p.

Trabalho de Conclusão de Curso (Graduação) - Bacharelado em Sistemas de Informação. Instituto Federal de Educação, Ciência e Tecnologia da Bahia - *Campus Vitória da Conquista* - BA, 2025.

1. Assistente de Atendimento. 2. Chatbot. 3. Inteligência Artificial. 4. Sistema de Certificados. I. Matos, Pablo Freire, orient. II.TÍTULO.

CDU:004