INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E

TECNOLOGIA DE SÃO PAULO

CAMPUS VOTUPORANGA

GUSTAVO MONTANHER DE OLIVEIRA

**AGENDAGORA – SISTEMA DE AGENDAMENTO DE SERVIÇOS**

VOTUPORANGA

2024

GUSTAVO MONTANHER DE OLIVEIRA

**AGENDAGORA – SISTEMA DE AGENDAMENTO DE SERVIÇOS**

Trabalho de Qualificação de Curso apresentado como exigência parcial para obtenção do diploma do Curso em Bacharelado em Sistemas de Informação do Instituto Federal de Educação, Ciência e Tecnologia, Câmpus Votuporanga.

Professor Orientador: Ivan Oliveira Lopes.

Votuporanga

2024

GUSTAVO MONTANHER DE OLIVEIRA

**AGENDAGORA – SISTEMA DE AGENDAMENTO DE SERVIÇOS**

Trabalho de Qualificação de Curso apresentado como exigência parcial para obtenção do diploma do Curso em Bacharelado em Sistemas de Informação do Instituto Federal de Educação, Ciência e Tecnologia, Campus Votuporanga.

Professor Orientador: Ivan Oliveira Lopes.

Aprovado pela banca examinadora em 16 de agosto de 2024.

BANCA EXAMINADORA:

­

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Éder Antonio Pansani Júnior

**DEDICATÓRIA**

*Dedico este trabalho a todos os colegas*

*de classe que estiveram presentes nessa trajetória de alegrias e superação.*

**AGRADECIMENTOS**

A todos os professores e servidores do IFSP Campus Votuporanga que contribuíram direta e indiretamente para a conclusão desse trabalho.

À minha família, que deu todo o apoio necessário para que eu chegasse até aqui.

Ao meu orientador, que me auxiliou a solucionar as dificuldades encontradas no caminho.

**EPÍGRAFE**

*"O que prevemos raramente ocorre; o que menos esperamos geralmente acontece".*

*Benjamin Disraeli*

**RESUMO**

Consultórios, barbearias, restaurantes e outros negócios e empresas muitas vezes precisam de sistemas de agendamento para que suas operações funcionem. Inspirado nos benefícios que agendamentos online proporcionam, dentro deste trabalho será apresentado a proposta de uma plataforma web de agendamentos que poderá ser usado por diferentes tipos de negócios, como consultórios, cabelereiros ou restaurantes. Este sistema visa criar uma alternativa simples para agendamentos online, pensando na facilidade de uso tanto para os possíveis donos de negócios quanto para seus clientes. Esta plataforma será programada em Python e será feito o uso da plataforma AWS para incorporar a tecnologia de *chatbots*, que será utilizada para administrar a conversa das reservas de clientes da plataforma.

**Palavras-chave:** agendamento de serviços, sistemas web, AWS, *chatbot*

**ABSTRACT**

Offices, barbershops, restaurants and other businesses and companies often need scheduling system systems for their operations to function. Inspired by the benefits that online booking provides, this paper will present a proposal for a booking web platform that can be used by different types of businesses, from dentists to hairdressers. This system aims to create a simple alternative to online bookings, with ease of use in mind for both the potential business owners and their customers. This platform will be programmed in Python and use the AWS platform to incorporate chatbot technology, which will be used to manage the conversation of the platform's customer bookings.

**Keywords:** service appointments, web-based, AWS, chatbot

**SUMÁRIO**

**1 INTRODUÇÃO*…………………………………………………………….......*** 7

**2 OBJETIVOS*……………………………………………………………….......*** 8

2.1 OBJETIVO GERAL***...................................................................................*** 8

2.2OBJETIVOS ESPECÍFICOS***……………………………………........………*** 8

**3 JUSTIFICATIVA*…………………………………………………...................*** 9

**4 TRABALHOS CORRELATOS *…………………………......……....……....*** 10

**5 REVISÃO DE LITERATURA*....................................................................*** 11

5.1 WEBSITES***................................................................................................*** 11

5.2 LINGUAGENS DE PROGRAMAÇÃO***.......................................................*** 12

**5.2.1** **Python*......................................................................................................*** 12

5.3 COMPUTAÇÃO EM NUVEM***....................................................................*** 13

**5.3.1** **AWS*..........................................................................................................*** 13

5.3.1.1 Amazon Lex***..............................................................................................*** 13

5.3.1.2 AWS Lambda***…………………………………………………………………..*** 14

5.3.1.3 Amazon Cloudfront***…………………………………………………………….*** 15

5.3.1.4 Amazon Simple Storage Service***…………………………………………….*** 15

5.3.1.5 BANCO DE DADOS***.................................................................................*** 16

**6 DESENVOLVIMENTO*.............................................................................*** 17

6.1 DIAGRAMA DE CASO DE USO***..............................................................*** 17

6.2 ARQUITETURA EM NUVEM***...................................................................*** 18

**7 RESULTADOS E DISCUSSÃO*...............................................................*** 20

**REFERÊNCIAS*........................................................................................*** 21

# 1 INTRODUÇÃO

Antes dos telefones e celulares, agendar serviços ou fazer reservas envolvia visitar pessoalmente o local de prestação de serviços, como consultórios médicos, salões de beleza, restaurantes, entre outros. Este método era limitado pela necessidade de deslocamento físico e pela disponibilidade restrita de horários para fazer agendamentos.

Com a popularização dos telefones, muitas empresas e prestadores de serviços começaram a oferecer a opção de agendamento por telefone. Isso proporcionou uma maior conveniência para os clientes, permitindo que agendamentos fossem feitos sem necessidade de visitas pessoais, mas ainda estava sujeito à disponibilidade de operadores e aos horários de funcionamento das linhas telefônicas.

Com a expansão da internet, surgiu a possibilidade de realizar agendamentos *online*, proporcionando assim maior acessibilidade e flexibilidade do que antes era possível. Este uso *online* de agendamento pode vir de diferentes meios:

* Redes sociais, como dentro do Whatsapp[[1]](#footnote-1);
* Websites pessoais da empresa ou negócio;
* Plataformas externas, como o Markey ou Reservio.

Dentro deste trabalho, será abordado o desenvolvimento de uma plataforma web similar ao dos últimos exemplos dados, porém em um foco na simplicidade e facilidade de uso.

Ele será feito com uma hospedagem e arquitetura AWS[[2]](#footnote-2) e poderá ser usado como uma plataforma para diferentes empresas e negócios diversos, dando a eles o poder de agendarem com seus devidos clientes.

# 2 OBJETIVOS

## 2.1 OBJETIVO GERAL

Criar uma plataforma de agendamentos de horários e/ou serviços que, utilizando o recurso de *chatbot* da plataforma AWS para administrar conversas, poderá ser usada por diferentes tipos de negócios, como consultórios, cabelereiros ou restaurantes.

## 2.2 OBJETIVOS ESPECÍFICOS

* Entender as diferentes necessidades de reserva que cada negócio pode ter;
* Desenvolver uma plataforma web donde é possível criar e acessar perfis pessoais;
* Criar entradas dentro da plataforma que informarão possíveis serviços e horários de atendimento;
* Criar um banco de dados para o armazenamento destas informações;
* Trabalhar com os sistemas e mecanismos da plataforma AWS;
* Criar o *chatbot* que será utilizado dentro da plataforma web;
* Integrar o *chatbot* junto a plataforma web.

# 3 JUSTIFICATIVA

Como já dito, antes das tecnologias de comunicação modernas, agendamentos tinham que ser feitos de forma presencial, assim limitando seus clientes pela necessidade de deslocamento físico até os seus devidos estabelecimentos. Porém, com o advento de telefones, foi possível fazer agendamentos mais convenientes desprovidos de visitas pessoais. Com a internet, podemos fazer agendamentos *online* por meio de algum *software* ou *website*.

O agendamento online é interessante por poder oferecer uma variedade de benefícios, como por exemplo:

* a disponibilidade de 24 horas, disponível a qualquer momento;
* a conveniência de fazer o agendamento de qualquer lugar, utilizando dispositivos móveis ou computadores, sem a necessidade de esperar em filas telefônicas ou físicas;
* o poder de escolher entre diferentes horários e datas disponíveis de forma mais clara e detalhada do que seria possível por telefone;
* um custo mais barato proveniente da redução da carga de trabalho administrativo associada ao agendamento por telefone;
* menor probabilidade de erros de comunicação ou de interpretação, já que as informações são inseridas diretamente pelo usuário de forma uniforme e previamente já pensada;
* a facilidade de consultar seu histórico de agendamentos anteriores.

A justificação deste trabalho vem com a maior flexibilidade que donos de negócios terão por terem mais uma alternativa para seus sistemas de agendamento. Com a plataforma web desenvolvida, eles poderão ter um sistema de fácil uso com todos os benefícios listados sem terem que pagar para um desenvolvedor construir um sistema pessoal para eles.

# 4 TRABALHOS CORRELATOS

Agora, já existem plataformas que providenciam um serviço semelhante ao que planeja ser desenvolvido dentro deste trabalho, como o Markey ou o Reservio por exemplo. Com isso em mente, ainda é beneficial desenvolver este tipo de plataforma por três motivos principais:

• prover uma alternativa focada na simplicidade e na facilidade a estas opções existentes é bom para aqueles que se interessam por adquirir um sistema de agendamento próprio sem muitas complicações;

• pela natureza deste trabalho, o desenvolvimento desta plataforma será documentado neste documento e logo poderá servir como base para o desenvolvimento de outras aplicações comerciais semelhantes;

• o desenvolvimento desta plataforma poderá ajudar aqueles que não estão dispostos a pagar pela criação de um sistema próprio e que buscam invés disso desenvolver sua própria plataforma pessoal.

# 5 REVISÃO DE LITERATURA

Dentro desta seção são discutidos os fundamentos básicos de alguns dos alicerces tecnológicos que serão empregados no desenvolvimento da plataforma de agendamentos deste trabalho. São falados de *websites,* linguagens de programação em geral mas especialmente sobre a língua Python, computação em nuvem e o uso dessa tecnologia dentro das diversas ferramentas providas pela plataforma da *Amazon Web Services*.

## 5.1 WEBSITES

De acordo com o dicionário inglês de Oxford fornecido pela Oxford University Press (2023), um website pode ser definido como uma coleção de páginas *web* que é identificado por um nome de domínio. Mas agora, o que é um “nome de domínio”? De acordo com o Suh (2024) em seu artigo da Cloudfare, um nome de domínio é basicamente o texto que um usuário digita em uma janela do navegador para acessar um site específico. Por exemplo, o nome de domínio do Google é "google.com".

Os *hyperlinks* (também popularmente conhecidos como links) entre as páginas da *web* orientam a navegação do site, que como no caso da plataforma deste trabalho, geralmente começa com uma página inicial.

Os websites desempenham um papel fundamental na era digital, atuando em várias áreas da vida moderna em sua presença online e alcance global. Desde seu uso no setor profissional quanto dentro da bolha casual do nosso cotidiano, websites funcionam como um meio valioso para obter informações e cuidar de uma miríade de funções.

Como esta plataforma web de agendamentos irá ter certas funções como o registro de usuários e interações com respostas em tempo real por meio de *chatbots,* ela terá irá se encaixar na definição de um site dinâmico. Um site dinâmico muda ou se personaliza automaticamente, podendo oferecer para seus usuários uma experiência mais interativa (Lyardet; Rossi; Schwabe, 1998).

Para o desenvolvimento de um site dinâmico, como na plataforma de agenda deste trabalho, será necessário fazer amplo uso de linguagens de programação.

## 5.2 LINGUAGENS DE PROGRAMAÇÃO

Uma linguagem de programação é um sistema de notação para escrever programas de computador (Aaby, 2007). Este tipo de tecnologia é uma ferramenta essencial que, segundo Morimoto (2007) serve como um meio de comunicação entre o ser humano e o computador, permitindo que estes últimos realizem tarefas específicas de acordo com as intenções do usuário.

Em outras palavras, linguagens de programação são os grandes alicerces que fazem o desenvolvimento de softwares possível. Como um paralelo demonstrativo para melhor entender o quão importante são estas línguas, é possível dizer que tentar desenvolver uma aplicação de computador sem tais ferramentas é como tentar construir uma casa sem ferramentas e materiais. Sem uma linguagem de programação, você não tem os recursos necessários para criar, estruturar, e executar um software de maneira eficaz. Ambos os processos requerem planejamento, ferramentas adequadas, execução cuidadosa, e manutenção contínua para garantir um resultado funcional e bem-sucedido.

Existe uma variedade de linguagens de programação, cada uma com seus pontos fortes e sintática, mas dentro do desenvolvimento da plataforma deste trabalho, será amplamente utilizado a linguagem de programação chamada Python.

### 5.2.1 Python

Python é uma linguagem de programação de propósito geral de alto nível (KHULMAN, 2009).

Linguagens de alto nível não são interpretadas diretamente pelo computador, sendo assim necessário traduzir suas palavras para a linguagem binária. Para esta conversão é utilizado um tipo de programa chamado compilador. Evaristo (2014) em seu artigo conclui que elas são mais próximas a linguagens naturais e logo são mais compreensíveis por usuários humanos. Por causa disto, linguagens de alto nível como o Python são de fácil entendimento e manuseio.

O Python foi escolhido para este projeto principalmente porque ele é uma língua de fácil uso e entendimento que pode ser usado dentro da plataforma de computação em nuvem da AWS, plataforma providenciará o uso de *chatbots*.

## 5.3 COMPUTAÇÃO EM NUVEM

É normal o uso de tecnologias e documentos que tomam espaço em nossos computadores e dispositivos pessoais. Podemos por exemplo assistir filmes que não possuímos em nossas máquinas por meio da Netflix, podemos também nos relacionar com jogos eletrônicos por meio da *Xbox Cloud Gaming* mesmo sem que estes próprios ocupem espaço em nossa memória ou utilizem de outros processos de nosso hardware. Isto é possível por meio da tecnologia de computação em nuvem.

A computação em nuvem é a disponibilidade sob demanda de recursos de um dado sistema de computador, como no armazenamento de dados (neste contexto também chamado de armazenamento em nuvem) ou na capacidade de computação sem a utilização ativa da máquina do usuário (Ray, 2018).

### 5.3.1 AWS

A *Amazon Web Services*, Inc. (AWS), é uma subsidiária da Amazon que trabalha com computação em nuvem, providenciando a seus clientes esta tecnologia *on-demand[[3]](#footnote-3)* e APIs em um formato de *pay-as-you-go[[4]](#footnote-4)* (AWS, 2024).

O uso dos serviços da AWS é amplamente usado em todo tipo de desenvolvimento atualmente. De projetos pequenos, como o desenvolvido neste trabalho, até grandes empresas, como o iFood ou a Nubank, fazem da utilização dela para o funcionamento geral de seus programas.

A AWS, via a estes serviços da *Web* de computação em nuvem, fornece uma variedade de ferramentas relacionados a redes, computação, armazenamento, middleware, IoT e outras capacidades de processamento (AWS, 2024). Uma das ferramentas que irá ser usado dentro da plataforma de agendamento será a aplicação de *chatbot* provido pelo AWS.

#### 5.3.1.1 Amazon Lex

Um *chatbot* (originalmente *chatterbot*) é um aplicativo de software ou interface da *Web* projetado para imitar a conversação humana por meio de interações de texto ou voz (Adamopoulou; Moussiades, 2020). Dentro deste projeto, será usado um *chatbot* de texto tradicional, diferente de como alguns *chatbots* modernos que usam sistemas de inteligência artificial generativa capazes de manter uma conversa com um usuário em linguagem natural e simular a maneira como um ser humano se comportaria como parceiro de conversação.

*Chatbots* são muito usados em serviços de atendimento ao cliente como uma escolha alternativa ou auxiliar do convencional atendente humano. Muitas organizações querem ou já incorporam *chatbots* em seus serviços como um passo a automatização geral de suas funções. Fazer utilização deste tipo de tecnologia tende a ser mais barato do que o sistema tradicional de atendimento, pois a redução da carga de trabalho administrativo associada ao agendamento por telefone é massivamente menor (Hingrajia, 2024).

O Amazon Lex é um serviço da Amazon Web Services (AWS) que facilita a criação de interfaces de conversação utilizando a tecnologia de processamento de linguagem natural (NLP) e reconhecimento automático de fala (ASR). Em termos mais simples, o Amazon Lex permite que desenvolvedores criem *chatbots* e assistentes virtuais que podem interagir com os usuários por meio de texto ou voz de maneira natural e intuitiva. Ele será a ferramenta usada no desenvolvimento da plataforma web para o uso de *chatbots* na administração de reservas e agendamentos.

#### 5.3.1.2 AWS Lambda

O AWS Lambda é um serviço de computação dentro da plataforma AWS que executa seu código em resposta a eventos e gerencia automaticamente os recursos de computação. Ele foi projetado para permitir que os desenvolvedores executem códigos sem provisionar ou gerenciar servidores.

Ele será usado dentro da arquitetura para auxiliar o AWS Lambda com funções mais complexas que o Amazon Lexem si não consegue proporcionar em resposta a certos eventos que aconteceram dentro da plataforma.Agora vem em questão como é possível integrar os serviços de nuvem da plataforma AWS para uma plataforma web terceira, como a desenvolvida dentro deste trabalho. Para isso, será necessário o uso de uma outra ferramenta da AWS, a Amazon Cloudfront.

#### 5.3.1.3 Amazon Cloudfront

Agora vem em questão como é possível integrar os serviços de nuvem da plataforma AWS para uma plataforma web terceira, como a desenvolvida dentro deste trabalho. Para isso, será necessário o entendimento de uma outra ferramenta da AWS, a Amazon Cloudfront. Para isso, é preciso entender antes o conceito de CDNs.

*Content Delivery Networks* (CDN), ou traduzida para o português como Redes de Distribuição de Conteúdo, são grupos de servidores distribuídos geograficamente que acelera a entrega de conteúdo da Web, aproximando-o de onde os usuários estão (Akamai, 2023).

Se você está no estado de São Paulo, por exemplo, usar uma CDN dessa região ou perto dela irá fornecer uma performance mais eficiente do que se o servidor fosse em uma região mais distante.

O Amazon CloudFront é um serviço de CDN fornecido pela Amazon Web Services (AWS). Ele é projetado para distribuir conteúdo, como arquivos de mídia, páginas da web, e outros recursos, para usuários finais com baixa latência e alta velocidade de transferência.

Um benefício que irá ser utilizado do Cloufront é como ele consegue se integrar bem com outros serviços da AWS, como o Amazon Simple Storage Service.

#### 5.3.1.4 Amazon Simple Storage Service

O Amazon Simple Storage Service (Amazon S3) é um serviço de armazenamento de objetos dentro da AWS. Clientes de todos os portes e setores podem armazenar e proteger qualquer quantidade de dados de praticamente qualquer caso de uso dentro dele.

Para fazer upload de objetos para o Amazon S3, primeiro é necessário criar um *bucket* do S3 em uma das Regiões da AWS. Um *bucket* é um contêiner para objetos armazenados no Amazon S3. Você pode armazenar qualquer número de objetos em um *bucket* e pode ter até 100 *buckets* na sua conta.

O S3 será usado como um baco de dados para armazenar perfis, agendas, imagens e outras informações importantes para o funcionamento da plataforma.

## 5.4 BANCO DE DADOS

Dentro de muitas aplicações é necessário guardar uma variedade de dados dentro de algo chamado de banco de dados. De acordo com Date (2004) um sistema de banco de dados é basicamente apenas um sistema computadorizado de manutenção de registros. O autor ainda contribui com a analogia de que, o banco de dados, por si só, pode ser considerado como o equivalente eletrônico de um armário 15 de arquivamento; ou seja, ele é um repositório ou recipiente para uma coleção de arquivos de dados computadorizados. Os usuários de um banco de dados podem realizar uma variedade de operações para o manuseio das informações contido nele, como por exemplo:

• Acrescentar novos arquivos ao banco de dados;

• Inserir dados a arquivos existentes;

• Buscar dados de arquivos existentes;

• Excluir dados de arquivos existentes;

• Alterar dados em arquivos existentes;

• Remover arquivos existentes do banco de dado.

# 6 DESENVOLVIMENTO

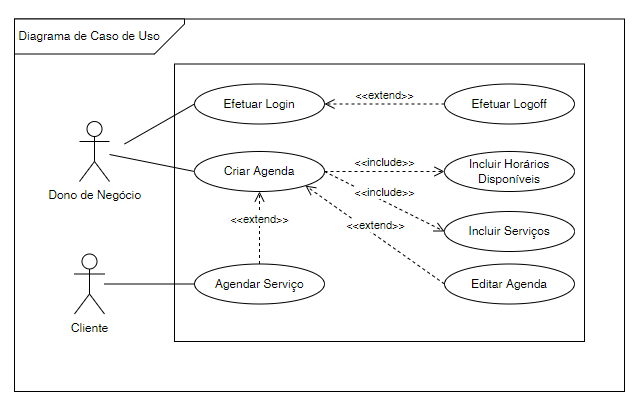
Nesta seção são apresentados o relatório do que se foi projetado e desenvolvido em direção à plataforma web de agendamentos deste trabalho, além das explicações técnicas de diagramas e modelos criados.

## 6.1 DIAGRAMA DE CASO DE USO

O diagrama de caso de uso tem como objetivo demonstrar as maneiras que usuários podem interagir com um sistema. Ele resume os detalhes dos usuários de seu sistema e as interações deles. (Lucidchart, 2024)

Em sua anatomia, podemos encontrar alguns componentes comuns:

* Atores: os usuários que interagem com o sistema. Eles podem ser uma pessoa, uma organização, ou qualquer tipo de agente consciente que interage com o sistema;
* Cenário: uma sequência específica de ações e interações entre os atores e o sistema;
* Caso de Uso*:* uma tarefa ou uma função feita pelo ator;
* Comunicação: ligação entre atores e casos de uso.

Figura 1 – Diagrama de Caso de Uso

Fonte: Elaborado pelo autor.

Dentro da Figura 1, podemos observar um diagrama de caso de uso composto por desenhos simples que descrevem de maneira bem concisa o que textualmente poderia ser mais confuso.

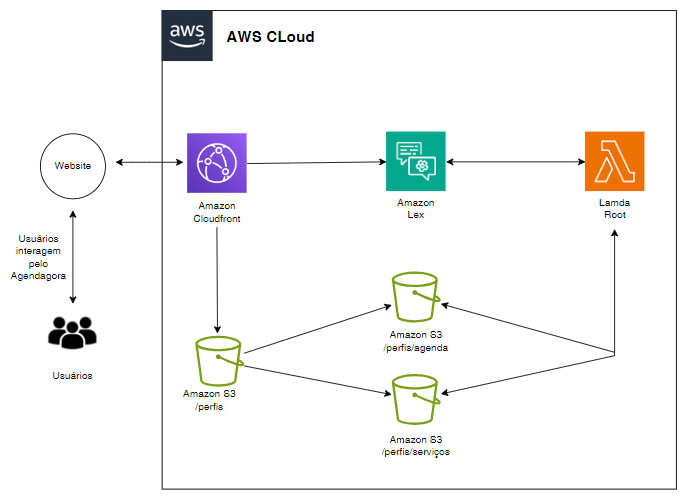
Podemos ver como os dois atores, o dono de negócio e o cliente, se interagem com os casos de uso do sistema. Mas dentro dele também se encontra algo ainda não discutido: as linhas de *include* e *extend.*

* *Include*: relação de um caso de uso que para ser executada precisa chamar outro caso de uso. Neste caso, ao criar uma agenda a relação de *include* mostra que dentro dessa ação o dono de cliente precisa necessariamente também incluir os seus horários de disponíveis e os tipos de serviços;
* *Extend*: quando um caso se estende para outro, isso significa que quando este segundo caso for executado, o primeiro poderá ser executado também ou não. No exemplo da Figura 1, o diagrama nos mostra que ao fazer *login[[5]](#footnote-5)* em um perfil, o usuário vai então ter a opção de fazer o *logoff[[6]](#footnote-6)* dele.

## 6.2 ARQUITETURA EM NUVEM

Tendo visto como o sistema será utilizado dentro do diagrama de caso de uso, e como já foi discutido o quão importante é a plataforma AWS para o funcionamento do sistema, é interessante ver de forma visual o modelo de arquitetura projetada para o funcionamento da plataforma web de reservas.

Figura 2 – Arquitetura dentro da plataforma AWS



Fonte: Elaborado pelo autor

Aqui podemos entender, de forma visual, como o sistema funcionará no nível nuvem dentro da plataforma AWS.

Tudo se inicia com a interação de usuários ao website. Daí, haverá a integração do website a plataforma de nuvem via o CDN da Amazon Cloudfront.

A Amazon Cloudfront consegue então se comunicar com a Amazon Lex, a ferramenta que irá prover o uso de *chatbot* ao nosso sistema. Ele também irá conseguir se conectar com os dados do agendamento por meio do banco de dados formados de *buckets* dentro da AWS.

Estes *buckets* contém a informação do perfil de que se está sendo interagido e logo de seus horários disponíveis em sua agenda e os serviços que ele providencia.

Com esses dados, o *chatbot* da Amazon Lex consegue saber que tipo de reserva ela proporcionará e logo consegue conversas com o cliente e agendar o que for necessário para aquele momento. Porém, o Amazon Lex só consegue pegar os dados dentro dos *buckets* por meio do Lambda.

O Lambda interage junto com o Amazon Lex e os *buckets* para fazer a interação de máquina e cliente desejada funcionar. O Amazon Lex não consegue fazer certas funções como pegar ou enviar dados para os *buckets,* logo dentro da plataforma será chamado uma função Lambda para certas ações.

# 7 RESULTADOS E DISCUSÃO

Dentro deste artigo foi traçado e estabelecido uma base de pesquisa de onde será desenvolvido uma plataforma de agendamento para os menores setores da economia. Porém, o esforço intelectual dentro de seu desenvolvimento ainda não está completo, no futuro terá que se pesquisar ainda mais da natureza deste software para sua criação e para a finalização deste trabalho.

# REFERÊNCIAS

AABY, Anthony A. Introduction to programming languages. Computer Science Department, Walla Walla College. http://www. worldcolleges. info/sites/default/files/aaby. pdf, 1996.

ADAMOPOULOU, Eleni; MOUSSIADES, Lefteris. An overview of chatbot technology. In: IFIP international conference on artificial intelligence applications and innovations. Springer, Cham, 2020. p. 373-383.

COMPUTAÇÃO em nuvem com a AWS. [S. l.], 25 jul. 2024. Disponível em: https://aws.amazon.com/pt/what-is-aws. Acesso em: 6 ago. 2024.

DATE, Christopher J. Introdução a sistemas de bancos de dados. Elsevier Brasil, 2004.

EVARISTO, Jaime. Aprendendo a programar na Linguagem C. 3ª ed. Revisada Ampliada. Edição Digital. 2014. Disponível em: http://www.das.ufsc.br/~jomi/das5334/Livro%20Aberto%20Aprendendo%20a%20Program ar%20naLinguagem%20C.pdf

FIELDING, Roy; RESCHKE, Julian. Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content. Rfc7231. [S. l.]: Adobe, 2014. Disponível em: https://datatracker.ietf.org/doc/html/rfc7231. Acesso em: 23 jul. 2024.

HINGRAJIA, Migrant. Understanding the Intelligent Chatbot for Banking System. [S. l.], 04 jun. 2024. Disponível em: https://marutitech.com/chatbots-transforming-wall-street-main-street-banks/. Acesso em: 6 ago. 2024.

LYARDET, Fernando; ROSSI, Gustavo; SCHWABE, Daniel. Patterns for dynamic websites. In: Proceedings of PloP. 1998.

MEDINA, Marcos; FERTIG, Cristina. Algoritmos e Programação –Teoria e Prática. 1ª ed. São Paulo. Ed Novatec, 2005

MELENDEZ, Steven. The Difference Between Dynamic & Static Web Pages. CHRON (ed.). [S. l.]: CHRON, 10 ago. 2018. Disponível em: https://smallbusiness.chron.com/difference-between-dynamic-static-pages-69951.html. Acesso em: 6 ago. 2024.

MITCHEL, John. What Programming Language Is Primarily Used on the Web?. CHRON (ed.), 25 fev. 2023. Disponível em: https://web.archive.org/web/20230425003024/https://smallbusiness.chron.com/programming-language-primarily-used-web-26635.html. Acesso em: 6 ago. 2024

MORIMOTO, Carlos E. Linguagens de Programação. [S. l.], 06 jul. 2007. Disponível em: http://www.hardware.com.br/artigos/linguagens/

OXFORD UNIVERSITY PRESS (org.). Oxford English Dictionary: Oxford UP, 2023. Disponível em: https://doi.org/10.1093/OED/1301261284. Acesso em: 4 ago. 2024.

RAY, P. P. An Introduction to Dew Computing: Definition, Concept and Implications. IEEE Access, v. 6, p. 723–737, 2018.

SANT’ANA, Gabriel. O que é abstração em programação? 2020. Disponível em: https://medium.com/@gabrielsantana765/o-que-é-abstração-em-programação-bf8ac2fcd96f

SOMMERVILLE, Ian. Engenharia de software. 6° ed. Tradução Maurício de Andrade. São Paulo: Ed Addison-Wesley, 2003

SUH, Jonathan. O que é um nome de domínio?: Nome de domínio X URL. Cloudfare, [s. l.], 29 fev. 2024. Disponível em: https://www.cloudflare.com/pt-br/learning/dns/glossary/what-is-a-domain-name/. Acesso em: 6 ago. 2024.

W3. What is the difference between the Web and the Internet?. [S. l.], 2024. Disponível em: https://www.w3.org/help/#webinternet. Acesso em: 4 jul. 2024.

LENOVO. What is a Dynamic Website & How Can It Benefit Your Business? [S. l.], 1 abr. 2024. Disponível em: https://www.lenovo.com/us/en/glossary/dynamic-website/#:~:text=Unlike%20static%20websites%2C%20which%20display,databases%2C%20and%20provide%20interactive%20features. Acesso em: 6 ago. 2024

SCHERMA, Mariana. Chatbot para WhatsApp: dicas para criar um e usar na sua estratégia — inclusive para fidelizar clientes. 16 ago. 2024. Disponível em: <https://www.rdstation.com/blog/conversacional/chatbot-para-whatsapp/>

KUHLMAN, Dave. A python book: Beginning python, advanced python, and python exercises. Lutz: Dave Kuhlman, 2009.

LUCIDCHART. Diagrama de caso de uso UML: O que é, como fazer e exemplos. 10 ago. 2020. Disponível em: <https://www.lucidchart.com/pages/pt/diagrama-de-caso-de-uso-uml>

AKAMAI. What Is a CDN (Content Delivery Network)? 12 mai. 2023. Disponível em: <https://www.akamai.com/glossary/what-is-a-cdn#:~:text=A%20content%20delivery%20network%20(CDN,closer%20to%20where%20users%20are>.

GALL, Bruno. O que é login? Conheça os principais métodos de acesso no meio digital. 31 ago. 2022. Disponível em: <https://tecnoblog.net/responde/o-que-e-login/#:~:text=Login%20(de%20%E2%80%9Clogging%20in%E2%80%9D,%2C%20e%2Dmail%20e%20afins>.

HARWOOD, Robin. Fazer logoff ou desconectar sessões do usuário. 30 ago. 2023. Disponível em: https://learn.microsoft.com/pt-br/windows-server/remote/multipoint-services/log-off-or-disconnect-user-sessions

1. Um chatbot dentro do WhatsApp é uma ferramenta que você mesmo configura para conversar com Leads e clientes para que essas pessoas tenham um atendimento mais rápido. As melhores soluções do mercado usam IA para um atendimento mais humano. (Scherma, 2024) [↑](#footnote-ref-1)
2. O *Amazon Web Service*, ou AWS, é uma plataforma on-line que oferece soluções econômicas e dimensionáveis de computação em nuvem. (AWS, 2024) [↑](#footnote-ref-2)
3. Sob demanda. [↑](#footnote-ref-3)
4. Um sistema de cobrir os custos à medida que eles surgem ou de pagar por um serviço antes que ele seja usado. [↑](#footnote-ref-4)
5. Login (de “logging in” ou “conectar”) é um conjunto de credenciais e procedimentos usados para identificar um usuário em um app, site, rede social, e-mail e afins. (Gall, 2023) [↑](#footnote-ref-5)
6. A desconexão de uma sessão. Sair de seu perfil. (Harwood, 2023) [↑](#footnote-ref-6)