Logotipo, nome da empresa

Descrição gerada automaticamente

**UNIVERSIDADE DE UBERABA**

**SAMUEL LUCAS RODRIGUES DE OLIVEIRA E SILVA**

**DESENVOLVIMENTO DE APLICAÇÃO ASSISTENTE NA FORMAÇÃO DE  
HÁBITOS BASEADA EM NEUROCIÊNCIA COMPORTAMENTAL**

UBERLÂNDIA – MG 2024

**SAMUEL LUCAS RODRIGUES DE OLIVEIRA E SILVA**

**DESENVOLVIMENTO DE APLICAÇÃO ASSISTENTE NA FORMAÇÃO DE  
HÁBITOS BASEADA EM NEUROCIÊNCIA COMPORTAMENTAL**

Trabalho apresentado à Universidade de Uberaba, como parte das exigências à conclusão do componente Trabalho de Conclusão de Curso, da 10ª etapa, do curso de graduação em Engenharia da Computação, da UNIUBE, Campus

Uberlândia.

Orientador: Prof. Me. Clênio Eduardo da Silva

UBERLÂNDIA – MG 2024

**SAMUEL LUCAS RODRIGUES DE OLIVEIRA E SILVA**

**DESENVOLVIMENTO DE APLICAÇÃO ASSISTENTE NA FORMAÇÃO DE HÁBITOS BASEADA EM NEUROCIÊNCIA COMPORTAMENTAL**

Trabalho apresentado à Universidade de Uberaba, como parte das exigências à conclusão do componente Trabalho de Conclusão de Curso, da 10ª etapa, do curso de graduação em Engenharia Elétrica, da UNIUBE, Campus Uberlândia.

Aprovado em \_\_ de junho de \_\_\_\_.

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

Prof. Me. Clênio Eduardo da Silva Universidade de Uberaba

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

Prof. Me. Stéfano Schwenck Borges Vale Vita Universidade de Uberaba

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

Prof. Me. Júlio Almeida Borges

Universidade de Uberaba

**DESENVOLVIMENTO DE APLICAÇÃO ASSISTENTE NA FORMAÇÃO DE HÁBITOS BASEADA EM NEUROCIÊNCIA COMPORTAMENTAL**

Samuel Lucas Rodrigues de Oliveira e Silva  
 samuellucasrdg@gmail.com

Clênio Eduardo da Silva clenio.silva@uniube.br

# RESUMO

Atualmente o usuário médio de internet passa cerca de seis horas de seu dia “online”, devido a facilidade de acessar o meio digital e aplicativos que tem como parte de seu “design” a adicção de seus usuários, por meio de técnicas como a seleção de conteúdos baseados no interesse do usuário. No entanto, o ambiente digital não foi criado com o intuito de ser um local maléfico que retira a capacidade produtiva de seus usuários. Esse trabalho tem como objetivo criar uma aplicação *mobile*, com o intuito de auxiliar na formação de novos hábitos utilizando de técnicas baseadas na neurociência comportamental, para facilitar a modulagem de novos comportamentos.

**Palavras-chave:** Dispositivos móveis, aplicativo, hábitos.

# ABSTRACT

Currently, the average internet user spends about six hours a day online, due to the ease of accessing digital media and applications that are designed to be addictive to their users, using techniques such as content selection based on user interest. However, the digital environment was not created with the intention of being a harmful place that diminishes users' productivity. This work aims to create a mobile application to assist in the formation of new habits using techniques based on behavioral neuroscience to facilitate the shaping of new user behaviors.

**Keywords:** Mobile devices, application, habits.

# INTRODUÇÃO

De acordo com o relatório da Global Web Index (2017), usuários da *internet* estimaram passar cerca de seis horas e meia *online*. Além disso, é estimado que o uso de telefones celulares, para acessar a *internet* aumentou 1 hora e 17 minutos para 2 horas e 30 minutos. Durante o mesmo período, computadores e tablets registraram uma queda, mesmo que mínima, em seu tempo de uso.

O Brasil está entre um dos países a qual o tempo de uso da *internet* é maior que o das mídias tradicionais, como rádios e jornais (Global Web Index, 2017). Ademais, de acordo com o Relatório do Acompanhamento do Setor de Telecomunicações da Agência Nacional de Telecomunicações (Anatel), vinculada ao Ministério das Comunicações (MCom), o Brasil registrou 234,07 milhões de acessos móveis em 2020, representando um aumento de 3,26% em relação a 2019 (BRASIL, 2021). Assim, é notável que o acesso aos meios digitais por meio de telefones é algo de suma importância na rotina do brasileiro.

Dito isso, é válido afirmar que, o uso do celular nem sempre é positivo, pois, no ano de 2012, pesquisadores encontraram uma correlação entre o uso intensivo dos celulares e sintomas de depressão e taxa de suícidio em adolescentes. Além disso, um estudo de caso com estudantes de negócios em Bangladesh encontrou que o aumento da impaciência e a perturbação na vida diária, decorrido do uso excessivo de telefones, afetaram negativamente o desempenho acadêmico dos estudantes (SHOUKAT, 2019). Um possível questionamento a ser feito é como um ambiente, mesmo ao trazer tantas adversidades a seus usuários, ainda é utilizado por tantas horas no dia.

De acordo com Mendelsohn (2019), a maioria dos comportamentos realizados em um dia são ditados pelos hábitos. Ademais, essa também descreve o hábito como responsável pela rotina, comportamentos e até mesmo alguns processos cognitivos. Acrescenta-se que, em um estudo realizado em laboratório, quando um rato, treinado para puxar uma alavanca com o objetivo de receber uma recompensa, recebe uma punição, o comportamento habitual se torna menos provável de acontecer uma próxima vez. Dessa maneira, pode-se inferir que o uso intensivo de celular, mesmo ao afetar negativamente o usuário em diversas áreas de sua vida, mantêm-se elevado por se tratar de um hábito que não traz punições imediatas.

De acordo com Ostlund e Balleine (2008), vícios são, de uma maneira simplificada, hábitos, pois a natureza compulsiva da busca e execução do vício é coerente com os estudos e evidências empíricas que se acumularam ao longo de anos. Dito isso, vícios não são punitivos durante a execução desses, pelo contrário, o cérebro entende a ação realizada como positiva devido à alta carga dopaminérgica liberada pelo gatilho responsável pelo impulso inicial. Essa carga é liberada antes de ações que são reconhecidas como positivas pelo cérebro e toda vez que essa ação é repetida, e obtém resultados positivos, ela é reforçada (SCHULTZ et al, 1993).

Para que uma ação seja considerada positiva pelo cérebro, esse tem que ter ciência dos efeitos positivos gerados por tal, um exemplo clássico, é o experimento realizado por Schultz (1993), em que primatas eram inicialmente ensinados que dentro de um recipiente existia suco de maçã, o que fazia com que esse reconhecesse a recompensa inicial. Após isso, uma luz e um botão eram colocados próximos ao recipiente, e esses primatas eram ensinados que sempre que a luz ligasse e o botão fosse apertado mais suco entraria no recipiente, formando assim um reforço para a criação daquele hábito. Mais tarde no teste, houve uma mudança, quando a luz acendesse e o primata apertasse o botão, o suco não seria despejado, devido a isso notou-se que os primatas continuavam repetindo a ação de apertar o botão quando a luz acendia. Porém, quanto mais tempo se passava sem a queda do suco, a ação de apertar o botão era menos recorrente. Ao analisar esse experimento, é notável que caso o macaco não fosse ensinado que o recipiente continhao suco de maçã, não existiria a motivação inicial para o aprendizado das próximas etapas, logo o reconhecimento da recompensa final é necessário para a motivação inicial.

Assim, é possível uma pessoa adicta “quebrar” o vício por meio do reconhecimento da recompensa final, e utilizar esse conhecimento como motivador, o que configura esse como uma comportamentos direcionados a um objetivo. Caso essa pessoa consiga reconhecer ações tomadas por hábito e utilizar dos comportamentos orientados ao objetivo, é possível moldar o comportamento associado ao gatilho por meio da degradação de contingência. No entanto isso é um desafio de ser feito no dia a dia (MENDELSOHN, 2019).

Desta forma, esse projeto tem como objetivo geral o desenvolvimento de uma aplicação *mobile* que dê assistência na formação de um hábito escolhido pelo usuário, por meio do reconhecimento da recompensa final, gerado por um modelo de linguagem de grande escala; técnicas de *gameficação* durante o uso da aplicação; e um ambiente social para a formação do hábito, além de buscar um design moderno e uma boa experiência de usabilidade para o usuário.

Para tanto, os objetivos específicos são: avaliação dos estudos sobre hábitos no tópico da neurociência comportamental e desenvolvimento de uma aplicação *mobile* e suas *API*s para formação de hábitos.

# FUNDAMENTAÇÃO TEÓRICA

Nesta sessão será explorado de forma completa os conceitos supracitados, trazendo embasamento teórico e técnico dos recursos explorados para desenvolvimento do projeto. Tais como: formação de hábitos, estruturas de projetos, bancos de dados, modelos de linguagem de grande escala e desenvolvimento de código com linguagens: *Flutter*, *Java* e *Python*.

## Ações voluntárias

Ações voluntárias são distintas, e definidas por teoristas do aprendizado, em duas variedades: ações orientada a um objetivo e hábitos (BOUTON, 2021).

### Ação orientada a um objetivo

Uma ação pode ser descrita como um comportamento instrumental, com a função de levar uma pessoa a um objetivo, dado que essa esteja mentalmente engajada durante o processo da ação, ou seja, com a representação do objetivo em memória (BOUTON, 2021).

Este tipo de ação é o que permite uma pessoa a tomar decisões que estão fora de sua rotina e que não irão trazer nenhum tipo de benefício a curto prazo. Normalmente essas ações têm uma baixa carga dopaminérgica e demandam um grande esforço mental para serem concluídas. Algumas dessas ações são, por exemplo: uma pessoa que não tinha o hábito de estudar, começa a estudar com o objetivo de passar em um concurso; uma pessoa que não tinha o hábito de ir na academia, começa a ir na academia e se alimentar melhor com o objetivo de se tornar mais saudável.

### Hábito

Um hábito pode ser caracterizado como uma rotina, comportamento ou processo cognitivo que começa espontaneamente, mas é repetido de maneira automática como resultado de uma experiência própria. Ademais, hábitos são dependentes de contextos, pois eles se tornam mais fortes por meio de repetição e associações a dicas do ambiente, de maneira que esses se tornam dependentes de pistas relevantes (MENDELSOHN, 2019).

Além disso, sabe-se que um hábito não requer muito valor cognitivo, visto que esses são performados de maneira rápida e automática, além disso eles tendem a ser inflexíveis. Dessa maneira, hábitos servem um propósito crítico em tornar nosso comportamento mais eficiente, reduzindo a carga mental em decisões de menor importância, o que faz com que tomadas de decisões mais importantes tenham mais energia mental disponível (MENDELSOHN, 2019).

Por fim, sabe-se que o hábito é um comportamento que pode, eventualmente, levar uma pessoa a um objetivo, mas é performado de maneira automática, sem que se tenha um objetivo em “mente”. Uma ação orientada a objetivo pode se tornar um hábito com repetição extensiva e prática (BOUTON, 2021).

### Dopamina

De acordo com Cleveland Clinic[[1]](#footnote-1) (2022), a dopamina é um neurotransmissor produzido no cérebro que tem a função de agir como o “centro de recompensa” e tem participação ativa em diversas funções corporais, como memória, movimento, motivação e humor.

Na formação dos hábitos, a dopamina tem uma função importante, pois esse hormônio é o responsável pela motivação necessária para a tomada das decisões de ações orientadas a hábitos. O reconhecimento da recompensa ao executar a ação é essencial para que a dopamina seja liberada antes da ação nas próximas vezes. Um hábito é formado pela repetição da ação, que por sua vez também aumenta a quantidade de dopamina liberada previamente à ação (SCHULTZ et al, 1993).

## Linguagens de Programação

Uma linguagem de programação é um conjunto de regras e sintaxes que permite aos programadores escreverem instruções que um computador pode entender e executar. Essas linguagens são usadas para criar *software*, *aplicativos*, *scripts* e outros tipos de sistemas computacionais.Nesse projeto foram utilizadas as linguagens de programação: *Dart*, *Java* e *Python*.

### *Dart*

### *Dart* é a linguagem de programação onde o framework *Flutter* é escrito e compilado. Essa linguagem foi primeiro lançada em 14 de novembro de 2013 e teve sua segunda versão lançada em agosto de 2018. Dart foi popularizado principalmente após o lançamento do *Flutter*, atualmente seu *framework* principal.

### *Flutter* é um framework de código aberto desenvolvido pelo Google, com o objetivo de criar telas modernas, de compilação nativa e multi plataforma com apenas um código base. Esse foi lançado em maio de 2017 e no ano de 2022 teve sua terceira versão lançada.

O *Flutter* consiste tanto de uma linguagem de criação de interface de usuário, quanto de um motor de renderização. Dessa maneira, aplicativos *flutter* podem ser escrito em apenas uma base de código e esses podem ser renderizados para diversas plataformas.  
 Dessa maneira, o flutter será a linguagem utilizada nesse projeto para o desenvolvimento das telas *frontend[[2]](#footnote-2)*, visto que com essa é possível abranjer um maior número de dispositivos com apenas uma base de código.

### *Java*

*Java* é uma linguagem de programação orientada a objetos desenvolvida pela Sun Microsystems, e lançada em 1995. Projetada para ser simples, robusta e de uso geral, *Java* permite a criação de aplicativos portáteis e seguros, que podem ser executados em qualquer plataforma que possua uma Java Virtual Machine (JVM). Uma das características mais destacadas de *Java* é o seu conceito de "*Write Once, Run Anywhere*" (WORA), que garante a compatibilidade de código em diferentes sistemas operacionais sem a necessidade de modificações.

Além disso, Java é amplamente utilizada em desenvolvimento de aplicações empresariais, sistemas web, aplicativos móveis (particularmente para a plataforma Android), e grandes sistemas de *backend[[3]](#footnote-3)*, devido à sua alta performance, escalabilidade e um ecossistema rico de bibliotecas e frameworks. A linguagem também se destaca por seu gerenciamento automático de memória, através do coletor de lixo (*garbage collector*), que ajuda a prevenir vazamentos de memória e melhora a segurança e estabilidade das aplicações.

Dessa maneira, nesse projeto o desenvolvimento do *backend* será realizado em sua maioria em *Spring[[4]](#footnote-4), que é um* *framework* desenvolvido em *Java*. Pois essa é uma linguagem segura e resiliente.

### Python

*Python* é uma linguagem de programação de alto nível, interpretada e de propósito geral, desenvolvida por Guido van Rossum e lançada pela primeira vez em 1991. Caracteriza-se por sua sintaxe clara e concisa, que promove a legibilidade do código e a facilidade de manutenção.

Além disso, Python é amplamente utilizado em diversas áreas, como desenvolvimento web, automação de scripts, análise de dados, inteligência artificial e aprendizado de máquina, graças à sua vasta biblioteca padrão e ao rico ecossistema de bibliotecas e frameworks de terceiros. Ademais, a linguagem adota paradigmas de programação orientada a objetos, funcional e imperativa, oferecendo flexibilidade e poder aos desenvolvedores.

Assim, nesse projeto haverá o desenvolvimento de um serviço *backend* em *FastAPI[[5]](#footnote-5),* que é um framework desenvolvido em *Python*. Isso pois, além de ser uma linguagem resiliente para o desenvolvimento de aplicações *backend,* essa tem uma biblioteca ofertada pela OpenAI, que facilita o desenvolvimento de requisições para o modelo de linguagem de grande escala desenvolvido por eles.

## Modelo de Linguagem de Grande Escala

Os modelos de linguagem de grande escala, como o GPT-4 desenvolvido pela OpenAI, representam um avanço significativo na área de processamento de linguagem natural (PLN). Esses modelos são treinados em enormes quantidades de dados textuais, utilizando redes neurais profundas, especificamente arquiteturas de transformadores, para capturar nuances complexas da linguagem humana.

Ademais, esses modelos são capazes de gerar texto coerente, responder a perguntas, traduzir idiomas e até mesmo criar conteúdo original. Assim, nesse projeto, serão feitas requisições a API da OpenAI, para que sejam gerados conteúdos específicos para cada usuário da aplicação.

## Banco de Dados

Um banco de dados é um sistema organizado para armazenar, gerenciar e recuperar informações de forma eficiente e estruturada. Entre os diversos sistemas de gerenciamento de banco de dados (SGBD) disponíveis, o MySQL destaca-se como uma solução robusta e amplamente utilizada, especialmente em aplicações web.

Portanto, nesse projeto, o MySQL será utilizado para que as informações do usuário sejam salvas, pois esse é conhecido por sua alta performance, escalabilidade e compatibilidade com diversas plataformas e linguagens de programação.

## Estrutura de Projetos

Na comunidade do desenvolvimento de software, discussões sobre conceitos e regras para garantia da qualidade do código são recorrentes e pertinentes. Durante a evolução da literatura da área de Engenharia de Software diversos conceitos se tornaram relevantes para o desenvolvimento.

### Código Limpo – *Clean Code*

O conceito de código limpo, descrito inicialmente por Robert C. Martin, criador e fundador da empresa Clean Coders e um dos 17 signatários originais do manifesto ágil. Esse termo foi citado e popularizado em sua obra Código Limpo: Habilidades práticas de Agile Software, no ano de 2008. Nela um conjunto de diretrizes e boas práticas eram definidas para que fossem aplicadas no dia a dia do desenvolvedor.

Assim, nesse projeto, princípios e diretrizes definidos na obra serão aplicados, como: nomenclaturas significativas, ou seja, nomes descritivos e sem abreviaturas; funções pequenas e objetivas; evitar código replicado, seguindo o padrão DRY (*Don’t repeat yourself*)[[6]](#footnote-6); simplicidade, assim como descrito em KISS (*Keep it Stupid Simple)[[7]](#footnote-7)* e YAGNI (*You aren’t gonna need it*)[[8]](#footnote-8); e separação de conceitos, mantendo classes relacionadas juntas e evitando a dependência entre módulos.

### *Bloc*

O padrão *Bloc* (*Business Logic Component)[[9]](#footnote-9)* é uma abordagem arquitetural utilizada no desenvolvimento de aplicações Flutter, que permite a separação entre a interface do usuário e a lógica de negócios. Inspirado nesse padrão, uma biblioteca, também chamada *Bloc*, foi criada com o intuito de gerenciar estado previsível em Dart. No *Bloc*, ações de usuários são registradas como eventos e são enviados para a camada de lógica da aplicação, essa que, por sua vez, processa esses eventos e emite estados para que a camada de interface do usuário consiga demonstrar o valor correto.

Dessa maneira, esse padrão foi o escolhido para o desenvolvimento do *frontend* do código, pois essa separação clara entre as responsabilidades contribui para um código mais modular e de fácil manutenção, assim como demonstrado na imagem 1.

**Figura 1 –** Arquitetura Bloc da aplicação *frontend*

A diagram of a program

Description automatically generated

Fonte: Autoria própria, 2024.

## *API*

De acordo com a AWS[[10]](#footnote-10) (Amazon Web Services), uma *API* (*Application Programming Interface)[[11]](#footnote-11)* é uma interface que disponibiliza o contrato de serviço entre duas aplicações, essas que são qualquer software de funções distintas. Esse contrato define como ambas as aplicações respondem a determinadas requisições.

Ademais, *API*s podem funcionar de quatro maneiras: SOAP, onde é seguido o protocolo de acesso a objetos simples; RPC, onde é seguido o padrão de chamadas de procedimento remoto; WebSocket, onde é oferecido suporte à comunicação bidirecional entre aplicativos cliente e o servidor; e REST, que são as *API*s mais populares e flexíveis do mercado atualmente.

Para esse projeto, o *backend*, que consta de *API*’s desenvolvidas em *Spring* e *FastAPI*, se comunicarão utilizando do padrão REST com o frontend e com API’s externas, como a *API* do modelo de linguagem de grande escala da OpenAI.

## Conteinerização

A conteinerização é uma tecnologia que permite empacotar uma aplicação e todas as suas dependências em um único contêiner, garantindo que ela possa ser executada de forma consistente em qualquer ambiente.

Os contêineres compartilham o mesmo *kernel[[12]](#footnote-12)* do sistema operacional, mas operam em ambientes isolados, oferecendo a vantagem de serem mais leves e eficientes em comparação com as máquinas virtuais. Esta abordagem promove uma maior portabilidade e escalabilidade das aplicações, facilitando o gerenciamento e a orquestração de múltiplos contêineres através de ferramentas como Docker Compose.

Assim, nesse projeto, para que seja possível a execução das aplicações em diversos ambientes, será utilizado do Docker para isolar o *backend* e do Docker Compose para orquestrar e gerenciar as múltiplas imagens.

## Padrões de Segurança

Citar aqui sobre os padrões usados para segurança na autenticação.

## *Github*

Explicar sobre o github, citar sobre o projeto ser mono repo. As vezes mudar esse tópico para Sistema de Controle de versão

# METODOLOGIA

## Projeto base

## Jornada da aplicação

# RESULTADOS E DISCUSSÕES

## Configuração do ambiente de testes

## Aplicação de testes de *software*

# CONCLUSÃO

# REFERÊNCIAS

BOUTON, M. E. Context, attention, and the switch between habit and goal-direction in behavior. Learning & behavior, v. 49, n. 4, p. 349–362, 2021. Disponível em: https://doi.org/10.3758/s13420-021-00488-z. Acesso em: 09 jun. 2024.

BRASIL. **Brasil tem mais de 234 milhões de acessos móveis em 2020**. Ministério das Comunicações, 2021. Disponível em: https://www.gov.br/pt-br/noticias/transito-e-transportes/2021/04/brasil-tem-mais-de-234-milhoes-de-acessos-moveis-em-2020. Acesso em: 9 jun. 2024.

CLEVELAND CLINIC.

GLOBAL WEB INDEX**. Digital vs Traditional Media Consumption: analyzing time devoted to online and traditional forms of media at a global level, as well as by age and across countries**. Insight Report, Q1 2017. Disponível em: https://www.gwi.com/hubfs/Digital\_vs\_Traditional\_Media\_Consumption.pdf. Acesso em: 09 jun. 2024.

MENDELSOHN, A. I. **Creatures of habit: the neuroscience of habit and purposeful behavior**. Biological Psychiatry, v. 85, n. 11, p. e49–e51, 2019. Disponível em: https://doi.org/10.1016/j.biopsych.2019.03.978. Acesso em: 9 jun. 2024.

OSTLUND, S. B.; BALLEINE, B. W. **On habits and addiction: an associative analysis of compulsive drug seeking**. Drug discovery today. Disease models, v. 5, n. 4, p. 235–245, 2008. Disponível em: https://doi.org/10.1016/j.ddmod.2009.07.004. Acesso em: 09 jun. 2024.

Schultz, W., Apicella, P., & Ljungberg, T. (1993). **Responses of monkey dopamine neurons to reward and conditioned stimuli during successive steps of learning a delayed response task**. The Journal of neuroscience : the official journal of the Society for Neuroscience, 13(3), 900–913. <https://doi.org/10.1523/JNEUROSCI.13-03-00900.1993>.

SHOUKAT, S. **Cell phone addiction and psychological and physiological health in adolescents**. EXCLI journal, v. 18, p. 47–50, 2019.

1. Disponível em < <https://my.clevelandclinic.org/health/articles/22581-dopamine>>. Acesso em: 10 jun. 2024. [↑](#footnote-ref-1)
2. *Frontend* é onde se encontra a parte visual da aplicação. Ou seja, o aplicativo que irá rodar no celular do usuário. [↑](#footnote-ref-2)
3. *Backend* é onde se encontra a parte lógica da aplicação. Ou seja, os servidores e bancos de dados que são responsáveis pelo processamento de dados. [↑](#footnote-ref-3)
4. Disponível em < <https://spring.io/> >. Acesso em 10 de jun. de 2024. [↑](#footnote-ref-4)
5. Disponível em < <https://fastapi.tiangolo.com/> >. Acesso em 10 de jun. de 2024. [↑](#footnote-ref-5)
6. Não repita a si mesmo [↑](#footnote-ref-6)
7. Mantenha estupidamente simples [↑](#footnote-ref-7)
8. Você não vai precisar [↑](#footnote-ref-8)
9. Componente de Lógica de Negócios [↑](#footnote-ref-9)
10. Disponível em < <https://aws.amazon.com/pt/what-is/api/#:~:text=API%20significa%20Application%20Programming%20Interface,de%20servi%C3%A7o%20entre%20duas%20aplica%C3%A7%C3%B5es>. >. Acesso em: 10 de jun. de 2024 [↑](#footnote-ref-10)
11. Interface de Programação de Aplicação [↑](#footnote-ref-11)
12. Núcleo do sistema operacional que gerencia os recursos do hardware e fornece serviços essenciais para a execução de todos os outros componentes de software. [↑](#footnote-ref-12)