API

(Application Programing Interface)

## **1.0 - Introdução: O que é uma API?**

Uma API é, de modo geral, uma interface (composta por ferramentas, definições e protocolos) que permite a comunicação entre dois componentes de software. Por meio dessa comunicação são trocados dados, serviços e recursos entre as partes de modo que este processo seja abstrato. Ou seja, apenas as informações relevantes são acessadas pela API.

Deste modo, as implementações, ou detalhes ocultos, dos componentes de software são alheias entre si. Deste modo, a API torna o acesso ao componente mais seguro e controlado. APIs permitem o compartilhamento de pequenos pacotes de dados e serviços isolados relevantes especificamente para a natureza das requisições(1).

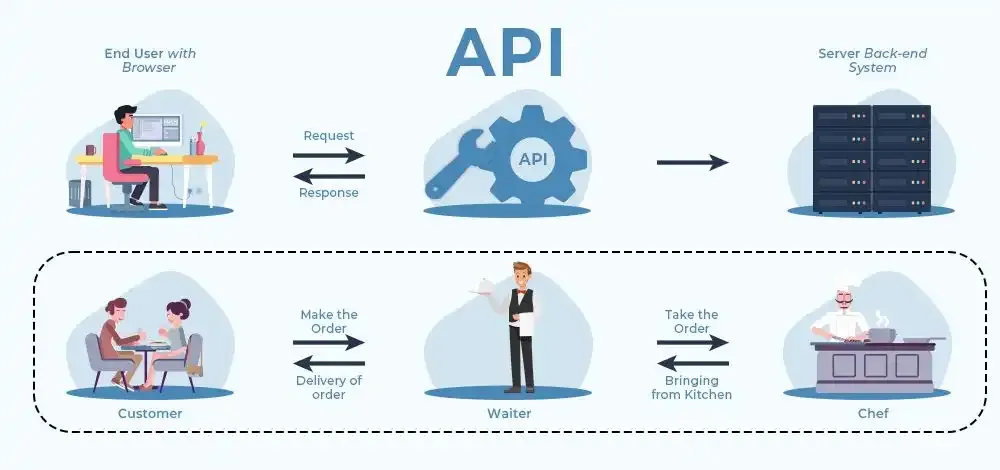
## **2.0: Desenvolvimento: Como uma API funciona?**

É útil pensar na comunicação da API em termos de solicitação e resposta entre um cliente e um servidor. O aplicativo que envia a solicitação é o cliente e o servidor fornece a resposta. A API é a ponte que estabelece a conexão entre eles(1).

Uma maneira simples de entender como as APIs funcionam é analisar um exemplo comum: o processamento de pagamentos de terceiros. Quando um usuário compra um produto em um site decomercio eletronico.o site pode pedir ao usuário para "Pagar com PayPal" ou outro tipo de sistema de terceiros. Essa função depende de APIs para fazer a conexão(1).

Para trocar informações entre APIs, os desenvolvedores devem fornecer a documentação da API que descreve os tipos de solicitações que uma API pode aceitar, os casos de uso para os quais ela foi projetada e quaisquer outras condições (por exemplo protocolos, esquemas e requisitos de segurança) que terceiros devem seguir(3).

É importante ressaltar que APIs são utilizadas em diferentes contextos e não necessitam ser apenas web APIs. Quando se utiliza qualquer sistema operacional, recursos deste são acessados por meio de APIs. O mesmo ocorre para SGBDs quando uma aplicação acessa um banco de dados. Por fim, as APIs web definem protocolos para clientes acessarem (via web) recursos e serviços localizados em servidores.

Figura 1 - Analogia sobre o funcionamento de uma API com serviço de atendimento de um restaurante(2).

## **2.1 End-points e requisições HTTP:**

Um endpoint de API aceita a chamada de API e retorna as informações solicitadas.Tanto os clientes quanto os endpoint de API referem-se a programas de software hospedados em servidores em vez de dispositivos de hardware diferentes.

Os servidores de API podem hospedar vários endpoints, sendo que cada um deles recebe um Uniform Resource Identifier (URI) que permite que seja localizado por um cliente de API. Na maioria dos casos, esse URI é um localizador uniforme de recursos (URL), que aponta para locais baseados na internet (por exemplo, um site)(3).

Hoje, a maioria das APIs são APIs da web. APIs da web são um tipo de API remota (o que significa que a API usa protocolos para manipular recursos externos) que expõem os dados e funcionalidades de uma aplicação pela internet. Estas APIs utilizam o protocolo HTTP.

Este protocolo define um conjunto de métodos de requisição responsáveis por indicar a ação a ser executada para um dado recurso. Embora esses métodos possam ser descritos como substantivos, eles também são comumente referenciados como HTTP Verbs (Verbos HTTP). Cada um deles implementa uma semântica diferente, mas alguns recursos são compartilhados por um grupo deles, como por exemplo, qualquer método de requisição pode ser do tipo safe, idempotent ou cacheable(4).

**GET**: O método GET solicita a representação de um recurso específico. Requisições utilizando o método GET devem retornar apenas dados.

**POST**: O método POST é utilizado para submeter uma entidade a um recurso específico, frequentemente causando uma mudança no estado do recurso ou efeitos colaterais no servidor.

**PUT**: O método PUT substitui todas as atuais representações do recurso de destino pela carga de dados da requisição.

**DELETE**: O método DELETE remove um recurso específico.

## **3.0: Conclusão:**

Em suma, APIs (Application Programming Interfaces) são interfaces essenciais que permitem a comunicação padronizada e segura entre diferentes componentes de software, abstraindo complexidades e facilitando a troca de dados e serviços. Elas funcionam como pontes entre clientes e servidores, utilizando protocolos como HTTP e métodos específicos (GET, POST, PUT, DELETE) para operações como recuperação, criação, atualização e exclusão de recursos.

Além de impulsionar integrações em sistemas web (como pagamentos terceirizados), APIs estão presentes em diversos contextos, desde sistemas operacionais até bancos de dados, garantindo interoperabilidade e eficiência. A documentação clara e a adoção de padrões são fundamentais para seu funcionamento adequado. Em suma, as APIs são pilares da tecnologia moderna, viabilizando inovações e conectividade em ambientes digitais.

referencias:  
  
1 - https://www.ibm.com/br-pt/think/topics/api

2 -<https://www.alura.com.br/artigos/api?srsltid=AfmBOoqJc46NVWKDz_lv_tyapDJNFPBCApuOvBWy69-L4jWVf0JBnNp5>

3 - <https://www.cloudflare.com/pt-br/learning/security/api/how-do-apis-work/>

4 - https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Reference/Methods