**O que é uma API?**

APIs são mecanismos que permitem que dois componentes de software se comuniquem usando um conjunto de definições e protocolos. Por exemplo, o sistema de software do instituto meteorológico contém dados meteorológicos diários. O aplicativo meteorológico em seu telefone “fala” com este sistema por meio de APIs e mostra atualizações meteorológicas diárias no telefone.

**O que significa API?**

API significa Application Programming Interface (Interface de Programação de Aplicação). No contexto de APIs, a palavra Aplicação refere-se a qualquer software com uma função distinta. A interface pode ser pensada como um contrato de serviço entre duas aplicações. Esse contrato define como as duas se comunicam usando solicitações e respostas. A documentação de suas respectivas APIs contém informações sobre como os desenvolvedores devem estruturar essas solicitações e respostas.

**Como as APIs funcionam?**

A arquitetura da API geralmente é explicada em termos de cliente e servidor. A aplicação que envia a solicitação é chamada de cliente e a aplicação que envia a resposta é chamada de servidor. Então, no exemplo do clima, o banco de dados meteorológico do instituto é o servidor e o aplicativo móvel é o cliente.

Existem quatro maneiras diferentes pelas quais as APIs podem funcionar, dependendo de quando e por que elas foram criadas.

**APIs SOAP**

Essas APIs usam o Simple Object Access Protocol (Protocolo de Acesso a Objetos Simples). Cliente e servidor trocam mensagens usando XML. Esta é uma API menos flexível que era mais popular no passado.

**APIs RPC**

Essas APIs são conhecidas como Remote Procedure Calls (Chamadas de Procedimento Remoto). O cliente conclui uma função (ou um procedimento) no servidor e o servidor envia a saída de volta ao cliente.

**APIs WebSocket**

[API Websocket](https://docs.aws.amazon.com/apigateway/latest/developerguide/apigateway-websocket-api-overview?pg=wianapi&cta=websocketapi) é outro desenvolvimento de API da Web moderno que usa objetos JSON para passar dados. Uma API WebSocket oferece suporte à comunicação bidirecional entre aplicativos cliente e o servidor. O servidor pode enviar mensagens de retorno de chamada a clientes conectados, tornando-o mais eficiente que a API REST.

**APIs REST**

Essas são as APIs mais populares e flexíveis encontradas na Web atualmente. O cliente envia solicitações ao servidor como dados. O servidor usa essa entrada do cliente para iniciar funções internas e retorna os dados de saída ao cliente. Vejamos as APIs REST em mais detalhes abaixo.

**O que são APIs REST?**

REST significa Transferência Representacional de Estado. REST define um conjunto de funções como GET, PUT, DELETE e assim por diante, que os clientes podem usar para acessar os dados do servidor. Clientes e servidores trocam dados usando HTTP.

A principal característica da [API REST](https://docs.aws.amazon.com/apigateway/latest/developerguide/http-api-vs-rest?pg=wianapi&cta=restapi) é a ausência de estado. A ausência de estado significa que os servidores não salvam dados do cliente entre as solicitações. As solicitações do cliente ao servidor são semelhantes aos URLs que você digita no navegador para visitar um site. A resposta do servidor corresponde a dados simples, sem a renderização gráfica típica de uma página da Web.

**O que é API Web?**

Uma API Web ou API de serviço da Web é uma interface de processamento de aplicações entre um servidor da Web e um navegador da Web. Todos os serviços da Web são APIs, mas nem todas as APIs são serviços da Web. A API REST é um tipo especial de API Web que usa o estilo de arquitetura padrão explicado acima.

Os diferentes termos que abrangem as APIs, como API Java ou APIs de serviço, existem porque, historicamente, as APIs foram criadas antes da World Wide Web. As APIs Web modernas são APIs REST e os termos podem ser usados de forma intercambiável.

**O que são integrações de API?**

As integrações de API são componentes de software que atualizam automaticamente os dados entre clientes e servidores. Alguns exemplos de integrações de API são quando os dados automáticos são sincronizados com a nuvem por meio da galeria de imagens do seu telefone ou a data e a hora são sincronizadas automaticamente no seu laptop quando você viaja para um local com outro fuso horário. As empresas também podem usá-las para automatizar com eficiência muitas funções do sistema.

**Quais são os benefícios das APIs REST?**

As APIs REST oferecem quatro principais benefícios:

**1. Integração**

As APIs são usadas para integrar novas aplicações com sistemas de software existentes. Isso aumenta a velocidade de desenvolvimento porque cada funcionalidade não precisa ser escrita do zero. Você pode usar APIs para aproveitar o código existente.

**2. Inovação**

Setores inteiros podem mudar com a chegada de uma nova aplicação. As empresas precisam responder rapidamente e oferecer suporte à rápida implantação de serviços inovadores. Elas podem fazer isso fazendo alterações no nível da API sem precisar reescrever todo o código.

**3. Expansão**

As APIs apresentam uma oportunidade única para as empresas atenderem às necessidades de seus clientes em diferentes plataformas. Por exemplo, a API de mapas permite a integração de informações de mapas por meio de sites, Android, iOS etc. Qualquer empresa pode fornecer acesso semelhante aos seus respectivos bancos de dados internos usando APIs gratuitas ou pagas.

**4. Facilidade de manutenção**

A API atua como um gateway entre dois sistemas. Cada sistema é obrigado a fazer alterações internas para que a API não seja afetada. Dessa forma, qualquer alteração futura de código feita por uma parte não afetará a outra parte.

**Quais são os diferentes tipos de API?**

As APIs são classificadas de acordo com sua arquitetura e escopo de uso. Já exploramos os principais tipos de arquiteturas de API, agora, vamos dar uma olhada no escopo de uso.

**APIs privadas**

Elas são internos a uma empresa e são usadas apenas para conectar sistemas e dados dentro da empresa.

**APIs públicas**

Estas são abertas ao público e podem ser usadas por qualquer pessoa. Pode ou não haver alguma autorização e custo associado a esses tipos de APIs.

**APIs de parceiros**

Estas são acessíveis apenas por desenvolvedores externos autorizados para auxiliar as parcerias entre empresas.

**APIs compostas**

Estas combinam duas ou mais APIs distintas para atender a requisitos ou comportamentos complexos do sistema.

**O que é um endpoint de API e por que ele é importante?**

Os endpoints da API são os pontos de contato finais no sistema de comunicação da API. Estes incluem URLs de servidores, serviços e outros locais digitais específicos de onde as informações são enviadas e recebidas entre sistemas. Os endpoints da API são fundamentais para as empresas por dois motivos principais:

**1. Segurança**

Os endpoints da API tornam o sistema vulnerável a ataques. O monitoramento da API é crucial para impedir o uso indevido.

**2. Performance**

Os endpoints da API, especialmente os de alto tráfego, podem causar gargalos e afetar a performance do sistema.

Como proteger uma API REST?

Todas as APIs devem ser protegidas por meio de autenticação e monitoramento adequados. As duas principais maneiras de proteger APIs REST incluem:

**1. Tokens de autenticação**

Eles são usados para autorizar os usuários a fazer a chamada de API. Os tokens de autenticação verificam se os usuários são quem afirmam ser e se têm direitos de acesso para aquela chamada de API específica. Por exemplo, quando você faz login em seu servidor de e-mail, seu cliente de e-mail usa tokens de autenticação para acesso seguro.

**2. Chaves de API**

As chaves de API verificam o programa ou a aplicação que faz a chamada de API. Elas identificam a aplicação e garantem que ela tenha os direitos de acesso necessários para fazer a chamada de API específica. As chaves de API não são tão seguras quanto os tokens, mas permitem o monitoramento da API para coletar dados sobre o uso. Você pode ter notado uma longa sequência de caracteres e números no URL do seu navegador ao visitar diferentes sites. Essa string é uma chave de API que o site usa para fazer chamadas de API internas.

**Como criar uma API?**

Diligência prévia e esforços são necessários para criar uma API com a qual outros desenvolvedores desejem trabalhar e na qual queiram confiar. Estas são as cinco etapas necessárias para o design de uma API de alta qualidade:

**1. Planejar a API**

As especificações da API, como OpenAPI, fornecem o esquema para o design da sua API. É melhor pensar em diferentes casos de uso com antecedência e garantir que a API esteja de acordo com os padrões atuais de desenvolvimento de API.

**2. Criar a API**

 Os designers de APIs prototipam APIs usando código boilerplate. Depois que o protótipo é testado, os desenvolvedores podem personalizá-lo de acordo com as especificações internas.

**3. Testar a API**

O teste de API é o mesmo que o teste de software e deve ser feito para evitar bugs e defeitos. As ferramentas de teste de API podem ser usadas para testar a resistência da API contra ataques cibernéticos.

**4. Documentar a API**

Embora as APIs sejam autoexplicativas, a documentação da API funciona como um guia para melhorar a usabilidade. APIs bem documentadas que oferecem uma série de funções e casos de uso tendem a ser mais populares em uma arquitetura orientada a serviços.

**5. Comercializar a API**

Assim como a Amazon é um marketplace online para varejo, existem marketplaces de API para desenvolvedores comprarem e venderem outras APIs. Catalogar sua API pode permitir que você ganhe dinheiro com ela.

## Exemplo de uso de API

Se você é desenvolvedor, é muito comum que tenha que conectar sua aplicação com um banco de dados para que a parte visual da sua aplicação se comunique com o servidor. Esse mesmo servidor, uma vez comunicando-se com o seu banco de dados, poderá levar informações do banco direto para o seu front, ou seja, para o cliente.

Alguns exemplos com esse fluxo, seu front fazendo o que chamamos de requisições para API e que interajam com o seu banco de dados são:

* Clicar em algum botão que crie uma conta, onde os dados inseridos por você terão que se comunicar de alguma forma com o banco de dados para serem inseridos nele; quem fará essa comunicação é a API;
* Listar produtos de um e-commerce, onde novamente os mesmo produtos estarão cadastrados no banco de dados, e serão visíveis para o cliente através da API.

