



PodCast

Interação com APIs

Para entender a parte de interação com APIs, nós temos que entender alguns conceitos como:

- O que é uma API?
- API Restful
- CRUD
- Como interagir com essas APIs?

Vamos responder à essas perguntas nesse podcast. Vamos lá?

O que é uma API?

API é uma sigla para *Application Programming Interface*, que traduzido significa interface de programação de aplicações.

API é uma interface entre diferentes aplicações e tem como função possibilitar a comunicação entre diferentes sistemas.

Com a API, podemos criar soluções que se comunicam com vários sistemas diferentes, construindo aplicações e soluções cada vez mais completas e complexas.

Para entender melhor, podemos fazer uma analogia com um sistema de restaurante.

A API seria o garçom, que consegue interligar a cozinha, onde são preparados os pedidos, à mesa onde vai ser servido ao cliente que está no restaurante.

Além disso, há a possibilidade de um sistema de delivery também solicitar um pedido para essa cozinha. Da mesma forma que o entregador não tem acesso direto com a cozinha, uma página da web não acessa diretamente o banco de dados: ela precisa passar por um intermediário, que no caso é a API, que traz os dados para essa base.

O cliente que está na mesa não busca o prato dele diretamente na cozinha; ele solicita ao garçom, que traz a refeição da cozinha e serve o cliente.

Então, a API tem esse papel intermediário de interligar diferentes sistemas, como um banco de dados, uma página da web, um aplicativo Android e outros tipos de sistemas.

Resumindo, a API serve para integrar camadas ou sistemas diferentes.

API Restful

REST é um dos padrões que permitem que a API interligue sistemas diferentes e desenvolvidos com tecnologias diferentes sem que haja perda de comunicação. Esse padrão é formado por regras que determinam como uma API realiza a comunicação dentro desses sistemas.

A sigla Rest significa *Representational State Transfer*, do inglês transferência de estado representacional. É uma comunicação que se utiliza das informações de estado, ou seja, não armazena informações principais do lado do cliente, e sim no servidor, transferindo somente dados necessários.

Isso traz como vantagem uma maior performance em comparação ao padrão SOAP, que já foi muito utilizado anteriormente. Além disso, o REST é mais simples de entender.

Uma das regras do REST é Uniformização da Interface, que determina a identificação clara dos recursos de uma API, para que os sistemas possam interagir com a API e utilizar dessas funcionalidades.

Um **recurso de uma API** é o tipo de informação que essa API vai disponibilizar às aplicações que estão interagindo com ela.

CRUD

Sigla para as operações básicas que a maioria dos sistemas utiliza, na qual a letra C representa *Create* (criar), R seria *Read* (ler), U seria *Update* (atualizar) e D seria *Delete* (deletar). Essas são todas ações realizadas em um registro.

O conceito de CRUD é utilizado em vários sistemas informatizados.

Acompanhe um exemplo de CRUD aplicado a uma situação de redes sociais:

Imagine como uma rede social interage com as informações de um usuário.

Aqui, tratamos usuário como um recurso porque a API trabalha com um tipo de informação dos recursos. As operações de CRUD seriam aplicadas da seguinte forma:

- **Criar** um registro do recurso usuário: quando nos cadastramos em uma rede social, por exemplo.
- **Ler**, exibir registros ou exibir um recurso usuário específico: quando listamos nossos amigos em uma rede social, por exemplo. Nesse caso, ele exibe os registros de usuários.
- **Alterar** o registro de um recurso usuário: quando alteramos nossa imagem de perfil ou endereço ou alguma outra informação do nosso perfil.
- **Excluir** registro: quando excluímos a nossa conta, excluímos nosso usuário.