

REST API

REST API (transferência de estado representacional) refere-se a um conjunto específico de regras que determinam como os aplicativos de serviço da Web se comunicam pela Internet. Portanto, as APIs RESTful são APIs que seguem essas regras, proporcionando flexibilidade, largura de banda e vantagens de velocidade para integração e comunicação de software.

Uma API REST atua como intermediária entre um cliente e um servidor host, atendendo solicitações de recursos e entregando respostas por HTTP. No design REST, o cliente inicia a interação — por exemplo, acessar uma página da Web por meio de uma URL é uma solicitação HTTP comum. Para entender como as APIs RESTful transferem dados entre clientes e aplicativos, é útil dividir a operação em dois processos.

Solicitação do cliente: Um cliente é um programa ou pessoa que acessa um serviço de API. Por exemplo, se você integrar com um aplicativo da web como Instagram ou Youtube, seu programa se tornará o cliente. Se você solicitar uma URL, seu navegador será o cliente.

Resposta do servidor: Um recurso refere-se às informações específicas que uma API fornece a um cliente. Isso pode ser qualquer coisa, desde hashtags a perfis, comentários, endereços da web, tweets, etc. Todos os recursos têm um nome exclusivo chamado identificador de recurso e todos são armazenados em um servidor.

Quando um cliente faz uma solicitação usando uma API RESTful, o servidor transfere uma representação padronizada do estado do recurso para o sistema do cliente. Isso significa que o servidor não envia ao cliente o recurso real, mas uma representação de seu estado – ou seja, o estado do recurso em um determinado timestamp.

Restrições da API REST: Embora as APIs REST sejam apreciadas por sua simplicidade e flexibilidade, elas ainda precisam aderir a seis restrições para serem consideradas RESTful. Estes são os princípios orientadores do design da API REST

Separação cliente-servidor: REST limita explicitamente a capacidade de comunicação do cliente e do servidor. Os clientes fazem solicitações exclusivamente e os servidores respondem exclusivamente a elas. Portanto, o cliente inicia cada interação e conhece o URI do recurso solicitado. O servidor responde transmitindo os dados solicitados, mas não pode fazer modificações independentes. O isolamento de cliente e servidor facilita o gerenciamento. Não há necessidade de se preocupar com componentes modificando uns aos outros inadvertidamente durante mudanças independentes, simplificando o processo de otimização e dimensionamento.

Armazenável em cache: Como as APIs sem estado são projetadas para aumentar a sobrecarga de solicitações ao lidar com um grande número de chamadas de entrada e saída, as APIs REST devem priorizar o armazenamento em cache de dados. O armazenamento em cache refere-se ao armazenamento de dados no dispositivo cliente para economizar recursos do servidor e aumentar a largura de banda. As APIs REST classificam os dados em suas respostas como armazenáveis ou não armazenáveis em cache. Se os dados puderem ser

armazenados em cache, a API incluirá um período de tempo específico para o qual as informações podem ser armazenadas em cache. Quando esse tempo expirar, o cliente fará ping no servidor para atualizações.

A implantação de APIs RESTful oferece a oportunidade de interagir com empresas globais. Assim como você registra nomes para seus domínios ai ou implementa uma estratégia de liderança inovadora para estabelecer credibilidade no campo, a integração de seus programas com aplicativos populares consolida seus negócios como credíveis e confiáveis.