REST API

Framework Requirements of Communication

The framework to be used for effective communication must:

A. Enable communication over the network
B. Language neutral
C. Light
D. Fast

REST Framework is used to enable communication between two applications. REST Frameworks provides all the above-mentioned services.

REST stands for “representational state transfer”.

What is REST API?

REST API is an application programming interface(set of protocols used for building a web application) and it is a set of functions using which requests can be initiated and responses received and transfer data over the network. It is the way to communicate with other applications. The client-server communication model is based on HTTP protocol.

Example Use Case: Suppose you are doing a survey on the COVID health facilities and want to analyse the statistics of a government hospital in your area. You have built an application for analysis using machine learning which would input the statistics of Government hospitals. To transfer data between the web hospital management system and your application, a communication framework/interface is required, which would be the REST API. The hospital management system has to expose data, and the analysis application would fetch the data over the network. To fetch the data, the analysis application would need a data address(IP Address of the machine on which target data is present). This address of data is known as URI(Unique Resources Identifier). The application must know the operation to be used from the CRUD operations.
CRUD stands for Create, Read, Update and Delete. The operations are performed using HTTP calls.

Types of Requests:

A. GET -> Read data
B. POST -> Create/insert data
C. PUT -> Update data
D. DELETE-> Delete data

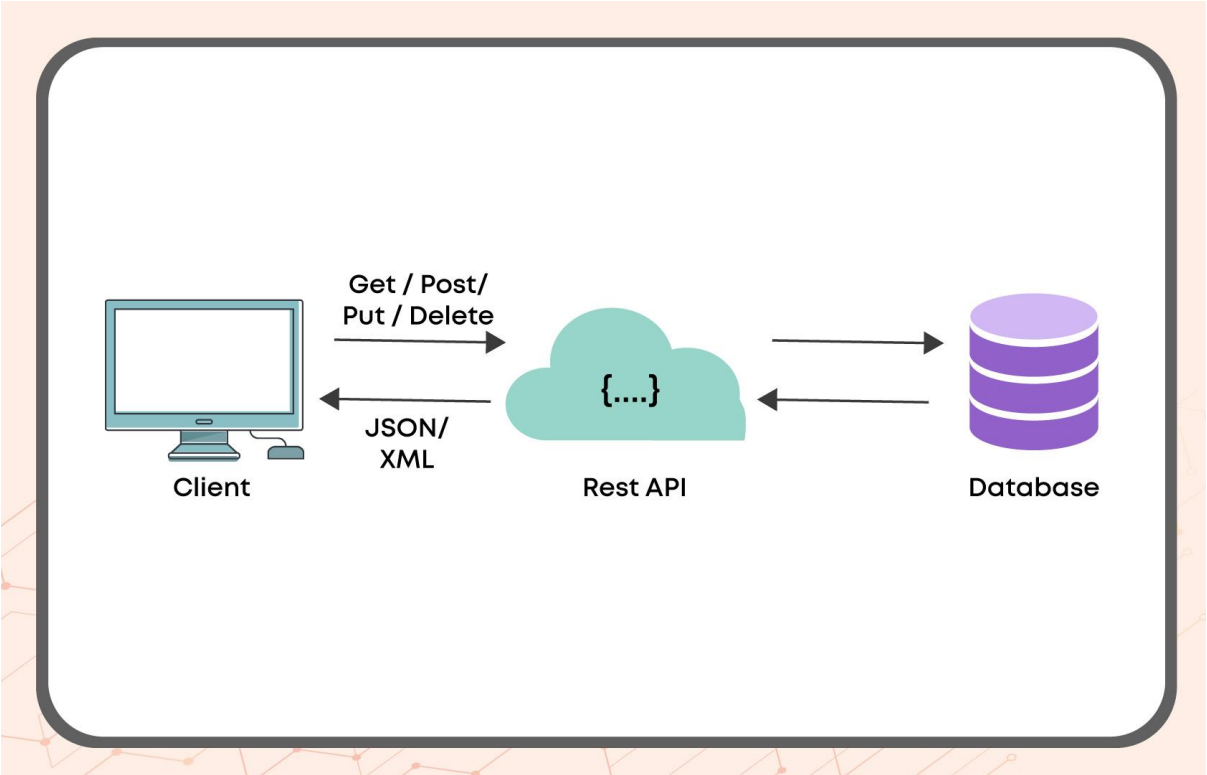

Fig: REST API

Advantages of using REST API

1. Simple: REST API is easier to adapt to the new syntax and build.
2. Resources: It uses fewer resources.
3. Reliability and Scalability: The client and server are separated, increasing scalability because the developers can scale the product better and less complex.

Drawbacks of REST API

1. REST typically relies on a few verbs (GET, POST, PUT, DELETE, and PATCH) which sometimes don't fit your use case.

For example, moving expired documents to the archive folder might not cleanly fit within these verbs.

2. Fetching complicated resources with nested hierarchies requires multiple round trips between the client and server to render single views, e.g. fetching content of a blog entry and the comments on that entry. For mobile applications operating in variable network conditions, these multiple roundtrips are highly undesirable.
3. Over time, more fields might be added to API response, and older clients will receive all new data fields, even those they do not need. As a result, it bloats the payload size and leads to larger latencies.