Lesson 9

API (Application Programming Interface) - REST API

What is a REST API?

A REST API (also known as RESTful API) is an application programming interface (API or web API) that conforms to the constraints of REST architectural style and allows for interaction with RESTful web services.

What is API?

An application programming interface (API) is a way for two or more computer programs to communicate with each other. It is a type of software interface, offering a service to other pieces of software.

REST?

REST is a set of architectural constraints, not a protocol or a standard. API developers can implement REST in a variety of ways.

REST?

In order for an API to be considered RESTful, it has to conform to these criteria:

- A client-server architecture made up of clients, servers, and resources, with requests managed through HTTP.
- Stateless client-server communication, meaning no client information is stored between get requests and each request is separate and unconnected.
- Cacheable data that streamlines client-server interactions.
- A uniform interface between components so that information is transferred in a standard form. This requires that:
  - resources requested are identifiable and separate from the representations sent to the client.
  - resources can be manipulated by the client via the representation they receive because the representation contains enough information to do so.
  - self-descriptive messages returned to the client have enough information to describe how the client should process it.
  - hypertext/hypermedia is available, meaning that after accessing a resource the client should be able to use hyperlinks to find all other currently available actions they can take.

REST?

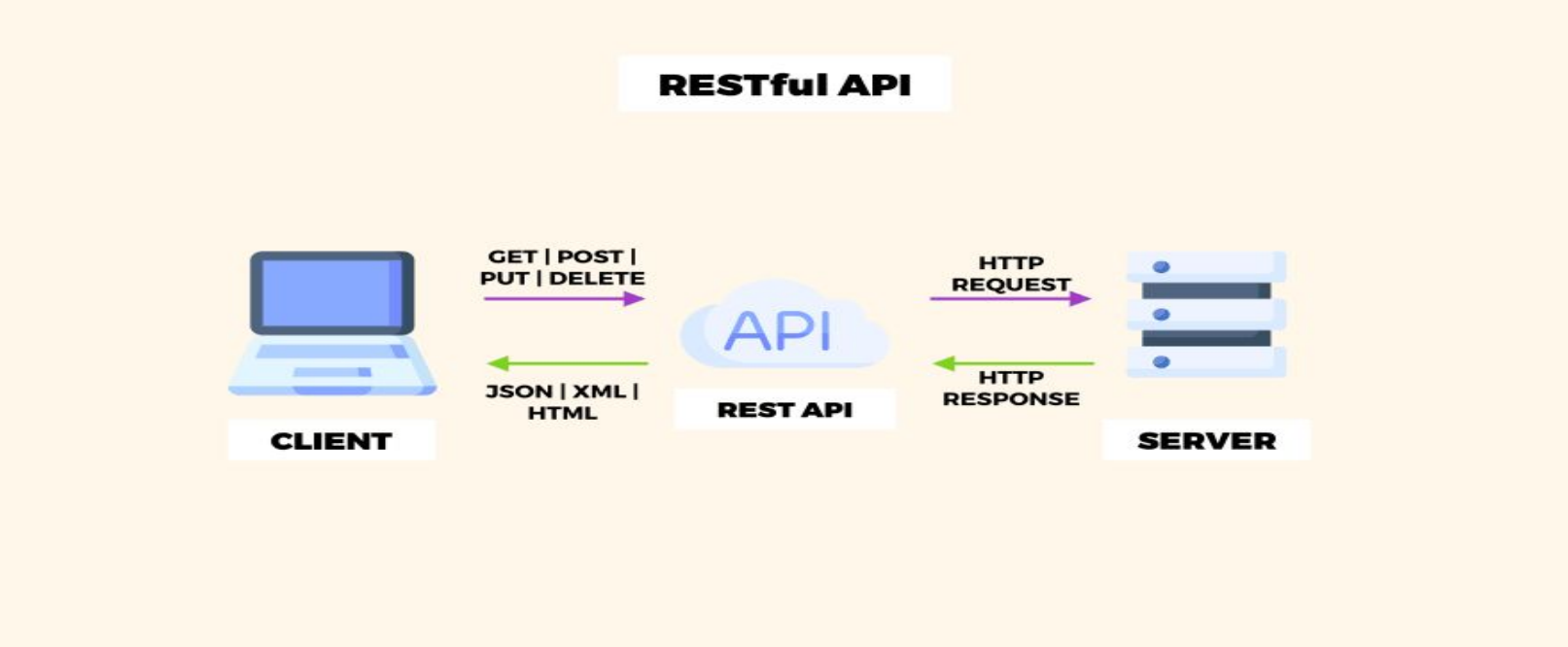
In order for an API to be considered RESTful, it has to conform to these criteria:
- A layered system that organizes each type of server (those responsible for security, load-balancing, etc.) involved the retrieval of requested information into hierarchies, invisible to the client.
- Code-on-demand (optional): the ability to send executable code from the server to the client when requested, extending client functionality.

REST?

Though the REST API has these criteria to conform to, it is still considered easier to use than a prescribed protocol like SOAP (Simple Object Access Protocol), which has specific requirements like XML messaging, and built-in security and transaction compliance that make it slower and heavier.
=> **SOAP refer: https://stoplight.io/api-types/soap-api**

In contrast, REST is a set of guidelines that can be implemented as needed, making REST APIs faster and more lightweight, with increased scalablity—perfect for Internet of Things (IoT) and mobile app development.

REST API


RESTful API
GET | POST |
PUT | DELETE
JSON | XML |
HTML
API
REST API
HTTP
REQUEST
HTTP
RESPONSE
CLIENT
SERVER

Communication between Client and Server

In the REST architecture, clients send requests to retrieve or modify resources, and servers send responses to these requests. Let’s take a look at the standard ways to make requests and send responses.

Refer: https://www.codecademy.com/article/what-is-rest

What is CRUD?

Refer: https://www.codecademy.com/article/what-is-crud

Authorization (3 Authorize)

- HTTP Basic
- JSON Web Token (JWT)
- OAuth2

CORS Policy

Refer: https://linuxpip.org/fix-access-to-xmlhttprequest-has-been-blocked-cors-policy/

API Document (Using Unit Test)

Call API

Tools Test API

- https://swagger.io/tools/swagger-ui/
- https://www.postman.com/

Exercises

- **Create CRUD API**