API Essentials

[Abd El Monem Mohamed Abdelmonem]

What is an API?

APIs are mechanisms that enable two software components to communicate with each other using a set of definitions and protocols. For example, the weather bureau's software system contains daily weather data. The weather app on your phone "talks" to this system via APIs and shows you daily weather updates on your phone.

What does API stand for?

API stands for Application Programming Interface. In the context of APIs, the word Application refers to any software with a distinct function. Interface can be thought of as a contract of service between two applications. This contract defines how the two communicate with each other using requests and responses. Their API documentation contains information on how developers are to structure those requests and responses.

How do APIs work?

API architecture is usually explained in terms of client and server. The application sending the request is called the client, and the application sending the response is called the server. So in the weather example, the bureau's weather database is the server, and the mobile app is the client.

What are REST APIs?

REST stands for Representational State Transfer. REST defines a set of functions like GET, PUT, DELETE, etc. that clients can use to access server data. Clients and servers exchange data using HTTP.

The main feature of REST API is statelessness. Statelessness means that servers do not save client data between requests. Client requests to the server are similar to URLs you type in your browser to visit a website. The response from the server is plain data, without the typical graphical rendering of a web page.

Source: https://aws.amazon.com/what-is/api/

[Beshoy Ayad AbdelMalak]

- **What is API**
  - The API, or the Application Programming Interface, is a set of tools that allow a user (or rather, a programmer) to access an application's functions without any knowledge of its inner workings

- **How does an API work**

- At its core, API is just a means of communication.
- An API of a program provides a language-agnostic interface for communicating with it. Whether the program is coded in machine language or Java, another programmer can use its functions through the API.

- **APIs protocols and architectures**
  - **a.** Low-Level APIs
    - The simplest APIs are typically reserved for function libraries.
  - b. SOAP
    - SOAP stands for Simple Object Access Protocol, a name that encapsulates its aims perfectly. SOAP APIs use XML data format, a stricter subset of HTML that is the standard for web pages. While XML is a far more robust format than HTML, it is also more complex and intolerant of even the most minor errors, which makes XML difficult to use. Consequently, only developers building their application on the .NET framework use SOAP based APIs due to easy XML generation within these platforms.
  - c. REST
    - Representational State Transfer (REST) may sound like a mouthful but is actually easier to use than SOAP.
    - The reason for this is that REST is an architecture, not a strict protocol. Instead of laying down hard rules to be followed in communications (like XML) REST only specifies an overarching structure of representing and passing on data, leaving the actual details of the implementation to every individual application itself.

- **API Methods**

a. GET - retrieves a specific piece of data or “resource”
b. POST - creates a new resource
c. PUT - updates an existing resource
d. DELETE - deletes an existing resource

[Esraa Abdullah Ahmed Shehab]

API definition:

API stands for (Application Programming Interface), it can be used to enable two SW components to communicate with each other using some protocols, so we can say it’s a helpful tool which enable the developer to access application’s functions and features without having to know how it works.

API mechanism:

Regarding APIs there is always client and server..

- **Client:** is the application sending the request.
- **Server:** is the application sending the response.

Example:

Google Maps database is the server and the mobile application is the client.

<u>APIs work according to what they created for:</u>

- **SOAP APIs:** where client and server use XML to exchange messages, these APIs aren’t flexible and not common.
- **RPC APIs:** where the client complete a function on the server then the server sends the output back to the client.
- **Websocket APIs:** it’s a web API uses JSON objects, it also supports two-way communication between client and server.
- **Rest APIs:** it’s also a web API and the most common and flexible API where the client sends request as data to the server which use it to start internal functions sending output back to the client.

Benefits of REST APIs

- Innovation
- Integration
- Expansion
- Maintenance

API types:

- Public APIs >> Can be used by anyone, no need for authorization.
- Private APIs >> Belong to an organization, and can only used by the authorized people for connecting different systems within the org.
- Partner APIs >> Used to support B2B partnerships and can only be accessed by authorized developers.
- Composite APIs >> Combination of two or more different APIs for sophisticated system requirements.

API creation:

- Planning: the blueprint for API design.
- Building: API prototype, test and customization.
- Testing: prevent defects and bugs.
- Documenting: improve usability.
- Marketing: listing API for monetizing.

API use/ implementation:

- Create an acc. With API provider to obtain an API key.
- Structure API request using API keys received by setting up an HTTP API client or simply do it yourself in browser.
- Use it in your code!.
