**1) What is API?**

**API** (*Application Programming Interface)*. It is an interface between two applications and allows the two software systems communicate with one another and data exchange between two software systems.

API works as; it takes a request from the source, takes that request to the database, fetches the request data from the database and returns a response to the source**. API takes the requests from the user and gives the response without exposing the internal details. API acts as Abstraction.**

It shows only essential things to the user and hides the internal details.

**Example:** Amazon API, Google Map API

**Example:** The most common example of API is searching for information on the internet using an app. If you use an application for a particular purpose, you use the internet. A connection is established between the internet and the application. The internet provides data to the server; it retrieves data, processes the information, and sends it back to the phone. The task of the application is to interpret data and present it to the user in a readable format. This process is facilitated using API.

**Example**- When you go to a restaurant and order food, the food comes from the kitchen. Hence the kitchen is the system. Your table is the place where the response is delivered. However, to deliver food on the table, there must be a channel of communication between the table and the kitchen. The waiter or API acts as a messenger between the table and the kitchen. It takes requests from you and tells the kitchen, which is a system, what has to be done. Then, the waiter delivers the response to the user, which is food in this case.

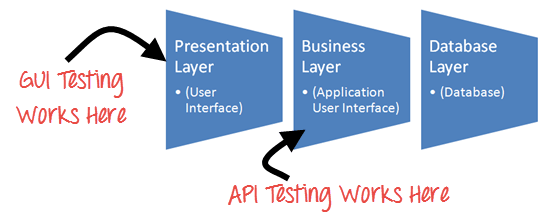
### 2) What is API testing?

**API testing** is a type of software testing that involves testing APIs directly. API is a part of integration testing to check whether the API meets customer expectations or not in terms of functionality, reliability, performance, and security of applications.

In API testing, our primary focus is on Business Logic Layer of the software architecture.

In API testing we don’t have graphical user interface (GUI). In this scenario, we will pass on parameter in given URI (Unified Resource Identifier) to check the response from the sever.

In API Testing, instead of using standard user inputs (keyboard) and outputs, you use software to send calls to the API, get output, and note down the system’s response. API tests are very different from GUI Tests and won’t concentrate on the look and feel of an application. It mainly concentrates on the business logic layer of the software architecture.



**3) What is Web Service?**

* **Web Service** is a service that will work over the network to communicate two devices is called web Service.
* It is the intermediate between source and target.
* A Web service is a collection of open source protocols and standards which are used for exchanging data between systems or applications.
* Software applications are written using various programming languages and running on multiple platforms. It allows you to use web services to exchange data over computer networks.

There are two types of web Service

1. SOAP
2. REST

|  |  |  |
| --- | --- | --- |
| **Sr.NO.** | **API** | **Web Service** |
|  | API is a part of integration testing to check whether the API meets expectations in terms of functionality, reliability, performance, and security of applications. | **Web Service** is a service that will work over the network to communicate two devices is called web Service. |
|  | API may or may not need network for its operations. | Web Services always need network for its operation. |
|  | API can work over the network as well as intranet process. | Web Service is compulsory work over the network. |
|  | API can be communicated through SOAP, REST, XML-RPC and CURL calls as well.  API can also be exposed in number of ways like JAR, DLL, XML over HTTP, JSON over HTTP etc. | Web service can be communicated through SOAP, REST, AND RPC. |
|  | API supports HTTP/HTTPS protocol. | Web service supports only HTTP protocol |
|  | API supports XML and JSON format. | Web service supports only XML format |
|  | API can perform all the operations which web service can't perform. | Web service can't perform all the operations like API. |
|  | All APIs are not web service. | All web services are API |

|  |  |  |
| --- | --- | --- |
| Sr. No. | **SOAP** | **REST** |
|  | **SOAP** stands as Simple Object Access Protocol. | **REST** stands as Representational State Transfer. |
|  | SOAP is protocol based. | REST is architectural based. |
|  | It is a XML-based protocol for communication between two applications. | (REST) is a software architectural style that follows the 6 constraints to create web services. |
|  | SOAP provides its own rules and regulations. | REST doesn’t provide any rules and regulations. |
|  | SOAP used only XML data format. | REST used different data format such as xml, Json, plain text, HTML, etc. |
|  | SOAP uses service interfaces to expose its functionality to client applications | REST used URI to exposed business logic. |

**SOAP**

**SOAP** (Simple Object Access Control) . It is an XML based protocol that helps in exchanging information among computers.

**Rest**

REST is an Architectural style in which a web service can only be treated as a RESTful service if it follows the 6 constraints such as

* Client Server
* Stateless
* Cacheable
* Layered System
* Uniform Interface
* Code on demand(optional)

REST used different data format such as xml, Json, plain text, HTML, etc.

4) What are the protocols used in API Testing?

Protocols used in API testing are:

* HTTP
* REST
* SOAP
* JMS

5) What are the tools used for API Testing?

Tools used for API testing are:

* ARC (Advanced rest client)
* Parasoft SOAtest
* PostMan
* AlertSite API monitoring

### 6) What exactly needs to verify in API testing?

In API testing, we send a request to API with the known data and then analysis the response.

1. We will verify the accuracy of the data.
2. Will see the HTTP status code.
3. We will see the response time.
4. Error codes in case API returns any errors.
5. Authorization would be check.
6. Non-Functional testing such as performance testing, security testing.

7) What are the major challenges faced during API testing?

The major challenges faced during the API testing are:

* Parameter Selection
* Parameter Combination
* Call sequencing
* Output verification and validation
* A major challenge is providing input values which are very difficult because GUI is not available.

8) What is a RESTFUL web services?

There are two kinds of web services

1. SOAP Web Services
2. RESTFUL Web Services

**1. SOAP (Simple Object Access Protocol) -** SOAP is a XML based method which is used in Web Services.

**2. RESTFUL Web Services -** To implement the concept of REST architecture HTTP method is used. RESTFUL Web Services defines URI (Uniform Resource Identifier), and also provides resource representation like JSON and a set of HTTP method.

### 9) What protocol is used by the RESTFUL Web Services?

RESTFUL Web Services uses the HTTP protocol. They use the HTTP protocol as a medium of communication between the client and the server.

### 10) What is the HTTP protocol supported by REST?

### RESTFUL Web Services Methods are

**GET**: - It is used to retrieve list of data from database then we will use get method

Return status code is 200.

**POST: - It is used for insert new element in our database the will used post method.**

**Return status code is 201.**

**PUT: It is used to update maximum fields then will use put method.**

**In case updating data is not present then put method will work as a post method.**

**Return status code is 200 or 201.**

**PATCH: - It is used to update minimum fields then will use patch method.**

**In case updating data is not present then patch method will throw exception.**

Return status code is 200.

**DELETE:** It is used for delete element from database.

Return status code is 200.

**OPTIONS:** OPTION is used to describe the communication option for the target resources.

**HEAD:** HEAD asks for response which is identical to GET requests, but without the response body.

10) Explain all kind of HTTP codes? 1xx 2xx 3xx 4xx 5x

: -

* [**1xx: Informational**](https://restfulapi.net/http-status-codes/#1xx) – Communicates transfer protocol-level information.
* [2xx: Success](https://restfulapi.net/http-status-codes/#2xx) – Indicates that the client’s request was accepted successfully.
* [**3xx: Redirection**](https://restfulapi.net/http-status-codes/#3xx) – Indicates that the client must take some additional action in order to complete their request.
* [**4xx: Client Error**](https://restfulapi.net/http-status-codes/#4xx) – This category of error status codes points the finger at clients.
* [**5xx: Server Error**](https://restfulapi.net/http-status-codes/#5xx) – The server takes responsibility for these error status codes.

|  |  |
| --- | --- |
| **Status Code** | **Description** |
| **200 OK** | Indicates that the request has succeeded. |
| **201 Created** | Indicates that the request has succeeded and a new resource has been created as a result. |
| **204 No Content** | The server has fulfilled the request but does not need to return a response body. The server may return the updated meta information. |
| **400 Bad Request** | The request could not be understood by the server due to incorrect syntax. The client SHOULD NOT repeat the request without modifications. |
| **401 Unauthorized** | Indicates that the request requires user authentication information. The client MAY repeat the request with a suitable Authorization header field |
| **402 Payment Required (Experimental)** | Reserved for future use. It is aimed for using in the digital payment systems. |
| **403 Forbidden** | Unauthorized request. The client does not have access rights to the content. Unlike 401, the client’s identity is known to the server. |
| **404 Not Found** | The server cannot find the requested resource. |
| **500 Internal Server Error** | The server encountered an unexpected condition that prevented it from fulfilling the request. |
| **501 Not Implemented** | The HTTP method is not supported by the server and cannot be handled. |
| **502 Bad Gateway** | The server got an invalid response while working as a gateway to get the response needed to handle the request. |

There are two popular formats for request and response **XML** and **JSON**.

**0:00/0**

**XML Format:** XML is the popular form as request and response in web services. Consider the following XML code:

<getDetail>

<id>DataStructureCourse</id>

</getDetail>

The code shows that user has requested to access the Data Structure Course. The other data exchange format is JSON. JSON is supported by wide variety of platform.

**JSON Format:** JSON is a readable format for structuring data. It is used for transiting data between server and web application.

[

    "employee":

   {

       "id": 00987

        "name":       "Jack",

       "salary":      20000,

  }

]