# **API Testing Basics**

**1.What are facets**?

APIs comes with 2 broad facets

1. **Request**– An encapsulation of action performed along with pertinent details to be sent to the server.

2. **Response**– An encapsulation of processed data along with supplement details sent back to client by the server.

Both request and response come with header attached. Take a look at any network call, you should be able to see headers quite evidently.

**2.How are API called?**

Each user interaction on UI is associated with an API that will be accessed through a URL. Let us say a user fills up a form and clicks on “Send” button, on trigger of this event respective API will be invoked.

**3.API Request Method?**

Every request is associated with the “Request Method”. This method specifies the desired action to be performed by the server. Below are the frequently used methods

1. **GET**: This method is used to retrieve the details, this performs a read-only operation.

2. **POST**: This method submits data by creating an entry into database. This method should be used wisely as it can change the state causing side effects if not handled properly.

3. **PUT**: This method is used to update any data, the data that needs to be replaced is sent as part of payload.

4. **DELETE**: This method just deletes the data.

5. **HEAD**: This is similar to get method, however there is no response body sent.

6. **PATCH**: This method specifies how to update the data, this can cause side effects if not handled with care.

**4.What is API Response?**

**Every API request is associated with response. The response have 2 facets**

**1. Status code :** Numeric representation of the web server response. There are pre-defined set of status code which can be reused. Or, teams can create their own status code as per convenience. However, appropriate status code range should be used.

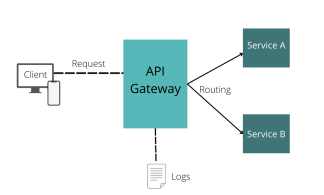
**2. Status Message :** Each status code is associated with detailed response message.

**for example:**

Status code 200 → Success

Status code 401 → Unauthorized

Status code 404 → Not found



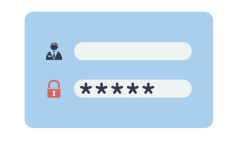
**5.What is API Gateway?**

API gateway helps establishing authentication to verify the calls before transferring it to further execution. The capability of API gateway is not just restricted to authentication, it also provides multitude of services like.

1. Routing
2. Rate Limiting
3. Analytics
4. Security
5. Policies etc

**6.what is API Authentication?**

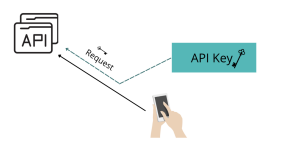
Authentication is a process of verifying the identity of the user it solely answers who you are, whereas authorization mostly deals with access management and comes into play only after the user is identified and verified successfully.

1. **Basic Authentication:**

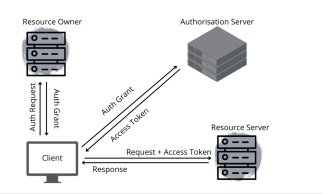
This technique involve providing username an

d password for user verification. When user enters their credentials the details are encoded in Base64 generating a Key which will be bundled in request header and sent to server for verification. Server Verifies the key with the stored username and password.

If the identity is verified the request is fulfilled else an error is sent back denying request sent.

1. **API Key Authentication**:

API key is a long encrypted string which identifies the application without any principal. These are sent either as a part of request header or URL. When client recognizes the API key server will process the request.

1. **OAUTH Authentication:**

This technique is considered quite powerful and secure way of authenticating the users. OAuth technique can also be used for authorization. Initially a user may have to login to the OAuth application using the credentials to generate a token. The generated token is attached as part of request header, which will be sent to authentication server in order to verify. If the token is recognized the API request will be processed.

s

1. **API Testing Basics?**

API testing is a [black box testing](https://testsigma.com/blog/key-differences-between-black-box-and-white-box-testing/" \t "https://testsigma.com/blog/api-testing-basics/_blank) technique.

For example consider the below [REST API](https://www.redhat.com/en/topics/api/what-is-a-rest-api" \t "https://testsigma.com/blog/api-testing-basics/_blank)

GET <https://gorest.co.in/public/v2/users>

The response returned will be in the JSON format. REST Apis generally use JSON structure to represent the response body.

* Let us go ahead churning the test cases for this API

1. On Successful execution the API should return status code 200 (message : Success)

2. Specific user details can be fetched by passing user id in URL (<https://gorest.co.in/public/v2/users/25>)

3. The JSON response should match the below schema

{

“id”:integer,

“name”:string,

“email”:string,

“gender”:string,

“status”:string

}

4. Wrong URL should respond with 404 (Not found)

5. Multiple User ID in the URL should respond with 404 error

1. **What is API Testing Approach?**

API Testing Approach:

There are various approaches to API testing, and the following points highlight some of the standard testing approaches:

1. **Functional testing**: This involves testing the functionality of the API to ensure that it meets the specified requirements.
2. **Security testing**: This involves testing the security of the API to ensure that it is not vulnerable to security threats.
3. **Load testing:** This involves testing the performance of the API under different loads to ensure that it can handle the expected traffic.
4. **Integration testing**: This involves testing the integration of the API with other software components to ensure that it works continuously.

# **How to Test API?**

# **Understand the API’s Functionalit**y:--

# Before you start testing an API, you need to understand its functionality. This includes the data it exchanges with the client, the expected format of the data, the input parameters, and the expected output. Understanding the API’s functionality will help you design your test strategy and ensure that your test cases cover all the required scenarios.

# **Test the API Endpoints:--**

# Once you understand the API well, you can begin testing the endpoints. This involves sending requests to the API using various HTTP methods (GET, POST, PUT, DELETE).

## 10.**Best Practices of API Testing?**

The best practices of API testing are essential to ensure the functionality and reliability of an application programming interface.

1. Plan and design the API tests before starting the testing process.
2. Create test data covering a wide range of scenarios .
3. Verify the API’s functional behavior, security, and performance.
4. Conduct regression testing to ensure that changes and updates do not impact the existing API functionality.
5. Integrate API testing with continuous integration and delivery pipelines.
6. Collaborate with developers and other stakeholders to identify and resolve issues early.
7. Use monitoring tools to track API performance and identify issues before they impact end-users.

## **Challenges of API Testing?**

## Here are some of the most common challenges:

* Complex API architecture and integration with other systems can make testing difficult and time-consuming.
* API testing requires programming skills and knowledge of HTTP protocols and methods.
* Test data management can be challenging, especially when dealing with large data sets.
* Ensuring API security and authentication is essential, but testing can be challenging.
* API testing requires close collaboration between developers, testers, and stakeholders to ensure that the APIs meet the business requirements and user expectations.