**API testing**:

API testing is a type of software testing that involves testing **application programming interfaces directly** and **as part of integration testing** to determine if they meet expectations for **functionality, reliability, performance, and security**. Since APIs lack a GUI, API testing is performed at the message layer.

**Types of API testing: Unit, Functional, Performance, Reliability, Security testing**

**API Architecture:**

1. HTTP Header
2. HTTP Request
3. Status code / Response code

**1.HTTP Header:**

which carries metadata (as key-value pairs) for the HTTP Request message. Metadata could be a client (or browser) type, format supported by the client, format of a message body format, cache settings, and so on.

**Authentication**

**Response time**

**2.HTTP Request**

**GET** is only used to request data from a specified resource.

**POST** is used to send data to a server to create/update a resource

**PUT** replaces all current representations of the target resource with the request payload

**DELETE** removes the specified resource.

**OPTIONS** is used to describe the communication options for the target resource.

**HEAD** asks for a response identical to that of a GET request, but without the response body.

**3.Status code /Response code**

**Success (2xx)**

200 - OK - request was fulfilled.

201 - Created - following POST command, indicates success.

202 – Accepted – request has been accepted for processing but not completed.

203 – Partial information – we received in the response to GET command.

204 – Server received request but there is no information to send back.

**Error (4xx, 5xx)**

400 – Bad request – request had bad syntax.

401 – Unauthorized –

402 – Payment Required

403 – Forbidden

404 – Not found – Server had not found anything matching he URI given.

500 – Internal Error

501 – Not implemented

**Redirection (3xx)**

301 – Moved

302 – Found

**Types of bugs expected**

Wrong status code

API validations

Components (modules) not interacting as expected

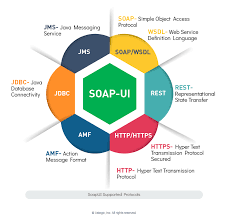
HTTPS request are not working

**REST (Representational State Transfer)** is an architectural style for developing web services which exploit the HTTP protocol and uses HTTP method to define actions. It revolves around resource where every component being a resource that can be accessed through a shared interface using standard HTTP methods.

**SOAP (Simple Object Access Protocol)** is defined as an XML-based protocol. It is known for designing and developing web services as well as enabling communication between applications developed on different platforms using various programming languages over the Internet. It is both platform and language independent.

Popular tools for API automation. **SoapUI, Postman** and **Katalon Studio.**

**SoapUI** and **Postman** specialize in API testing while Katalon Studio provides a complete solution for API, Web and mobile app testing.



**SOAPUI**

[SoapUI](https://huddle.eurostarsoftwaretesting.com/how-to-soap-ui-with-docker-using-docker-to-execute-tests/) is widely cited as a top choice when it comes to API testing. It is a headless functional testing tool specifically designed for API testing. SoapUI supports both REST and SOAP services. API automation testers can either use open-source or pro versions. The pro edition has a user-friendly interface and several advanced features such as assertion wizard, form editor, and SQL query builder. [SoapUI](https://en.wikipedia.org/wiki/SoapUI) is a tool of ReadyAPI suite, offered by SmartBear.

The tool provides many advanced features for API testing, including

* Generates tests easily using drag and drop, point-and-click
* Powerful data-driven testing with data from files and databases
* Scripts can be reused easily
* Mocks services with RESTful mocking
* Asynchronous testing

SoapUI is distributed as **Open-source** and **Pro versions** ($659/year for medium and large teams).



**POSTMAN**

Postman emerged as a popular automation tool for API testing after having only been known as a browser extension for API validation. [Postman](https://huddle.eurostarsoftwaretesting.com/api-testing-api-code-snippets/) can be installed as a browser extension or a desktop application on Mac, Linux, and Windows. It is used not only by automation testers for API tests but also by developers for developing and validating APIs. It has evolved, in fact, as an environment to develop and test APIs. Some notable features include:

* A comprehensive feature set for designing, debugging, testing, documenting, and publishing APIs
* Supports both automated and exploratory testing
* A friendly and easy-to-use user interface
* Accepts Swagger and RAML API formats

Postman is affordable as the product is offered in three editions: **Postman** (free), **Postman Pro**($8/month), and **Postman Enterprise** ($21/month).



**KATALON STUDIO**

Katalon Studio is an integrated environment to generate and execute API, Web-based GUI, and mobile apps automation tests. It has a rich feature set for these types of testing and supports multiple platforms including Windows, Mac OS, and Linux. By integrating Selenium and Appium engines with all needed components, built-in keywords, and templates, Katalon Studio provides a unique environment for both testers and developers to perform API and Web automation tests. Some notable highlights of the tool:

* Handles API, Web, and mobile tests across platforms
* Allows testers and developers to collaborate and share test cases easily
* Hundreds of built-in keywords for creating test cases
* Supports AssertJ to create fluent assertions using the BDD style
* Seamless integration with other ALM tools for CI/DevOps practices

[**Katalon Studio**](https://huddle.eurostarsoftwaretesting.com/automation-web-testing/)**is free of charge** although it is not open-source.

Dedicated support services, for instance, troubleshooting and consulting services are offered in forms of [Business Support](https://www.katalon.com/support-service-options) and Enterprise Support.

**COMPARING SOAPUI VS POSTMAN, KATALON STUDIO**



**REST and SOAP** are the dominant API types, accounting for more than 95% of all API/Web services according to the [State of API Integration Report](https://jaxenter.com/state-of-api-integration-report-136342.html). Katalon Studio and SoapUI support both REST and SOAP APIs while Postman handles only REST requests. It should be noted that the overwhelming majority of API implementations (83%) follow the REST protocol.

**Automated assertion generation**: this capability involves analyzing APIs and generating assertions automatically. It is expected to save time from manually generating assertions for APIs. All three tools provide basic functions for generating assertion scripts based on some predefined templates or rules.

**BDD Cucumber support:**this capability allows test cases to be written in natural languages, which helps improve the collaboration between technical staff and business stakeholders. This is a Katalon Studio’s distinct feature not seen in the other tools.

**Test reports:** all three tools provide capabilities for reporting API test results. Postman generates reports in JSON and HTML formats; Katalon Studio reports results and logs in different forms such as console report, log, HTML and analytics via Katalon Analytics. For SoapUI, the capability to produce detailed test reports is in the commercial edition.

**Scripting languages:** all tools support languages based on Java.

**Web UI and Mobile app testing**: in developing a multi-platform application, the team performs Web UI and mobile app tests besides testing APIs. So, supporting these types of test is an advantage of Katalon Studio over Postman and SoapUI, as it allows developers and testers use the same tool, share and collaborate on the same test artifacts.

**Test execution analytics:** test exaction logs and results from Katalon Studio can be automatically exported to Katalon Analytics, allowing teams to analyze, diagnose, and report bugs.

**Rest-Assured:**

Rest-Assured is an open-source Java Domain-specific language that makes testing REST service more simple.

Have a bunch of baked-in functionalities, which means users don’t have to code things from scratch.

Integrates seamlessly with Serenity automation framework, so that users can combine the UI and REST tests all in one framework that generates awesome reports.

Support BDD Given/When/Then syntax

Users don’t necessarily need to be an HTTP expert

[Karate DSL](https://github.com/intuit/karate)

Karate DSL is a new API testing tool which helps create scenarios for API-based BDD tests in a simple way without writing step definitions. Those definitions have been created by KarateDSL so that users can kickstart the API testing quickly.

Build on top of Cucumber-JVM

Can run a test and generate reports like any standard Java project

A test can be written without any Java knowledge required

Tests are easy to write even for non-programmers

Supports configuration switching/staging, multithreaded parallel execution

Apigee.com >

To prase JSON file

Json.praser.online.fr