

API TESTING

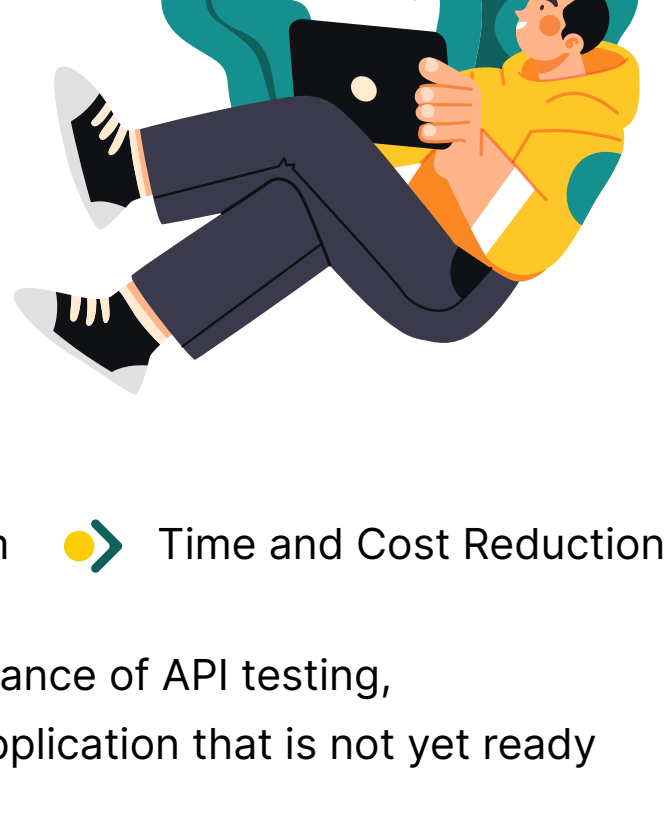
WHAT, WHY, AND HOW?



Application Programming Interface (API) enables two or more computer programs to interact with each other and exchange data between them without any user intervention.

What Are The Benefits Of API Testing?

It helps to validate and qualitatively measure the success of business requirements through a defined test case process.



- Early Evaluation
- CI/CD Compliance
- Tight Security
- Platform Agnosticism
- Time and Cost Reduction

All these aspects serve as examples of the relevance of API testing, especially in reducing the risk of delivering an application that is not yet ready for the market.

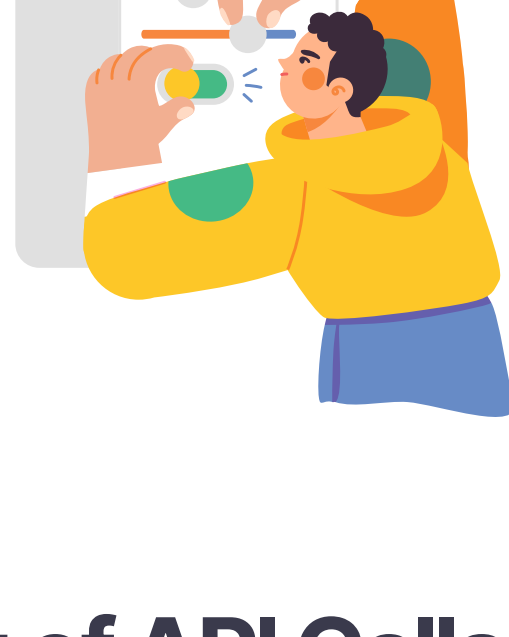
Challenges in API Testing

- Initial Setup of API Testing
- Sequencing of API Calls
- Selecting, Testing, and Validating the Parameter Combinations
- Updating the Schema of API Testing

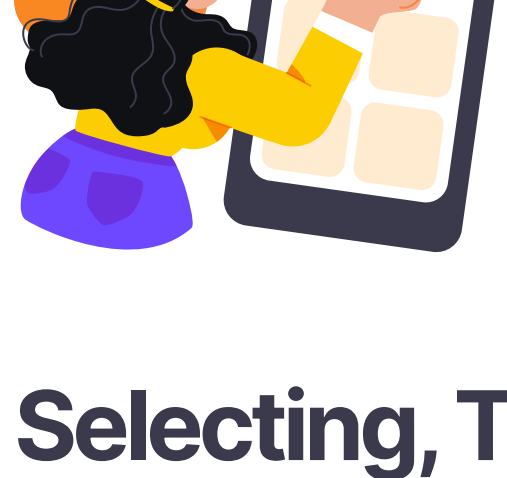


Initial Setup of API Testing

- 01 It's different from other software testing environments as there's no GUI available.
- 02 The database and server must be configured per the application's requirements.
- 03 The QA team has to get access to the testing system, get security approvals.



Sequencing of API Calls



The API calls may not appear in the sequence expected. This is especially a problem with multi-threaded apps. Haphazard call sequencing could lead to confusion as the tester will not be able to check if the system is working correctly.

Selecting, Testing, and Validating the Parameter Combinations

APIs assign data values to parameters. These parameters pass through the data requests to manage communication between the systems. Sometimes the API would assign two different values to the same parameter. The QA team has to test all the possible parameter combinations in API to check for any issues in the specific configurations.

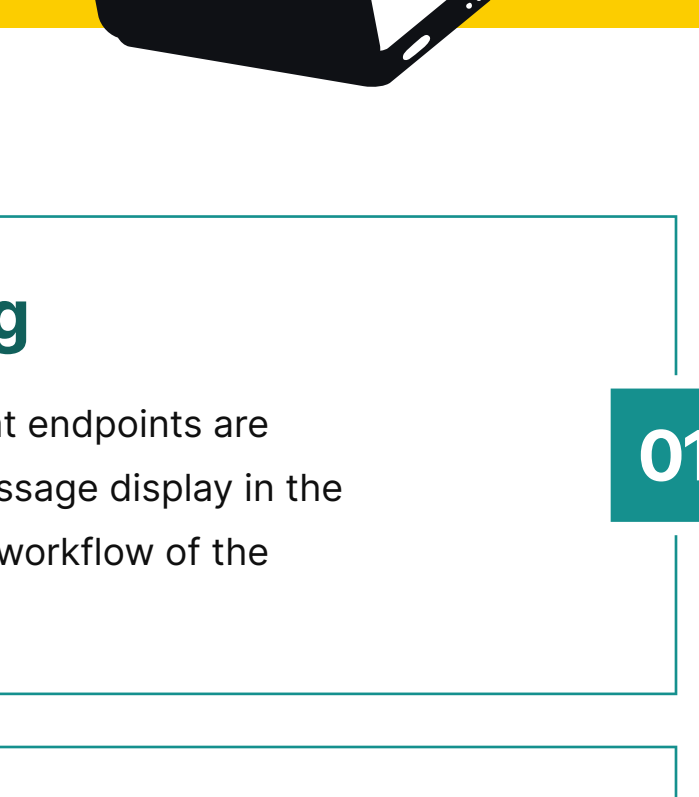


Updating the Schema of API Testing

The QA team needs to ensure that any change in the program that creates an additional parameter for API calls reflects in the schema configuration.



How To Approach API Testing?



Define the Scope of Testing

This includes asking critical questions like what endpoints are available for testing, what should the error message display in the event of an unsuccessful request, what is the workflow of the application, etc.

01

Set Up the Environment and Integrate Application Data

Configure the database and server according to the application requirements, and integrate application data with the API tests to test the API against all configurations.

02

Apply Various Testing Techniques

Perform various testing techniques such as functionality testing, load testing, and security testing to ensure that the API works efficiently. Each serves a specific purpose, so test thoroughly before sending to production.

03

Choose the Right API Testing Tool

Discuss the different tools with developers and the QA team, analyze their pros and cons, check the programming language they are based on, and evaluate if they fit the budget.

04

Develop Test Cases and Analyze the Results

Write and execute test cases and check if the actual results match the expected results.

05

Document Maintenance

When the test cases are executed, document the results for future reference. Also, make it a point to update the document regularly in case there are changes in endpoints or the backend impacting the API.

06

Best Practices of API Testing



01 Develop a flowchart to help the QA team visualize the API calls.

02 Test for both positive and negative results.

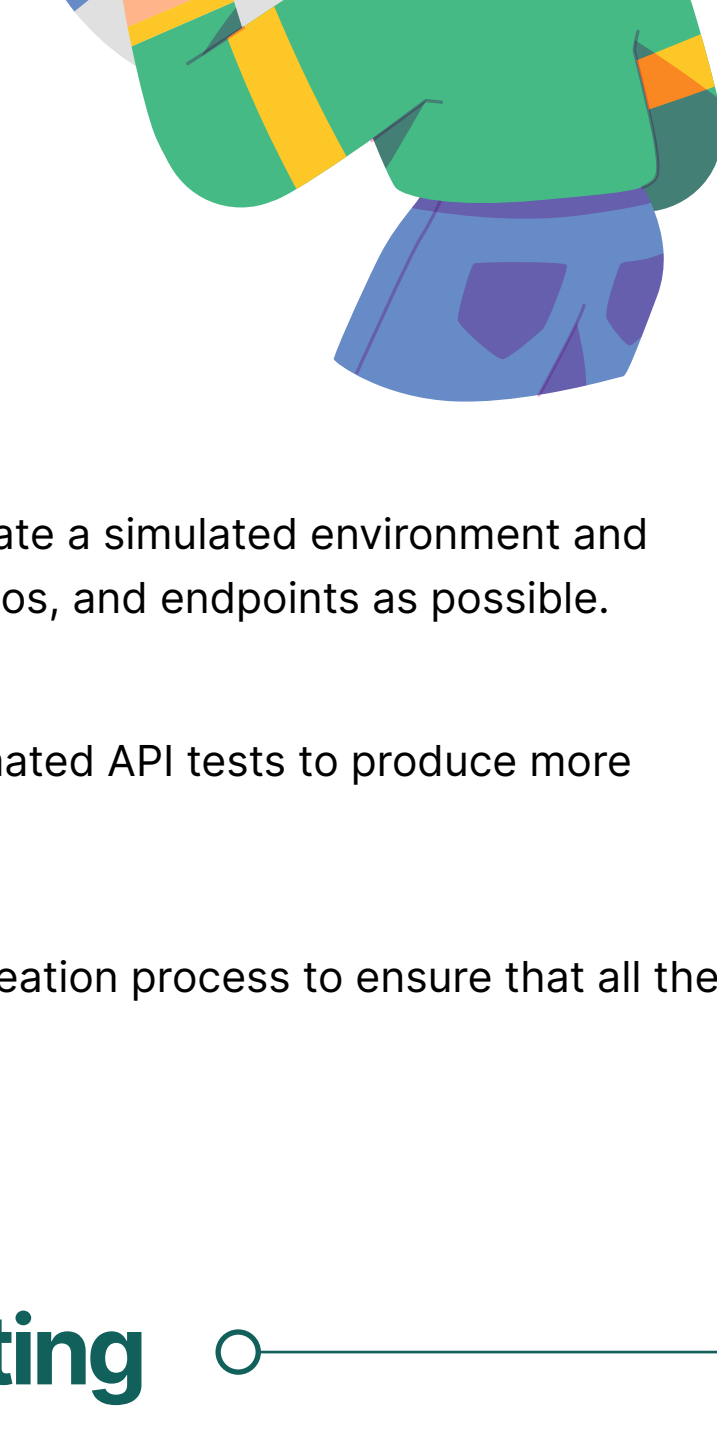
03 Perform API testing for all API input combinations to widen the scope of test coverage.

04 Use data that would reflect the conditions the API would face in the production phase.

05 Use service virtualization to create a simulated environment and test as many test cases, scenarios, and endpoints as possible.

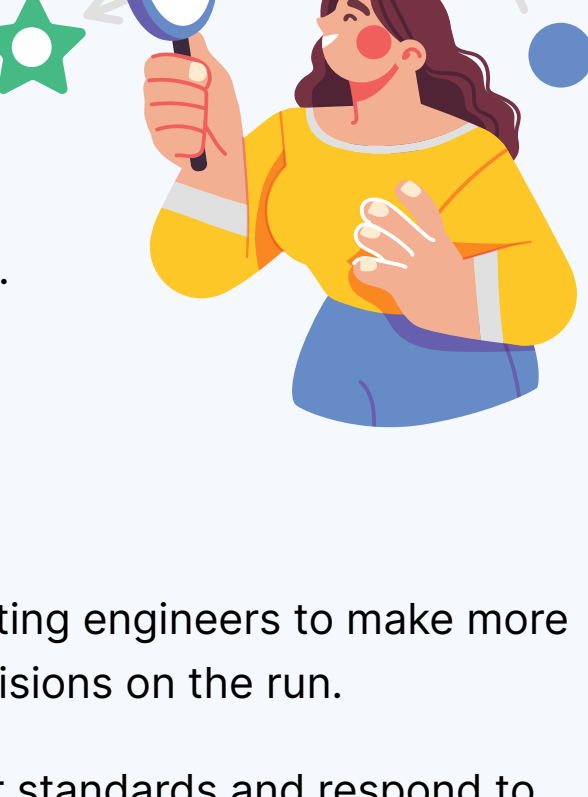
06 Perform both manual and automated API tests to produce more trustworthy results.

07 Automate the documentation creation process to ensure that all the test cases are documented.



Examples of API Testing

- Testing if a mobile app can pull out the necessary information about the user from Google or Facebook.
- Testing the payment integration in a mobile app.



Why ACCELQ for API Test Automation?

- 01 Speed up the continuous delivery cycle by reducing the dependency on manual testing.
- 02 Facilitate end-to-end API validation through chain API tests.



- 03 Empower testing engineers to make more informed decisions on the run.
- 04 Maintain your standards and respond to business requirements rapidly.
- 05 Reduce the cost of development and maintenance of the codebase with a lean testing strategy