|  |  |
| --- | --- |
| Text, logo  Description automatically generated | Rest API Testing Guide |

AppDev QA Management

Rest API Testing

Version 1.0

Author: Parimala Divya G

Contents

[API 3](#_Toc161764139)

[API Methods 3](#_Toc161764140)

[Types of REST API Tests 4](#_Toc161764141)

[Sections in Postman 4](#_Toc161764142)

[**Header:** 4](#_Toc161764143)

[**Response:** 5](#_Toc161764144)

[**Sidebar:** 6](#_Toc161764145)

[**Builder:** 6](#_Toc161764146)

[Workflow in Postman: 7](#_Toc161764147)

[**Creating and triggering request:** 7](#_Toc161764148)

[**Validating the response:** 8](#_Toc161764149)

[Collection Runner in Postman: 8](#_Toc161764150)

# API

API (Application Programming Interface) enable communication and interaction between different software systems, allowing them to exchange data and functionalities.

# API Methods

                 REST APIs use five HTTP methods to request a command:

* GET: Retrieve a resource
* POST: Create a new resource
* PUT: Update an existing resource
* PATCH: Modify an existing resource
* DELETE: Delete an existing resource

# Types of REST API Tests

* Unit Testing: Testing the functionality of individual operations
* Integration Testing: Testing the interaction between multiple software modules
* Functional Testing: Ensuring that REST APIs behave exactly as it should
* Load Testing: Measuring how many calls REST APIs can handle
* Security Testing: Validating REST API encryption methods and access control

# Sections in Postman

To start working with Postman, we have the navigations as shown below. It primarily consists of four sections:

## **Header:**

Under Header, New menu is used to create a new Environment, Collection or request. The Import menu helps to import an Environment/Collection



A screenshot of a computer

Description automatically generated

We can import from a File, Folder, Link, Raw text or from Code repository options which are also available under Import.

A screenshot of a computer

Description automatically generated

## **Response:**

Response section shall have values populated only when a request is made. It generally contains the Response details.

A screenshot of a computer

Description automatically generated

## **Sidebar:**

Sidebar consists of Collections (used to maintain tests, containing folders, sub-folders, requests), History (records all API requests made in the past), and APIs.

A screenshot of a computer

Description automatically generated

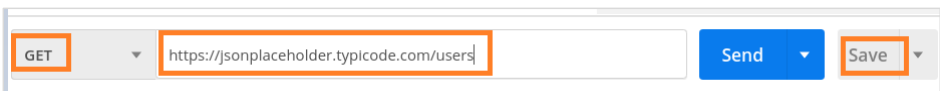
## **Builder:**

Builder is the most important section of the Postman application. It has the request tab and displays the current request name. By default, Untitled Request is mentioned if no title is provided to a request.

A screenshot of a computer

Description automatically generated

The Builder section also contains the request type (GET, POST, PUT, and so on) and URL. A request is executed with the Send button. If there are any modifications done to a request, we can save it with the Save button.



The Builder section has the tabs like Param, Authorization, Headers, Body, Pre-req., Tests and Settings. The parameters of a request in a key-value pair are mentioned within the Params tab. The Authorization for an API with username, password, tokens, and so on are within the Authorization tab.

The request headers, body are defined within the Headers and Body tab respectively. Sometimes, there are pre-condition scripts to be executed prior to a request. These are mentioned within the Pre-req. tab.

The Tests tab contains scripts that are run when a request is triggered. This helps to validate if the API is working properly and the obtained data and Response code is correct

A screenshot of a computer

Description automatically generated

# Workflow in Postman:

## **Creating and triggering request:**

Below are the steps to test an API in postman:

* Create a collection under workspace
* Under the collection, create the API request
* Click on the request to configure the
* Method: GET/POST/PUT/PATCH/DELETE
* URL
* Body: Raw-> Text/JavaScript/JSON/HTML/XML
* Authorization: select authorization type and set the values
* Click on save
* Click on send button to trigger the API
* Once the API request is triggered successfully, the response shall be visible under response tab

## **Validating the response:**

Once a request has been sent, we can see the Response code, response time and response size and the response body.

A screenshot of a computer

Description automatically generated

# Collection Runner in Postman:

Postman Collection Runner is used to execute a Collection having multiple requests together. All the requests within a Collection will be executed simultaneously.

Collection Runner does not produce any Response Body. The Collection Runner console displays the test results for individual requests. It is mandatory to have more than one request within the Collection to work with Collection Runner.

Follow the steps given below to execute the tests with Collection Runner in Postman:

* Click on the Runner menu present at the top of the Postman application.
* Select the Collection name from Choose a collection or folder
* Specify the number of times we need to iterate the request. We can also set a delay time in milliseconds for the requests
* Click on the Run Collection button

The Run Results page shall come up. Depending on the delay time provided, the tests should get executed. The test results (Pass/Fail) should be displayed for each iteration. The pass status is represented in green and failed ones are represented in red. If there is no test implemented for a particular request, then it shall display the message as - This request does not have any tests.