Hashedin

API TESTING

AGENDA

Hashedin

Introduction for API Testing

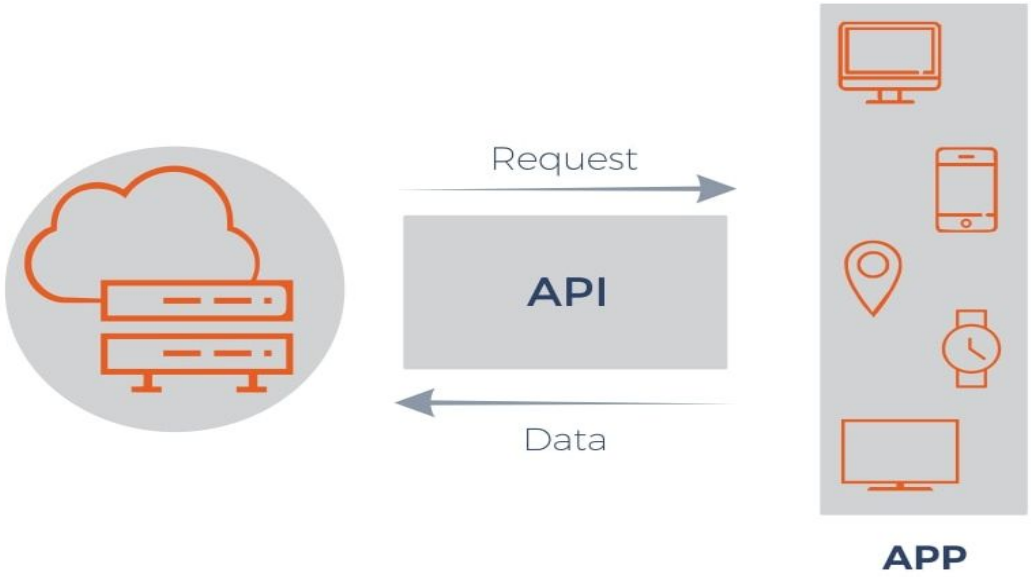
Postman

API Automation Testing using Python - Pytest

API - Application Programming Interface

Hashedin

- A set of rules and mechanisms that enable and specifies the interaction between various software programs or applications
- A set of rules that tell your application how to behave and where to go



Why API Testing ?

Hashedin

- Faster and more reliable than UI-based tests
- Rapid record manipulation
- To find bugs earlier in the development process, often before the UI has been created
- To make requests that might not be allowed through the UI, which is crucial for exposing potential security flaws in an application.

Types of APIs

Hashedin

- SOAP (Simple Object Access Protocol) APIs
- REST (REpresentational State Transfer) APIs
- RPC (Remote Procedure Call) APIs

- Private/ internal APIs
- Public / open APIs
- Partner APIs

https://searchapparchitecture.techtarget.com/definition/SOAP-Simple-Object-Access-Protocol
https://blog.rapidapi.com/types-of-apis/

Hashedin

How does a REST request works ?

Hashedin

A REST request is made up of the following parts:

- An HTTP Method that describes what action should be taken
- A Uniform Resource Locator (URL) that defines the location of the request
- HTTP headers that provide information to the server about the request
- A request body that provides further details for the request (this can sometimes be empty)
- Authorization

HTTP METHODS

Hashedin

- To indicate the desired action to be performed for a given resource
- The mostly used HTTP Methods are
  - GET : To retrieve data from a specific resource, reads information only
  - POST : To send data to a server to create a resource
  - DELETE : To delete the specified resource
  - PUT : Replaces all current representations of the target resource with the request payload

  - References
  - https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods
  - https://assertible.com/blog/7-http-methods-every-web-developer-should-know-and-how-to-test-them

HTTP Headers

Hashedin

- Request header: Headers containing more information about the resource to be fetched or about the client itself.
- Response header: Headers with additional information about the response, like its location or about the server itself (name and version etc.)
- HTTP headers can provide information to the server such as:
    - Host: the domain and port number of the user making the request
    - Authorization: the credentials of the user making the request
    - Content-Type: the format of the information provided in the body of the request

XML - Extensible Markup Language & JSON - JavaScript Object Notation

Hashedin

The body specifies exactly what information should be added to the database.

```
{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021"
  },
  "phoneNumbers": [
    {
      "type": "home",
      "number": "212 555-1234"
    } ],
  "sex": {
    "type": "male"
  }
   }
```

```
<person firstName="John" lastName="Smith" age="25">
  <address streetAddress="21 2nd Street" city="New York" state="NY"
postalCode="10021" />
  <phoneNumbers>
    <phoneNumber type="home" number="212 555-1234"/>
    </phoneNumbers>
  <sex type="male"/>
    </person>
```

STATUS CODES

Hashedin

- 1xx: Informational : It means the request has been received and the process is continuing.
- 2xx: Success : It means the action was successfully received, understood, and accepted.
- 3xx: Redirection : It means further action must be taken in order to complete the request.
- 4xx: Client Error : It means the request contains incorrect syntax or cannot be fulfilled.
- 5xx: Server Error : It means the server failed to fulfill an apparently valid request.

Reference: https://www.tutorialspoint.com/http/http_status_codes.htm

Hashedin

API Testing Tools

Hashedin

- Postman
- SOAP UI
- JMeter
- Rest Assured
- Katalon Studio
- Apigee
- HttpMaster

API TESTING USING POSTMAN

Postman is one of the most popular tool used to test rest API's

Why Postman

1. Accessibility
2. Use of Collections = Helps in organizing test suite
3. Automation Testing
4. Collaboration
5. Creating Environments
6. Debugging

How to use postman

Hashedin

1) Requests:
    - Http Methods (GET , POST, PUT, DELETE)
    - Request Url
2) Json Body
3) Authorization
4) Environment
5) Parameterization
6) Query param(Key and value pair)
7) Headers
8) Test Script
9) Assertion
10) Collection
11) Runner

- Introduction for Trello - https://trello.com
- Introduction for Trello API - https://developers.trello.com
- Working with GET Request
- Working with POST Request

API Automation Using Pytest

- Create new project
- Create new virtual environment
  From cmd prompt add **“pip install -U requests”** library and this library will be present inside the venv folder

API Automation

Hashedin

- Requests Library
- Json dumps and loads

Reference doc:

https://realpython.com/python-requests/#query-string-parameters

https://realpython.com/python-json/

https://www.guru99.com/pytest-tutorial.html

ASSESSMENT

- API AUtomation for Trello
  - Organisation
  - Members
  - Boards
  - Lists
  - Cards
- Generate HTML Reports
- Log on assertion failure
- POM
- Usage of fixtures