BASICS OF API TESTING

What is an API?

- An API (Application Programming Interface) allows communication between software components. It defines rules and protocols for accessing a web-based software application.

What is API Testing?

- It is a type of software testing that validates APIs directly to ensure they return correct responses for various requests and inputs.
-

What are the types of APIs?

- Open APIs (Public)
- Partner APIs (Restricted)
- Internal APIs (Private)
- Composite APIs (Combine multiple APIs)

What protocols are used in API testing?

- HTTP, HTTPS, SOAP, REST, GraphQL, WebSocket, JMS, etc.



BASICS OF API TESTING

What are RESTful APIs?

- APIs that follow REST (Representational State Transfer) principles using standard HTTP methods like GET, POST, PUT, DELETE.

What are HTTP methods?

- GET – Retrieve data
- POST – Create data
- PUT – Update data
- DELETE – Delete data
- PATCH – Partially update data

What is an HTTP status code?

It shows the result of the request:

- 200: OK
- 201: Created
- 400: Bad Request
- 401: Unauthorized
- 404: Not Found
- 500: Server Error



BASICS OF API TESTING

What are headers in API?

- Metadata passed in API requests/responses (e.g., Content-Type, Authorization).

What is payload in an API?

- The data body sent in POST/PUT requests.

What tools are used for API testing?

- Postman
- REST Assured
- SoapUI
- JMeter
- Newman
- Swagger

What is schema validation?

- Ensuring the API response follows a defined structure (JSON Schema, XML Schema).



BASICS OF API TESTING

What is a base URI in API testing?

- The root address of the API. Example: https://api.example.com/.

What is an endpoint in API testing?

- A specific URL that performs a function (e.g., /login, /users).

What is query parameter vs path parameter?

- Path: /users/{id}
- Query: /users?id=123
- What is JSON and why is it used?
- JavaScript Object Notation — used for sending structured data due to its readability and compatibility.

What is response time in API testing?
Time taken by the server to respond to a request.



BASICS OF API TESTING

What is rate limiting?

- Restriction on number of API calls allowed in a specific timeframe.

What is API throttling?

- Controlling traffic to avoid overloading the server.

What are negative test cases in API testing?

- Invalid method
- Missing headers
- Invalid input
- Unauthorized access



POSTMAN & COLLECTIONS

What is Postman?

- A GUI-based tool used to test APIs with support for automation and scripting.

What is a Postman Collection?

- A group of saved requests that can be executed manually or via automation.

What is the Collection Runner?

- Tool to run multiple API requests in a sequence with data files (CSV/JSON).

What is an environment in Postman?

- A set of key-value pairs (e.g., baseURL) used for dynamic test setup.

What are global variables in Postman?

- Variables accessible from any workspace or collection.



POSTMAN & COLLECTIONS

How to write test scripts in Postman?

- Using JavaScript in the Tests tab:
- Example:

```
pm.test("Status code is 200", function () {
    pm.response.to.have.status(200);
});
```

How to use pre-request scripts in Postman?

- Run JavaScript before sending a request (e.g., generate token).

How to test response body in Postman?

- Sample Javascript code:

```
pm.test("Name is John", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData.name).to.eql("John");
});
```



POSTMAN & COLLECTIONS

How to set dynamic variables in Postman?

- Sample Javascript code:

```
pm.environment.set("userId", jsonData.id);
```

How to pass data from CSV in Postman?

- Use {{variableName}} inside the request and link the CSV in Collection Runner.

What is Newman?

- CLI companion of Postman used to run collections via command line.

How to run Newman command?

- Sample bash code:

```
newman run collection.json -e environment.json
```

Can we integrate Postman/Newman with CI tools?

- Yes, using Jenkins, GitLab CI, Azure DevOps pipelines.



POSTMAN & COLLECTIONS

What is a monitor in Postman?

- A scheduled collection run used to monitor API performance and correctness.

How to validate response time in Postman?

- Sample Javascript code:

```
pm.test("Response time < 1000ms", function ()
{
pm.expect(pm.response.responseTime).to.be.below(
1000);
});
```

Can we chain API calls in Postman?

- Yes, by storing response data into variables and using them in subsequent requests.

How to handle authentication in Postman?

- Options:
- Bearer Token
- OAuth 2.0
- Basic Auth
- API Keys



POSTMAN & COLLECTIONS

What is dynamic data testing in Postman?

- Using random values (e.g., email, ID) using faker.js or Postman's dynamic variables.

Can Postman validate JSON schema?

- Yes, using the tv4 or ajv library in test scripts.

What is Postman sandbox?

- The JavaScript execution environment where Postman scripts run.



REAL-TIME SCENARIOS, ERRORS & VALIDATIONS

How do you validate response data in API testing?

- Check status code
- Validate headers
- Check response body content
- Verify schema
- Response time check

What is the difference between 401 and 403?

- 401 Unauthorized: Authentication failed
- 403 Forbidden: Authenticated but access denied

How do you handle token expiration in API tests?

- Use Pre-request Script to refresh token before each request
- Store updated token in environment variables

What is a mock server in API testing?

- A fake server that mimics real API responses for testing when backend is not ready.



REAL-TIME SCENARIOS, ERRORS & VALIDATIONS

What is the difference between PUT and PATCH?

- PUT replaces the entire resource
- PATCH modifies only specific fields

What is contract testing?

- Testing whether the API request and response structure follows the agreed contract (e.g., OpenAPI/Swagger).

What is the use of Swagger?

- API documentation
- Try out endpoints directly
- Generate client/server code
- Validate request/response schema

What are the main components in a REST request?

- Method (GET, POST, etc.)
- URL (with parameters)
- Headers
- Body (Payload)



REAL-TIME SCENARIOS, ERRORS & VALIDATIONS

How do you validate API pagination?

- Check limit, offset or page in response
- Ensure no duplicates
- Validate total count

How to test file upload API?

- Use form-data in Postman or use a curl command:

```
curl -F "file=@sample.pdf" http://api.example.com/upload
```

How to test API rate limiting?

- Send rapid requests
- Expect 429 Too Many Requests
- Validate retry-after header

What is response caching in APIs?

- Server caches the response
- Helps reduce load
- Controlled via headers like Cache-Control



REAL-TIME SCENARIOS, ERRORS & VALIDATIONS

What is CORS in APIs?

- Cross-Origin Resource Sharing — controls resource sharing across different domains.

What is the difference between synchronous and asynchronous API?

- Synchronous: Immediate response (e.g., REST)
- Asynchronous: Processed in background (e.g., Webhooks)

What is an API gateway?

- Manages API traffic, authentication, routing, rate limiting, caching.

How to validate date and time fields in APIs?

- Use regex or schema validation to check format (e.g., ISO 8601 yyyy-mm-ddTHH:MM:SSZ)



REAL-TIME SCENARIOS, ERRORS & VALIDATIONS

How to ensure API security during testing?

- Use OAuth2
- Validate access roles
- Test for SQL injection, XSS
- Avoid exposing sensitive data

What is idempotency in APIs?

- Multiple identical requests result in the same outcome (e.g., PUT should be idempotent).

What is an API key and how is it different from Bearer Token?

- API Key: Static string for identification
- Bearer Token: Dynamic token for authorization (via OAuth2)

How do you validate API logs and traces?

- Use API logging tools or server logs to check request/response, errors, latency, etc.



AUTOMATION TOOLS & FRAMEWORKS

What is REST Assured?

- A Java library used for testing REST APIs with a fluent java syntax:

```
given().get("/users").then().statusCode(200);
```

What is the structure of a REST Assured test?

- given(): Setup
- when(): Action
- then(): Validation

How to test headers using REST Assured?

- Java sample:

```
given().header("Authorization", "Bearer token")
```

What is a BDD framework in API testing?

- Behavior-Driven Development using Gherkin syntax (Given, When, Then) with Cucumber or Karate.



AUTOMATION TOOLS & FRAMEWORKS

What is Karate DSL?

- An API testing tool that uses Gherkin syntax.

Example(gherkin):

Given url 'https://api.com/users'
When method GET
Then status 200

What is the use of assertions in API automation?

To validate status, body values, headers, response time, etc.

What is schema validation in REST Assured?

- Java Sample:

response.then().body(matchesJsonSchemaInClasspath("schema.json"));

What are data-driven tests in API automation?

- Running tests with multiple data sets using external files like Excel, CSV, or JSON.



AUTOMATION TOOLS & FRAMEWORKS

What is the difference between manual and automated API testing?

- Manual: GUI tools like Postman
- Automated: Scripts executed repeatedly via CI/CD pipelines

What are some common assertions in API automation?

- Status code = 200
- JSON field = expected value
- Response contains array/object
- Response time < threshold

What tools are used for API performance testing?

- JMeter
- Gatling
- Locust
- k6



AUTOMATION TOOLS & FRAMEWORKS

What is parallel execution in API testing?

- Running multiple API tests simultaneously to reduce execution time.

How do you generate reports in API automation?

- Allure Reports
- ExtentReports
- Cucumber HTML Report
- JUnit/TestNG default reports

How do you mock APIs during automation?

- Use WireMock, MockServer, or Postman Mock Server to simulate real responses.

What are flaky tests in API automation?

- Tests that randomly pass/fail due to network delays, unstable responses, or race conditions.

How to handle dynamic values in API responses?

- Extract them using JSONPath or regex and pass into next request.



AUTOMATION TOOLS & FRAMEWORKS

What is JSONPath?

- A syntax to extract specific parts from JSON responses (like XPath for XML).
- Example: $.data[0].name

How to run API tests in CI/CD pipeline?

- Integrate using:
- Jenkins
- GitHub Actions
- GitLab CI
- Azure DevOps

How to handle large payloads in testing?

- Use external files or chunk the payload using streaming or paging.

How to version APIs and test them?

- Use versioning in URI (/v1/users) or headers. Validate backward compatibility.



ADVANCED & REAL-TIME EXPERIENCE BASED

What is backward compatibility in APIs and how do you test it?

- Backward compatibility ensures that old clients still work when the API is updated.
- ✅ Test by running older test cases against the new API version.

How do you handle chained API requests in testing?

- Extract data (like id, token) from one API response and pass it into the next request as a parameter or header.
- Bash script:
- Login → get token
- Pass token in header to fetch profile

How do you test API dependencies (microservices)?

- Validate that service A fails gracefully if service B is down
- Use mocks for unavailable services
- Ensure retries and fallbacks work



ADVANCED & REAL-TIME EXPERIENCE BASED

What is HATEOAS in REST?

- HATEOAS (Hypermedia as the Engine of Application State) provides links in responses for navigation.
- Json example:

```
{
  "userId": 1,
 "links": [
    { "rel": "self", "href": "/users/1" },
    { "rel": "orders", "href": "/users/1/orders" }
  ]
}
```

What is the difference between JSON and XML in API testing?

- JSON is lightweight, preferred in REST
- XML is verbose but allows schema validation (XSD) and namespaces



ADVANCED & REAL-TIME EXPERIENCE BASED

How do you test GraphQL APIs vs REST APIs?

- GraphQL uses a single endpoint, flexible queries
- Test query, mutation, and schema
- Use tools like Postman, Insomnia, Apollo

How do you manage environments in API testing tools?

- Use environments in Postman or config files in code frameworks. Store base URLs, auth tokens, etc.

What is the role of API documentation in testing?

- Defines endpoints, parameters, and response structure
- Swagger/OpenAPI enables testing and contract validation
- Reduces ambiguity



ADVANCED & REAL-TIME EXPERIENCE BASED

What is latency testing in APIs?

- Measures response time under normal and load conditions. Ensures SLA adherence.
- Bash script:

Start Timer → Send Request → Stop Timer → Assert Time < 200ms

What is an API sandbox?

- A safe test environment with mock data for testing API integrations without affecting production.

How do you validate retry mechanisms in APIs?

- Simulate a failure (timeout, 5xx) and validate if retries happen as expected using headers/logs.

How do you test for data integrity in APIs?

- Create → GET → Compare values
- Update → Validate values
- Delete → Ensure removal



ADVANCED & REAL-TIME EXPERIENCE BASED

What's the best way to test multi-language APIs?

- Set Accept-Language in headers and validate localized content in the response.

example:Accept-Language: fr-FR

What are some good API testing practices?

- Validate both happy and negative scenarios
- Automate smoke and regression tests
- Use data-driven tests
- Version your test cases

What is an API probe test?

- A lightweight periodic check to verify if APIs are up and responsive (used in monitoring systems).

What's the difference between integration testing and API testing?

- API Testing: Tests a single API's request/response
- Integration Testing: Verifies communication between multiple components or services

ADVANCED & REAL-TIME EXPERIENCE BASED

What is schema drift and how do you detect it?

- A mismatch between expected and actual API schema.

✅ Use contract/schema validation tools (e.g., Swagger Validator).

What is synthetic monitoring for APIs?

- Monitoring APIs using scheduled test calls from different locations to ensure availability and performance.

How do you test APIs that interact with databases?

- Validate DB after POST/PUT
- Compare DB state before and after DELETE
- Use SQL queries for backend validation



ADVANCED & REAL-TIME EXPERIENCE BASED

What are the common API testing challenges you've faced?

✅ Real-world answers could include:

- Unstable environments
- Unavailable services
- Delayed documentation
- Token expiration
- Rate limiting
- Payload complexity



THANK YOU

Created by

Besant Technologies Velachery