**Webservice basic interview Question**

1. **What is SOA**

SOA – Service Oriented Architecture

****

1. **How to test the Application without GUI**

Using API testing

1.WBT testing

2.Webservice Testing

1. **What is middleware testing?**

The developers would have created an interface. With the help of this Interface the test engineer is able (getting access) to test the Source code.

1. **What are main differences between API and Web Service?**

API [ Application Programming Interface] – It is an interface between two Programs

Web Service is the mechanism or medium of communication through two applications exchange the data irrespective of their underline technology.

1. **Difference between Unit testing & Web Service API testing**

Testing each and every line of the source code using Unit Testing tools is called as Unit Testing.

Testing the request and response between two applications is called web services testing.

1. **Difference between frontend & Backend testing**

|  |  |
| --- | --- |
| **Front End Testing** | **Back End Testing** |
| Front End Testing is always performed on GUI | Back End Testing is done on business layer |
| Testing becomes slower because it have to wait for browser rendering time | It is faster and saves time |
| Performance testing is not easy | Performance testing is easy and faster |
| Adhoc, Compatibility and Usability testing is possible | Adhoc, Compatibility and Usability testing is difficult |
| End to end testing is easy | End to end testing is difficult |

1. **Difference between Soap Web Service & Rest Web Service**

|  |  |
| --- | --- |
| **SOAP Web service** | **REST web service** |
| Simple object access protocol | Representational state transfer |
| SOAP is a Protocol | REST is architectural style |
| It supports XML data format only | It supports JSON, XML, JavaScript, HTML, Text data formats |
| SOAP can’t use REST web services as it is a protocol | REST can use SOAP since it is concept |
| SOAP APIs/ services are exposed via .wsdl file | REST APIs/Services are exposed via URI |
| SOAP is designed using too many standards | REST does not define too many standards |
| SOAP defines its own Security | Inherits security measures like Oauth-1.0, Oauth-2.0, Bearer token etc.., |
| SOAP is less preferred than REST | REST is more preferred than SOAP |

1. **Difference between XML & JSON language**

|  |  |
| --- | --- |
| **JSON** | **XML** |
| JSON data has a data type | XML data is type less |
| light weight | consumes more memory |
| Supports various data types like string, number, array, Boolean, null, Object | All XML data should be string |
| JSON is extension of JavaScript | XML is an extension of HTML |
| JSON is support by most browser | Cross-browser XML parsing can be tricky |
| Retrieving value in JSON is easy compared to xml | Retrieving value in XMLis difficulty compared to JSON |
| JSON is faster & easier for transporting the data | XML is slower |
| It doesn’t support comments | It supports comments |
| JSON files are easy to read as compared to XML | XML documents are relatively more difficult to read and interpret |
| It is less secured | It is more secure than JSON |
| Content type of JSON is application/json | Content type of XML is application/XML |
| Data is readily accessible as JSON Objects | XML data needs to be parsed |

1. **What is API?**

Application programming interface acts like a bridge/ interface/ a connection between two programs/machines/applications.

1. **Why API testing is required?**

to ensure that APIs are reliable, secure, and performant, thereby contributing to the overall quality of the software.

1. **What is Web Service?**

●Web Services is the mechanism or the medium of communication through which two applications/machines exchange the data irrespective of their underlying technology.

● Web services help two applications exchange information with each other when running on the same or different platforms.

● Web service helps us to share the functionality of one application to any other application without sharing the source code & database data.

● All web services are exposed via API.

● **All web services are API, but all APIs are not web services.**

1. **Why Web Service testing required?**

Web services testing is required in order to make sure that all the services exposed by the provider are working as expected with respect to functionality, performance, scalability, security etc.

(OR)

The purpose of web services testing is to verify that all of the APIs exposed by your application are working as expected WRT functionality, performance, scalability & security.

1. **What is the Advantages of Web service?**

**1. Web Services Interoperability (WS-I)**

● Web Services are “Application, Platform and technology Independent”.

● **Ex:** Uber/ OLA and Google Maps share the data among each other.

**2. Loosely Coupled**

● Each application is independent of one another. Hence changes done to one application will not impact the “unrelated areas”.

**3. No need of re-inventing the wheel/ build the feature from scratch**

● Web Services reduces the software development time.

● This helps the other business partners to quickly develop applications and start doing business.

● This helps businesses to save time and money by cutting development time.

● **Ex:** Uber/ OLA can make use of Google Maps.

**4. Business Opportunity**

● Web Services will open the door for new business opportunities by making it easy to connect with partners.

● **Ex:** Dominos can get the order from Food Panda/ Swiggy along with getting orders from its own site.

**5. Service Reuse**

● Web Services takes code reuse a step further.

● **Ex:** An organization can have a “Single Payment Gateway Services” which helps other web applications of the organization to interact.

**6. Scalability**

● Balancing the load by scaling up and down the servers.

1. **Types of Web Service testing**

**Types of web services Testing**

1. SOAP Web Service Testing

2. REST Web Service Testing

1. **What is Restful**

"RESTful" refers to web services that adhere to the principles of REST (Representational State Transfer), an architectural style for designing networked applications.

1. **Which scripting language will use in REST and SOAP Web Service**

JavaScript, Python, Ruby, PHP

1. **What is the syntax of URL & explain?**

**WEB URL:**

● Web uniform Resource locator is used to uniquely identify the specific web resource within the web application.

● Every web application should have its unique address in the form of a URL.

● URL is the one & only way to access web applications via a browser.

**Syntax: - Protocol://DomainName:PortNumber/ResourcePath?QueryString#FragmentId**

**Protocol:**

● When one application wants to communicate with another application there needs to be a common language which both the application understands, hence we use protocol.

● Protocol acts like a common language between two different applications/programs/machines to exchange information between them.

● It is a set of rules / instructions.

● Protocol is optional information in the URL & it is not case sensitive.

● The browser always sends a request & receives a response via HTTP protocol hence it is called HTTP request/ HTTP response.

**Types of protocol:**

1. HTTP-> Hypertext Transfer Protocol

2. HTTPS-> Hypertext Transfer Protocol Secure

3. FTP-> File Transfer Protocol. Ex: Google Drive

4. SMTP-> Simple Mail Transfer Protocol. Ex: Gmail

**Domain Name:**

● It is used to uniquely identify the specific server or computer within the network or area in which a web application is present.

● A domain name might be a computer name or IP address.

● It's mandatory information.

Ex: www.amazon.com -> commercial

.org -> Organization

.edu -> education

.gov -> government

.in -> India

**Port Number:**

● It is used to uniquely identify the specific software/ application inside the computer.

● In the URL this is optional information.

● If we are using the IP Address, then only the Port Number is required.

● Mobile automation-4000 series

● WebLogic- 2digits

● Apache J-boss – 3digits

● Apache tomcat- 4digits (in FireFlink we make use of Apache Tomcat)

● **Ex:** localhost:8888 -> V-Tiger

localhost:8080 -> Jenkins

localhost:4723 -> Appium

**Resource Path:**

● It is used to uniquely identify the specific web resource within the application server (web application).

**Query String:**

● Used to filtering the condition.

● It is one of the parameters/components of the URL which is used to specify/filter a condition / pass a value to a specific parameter.

● It always begins with “?”

● It is always written in “name” = value pairs.

● We can have any number of name value pairs separated by “&”

● **Ex:** http/localhost:8888/index.php?module=leads&action=index

**Fragment ID:**

● It is used to uniquely identify the specific fragment or section in the webpage.

● It’s not a mandatory field in URL.

● It should always begin with #

● **Ex:** #inbox, #Drafts etc..,

1. **Difference between URL and URI**

URL - uniform Resource locator is used to uniquely identify the specific web resource within the web application

URI – A URI is a string of characters used to identify a resource either by location, name, or both.

URI = URL+URN

1. **What is http Method? explain https method required in web service testing**
2. **What is CRUD operation**
3. **What is status code & list out the status Code?**
4. **What exactly you verify in Webservice API testing**
5. **Prerequisite for WebServcie testing ?**
6. **What is Query parameter & why its required ?**
7. **What is the properties of the HttpRequest**
8. **What is the properties of the HttpResponse**
9. **How you used to verify your API using Front End? Are you using any Tool?**
10. **Any API debugging tool which you used**
11. **. How are you passing your data (required info) while testing APIs?**
12. **What is Caching? Advantages and Disadvantages**
13. **What are cookies? Advantages and Disadvantages**

**Rest Assured Introduction Interview Question:**

1. **What is RestAssured**

RestAssured is an open source Java Library which is used for testing Restful web services

1. **Why RestAssured for API testing and why not Postman?**

Because in Postman we can only test business layer, but in RestAssured it can be Integrate seamlessly with existing java based framework and we can automate E2E business workflow which include all the layers of the Application.

1. **Why RestAssured is popular**

RestAssured it can be Integrate seamlessly with existing java based framework and we can automate E2E business workflow which include all the layers of the Application.

1. **Pre requisite for RestAssured**

* IDE is required [ Eclipse]
* Programming Knowledge
* Maven Project
* TestNG
* JSON language knowledge
* API documents
* API test cases

1. **Rest Assured supported languages?**

Java, Kotlin, Groovy, Scala

1. **Advantages of Rest Assured?**

* Supports all http methods
* Supports for BDD
* Use hamcrest Matchers for validation/Assertions
* Provide inbuilt method to validate response header and body
* Handles various authentication like Oauth1.0, Oauth2.0, token, Basic auth
* Integrates seamlessly with existing java based framework like selenium
* Used for complete Backend automation
* Open source headless client
* Framework can be integrate with CI/CD pipeline

1. **What is use of RestAssured class in restAssured?**

The RestAssured class in the Rest Assured library serves as the primary entry point for configuring and making HTTP requests to RESTful web services. It provides a fluent interface to set up and execute requests, assert responses, and manage the overall testing process. Here are the main uses and features of the RestAssured class:

**RestAssured Architecture Interview Question:**

1. **RestAssured is a class or interface?**

RestAssured is a class

1. **What is the use of RequestSpecification API in restAssured;**

Used to set specification of the request like Content Type, Body, Parameters, Authentication, Header

1. **What is the use of RequestSpeBuilder API in restAssured;**

Used to set precondition for the request

This is used to build request specification before sending request, we can define various aspects like Content Type, Body, Parameters, Authentication, Header

RequestSpecBuilder reqBuilder = new RequestSpecBuilder();

reqBuilder.setContentType(*ContentType*.***JSON***);

reqBuilder.setBaseUri(fLib.getDataFromPropertiesFile("BASEURL"));

*reqSpecObj* = reqBuilder.build();

1. **What is the use of Response API in restAssured;**

Used to receive the response of the request

1. **What is the use of ResponseSpeBuilder API in restAssured?**

Used to set the specifications of the response

This is used to build response specifications for validating responses, we can define expected status code, headers, body using this builder

1. **What is the use of ValidatbaleResponse API in restAssured?**

Used to Validate the response

1. **What is the return type of all CRUD Methods?**

Response Interface

1. **Why HTTP methods available in Both RestAssured as well as RequestSpecification**

For Code Optimization

**RestAssured- CRUD Operation**

1. **How to perform CRUD operation in restAssured?**

By using http methods like post, get, put, patch, delete

1. **How many ways we can create a post Request in RestAssured?**

* **Static Methods** in the RestAssured class for quick requests.
* **given() Method** for more configuration.
* **RequestSpecification** for reusable and detailed configurations.
* **Form Parameters** for form data submissions.
* **Multi-Part Form Data** for file uploads.
* **Authentication** for secure requests.
* **Query Parameters** for additional parameters in the request.

1. **How to post complex Request In RestAssured?**

Using POJO classes

1. **Advantages of BDD in RestAssured**

* Enhanced Readability
* Improved Collaboration
* Easier Maintenance
* Encourages Best Practices

1. **How to use BDD feature in RestAssured?**

To use BDD features in Rest Assured, you follow the Given-When-Then structure to write your tests in a readable and structured format. This approach enhances the readability of the tests and aligns them closely with the behavior-driven development (BDD) methodology

1. **What is the use given(), when() , then() in RestAssured?**

**given():** Used to set up the request, including headers, parameters, and body.

**when():** Specifies the action to perform (e.g., GET, POST, PUT, DELETE).

**then():** Used to validate the response.

**Serilization**

1. **What is serialization and deserialization?**

Process of converting java Object to JSON is called serialization.

Process of converting JSON to Java Object is called deserialization.

1. **Why serialization required in API testing**

Because in API testing for ensuring data integrity, efficient communication, and compatibility between different systems.

1. **Have you used any Parsing tool in JSON serialization**

jackson-databind, gson

1. **How to convert JAVA🡺 JSON and JSON 🡺JAVA**

Using Jackson parser tool

1. **What is ObjectMapper?**

ObjectMapper in Java, provided by the Jackson library, is essential for converting Java objects to JSON and vice versa

1. **What is POJO class?**

**POJO** [Plain Old Java Object]

POJO classes are business specific java class is used to get the complex JSON Object during the Post API request.

1. **Can you write one POJO class**

**Yes,**

public class Project {

private String createdBy;

private String projectName;

private String status;

private int teamSize;

public Project() {

}

public Project(String createdBy, String projectName, String status, int teamSize) {

super();

this.createdBy = createdBy;

this.projectName = projectName;

this.status = status;

this.teamSize = teamSize;

}

public String getCreatedBy() {

return createdBy;

}

public void setCreatedBy(String createdBy) {

this.createdBy = createdBy;

}

public String getProjectName() {

return projectName;

}

public void setProjectName(String projectName) {

this.projectName = projectName;

}

public String getStatus() {

return status;

}

public void setStatus(String status) {

this.status = status;

}

public int getTeamSize() {

return teamSize;

}

public void setTeamSize(int teamSize) {

this.teamSize = teamSize;

}

}

**Complex post request**

1. **How many ways we can post a request via RestAssured?**
2. **How to post a complex request**
3. **Can you write one restassured program to post complex request**
4. **Where do you use hashmap in restassured and why**
5. **Why JobjectObject class is required in restassured**
6. **Why pojo class required in restassured**

**Response validation with RestAssured Assertion**

1. **What exactly you verify in response**

Status Code, Response Time and Expected Results

1. **How to put assertion in restassured?**

By using then() method

1. **What exactly you verify in response header**

Status Code, Status line, Response time

1. **What exactly you verify in response Body**

Business requirement

1. **How do we verify the response time in response**

then().time(Matchers.*lessThan*(2000l))

1. **How do we verify the complex response in jsonbody**

By using jsonpath

1. **What is Jsonpath and write few example?**

JsonPath is used to traverse to specific key from root object

1. **How do we verify the dynamic response**

By using Jayway JsonPath

1. **How to capture the multiple data from Complex jsonbody**

By using JsonPath class

1. **How to extract required data from response**

JsonPath

1. **Write one complex response body**
2. **Why JSONPATH libraries required in RestAssured**

To extract the data based on our condition

1. **Write one Complex JSonXpath for the response body**
2. **Any open source tool used to write jsonpath?**

JsonPath Finder

1. **How to traverse in JSON response body**

By using .

1. **What is connection = alive**

This means that the server intends to keep the connection open after the current request/response cycle, allowing for multiple requests and responses to be sent over the same connection without reopening it each time.

1. **What did you verify in response header and body**

**Request chaining**

1. **What is request chaining ?**

Getting the data from one API response and passing the data into the API request is called request chaining

1. **Write one sample program for request chaining**
2. **How to automate end to end scenario in API testing**

By using request chaining

1. **How to automate micro webservice architecture based application**
2. By using request chaining
3. **Write one sample restassured program for request chaining**

**Type of parameters**

1. **What is API parameter and explain the types parameter?**

An API parameter is a variable part of an API request that allows users to customize and control the behavior of the API call

**Types of Parameters**

**1.Path Parameter:** Used to set the resource path/ end point.

**2.Query Parameter:** Used to set query string

**3.Form Parameter:** Used to pass the sensitive data in Request body

Not applicable for get, only applicable for put, patch, post

**4.Param Parameter:** It is a generic parameter for query and form parameter

**-** It acts like query parameter if it contains query data for get method

**-** It acts like form parameter if it contains form data for put, patch and post method

1. **Difference between path and Query parameter**

**1.Path Parameter:** Used to set the resource path/ end point.

**2.Query Parameter:** Used to set query string

1. **Difference between form and Query parameter**

**Authentication**

1. **What is Authentication and authorization in API testing**

**Authentication:** used to check whether the user is valid or not?

**Authorization:** Used to check the user permission across to the resource or not?

1. **Explain type of authentication**

**1.** Basic authentication

-Digest authentication

- Pre-Emptive authentication

**2.** Bearer Token

**3.** Oauth1 authentication

**4.** Oauth2 authentication

1. **What is oauth protocol in api testing**

It is an open standard for authorization protocol, widely used in web application which allows API services to access user data without sharing password

1. **Difference between digestive and preemptive auth**

**Digestive Auth:** In this client sends the username and password in encrypted with MD5[Message Digest Algorithm Technique]. Password is not exposed in the Network

**Preemtive Auth:** Password is exposed in the network

1. **Difference between Outh1.0 and Outh2.0**

|  |  |
| --- | --- |
| **Oauth 1.0** | **Oauth 2.0** |
| In order to access the Oauth1.0 we need to pass Consumer Key and Secret, Access Token and Secret | In Order to access the Oauth2.0 we need to pass Client ID and Client Secret |
| Provide 2 level of authentication | Provide 1 level of authentication |
| Token will not have expiry | Token will have expiry |
| Scope: Doesn’t have feature of scope | Scope: Client credentials, Password Credentials, Authorization Code etc |
| Complex and Highly secure | Simple and Secure |

1. **What is bearer token authentication?**

Bearer token is a type of token used for authentication and authorization, the bearer token is cryptic String and client must send this token in the authorization header when making requests for protected requests

1. **What is the status code of Forbidden and unauthorized**

**Unauthorized: 401**

**Forbidden: 403**

**Encryption and Decryption**

1. **What is encryption ?**

Encryption is a process used to convert plaintext (readable data) into ciphertext /cryptographic (unreadable data) using an algorithm and an encryption key.

1. **What is encoding**

Encoding is the process of converting data from one format to another

Rest Assured all basic authentication by default uses Base64 encoding facility

1. **Different type of encryption**
2. Symmetric Encryption
3. Asymmetric Encryption
4. **What is public and private key in encryption**

A public key is a cryptographic key that can be freely distributed and shared with anyone. It is used to encrypt data.

A private key is a cryptographic key that is kept secret and is only known to the owner. It is used to decrypt data.

1. **Which encryption technique used in restassured in basic auth**

Advanced Encryption Standard [ AES]

1. **What is BASE-64 encryption**

Base64 is a method for encoding binary data into ASCII characters and send encoded data over the network

**Data driven testing**

1. **Why data driven testing in rest api testing**

* Reusability of Data
* Suites can be executed in different Environments easily
* Running Test Script with different Test Data
* Maintenance and modification of Test Data is easy

1. **How to execute same API request with multiple set of data**

Using DataProvider

1. **Write one sample program to execute API with multiple data**

public class AddMultipleProjects\_DataProvider {

*@Test*(dataProvider = "getData")

public void addProject(String projectName) {

Project pObj = new Project("Raj", projectName, "Created", 10);

*given*()

.contentType(*ContentType*.***JSON***)

.body(pObj)

.when()

.post("http://49.249.28.218:8091/addProject")

.then()

.assertThat().statusCode(201)

.log().all();

}

*@DataProvider*

public Object[][] getData() {

Object[][] data = new Object[3][1];

data[0][0] = "V-Tiger\_21";

data[1][0] = "V-Tiger\_22";

data[2][0] = "V-Tiger\_23";

return data;

}

}

1. **Why random class in API testing**

To Generate and Pass the Random data to our request

1. **Advantages of data driven testing**

**Data Simulation**

1. **What is synthetic data**

Synthetic data is artificially generated data that acts real-world data but does not directly originate from actual events

1. **What is data simulation tool, and why required?**

Data Simulation is a tool where it takes care of creation artificial data and same data to be inserted in to Application via API or DB Query.

1. **How many ways to develop data simulation tool**

2 Ways

1. API Based Simulation tool
2. DB-Query Based Simulation tool
3. **What to create data simulation tool**

**Mocking**

1. **What is mock/Stub/ data virtualization**

Stubs are used during Top down integration Testing in order to simulate the behavior of lower-level modules or Applications that are not yet integrated.

Stubs are the modules or Applications that acts as a temporary replacement for a called module or Application and give the same output as that of the actual product.

1. **Why mock is required in testing**

Mocks are an essential component in testing, particularly in unit testing, integration testing, and end-to-end testing. Mocks simulate the behavior of real objects and are used to isolate the functionality being tested

1. **Name any tool to develop mocking**

Mockito, Spark

1. **What is mock data and mock server?**

**Mock data** refers to artificially created data that is used to simulate real data during testing.

**Mock server** is a server that mimics the behavior of a real server by responding to client requests with pre-defined responses. It is used to simulate interactions with external services or APIs.

1. **How to create mock server**

Write Java Code to Create the Mock Server

Convert that to Runnable jar file

1. **Difference between data simulation and mocking**

|  |  |
| --- | --- |
| **Data Simulation** | **Data Mocking** |
| Data simulation involves generating data that closely resembles real-world scenarios or conditions. It aims to replicate the characteristics, distributions, and variability of actual data. | Mocking involves creating objects or components that mimic the behavior of real objects or components within an application. It is used to isolate and test specific functionalities or interactions without relying on actual implementations. |
| Data simulation is more prevalent in data-centric applications | mocking is widely used in software testing, especially for API testing and unit testing. |
| Use when real data is impractical or unavailable, or when testing requires specific data scenarios and distributions. | Use when testing requires isolating and simulating interactions with external dependencies or when testing needs to focus on specific behaviors or functionalities within an application. |

1. **Where did you used mocking solution in your previous project?**
2. **Can we develop mocking solution for dependent business module?**

Yes

1. **Can we develop mocking solution for third party application which integrated via API?**

Yes

1. **How run mock server?**

Write Java Code to Create the Mock Server

Convert that to Runnable jar file

Run it in CMD using ---> java –jar filepath

1. **Write one sample rest assured to program to use mock server**

public class CreditCardMock {

public static void main(String[] args) {

Spark.*port*(8889);

Spark.*post*("/credit-card", (req, res) -> {

String response = "";

String card = JsonPath.*read*(req.body().toString(), "$.creditCard");

if (StringUtils.*equalsAny*(card, "12345678901234", "98765432101234")) {

response = "{\r\n" + " \"status\" : \"Payment Success\"\r\n" + "}";

res.status(200);

} else {

response = "{\r\n" + " \"status\" : \"Payment Failed\"\r\n" + "}";

res.status(404);

}

res.type("application/json");

return response;

});

System.***out***.println("Running.....");

}

}

**Utility**

1. **Explain the type of utility used in your API Framework?**
2. **What generic and business utility**
3. **Write few sample rest API utility used in framework?**
4. **Explain the advantages of utilities**

**Unit testing**

1. **Which unit testing tool used in your Framework**
2. **Explain the annotation used in your framework**
3. **How to execute batch in api testing?**
4. **How to execute same suite in difference environment**
5. **Where do SET BASE\_URI and AUTH**
6. **What is use RequestSpecBuilder and ResponsespecBuilder in API testing and where its being used in you framework**

**END points**

1. **. what is endpoints**

1. **. Where did you kept end points in your framework**
2. **. How may end points you tested in your framework**

**Rest Assured FrameWork**

1. **. explain the restassured framework components**
2. **. Explain the execution flow of the framework components**
3. **. Can we execute restAssured framework via Jenkins**

**Others**

1. **. How to handle SSL & TSL certification in restAssured?**
2. **. what is GRPC in API testing**
3. **. what is GRPH-QL in api testing**
4. **. how to handle cookies in api testing**
5. **Can we handle OTP in api testing**
6. **. how to upload image in API testing**