## 1.How do you identify whether it is a FE or BE issue?

**FE**

. performance issue

. UI Issue

. UX Issue

BE

. API or Backend issue

**FROUNT END ISSUE:**

**We check for UI is alinement is proper or not or front problem**

**API OR BACKEND ISSUE:**

**When click any button it has to open or redirect and page should be open and in network tab it if any issue that time we consider as API or Back end issue**

## 2. What will be the response when we use PUT instead of POST method?

When we use send **put** request **creates a new resource or updates**

**When it create we get 201, when is update 200 or 204**

**[**If the target resource does not have a current representation and the `PUT` request successfully creates one, then the origin server must inform the user agent by sending a `201` (`Created`) response.**]**

**When we use send Post request it will create we will get 201**

**[**If the target resource does have a current representation and that representation is successfully modified in accordance with the state of the enclosed representation, then the origin server must send either a `200` (`OK`) or a `204` (`No Content`) response to indicate successful completion of the request.**]**

### 3. What will be the response when we use GET instead of POST method?

**When we use GET instead of POST method we will get response 200 ok**

### 4. What will be the response if we provide a Request Body while using the GET method?

**we not provide any request body when we using get Method ,GET method response 200 ok**

### 5. What is the difference between PUT and PATCH methods?

**PUT: When we using PUT Method it will Create 201 or update 200 or 204.**

**PATCH: When we using PATH Partially update 200 response**

### 6. What will happen when we use PUT instead of PATCH method?

**When we using PUT instead of PATCH it creates and response 200 ok**

### 7. What is the difference between Cookies and Cache?

| S.NO | Cache | Cookies |
|---|---|---|
| 1. | Cache is employed to store the web site content for the long run purpose. | While cookie is employed to store user choices. |
| 2. | Cache's website contents are stored in browser only. | While cookie's contents are stored in both server and browser. |
| 3. | It expires manually. | While it expires automatically. |
| 4. | It consumes large space in terms of capacity. | While it consumes less space in terms of capacity. |
| 5. | the types of cache are: Browser cache and proxy cache. | While the types of cookies are: Transient and persistent cookies. |
| 6. | Cache stores the contents like html pages, images, JavaScript, CSS etc. | While cookie store the contents like browsing sessions and temporary tracking data. |
| 7. | Cache does not send the response with requests. | While cookie sends the response with requests. |
| 8. | Caches are less memory efficient. | Cookies are more memory efficient. |
| 9. | In Cache, content of the website is save only on browser. | In Cookies, content of the website is save on both server and browser. |

### 8. Types of Cookies.

**HTTP cookies:**

A small piece of data that a server sends to a user's web browser. The browser may store the cookie and send it back to the same server with later requests. Types of Cookies:

. **Session.**

**. Persistence.**

**. Third Party**

**Session cookies:** Session cookies are mostly use in secure application it is based on inactive time or session timeout. once inactive timeout over it shows pop up able to continue on session, if it's not it will logout the application automatically it shows the login page. Mostly inactive time 3 to 5 mints its bases on application **Ex: Banking, healthcare application.**

**Persistence cookies:** Persistent cookies are stored in your web browser once you closed it. Persistence cookies are mostly use in moderate secure and non-securer application it is based on longer inactive timeout or longer duration timeout in the type of cookie after longer duration it logout application without showing any pop up or an expiration date and are automatically removed when that date is reached from browser**. Ex: Gmail, fb**

**Third party cookies**: Third-party cookie is placed on a website by someone other than the owner and collects user data for the third party the user information that they can gather. Online advertising is the most common use of third-party cookies. A third-party cookie is created by a domain separate from the website. **Ex: advantaging, online offers,**

**9. Two mobiles with same hardware and software configuration, in one mobile application home screen is displayed once app is launched,**

**in other mobile blank screen is displayed once app is launched. What is the reason behind it and how will you debug?**

**10. What is an Idempotent method?**

An HTTP method is **idempotent** if an identical request can be made once or several times in a row with the same effect while leaving the server in the same state. In other words, an idempotent method should not have any side effects — unless those side effects are also idempotent. Implemented correctly, the `GET`, `HEAD`, `PUT`, and `DELETE` methods are **idempotent**, but not the `POST` method. All safe methods are also idempotent.

**11. Difference between status code 401 and 403?**

**401 unauthorized ----> request is not authorized to access the resource inside the server**

**403 Forbidden response status code indicates that the server understands the request but refuses to authorize it**

**12. Why do we get 500 Internal Server Error?**

**500 internal server side error --->problem from server side**

**13. What http method is invoked when we use search/filter?**

**14. Can we create a resource using the PUT method?**

**When we using PUT Method it will Create 201 or update 200 or 204**

**15. Can we create a resource using the GET method?**

**When we using GET Method we can't create resource**

### 16. What is the difference between API and Web Services?

| Web Services | Web API |
| --- | --- |
| Web services are a type of API, which must be accessed through a network connection. | APIs are application interfaces, implying that one application can communicate with another application in a standardized manner. |
| Web service is used for REST, SOAP and XML-RPC for communication. | API is used for any style of communication. |
| All Web services are APIs. | APIs are not web services. |
| It doesn't have lightweight design, needs a SOAP convention to send or receive data over the system. | It has a light-weight architecture furthermore, useful for gadgets which have constrained transmission capacity like smart phones. |
| It provides supports only for the HTTP protocol. | It provides support for the HTTP/s protocol: URL Request/Response Headers, and so on. |
| It is not open source, however, can be devoured by any customer that comprehends xml. | It is an open source and also ships with .NET framework. |
| Web service supports only XML. | API supports XML and JSON. |
| Web Services can be hosted on IIS. | Web API can be hosted only on IIS and self. |

### 17. What are the advantages of API testing?

a. API testing very Faster (because no to wait for Browser rendering time)

b. testing the functionality without GUI(Browser)

c. testing the functionality early stages (We can start API testing in Sprint-1)

d. find defect in early stages

e. Time effective & fast to release

f. whenever API provider develop an API, every API should be tested the Functionality, performance security, reliability before exposing those API to consumer

### 18. What are the challenges you face while doing API testing?

a. End to End Scenario testing is challenging because we have to do API Chaining

EG: Scenario →Search Product + ADD to cart + Billing + Logistic

API : API -1 for Search Product

API-2 for ADD to cart

API-3 for Billing

API-4 for Logistic

b. API Document is not clear, API testing challenging
c. Negative testing is challenging because in api document will not have complete requirement
d. Validation of Complex response is challenging
e. Deriving api test scenario is challenging ,

### 19. What is the maximum payload size that can be sent in the POST method?

**The HTTP protocol does not specify a limit**. The POST method allows sending far more data than the GET method, which is limited by the URL length - about 2KB.

### 20. Difference between HTTP and HTTPS?

| S.No. | HTTP | HTTPS |
|---|---|---|
| 1. | HTTP stands for HyperText Transfer Protocol. | HTTPS for HyperText Transfer Protocol Secure. |
| 2. | In HTTP, URL begins with “http://”. | In HTTPs, URL starts with “https://”. |
| 3. | HTTP uses port number 80 for communication. | HTTPs uses 443 port number for communication. |
| 4. | HTTP is considered to be unsecure. | HTTPs is considered as secure. |
| 5. | HTTP works at Application Layer. | HTTPS works at Transport Layer. |

| S.No. | HTTP | HTTPS |
|---|---|---|
| 6. | In HTTP, Encryption is absent. | Encryption is present in HTTPS. |
| 7. | HTTP does not require any certificates. | HTTPS needs SSL Certificates. |
| 8. | HTTP does not improve search ranking | HTTPS helps to improve search ranking |
| 9. | HTTP faster than HTTPS | HTTPS slower than HTTP |
| 10. | HTTP does not use data hashtags to secure data. | While HTTPS will have the data before sending it and return it to its original state on the receiver side. |

**21. While testing a web application we get a blank screen; how will you debug it?**

**22. For the given JSON payload, how will you identify the scenarios**

```
{
   first_name : "string",
   last_name : "string",
   phone_no : integer,
   email_id : "string"
}
```

**23. What is the difference between Authentication and Authorization?**

Authentication: used to check whether you are valid user or not?

Authorization: used to check your permission / accesses to the resource

**24. Types of Authorization?**

. Basic Auth (send a request using username/ password)

. Bearer Token (send a request using tokenID, but token is fixed)

. OAuth-1.0(older)

. OAuth-2.0 (send a request using tokenID , but token ID is dynamic created via Outh-cleintAPP) or (Gmail app , allow grant permission to skillRaray app without sharing your username/password via Oauth-2 Protocol )

## 25. Difference between O_Auth1.0 and O_Auth2.0

| Oauth1.0 | Oauth2.0 |
|---|---|
| older version of protocol for authentication | latest version of protocol for authentication |
| Two level authentication required for every api | One level authentication required for every api |
| complex authentication approach | Simple authentication approach compare to Oath1.0 |
| To get bearer token , should pass consumerID , consumer secret & accessID , access secret | To get bearer token , should pass Client & Client Secret |