In simple words, API stands for **A**pplication **P**rogramming **I**nterface. API acts as an interface between two software applications and allows the two software applications to communicate with each other. API is a collection of software functions that can be executed by another software program.

API works as; it takes a request from the source, takes that request to the database, fetches the request data from the database and returns a response to the source. API takes the requests from the user and gives the response without exposing the internal details. API acts as Abstraction.

Assume an **API** as a **Waiter** at a **Restaurant**.

At a restaurant, you give an order based on the items available on the menu. A waiter in the restaurant writes down your order and delivers it to the kitchen who prepares your meal. Once the meal is ready, the waiter picks up your food from the kitchen and serves it to you at your table.

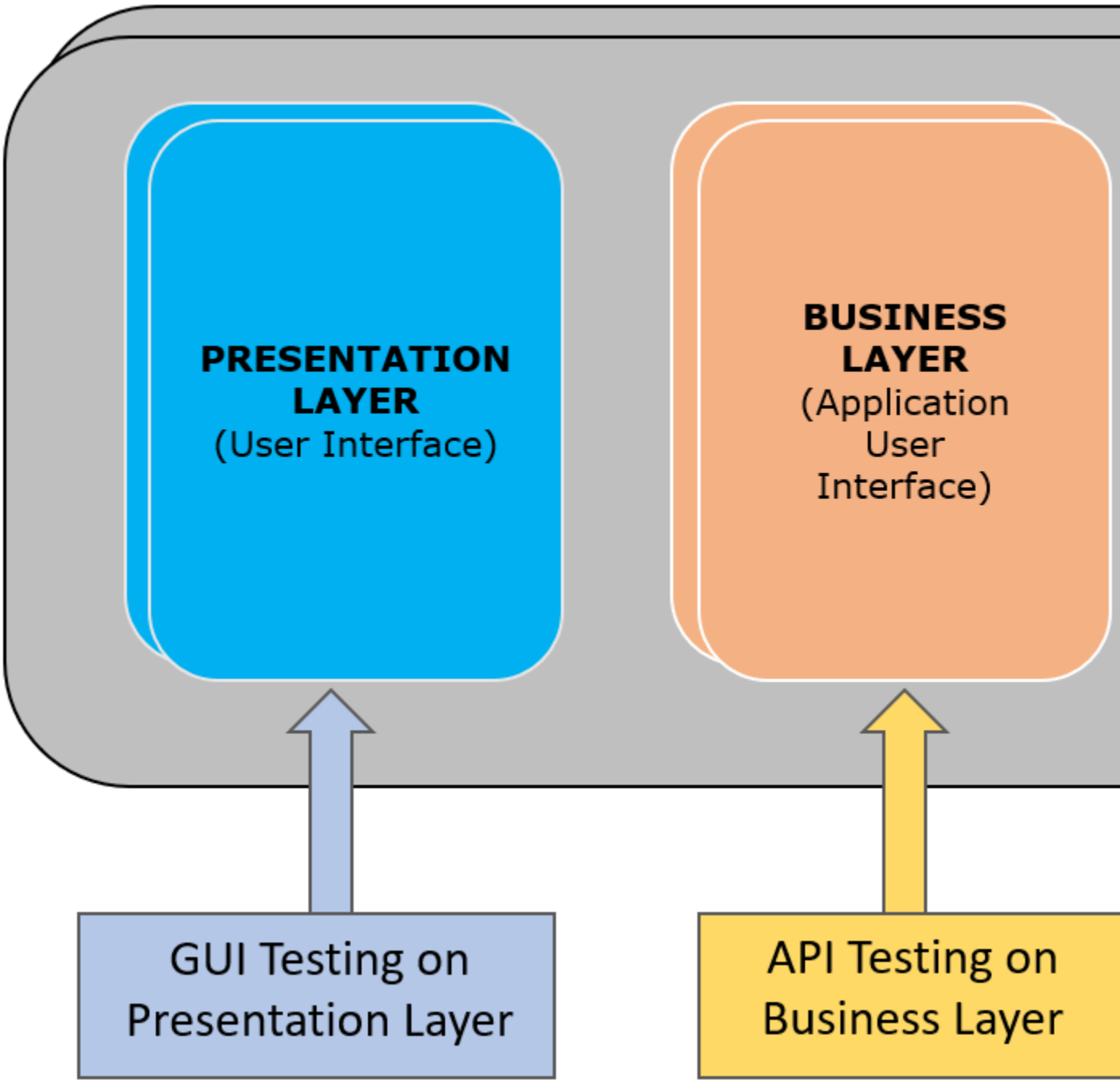
In this scenario, the waiter's role is similar to an API. As a waiter, the API takes a request from a source, takes that request to the database, fetches the requested data from the database, and returns a response to the source.

Now let's see another example.

If you are using a flight service engine say Expedia, where you search for flights on a specific date. Once you pass the data such as Source, Destination, Onward Date, and Return Date and click on search. Expedia sends a request to airlines through an API as per your search details. The API then takes the airline's response to your request and delivers it right back to Expedia.

API gets the request from the user and gives the response without exposing internal logic. API acts like an Abstraction in the OOPs concept.

What is API Testing?



API testing is a type of software testing that involves testing APIs directly and also as a part of integration testing to check whether the API meets expectations in terms of functionality, reliability, performance, and security of an application. In API Testing our main focus will be on a Business logic layer of the software architecture. API testing can be performed on any software system which

contains multiple APIs. API testing won't concentrate on the look and feel of the application. API testing is entirely different from GUI Testing.

Let's see how is UI testing is not similar to API testing?

UI (User Interface) testing is to test the graphical interface part of the application. Its main focus is to test the look and feel of an application. On the other hand, API testing enables communication between two different software systems. Its main focus is on the business layer of the application.

API Testing Types?

API testing typically involves the following practices:

- **Unit testing:** To test the functionality of individual operation
- **Functional testing:** To test the functionality of broader scenarios by using a block of unit test results tested together
- **Load testing:** To test the functionality and performance under load
- **Runtime/Error Detection:** To monitor an application to identify problems such as exceptions and resource leaks
- **Security testing:** To ensure that the implementation of the API is secure from external threats
- **UI testing:** It is performed as part of end-to-end integration tests to make sure every aspect of the user interface functions as expected
- **Interoperability and WS Compliance testing:** Interoperability and WS Compliance Testing is a type of testing that applies to SOAP APIs. Interoperability between SOAP APIs is checked by ensuring conformance to the Web Services Interoperability profiles. WS-* compliance is tested to ensure standards such as WS-Addressing, WS-Discovery, WS-Federation, WS-Policy, WS-Security, and WS-Trust are properly implemented and utilized
- **Penetration testing:** To find vulnerabilities of an application from attackers
- **Fuzz testing:** To test the API by forcibly input into the system in order to attempt a forced crash

12) What are the common tests that performed on API?

Here, are the common tests that performed on API are as:

1. Response of the API should be verified based on the request. We will verify that the return value is based on request.
2. When API is updating any data structure we should verify the system is authenticating the outcome.
3. We will verify whether the API is trigger other event or request another API.
4. We will verify the behavior of the API when no value is return.

What exactly needs to verify in API testing?

In API testing, we send a request to API with the known data and then analysis the response.

1. We will verify the accuracy of the data.
2. Will see the HTTP status code.
3. We will see the response time.
4. Error codes in case API returns any errors.
5. Authorization would be check.
6. Non-Functional testing such as performance testing, security testing.

Difference between API testing and Unit Testing?

UNIT TESTING:

- Unit testing is conducted by the Development Team
- Unit testing is a form of White box testing
- Unit testing is conducted prior to the process of including the code in the build
- Source code is involved in Unit testing
- In unit testing, the scope of testing is limited, so only basic functionalities are considered for testing

API TESTING:

- API testing is conducted by QA Team
- API testing is a form of Black box testing
- API testing is conducted after the build is ready for testing

- Source code is not involved in API testing
- In API testing, the scope of testing is wide, so all the issues that are functional are considered for testing

Advantages of API Testing:

- API Testing is time effective when compared to GUI Testing. API test automation requires less code so it can provide faster and better test coverage.
- API Testing helps us to reduce the testing cost. With API Testing we can find minor bugs before the GUI Testing. These minor bugs will become bigger during GUI Testing. So finding those bugs in the API Testing will be cost-effective to the Company.
- API Testing is language independent.
- API Testing is quite helpful in testing Core Functionality. We can test the APIs without a user interface. In GUI Testing, we need to wait until the application is available to test the core functionalities.
- API Testing helps us to reduce the risks.

21) What are the differences between SOAP and REST API?

Sr. No.	SOAP API	REST API
1.	**SOAP** stands as Simple Object Access Protocol.	**REST** stands as Representational State Tra
2.	**SOAP** is a protocol.	**REST** is an architectural pattern.
3.	**SOAP** can work with XML format. In SOAP all the data passed in XML format.	**REST** permit different data format such as etc. But the most preferred format for tran