Burcu Kündüm

Q2 SOA – Web Service - Restful Service – Http Methods

1) SOA – Service-Oriented Architecture

Applications are brought together in pieces with the Service-Oriented Architecture (SOA), where different services are created separately. These services can be built using different platforms and even different programming languages, communicating with each other to form the entirety of the application. Java is suitable for SOA because it is independent of the platform.

Applications are not composed of a single piece; rather, they consist of different services, allowing these services to be used in different applications, thereby facilitating development processes. Additionally, designing the structure in modular components simplifies development and modification processes.

When we consider the structure of a bank, from the moment a customer enters the bank, there are a plethora of financial products offered by the bank throughout the customer's lifetime. The customer can use credit to borrow, opt for a credit card to facilitate daily life, or utilize asset products to grow their savings.

For each product, there are specific documents that the bank needs to collect in the workflows. The flow to be established for these documents can be written into each product, however; in this case, the effort required for development and updating will be significant. By using the SOA architecture with a document service, repetitive processes are prevented, and in regulatory changes, updating a single service reduces effort and prevents errors.

2) Web Service

This method facilitates communication between different applications, enabling them to access each other's data or utilize each other's functions. Web services operate platform-independently, meaning it doesn't matter which programming language they are written in. This allows them to work seamlessly across different devices and operating systems.

There are generally two main ways web services work:

a) REST (Representational State Transfer):

REST enables data exchange using web standards. It is easy to use and flexible and the more frequently preferred method

b) SOAP (Simple Object Access Protocol)

It is used for more formal situations. SOAP has an XML-based protocol and is more suitable for complex operations.

While the REST method is preferred for convenience in applications such as weather forecasts and e-commerce, the SOAP method is preferred for financial transactions. SOAP provides the necessary security standards for processes requiring high security, such as money transfers.

3) Restful Service

It is an architectural approach used to create web services based on Rest principles. They use Http methods. RESTful services are a widely used approach in the development of modern web applications. The use of these services provides advantages such as flexibility, scalability and compatibility across different platforms.

In this structure, data is represented as resources accessed through Uniform Resource Identifier (URI) URLs. Operations on these resources are executed using HTTP methods. Clients communicate with resources using JSON and XML formats.

RESTful services are stateless, implying that each request must contain all necessary identifying information. The server does not retain any state about the client.

RESTful services play a crucial role in facilitating data exchange between mobile applications and servers. Within these applications, HTTP methods like PUT and GET are utilized for receiving input from the client and providing output to the client, ensuring efficient communication and interaction between the mobile app and the backend server.

4) HTTP - Hypertext Transfer Protocol

It is an application layer protocol used to request and serve hypermedia resources, such as web pages, images, videos, and other files on the Internet. Simply put, it manages communication between your web browser and web servers. It is one of the fundamental technologies that makes the web work. HTTP is mainly used to establish the relationship between client and server, transfer and verify data.

HTTP requests operate with a stateless structure, meaning each request is independent of others. However, certain elements like connection information or cookies can introduce stateful behavior. Despite the stateless nature of individual requests, HTTP can exhibit stateful characteristics due to the retention of such data in the background.

There are two primary versions of HTTP:

a) HTTP/1.1: This version is the most commonly used. It enables the transmission of multiple requests and receipt of responses within a single connection, thereby improving efficiency compared to its predecessor, HTTP/1.0.

b) HTTP/2: Introduced as a more advanced iteration, HTTP/2 boasts enhanced speed and efficiency over HTTP/1.1. It achieves this through features like multiplexing, header compression, and server push, which collectively optimize data transfer and reduce latency.