

6. SOA - Web Service - Restful Service - HTTP methods kavramlarını örneklerle açıklayınız.

(5 Puan)

### **SOA(Service Orianted Architecture-Hizmet Yönelimli Mimari)**

Bir yazılım mimarisidir. Bu mimari en temelde her hizmetin farklı birimler tarafından birbirlerinden bağımsız olarak çalıştırılmasını baz almaktadır. Yazılım dünyasında yoğunlukla kullanılan ve değiştirilmesi zor olan monolitik mimariye alternatif bir mimaridir. Özet olarak büyük yazılımların parçalara ayrılarak ağ üzerinden iletişim kuran servis yapılarına bölümlenmesidir.

Monolitik mimari ile tasarlanan uygulamalarda tüm fonksiyon ve modüller tek bir paket içerisinde bulunur. Bu sebeple uygulama yaşamı boyunca tek bir parça halinde hareket eder. Herhangi bir fonksiyonun güncellenmesi durumunda tüm uygulamanın derlenmesi gerekmektedir.

SOA ise yazılım bileşenlerini hizmet arayüzlerini kullanarak yeniden kullanılabilir hale getiren dağıtık sistem mimarisidir. Bu arayüzler yeni uygulamalara hızla entegre edilebilecek şekilde ortak iletişim standartlarını kullanır.

Bir SOA içerisindeki her hizmet bağımsız bir iş fonksiyonunu yürütmek için gerekli olan kodu ve veri entegrasyonunu kapsar. Hizmet arayüzleri temel yapıya gevşek bağlama şeklinde entegre edilir. Bu sayede entegrasyon kolay bir şekilde yapılır ve arayüzler kolayca çağırılabilir. Hizmetler SOAP/http veya JSON/http gibi standart ağ protokollerini kullanır. Web servislerin SOA mimarisine göre tasarlanması faydalı olacaktır.

### **Web Services**

Farklı platformların birbirleri ile iletişim kurabilmesi için ortak bir biçim kullanmaları gerekmektedir. Web servisler uygulamalarımızı platform bağımsız hale getirmek için kullandığımız temel yapılardan bir tanesidir. Bu servis türü farklı uygulama veya cihazların WWW(World Wide Web) altyapısı ve http protokolünü kullanarak iletişim kurabilmelerine olanak sağlar. Bu iletişim XML, JSON vb. ortak bir biçim kullanarak veri alışverişini sağlamaktadır. Yapılacak tüm iletişim belirtilen biçimler kullanılarak gerçekleştirildiğinden web servisi herhangi bir işletim sistemi veya programlama diline bağlı değildir. Web servis oluşturabilmek için en çok tercih edilen web servis yapıları SOAP ve REST yapılarıdır.

SOAP - Simple Object Access Protocol - Basit Nesne Erişim Protokolü

REST - Representational State Transfer - Temsili Durum Aktarımı

## Restful Services

Restful servislerden önce bu kavramın temelini oluşturan REST kavramını bilmek gerekmektedir. REST (Temsili Durum Aktarımı) SOA(Hizmet Yönelimli Mimari) temelli oluşturulan yazılımlarda kullanılan bir yazılım mimarisidir. Kullanıcı ile sunucu arasında hızlı ve verimli bir şekilde iletişim gerçekleştirilebilmesi amacıyla kullanılır. Bu iletişim için http protokolünü kullanmaktadır. İletişim sırasında taşınan verilerde istemciye ait detaylar mevcut değildir, ekstra başlık bilgileri bulunmaz. REST standartlarına uygun olarak oluşturulan web servisleri Restful servisler olarak adlandırılır. Günümüzde sağladığı hızlı ve verimli iletişim sebebiyle en çok tercih edilen web servisi konumuna gelmiştir. Restful servisler veri iletiminde farklı http methodlarını kullanmaktadır.

### Http Method

Http yöntemleri kullanıcı(web browser) ile sunucu(web server) arasındaki iletişim yöntemini belirtmek amacıyla kullanılmaktadır.

Get: Bu metod sunucudaki kaynaklara erişerek sunucudan veri almak kullanılırlar. En çok kullanılan metotdur. Tarayıcı URL'si üzerinden gönderilir. URL ekranda görüldüğünden gizlilik istenilen istekleri Get methodu kullanarak göndermek doğru olmayacaktır.

<http://servise.example.com/users> (Bütün kullanıcıları getir)  
<http://servise.example.com/users/1> (Id'si 1 olan kullanıcıyı getir)

Post: Bir kaynak oluşturmak ve kaynağa ait veri göndermek amacıyla kullanılır. Bu metod ile gönderilen veriler http isteğinin hem url'sinde hem de istek gövdesinde saklanabilir. Ancak sadece istek gövdesinin kullanılması güvenlik açısından daha doğru bir kullanım türüdür.

<http://servise.example.com/users> (yeni bir kullanıcı bilgisi oluşturulur)

Put: Belirli bir kaynaktaki verinin tamamının ya da bir kısmının değiştirmesi için kullanılır. Güncellenecek kaynağın id'sinin url ile gönderilmesi gerekmektedir. Kullanıcının sadece belli verileri değiştirilmek istenirse bile istek gövdesinde kullanıcı verilerinin tamamının bulunması gerekmektedir. Put metodu birden fazla kez çağırıldığında her zaman aynı sonucu verecektir.

<http://servise.example.com/users/1> (Id'si 1 olan kullanıcıyı güncelle)  
{ "name": "Şafak", "surname": "Akın" }

Patch: Belirli bir kaynaktaki verilerin bir kısmının değiştirilmesi için kullanılan metoddur.  
<http://servise.example.com/users/1> (Id'si 1 olan kullanıcıyı kısmen güncelle)  
{ "name": "Şafak" }

Delete: Sunucudaki bir kaynağı silmek için kullanılır.  
<http://servise.example.com/users/1> (Id'si 1 olan kullanıcıyı sil)

Head: Get metoduna benzer şekilde çalışır. Get metodundan farklı olarak ise sadece başlığı olan bir istek göndermesidir. Genellikle sunucuda bir kaynak mevcut mu veya kaynağın en son güncellenme bilgisi için kullanılır.

Connect: Sunucu ile bağlantı oluşturma isteği gönderir. Sunucu bağlantılarını minimum yük ile test etmeye olanak sağlar.

Options: Sunucunun desteklediği metotları kontrol etmek için kullanılır.

Trace: Bu metod ile bir sunucuya istek gönderdiğinizde, aradaki tüm vekil sunucular (Proxy, Gateway) isteğin başlığına kendi IP veya DNS bilgilerini eklerler. Genellikle hata ayıklama/bakım işleri için kullanılır.

Search: Bir dizinin altındaki kaynakları sorgulamak için kullanılır.