* Tüm platformlar benim kullandığım metotları kullanmak

isterse bunlara dll formatında veremeyeceğim için service

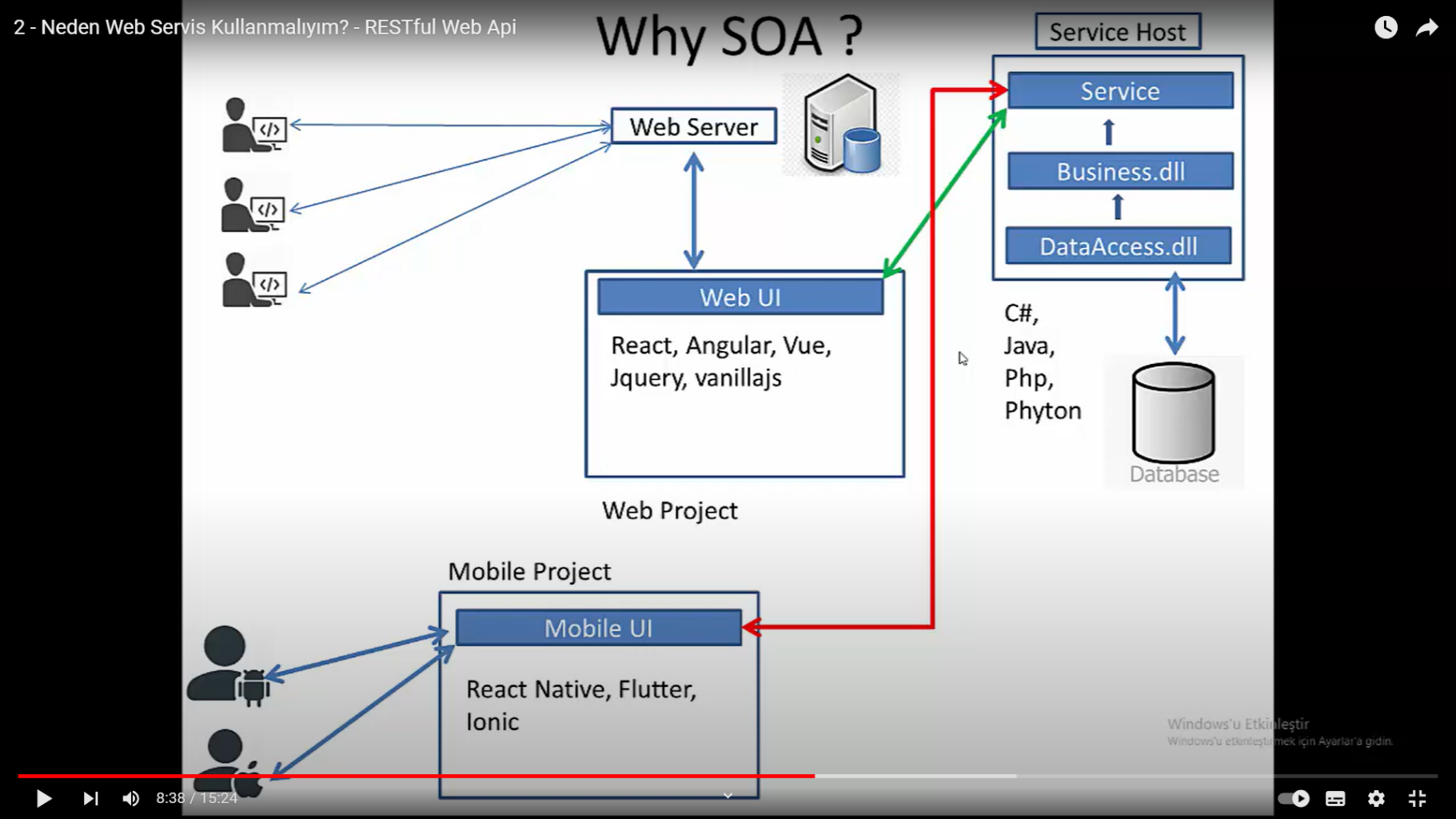
olarak sunmam gerekiyor.

SOP(SOA) -> Hizmet odaklı programlama ->

**1)** Mobil Arayüzü,Web arayüzü,masaüstü yani tüm platformlar tek bir yerden yönetiliyor ve yapılan değişiklikler(CRUD işlemleri) tüm platformlara anında yanstılıyor böylece tüm platformlarda veriler aynı kalarak tutarlılık sağlanıyor.

**2)** Servisin Hangi programla dilini ve teknolojiyi kullandığının önemi yok service olarak paylaşıldığı için.

**3)** Servisin nereye kurulduğunun da bir önemi yok.

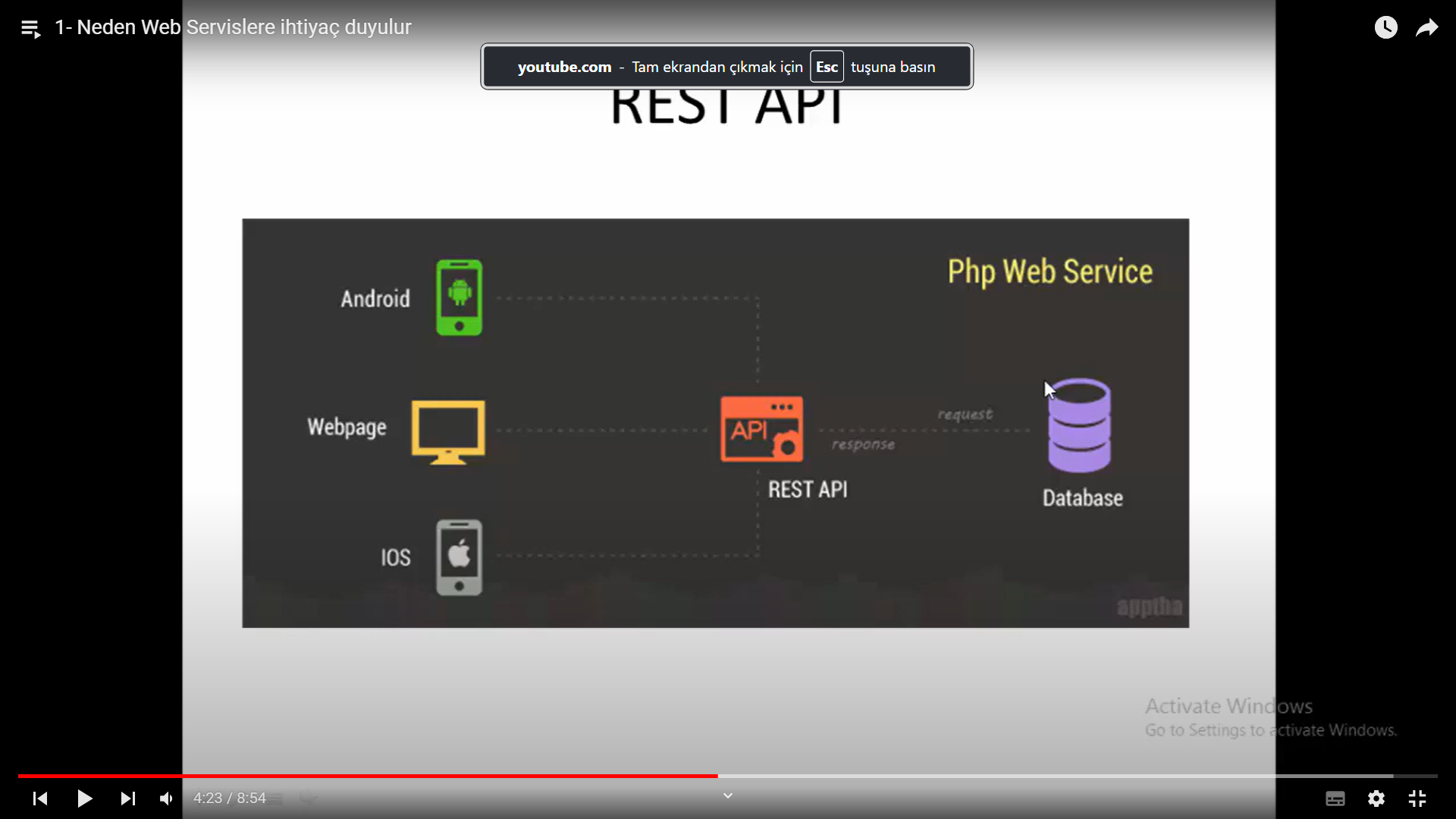


-REST-ful Services'lerde Web Apiler-

İnternete bağlanan tüm cihazlarda ve cihazların kullandığı tüm işletim sistemlerinin API aracılığıyla veri tabanına erişip veritabanı ile karşılıklı veri alışverişi yapıp tüm platformlara aynı verinin yansıtılmasını sağlayarak veri tutarlılığını sağlar.

Yani özetle veri tabanı API aracılığıyla tek bir yerden yönetim tüm platformlardaki cihazların veritabanı üzerindeki yaptığı değişimler diğer tüm cihazlara yansıtılması.

Örneğin, bir kullanıcının profili mobil uygulama veya web uygulaması üzerinden güncellendiğinde, bu değişikliklerin merkezi bir veritabanına kaydedilip diğer platformlara da yansıtılması sağlanabilir.



 Veritabanı API aracılığıyla tek bir merkezi yerden yönetilir.

 Tüm platformlardaki cihazlar, bu API'leri kullanarak veritabanına erişir.

 Bir cihazda yapılan veritabanı değişiklikleri, diğer tüm cihazlara API aracılığıyla hızlı bir şekilde yansıtılabilir.

 Bu sayede veri tutarlılığı sağlanır ve farklı platformlardaki uygulamalar arasında senkronizasyon mümkün olur.

Controller -> denetleyici,yönetici

Configure -> Yapılandır

Request -> talep,istek

Response -> Yanıt

URL -> Kaynağın web adresi

Routing -> Yönlendirme

Use -> kullanmak

Startup -> Başlangıç

Environment -> Ortam

Exception -> İstisna.

Example -> Örnek.

Page -> Sayfa.

Attribute -> Nitelik.

From -> -den,-dan, itibaren.

FromBody-> Gövdeden.

Find() -> Bulmak.

Varying -> Değişen.

Schemas -> Şemalar.

Definition -> Tanım,Tanımlama.

Implement -> Uygulamak.

Able -> Yapabilen.

Nullable -> Gerçesiz kılınabilen.

Disable -> Devre dışı bırakmak.

Enable -> Olanak vermek, izin vermek..

Access -> Erişim.

Layer -> Katman.

Repository -> Depo.

Complete -> Tamamlamal, Bitirmek.

Disposable -> Tek kullanımlık.

Dispose -> Elden çıkarmak.

Exception -> İstisna

Enumerable -> Sayılabilir.

Enum -> Sıralama.

Tracking -> İzleme.

Any -> Herhangi.

İnit -> içinde.

Layout -> Düzen.

Result -> Sonuç.

Action -> Eylem.

Temel Farklar

Kapsam: Find() metodu sadece List<T> için geçerlidir, FirstOrDefault() metodu ise tüm IEnumerable<T> ve IQueryable<T> koleksiyonları için kullanılabilir.

Kullanım Alanı: FirstOrDefault(), LINQ sorgularında yaygın olarak kullanılır ve daha geniş bir kullanım alanına sahiptir.

Performans: Find(), sadece List<T> için optimize edilmiştir ve bu nedenle List<T> üzerinde biraz daha hızlı olabilir. Ancak FirstOrDefault(), LINQ kullanarak daha geniş bir veri kaynağı üzerinde çalışabilir.

Her iki metodun da belirli senaryolar için kullanımı uygundur ve koleksiyonun türüne ve kullanım senaryosuna bağlı olarak tercih edilmelidir.