-Javascriptin web tarafında bu kadar popüler olma sebebi nedir? Kendi düşüncelerinizle belirtiniz.

Dinamik veb uygulamalarını html,css veya farklı backend uygulamalarında tasarlayabiliriz fakat,sayfanın olduğu gibi tekrar yüklenmesini(postback) beklemek zorunda olmadan mobil uygulama kullanır gibi yanıt alabiliriz. Js ile bir api ile iletişimde olup, dinamik bir web sitesi tasarlayabiliriz.Bckend bir programlama dili olmasının yanında,kütüphane desteği de oldukça zengindir.

-Javascript frameworkleri nedir ve ne işe yarar?

Framework, yazılım geliştiricilerin kullandığı önceden hazırlanmış kütüphanelerin bunluduğu ve bunlara yenilerini ekleyebileceği yapıların adıdır. Framework size iş kurallarıyla tam uyumlu, yapılandırılmış ve hem bakımı hem de yükseltilebilir bir uygulama geliştirdiğinize dair kesinlik sağlar.

-Http nedir?

1990 yılından beri dünya çapında ağ üzerinde kullanılan bir iletişim protokolüdür. HTTP‘nin açılımı “Hyper Text Transfer Protocol” yani “Hiper Metin Transfer Protokolü“dür. HTTP protokolü ağ üzerinden web sayfalarının görüntülenmesini sağlayan protokoldür.

-Cors nedir?

Cross-Origin Resource Sharing (Kökenler arası kaynak paylaşımı) anlamına gelen CORS, web tarayıcısı tarafından yönetilen ve ek HTTP başlıkları kullanılarak, bir kökende çalışan web uygulamasının, farklı bir kökende yer alan web uygulamasına erişim izni kontrolünü sağlayan mekanizmadır. Web uygulaması, internet tarayıcısı üzerinden farklı bir kökene (protokol, domain ve port) herhangi bir istek gönderirse cross-origin HTTP isteği oluşturmuş olur.

-DOM nedir?

Document Object Model, "DOM", "Belge Nesnesi Modeli" anlamlarına da gelmektedir. İnternet tarayıcıları girilen internet sitesini bir belge, bu belge içerisinde bulunan tüm elemanları da nesne olarak kabul eder. Buna göre resim, yazı, form gibi tüm elemanlar nesnedir.

-JWT nedir?

JSON Web Token, tarafların birbirleri arasındaki veri alışverişini ve bunun doğrulamasını sağlayan JSON tabanlı RFC 7519'de tanımlanmış açık bir standarttır. Örneğin bir sunucu, kullanıcının yönetici ayrıcalıklarına sahip olduğunu belirten bir anahtar oluşturabilir ve bunu kullanıcıya gönderebilir.

-Local storage ve cookienin farkları nelerdir?

Cookie, en sık kullanılan storage yöntemlerinden biridir. Bilgiler client tarafında saklanır bu yüzden yük clientlardadır. Burada her client kendi cookie bilgisini tutar. Yani chrome da tutulan çerez bilgisi chrome özelindedir. Local Storage, Local storage çerez mantığına (kalıcı çerez) benzer şekilde client tarafında tarayıcıda saklanır. Yani yük yine istemci tarafında saklanır. Çerezler ve yerel depolama farklı amaçlara hizmet eder. Çerezler öncelikle sunucu tarafını okumak içindir, yerel depolama yalnızca istemci tarafı tarafından okunabilir.

-Frontend, Backend, Veritabanı birbiriyle nasıl iletişim kurar?

Bugün, front-end ve backendin nasıl etkileşime girdiğini tanımlayan iki ana mimari vardır.

Sunucu tarafından oluşturulan uygulamalar

Birincisi, bir \*\*sunucu tarafından oluşturulan \*\*uygulamaya doğrudan HTTP istekleridir. Bu, tarayıcının bir HTTP isteği gönderdiği ve sunucunun bir HTML sayfasıyla yanıt verdiği bir sistemdir.

İsteği alma ve yanıtlama arasında, sunucu genellikle veritabanını sorgular ve onu bir şablona (ERB, Blade, EJS, Gidonlar) besler.

Sayfa tarayıcıya yüklendikten sonra, HTML onların ne olduğunu, CSS nasıl göründüklerini ve JS herhangi bir özel etkileşimi tanımlar.

AJAX kullanarak iletişim

AJAX, Asenkron JavaScript ve XML anlamına gelir. Bu, tarayıcıya yüklenen JavaScript'in \*\*sayfa içinden \*\*bir HTTP isteği (XHR, XML HTTP İsteği) gönderdiği ve geçmişte bir XML yanıtı aldığı anlamına gelir.

Günümüzde yanıtlar da JSON formatında yapılmaktadır.

Bu, sunucunuzun isteklere JSON/XML yanıtlayan bir bitiş noktasına\* sahip olması gerektiği anlamına gelir. Bunun için iki protokol örneği REST ve SOAP'tır.

İstemci tarafı (tek sayfa)

AJAX, sayfayı yenilemeden veri yüklemenizi sağlar. Bu, Angular ve Ember gibi çerçevelerle maksimuma çıkarıldı. Bu uygulamalar paketlenir, tarayıcıya gönderilir ve sonraki\*\* işleme \*\*istemci tarafında (tarayıcı) yapılır.

Bu ön uçlar, JSON/XML yanıtlarını kullanarak HTTP üzerinden arka uçla iletişim kurar.