TTFB:time to first byte:İlk gelen içeriğin zamanı.

TTI:time to interactive:Kullanıcının domda işlem yapabileceği an.

FCP:İlk içeriğin hazır olacağı zaman.

CSR:Client Side rendering. =SPA single page application

1-Kullanıcı istek atar

2-sunucu basit bir html dosyası döner.Css ve js linkleri ile beraber.

3-Browser gerekli js dosyalarını indirir.

4-İndirilen vue veya react ,javascript ile dom oluşturur.(ssr da html domu oluşturur.)

5-Sonraki isteklerde kullanılan react-router veya vue router ile istekler client kısmında yakalanır ve tekrardan vue veya react istenen domu oluşturur.Sadece fetch,axios gibi yöntemlerle sunucuyla bağlantı kurulur.

Avantajlar:

İlk yüklemeden sonra sayfa geçişleri çok hızlı olacaktır.

Fetch,axios istekleri dışında backende ihtiyacımız olmaz.Hatta backend hiç kullanılmayadabilir.

TTFB:hızlı İlk içerikle içeriğin hazır olması arasındaki zaman fazladır.

Dinamik routing işlemleri yapılabilir.

Dezavantajlar:

Seo (Search Engine Optimization) işlemlerini kötü etkiler.Dinamik meta etiketleri oluşturulamaz.

İlk yükleme büyük bir sorun olsada vue-router veya react-router kullanılarak bu çözülebilir.Vue router kısmında lazy load işlemi sadece istenilen dizindeki bilgileri yüklemeyi sağlar ve önbelleğe alır.Böylece tüm uygulamayı yüklemek yerine istenilen dizin eklenmiş olur ve uygulama hızı olduça artar.

SSR=SERVER side Rendering:

1-kullanıcı istek atar

2-sunucu işlenmeye hazır html döndürür.

3-browser html doma dönüştürür.Ancak js dosyaları eklenmemiştir.

4-js dosyaları eklenir ve sayfaya uygulanır.

Avantajlar:

-İlk yükleme hızlıdır.

-Seo dostudur.Dinamik meta etiketleri kullanımı sağlar.

TTFB:yavaş FCP ve TTI arasında min zaman dilimi vardır.

Dezavantajlar:

-Her sayfa için backende istek gider.

- SSR kısmında fetch işlemlerinin server tarafında yapılabilmesinden dolayı ve server internetininde kullanıcı internetinden daha hızlı olmasından dolayı istekler daha hızlı döner ve zaman kazanımı olur.

PRERENDERİNG:İsteği yapan bot ise statik html backendde verir.Kullanıcı ise csr gibi davranır.Belirli rotalar için html i server tarafında oluşturmayı sağlar.Seo dostu olması yüzüdnden ssr seçilecekse prerendering daha mantıklıdır.

<https://github.com/chrisvfritz/prerender-spa-plugin> Webpack ile kullanılır.Vite ile değil.

-