WEB UYGULAMALARI

Arayüz olarak web sitelerinin kullanıldığı uygulamalar web uygulaması olarak adlandırılır. Daha açık bir ifadeyle dinamik web sayfalarının ve genellikle bir veritabanının kullanıldığı sistemlere web uygulamaları denir.

Internet üzerinde sörf yapan bir kullanıcının bir web uygulamasıyla karşılaşma olasılığı çok yüksektir. Başta arama motorları olmak üzere, forum, elektronik ticaret, eğlence, belge yönetimi, kütüphane hizmetleri vs. gibi bir çok site web uygulaması şeklinde dizayn edilirler.

Dünyada yaygın olarak kullanılan arayüzler php, asp, cgi uygulaması, java ya da .net kullanılarak tasarlanırlar. Temelde yaptıkları iş bakımından birbirlerinden farklı olmasalar bile php'nin diğer programlama dillerine göre daha fazla kullanıldığını söylemek yanlış olmayacaktır.

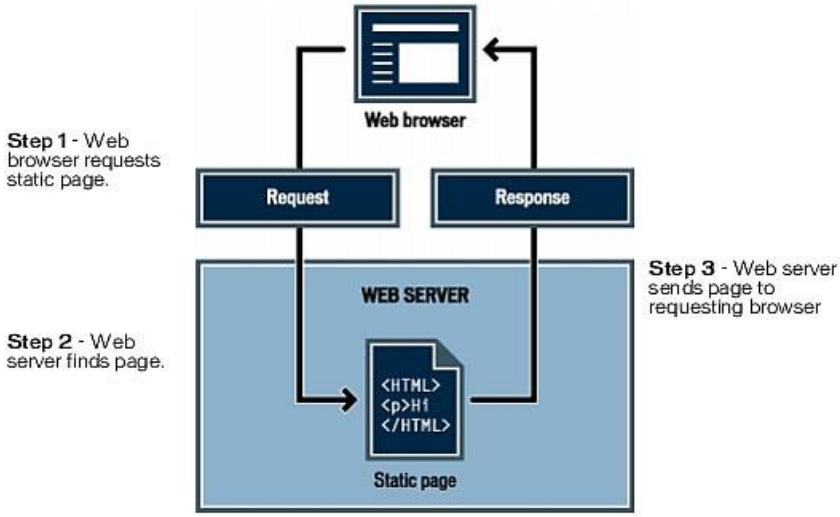
Web uygulamalarından önce server-client yazılımlarının kullanıldığı sistemler bulunuyordu. Bir uygulamanın kullanılması, sunucu üzerinde çalışan bir programa, istemci bilgisayara yüklenen özel bir program sayesinde ulaşılması şeklinde oluyordu. Ancak web uygulamaları ile sadece sunucu taraflı bir uygulama ile Internet ya da Intranet üzerinden browser aracılığıyla buna ulaşan bir istemci yeterli olmaktadır. İstemci bilgisayar üzerinden yapılan her istek, sunucu tarafından yorumlanır, sonuç html kodları içeren dinamik sayfa yardımıyla browser'a gönderilir.

1995 yılında Netscape firması istemci taraflı script (betik) dili olan Javascript'i duyurdu. Bu dil istemci tarafından programcıların bazı bileşenleri kullanarak dinamik sayfalar oluşturmaya yaramaktadır. Sunucu tarafından herhangi bir veri yorumlanmadığından bu bir web uygulaması olarak nitelendirilemez.

WEB UYGULAMALARI NASIL ÇALIŞIR?

Bir web uygulaması, dinamik ve statik sayfaların bir derlemesidir. Statik sayfalar ziyaretçinin (kullanıcının) isteğine göre değişmezler; sunucu web sayfası üzerinde herhangi bir değişiklik yapmadan sayfayı kullanıcının kullandığı browser'a gönderir. Dinamik sayfalar ise statik sayfaların tam tersine browser tarafından yapılan isteğe uygun olarak sunucu tarafından üretilen web sayfalarıdır. Bu sayfalara dinamik denmesinin sebebi de bu özellikleridir.

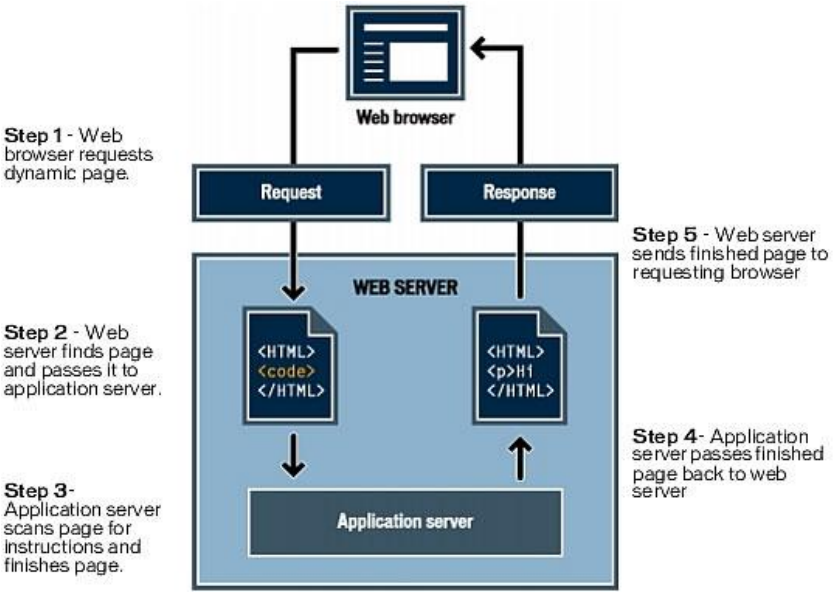
Statik bir web sitesi, içinde bir takım html sayfalarını barındıran ve web sunucu hizmeti bulunan bir server bilgisayar üzerinde bulunur. Web sunucu hizmeti, browserdan yapılan istek üzerine ilgili sayfaları gösteren bir yazılımdır. Tasarımcı tarafından hazırlanan sayfaların, kullanıcının yaptığı istek üzerine değiştirilmesi söz konusu değildir. Sayfa üzerinde çeşitli flash (swf) veya gif animasyonlarının olması teknik açıdan sayfanın statik olduğu sonucunu değiştirmez.



Yukarıdaki şekilde statik bir web sayfası isteğinde bulunan web browser'a, web sunucu hizmeti tarafından ne şekilde yanıt verildiği gösterilmiştir. Web browser tarafından istenen statik sayfa, sunucu tarafından bulunur ve gönderilir.

Statik bir sayfa isteğinde bulunan web browser'a ilgili sayfa doğrudan gönderilir. Web sunucusuna dinamik bir sayfa isteği gelmişse durum bundan daha karmaşıktır. İstek yapılan sayfanın gönderilebilmesi için isteği işlemekten sorumlu özel yazılımın bu sayfayı üretmesi gerekir. Burada bahsi geçen özel yazılım uygulama sunucusudur (application server).

Uygulama sunucu dinamik sayfa üzerindeki kodları okur, buradaki emirleri yerine getirir ve en sonunda bu sayfa içerisinde bulunan kodları kaldırır. Sonuçta karşı tarafa gönderilen bir statik sayfadır. Dolayısıyla istekte bulunan browser sadece HTML kodlarını görebilir. (Web browserlar bilindiği gibi sadece HTML kodlarını yorumlayabilirler.)



Bir uygulama sunucusu, sunucu taraflı (server-side) kaynaklarla çalışır. Örneğin, dinamik bir sayfa uygulama sunucusuna, veritabanındaki bilgileri buradan alıp HTML belgesi içerisine yazma komutu verebilir. Veritabanındaki bilgileri alma emri veritabanı sorgusu (database query) olarak adlandırılır. Bir sorgu SQL (Structured Query Language – Yapısal Sorgu Dili) olarak adlandırılan veritabanı arama kriterlerinden oluşur. SQL sorgusu sayfaya sunucu taraflı scriptler ya da taglar şeklinde yazılır.

Uygulama sunucusu veritabanına doğrudan erişemez. Çünkü farklı üreticilere ait veritabanlarının aynı yolla sorgulanması mümkün değildir. Bunu bir word belgesinin not defteri uygulamasında açılamamasına benzetebiliriz. Uygulama sunucusunun veritabanına erişimi bir aracı yazılım ile mümkün olabilir. Bu aracı yazılım veritabanı sürücüsü (database driver) olarak adlandırılır. Veritabanı sürücüsü veritabanı ve uygulama sunucusu arasında yorumlayıcı vazifesi görür.

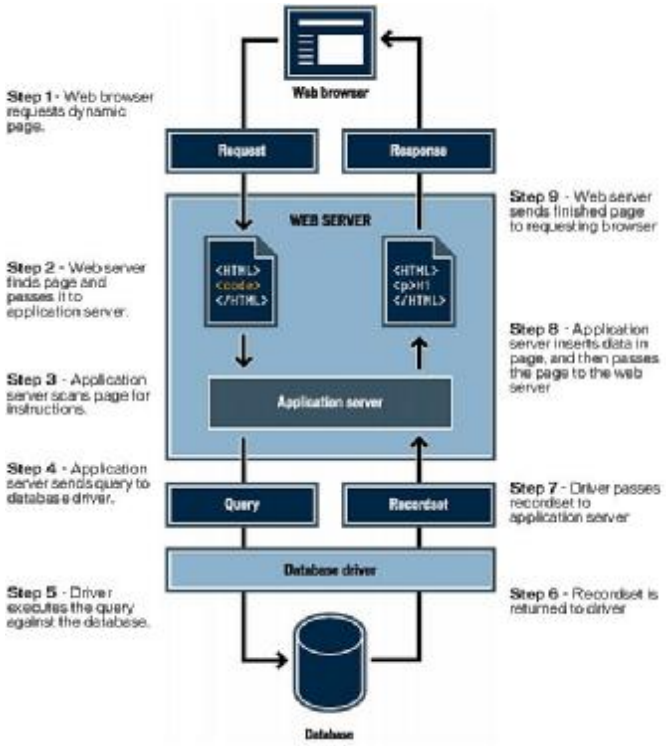
Sürücü bağlantıyı sağladıktan sonra, veritabanı üzerinde çalıştırılan sorgu bir kayıt dizisi (recordset) oluşturur. Kayıt dizisi (recordset), veritabanı içinde bulunan bir ya da daha fazla tablodan çıkarılmış dataların serisidir. Kayıt dizisi uygulama sunucusuna sayfayı oluşturabilmesi için gönderilir.

SQL için basit bir sorgu aşağıdaki gibidir;

SELECT soyad, ad, yas

FROM calisanlar

Yukarıdaki anlatım üç sütundan oluşan bir kayıt dizisi oluşturur. Bu üç sütundaki her satır soyad, ad ve yas alanlarında bulunan kayıtları içerir.



Yukarıdaki şekil veritabanı bağlantısı bulunan bir web uygulamasının çalışma şeklini göstermektedir.

Bir web uygulaması için hemen hemen bilinen tüm veritabanları kullanılabilir. Daha açık bir ifadeyle veritabanı sürücüsü bulunan tüm veritabanı yazılımlarını kullanmak mümkündür. Güçlü web uygulamaları için sağlam bir veritabanı planlaması yapılmalıdır. Bu tür bir plan içerisinde Microsoft SQL Server, Oracle yada MySQL veritabanı seçenekleri mutlaka göz önünde bulundurulmalıdır. Küçük ve orta ölçekli işletmeler için uygun bir veritabanı yazılımı olan FileMaker web uygulamaları içinde veritabanı desteği vermektedir.

Web uygulaması ile veritabanının aynı sunucuda bulunması gibi bir şart olmamakla birlikte, eğer böyle bir yönteme başvurulacaksa iki sunucu arasında hızlı bir bağlantının olmasına dikkat edilmelidir.

WEB UYGULAMALARININ KULLANIMINI ZORUNLU KILAN DURUMLAR

Web sitesinin tasarlanma amacı ve yapısı web uygulamalarını kullanmayı zorunlu kılabilir. Web uygulamalarının kullanılmasını zorunlu kılan durumlar şunlardır;

- Web sitesinin içeriği çok zenginse, kullanıcıların bilgiye kolay ve hızlı ulaşmasını sağlamak için web uygulamalarının kullanılması şarttır. www.msdn.microsoft.com ve www.amazon.com buna örnek verilebilir.
- Site ziyaretçileri tarafından sağlanan verinin toplanması, saklanması ve analiz edilmesi için web uygulaması kullanmak gereklidir. Web kullanıcıları tarafından girilen verilerin doğrudan veritabanına atılması ve yine bu verilerin özel bir web sayfasında görüntülenmesi isteniyorsa sitenin web uygulaması şeklinde tasarlanması gerekir. Örneğin banka, elektronik satış yapan mağaza, anket siteleri ile geri dönüş beklenen formlar için web uygulaması kullanılır.
- Sıksık içeriği değişen siteler için web uygulaması kullanılır. Bir web uygulaması kullanan sitede sayfalar bir tasarımcının desteği olmadan değişir. Bir editör tarafından girilen içerik, web uygulaması tarafından otomatik olarak sitede uygun yere yerleştirilir. Örneğin, www.economist.com, www.hurriyet.com , www.cnn.com

WEB UYGULAMALARINDA GÜVENLİK

Web uygulamalarının en büyük sorunu güvenliktir. Çok kullanışlı olmalarına karşın gerekli güvenlik tedbirleri alınmadığında önüne geçilemez sorunlarla karşılaşılabilir. Şirketlerin ve kurumların web uygulamalarıyla sağladığı platform bağımsız uygulama özgürlüğü, hackerların tehdidi altındadır. Genellikle hosting hizmeti veren firmaların sorumluluğunda olan web uygulaması güvenliği tasarımdan kaynaklanan hatalarla delinebilir.

En çok rastlanan web uygulamaları güvenlik sorunlarını şu şekilde sıralayabiliriz;

- Sunucu Yazılımından Kaynaklanan Güvenlik Açıkları: Hackerlar bir şekilde sunucu yazılımından kaynaklanan açıklardan faydalanarak veritabanına ya da diğer kaynaklara erişebilirler. Amaç her zaman siteyi çalışmaz hale getirmek olmayabilir. Çoğu zaman bu tür girişimler bilgi hırsızlığı amaçlıdır. Bu tür güvenlik açıkları düzenli olarak yayınlanan yamalarla giderilebilir ama kesin çözüm olduğu söylenemez.
- Kimlik Denetimi: Kimlik denetimi gerektiren web uygulamaları her zaman risk altındadır. Daha çok kullanıcıyla ilgili olan bu sorun https protokolü, karmaşık şifre politikası, login için zaman kısıtlama ve erişim kontrolü ile daha az risk taşıyabilir.

- Oturum Güvenliği: Bir web browser, uygulama sunucusunun bulunduğu bilgisayara her erişim yaptığında, sunucu tarafından bu web browser'a bir defaya mahsus benzersiz bir Oturum ID'si atanır. Sayı ve harflerden oluşan bu ID sayesinde sunucu bilgisayar hangi kullanıcıya hangi bilgiyi göndereceğini bilir. Bu ID'yi taklit eden hacker istediği bilgiye ulaşabilir. Bu sorunun önüne geçmek için; sıralı olmayan ID atama, çerez kontrolü, kullanıcı sistemi terk ettiğinde oturum belirticisinin silinmesi, browser geçmişinin kullanılmasının engellenmesi gibi tedbirler alınabilir.
- SQL Enjeksiyon Açığı: Genellikle tasarım tarafını ilgilendiren bir sorundur. SQL sorgularının özel karakterler kullanılarak sanki kullanıcı istekte bulunmuş gibi veritabanına yapılan ataklar şeklindedir. SQL Enjeksiyon kullanılarak veritabanındaki verilere ulaşılabilir. Bu sorunu çözmek için hem tasarımcı ve hem de web sunucusu üreticisinin üzerine düşen görevler bulunmaktadır. Tasarımcı kullanıcı taraflı herhangi bir SQL sorgusunun yapılamayacağından emin olmalı, üretici firma SQL komutu içeren ve kullanıcı tarafından gönderilen ifadelerin web sunucusundan uygulama sunucusuna erişmesini engellemelidir.
- Tampon Taşması (Buffer Overflow): Birçok web uygulaması verileri toplamak için tampon belleği kullanır, bunları daha sonra ilgili yerlere gönderir. Hackerlar bir web uygulamasını çalışmaz hale getirmek için büyük miktardaki verinin tampon belleğe yüklenmesini sağlarlar. Bu sorunun önüne geçmek için, formlardan gelen veri uzunluğu kontrolü, doğruluğu denetlenmemiş kaynaklardan gelen verinin engellenmesi gibi yöntemler kullanılabilir.
- Çapraz Site Betikleme (Cross-Site Scripting): Sayfayı görüntüleyen diğer kullanıcıların Web browser'larını kullanarak yapılan saldırılardır. Hacker'ın sitesini ziyaret eden bilinmeyen bir kullanıcının kendisi fark etmeden bilgisayarına bulaşan zararlı bir yazılım, web uygulamasına saldırıya geçer. Saldırı kullanıcı sisteme giriş yaptığında ya da giriş gerektirmeyen bir uygulamada betik çalıştırma şeklindedir. Bu sorunu çözmenin en mantıklı yolu sayısal verilerin uygulama sunucusuna geçişini filtre etmektir. Web sunucu yamalarında bunun için özel tedbirler bulunmaktadır.