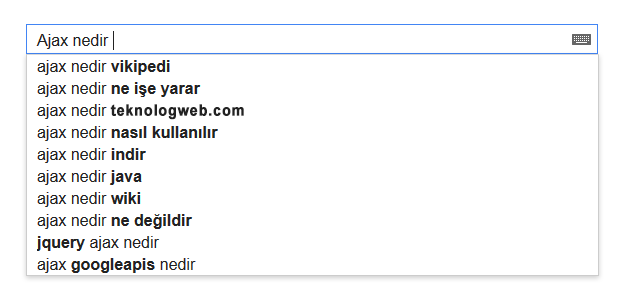
AJAX



İNTERNET PROGRAMCILIĞI 2

2.VİZE

SALİH MANDAL

MEHMET ALİ ER

Ajax nedir ?

1. **Tanım ve Tarihçe**

Asynchronous JavaScript and XML sözcüklerinin baş harflerinden oluşan AJAX, HTML/XHTML, XML, CSS, DOM, JS ve XMLHTTPREQUEST nesnesi ile sunucu taraflı dillerden birisinin (PHP, ASP, JSP vb.) beraber kullanılarak masaüstü yazılımlara benzer etkileşimli web arayüzleri/uygulamaları hazırlamak için kullanılmakta olan bir web geliştirme yaklaşımıdır.

XMLHTTPREQUEST nesnesi her ne kadar çok eskiden beri kullanılmakta olsa da AJAX sözcüğünün kullanılması çok eskilere dayanmıyor. Bu sözcüğü ilk kullanan Adaptive Path adlı bir web firmasının kurucularından ve de yöneticilerinden olan Jesse James Garrett. Her ne kadar bu sözcüğü kendisinin bulmadığına yönelik eleştiriler ve itirazlar olsa da internet ve web uygulamaları dünyasında sözcüğü icat eden olarak çoktan yerini aldı.

XMLHTTPREQUEST nesnesi ilk olarak Microsoft tarafından bir ActiveX nesnesi olarak kullanıldı ve yaklaşık olarak Internet Explorer’ın 4.0 sürümünden beri bir ActiveX nesnesi olarak kullanılmaya devam ediliyor. ActiveX nesnesi olduğu için de doğal olarak Javascript, VBScript ve IE’nin desteklediği diğer diller tarafından ulaşılabilir ve kullanılabilir oldu. Bir süre sonra Mozilla geliştiricileri bu gelecek vaadeden nesneyi alıp Mozilla 1.0’da kullanmaya başladılar. Apple’da bu arada boş durmayıp Mac OS platformu için geliştirmiş olduğu kendi browser’ına bu nesneyi entegre etti ve böylece Safari’ye AJAX desteği vermiş oldu. Internet dünyasının en çok kullanılan browserlarından olan ve hep en hızlı olmayı bilen Opera da 8.0 sürümü ile bu alanda eksik kalmayarak kendi kullanıcılarına AJAX’lı uygulamaları kullanma ve geliştirme imkanı sunmuş oldu.

XMLHTTPREQUEST nesnesinden bu kadar çok bahsetmemizin nedeni AJAX uygulamalarının şüphesiz belkemiğini oluşturmasıdır. Ne işe yaradığını açıklamak gerekirse, bu nesne Javascript ile uzaktaki veya uygulamanın çalıştığı sunucuya HTTP talepleri yapılmasına ve sunucudan da bu taleplerin sonuçlarının veya çıktılarının alınmasına imkan tanıyor. Ancak normal HTTP taleplerinden farklı olarak ve bu nesneyi bu kadar popüler kılan özelliği ise tüm bu talepleri çalıştığı web sayfasının yeniden yüklenmesine ihtiyaç bırakmadan ve kullanıcı farketmeden arka planda yapıyor olmasıdır.

# AJAX Nedir?

AJAX;

* + bir dil değildir.
  + bir program veya bir geliştirme ortamı değildir.
  + bir yaklaşımdır.
  + bir kısaltmadır.
  + bir web teknolojileri ve standartları ortak kullanımıdır.
  + etkileşimli zengin web arayüzleri oluşturulmasına imkan sağlar.
  + web geliştirme tekniği ve konseptidir.
  + Gmail’in veya Google’ın icadı değildir.
  + indirilebilir bir şey değildir.
  + XMLHTTPREQUEST’in diğer adı değildir.
  + XMLHTTPREQUEST ile aynı şey değildir.

# Çalışma Mantığı

AJAX uygulamalarında temel çalışma mantığı ve işlem sırası aşağıdaki gibidir:

Sayfada varolan işlemlerden biri tetiklenince, sayfada hazırda bekleyen javascript fonksiyonlarıyla xmlhttprequest nesnesi oluşturulur. Bu nesne arka planda gönderdiği bir HTTP talebiyle, istenen işlem için sunucu tarafında bulunan ve sunucu tabanlı bir dille yazılmış ilgili dosyadan o işlemin yapılmasını talep eder ve bu dosyanın çıktısını alarak kullanıcı tarafında işlenmesi ve istenen şekilde sunulması için diğer bileşenlere aktarır.

Bu işlem akışı sırasında, bir AJAX uygulamasını tetiklemek için HTML ve JS, XMLHTTPREQUEST nesnesini oluşturmak için JS, arkaplanda sunucuya asenkron (eşzamanlı olmayan/gerçek zamanlı) HTTP talebi göndermek için XMLHTTPREQUEST nesnesi, sunucu tarafında yapılması istenen işlem için PHP, ASP/.NET, JSP/JAVA vb. sunucu tabanlı bir dil, sunucu

tarafında üretilen çıktıyı almak için gene XMLHTTPREQUEST nesnesi, elde edilen bu çıktıyı işlemek ve sayfada istenen alana yerleştirmek için DOM, DHTML ve JS, işlenmiş çıktının renk, çizgi vb. şekilsel sunumu için de (X)HTML ve CSS kullanılmaktadır.



Şekil.1 Geleneksel yapıda web sayfasın isteği.

Geleneksel web uygulamalarında, internet tarayıcıları sunucudan, sayfanın tüm içeriğini getirmesi için istekte bulunur. Kullanıcı gelen sayfa üzerinde bulunan bir bağlantıyı tıklayarak ya da doldurduğu bir formu göndererek yeni bir istekte bulunabilir. Sunucu gelen bu yeni istek için yine kullanıcıya sayfanın tamamını gönderir. Bu durum Şekil 1 de gösterilmiştir



Şekil.2 AJAX Tekniği İle Web Sayfasının İsteği

Ajax tekniğinde ise, Şekil 1’de gösterilen durum farklıdır. Kullanıcının tıkladığı bağlantı veya gönderdiği forma karşılık, sayfanın tamamı yerine mevcut sayfa içerisinde değişmesi gereken içeriğin bulunduğu bölüm, gelen cevap ile güncellenir. Bu durum Şekil 2’de gösterilmiştir. Şekil 2’de gösterilen yapıyı gerçekleştirebilme, başka bir değişle sayfanın ilgili yerlerinin güncellenebilmesini sağlayabilmek için Microsoft’un geliştirdiği, XMLHttpRequest nesnesini kullanan bir arabirime ihtiyaç vardır.

# Destekleyen Browserlar

Bir browserın AJAX uygulamalarını desteklemesi için en temel şart javascript destekliyor olmasıdır. Bundan sonra ikinci şart ise XMLHTTPREQUEST nesnesini destekliyor olmasıdır. Şuanda AJAX uygulamaları belli başlı tüm browserlarda kullanılabilmektedir. Bir liste olarak vermek gerekirse;

* + Mozilla Firefox 1.0 ve üstü,
  + Netscape 7.1 ve üstü,
  + Konqueror,
  + Microsoft Internet Explorer 4.0 ve üstü,
  + Opera 7.6 ve üstü,
  + Apple Safari 1.2 ve üstü,
  + Google Chrome

# Avantajları

* + HTTP talepleri arka planda gidip geldiği için sayfa yeniden yeniden yüklenmemektedir.
  + Sayfalar tekrar tekrar yüklenmediği için ve sadece gerekli veri alışverişi yapıldığından hem kullanıcı/ziyaretçi için hem de site sunucusu için yüksek miktarda trafik/bandwith tasarrufu sağlar.
  + Masaüstü yazılımlarda bulunan zengin etkileşimin web’te bulunmaması geçmiş zamanlardan beri webin en büyük eksiklerinden sayılmış ve her zaman masaüstü yazılımlar karşısında webin dezavantajı olarak görülür. Ancak bu etkileşim artık AJAX uygulamaları sayesinde gerçekleştirilebilir görünmektedir. Talepler kullanıcı farkında olmadan arka planda gönderilip alındığından ve hızlı olduğundan (sadece gerekli veriler gidip gelmekte), kullanıcıya gerçek bir masaüstü yazılımın yaşattığı etkileşimli güzel deneyimleri web üzerinde yaşatmak mümkün olabilmektedir.
  + Aynı şekilde, masaüstü yazılımların dezavantajı olarak görülen heryerden ulaşılamama ve sürekli veri ve ayar taşıma sorunu, web’in doğuşundan beri sahip olduğu her yerden erişilebilir özelliği ve sonradan kazandığı dinamik uygulamaları sayesinde çözülmüş ve artık kullanıcılar masaüstü sorunlarından uzak ancak masaüstü deneyimlere benzer etkileşimli deneyimleri yaşayabilir hale gelmektedirler.
  + AJAX, bir çok web tasarımcısının ve uygulama geliştiricisinin bildiği teknolojilerden oluşturdugundan geliştiricilerin bu yaklaşımı öğrenmesi ve bu konsepte adapte olması çok kolay ve hızlı şekilde gerçekleşmektedir.
  + AJAX’ın yokluğunda, karmaşık web uygulamaları için kullanılan ve her birinin ayrı ayrı bir çok dezavantajı bulunan JAVA ve Flash teknolojilerin yerine daha yüksek ve gerçek zamanlı etkileşim sunmakta
  + Sunum ve çalışma biçimi açısından hiçbir zaman tam bir standart olamayan ve uygulama açısından sürekli sorunlu olan IFRAME ve POPUP pencerelerin en azından bu tür uygulamalarda ortadan kalkmasına ve dolayısıyla azalmasına neden olmuştur. Buna da en çok sevinenler benim gibi IFRAME ve POPUP antipatisi bulunan insanlar olmaktadır.
  + AJAX, CSS, DOM, XHTML, JS benzeri teknolojilerin kullanımını gerektirdiği için bu teknolojilerin ve standartların yayılmasını ve gelişmesine de katkıda bulunmaktadır.
  + Güvenlik nedeniyle güvenilmeyen kullanıcı tabanlı js denetimleri yerine sunucu tabanlı denetimler hızlıca ve basitçe sayfa yenilenmeden ve böylece kullanıcıya zaman kaybı yaşatmadan yapılmaktadır.

# Dezavantajları

* + Veriler ve sayfalar daha çok kullanıcı tarafında işlendiği için, zaten yüksek miktarlarda işlemci ve bellek kullanan browserlar daha yüksek miktarlarda işlemci ve bellek ihtiyacı duymaktadır.
  + AJAX, web sayfalarında kullanımına oldukça alıştığımız ve bir kullanılabilirlik standardı kabul edilen ve olmazsa olmaz şeklinde tabir edilen browser fonksiyonlarından durdur, geri ve ileri (Stop/Back/Forward) butonlarını kullanılamaz hale getirmektedir. Geri ve ileri butonları sayfanın tekrar yüklenmesi yani normal HTTP başlıkları ve talepleri gönderilmesi ile işlevsel hale gelir. Ancak XMLHTTPREQUEST ile gönderilen talepler geri ve ileri butonlarına etki etmediğinden, ileri ve geri butonlarını kullanmaya alışan tüm kullanıcılar bir önceki sayfaya veya bir sonraki sayfaya gitmekte sorun yaşamasına neden olur.
  + IE’de XMLHTTPREQUEST nesnesi bir ActiveX nesnesi olduğundan ve ActiveX bazı firma veya kullanıcı bilgisayarlarında varsayılan olarak güvenlik gerekçeleriyle engellenmiş olduğundan AJAX uygulamaları çalıştırılamayabiliyor. bunun için de çözüm çalışmaları yapılmış olsa da bu konu şuanda hala en çok kullanılan browser olarak görünen IE’nin kullanıcıları için sorun teşkil etmektedir.
  + Kullanılabilirlik ve erişilebilirlik kurallarına ve standartlarına göre, yapılan uygulamanın hazırlandığı teknolojilerin varolmadığı browser türlerinde veya özelliklerinde çalıştırılabilmesi için uyumluluk çalışmaları yapılması gerekmektedir. Bu da fazladan zaman, emek ve nakit harcanması anlamına gelmektedir. Bu da özellikle devlet kurumlarına veya bu standartlara ve kurallara uygun veya yüksek sayıda kullanıcısı olan web siteleri isteyen firmalara hazırlanacak uygulamalar ve sitelerde dikkat edilmesi gereken bir konudur.
  + Buna benzer şekilde, bazı durumlarda düzgün bir statik sayfa gibi çıktı alamama ve çıktı sorunları yaşatabilmektedir. Ayrıca sayfaların kaydedilmesinde de istenen sonuçlar elde edilemeyecek.
  + Klasik web uygulamalarında kullanıcı profili ile ilgili veri toplamak biraz da kullanıcıya bağlıydı. Ancak AJAX ile biz kullanıcılar farkında olmadan hakkımızda ve özellikle web hareketlerimiz ile ilgili oldukça fazla bilgi toplanabilme olanağı doğmuş oldu. Bu web çalışanları için yeni bir avantaj, kullanıcılar için de bir dereceye kadar dezavantaj sayılabilir.

# Hızlı Yayılmasının Nedenleri

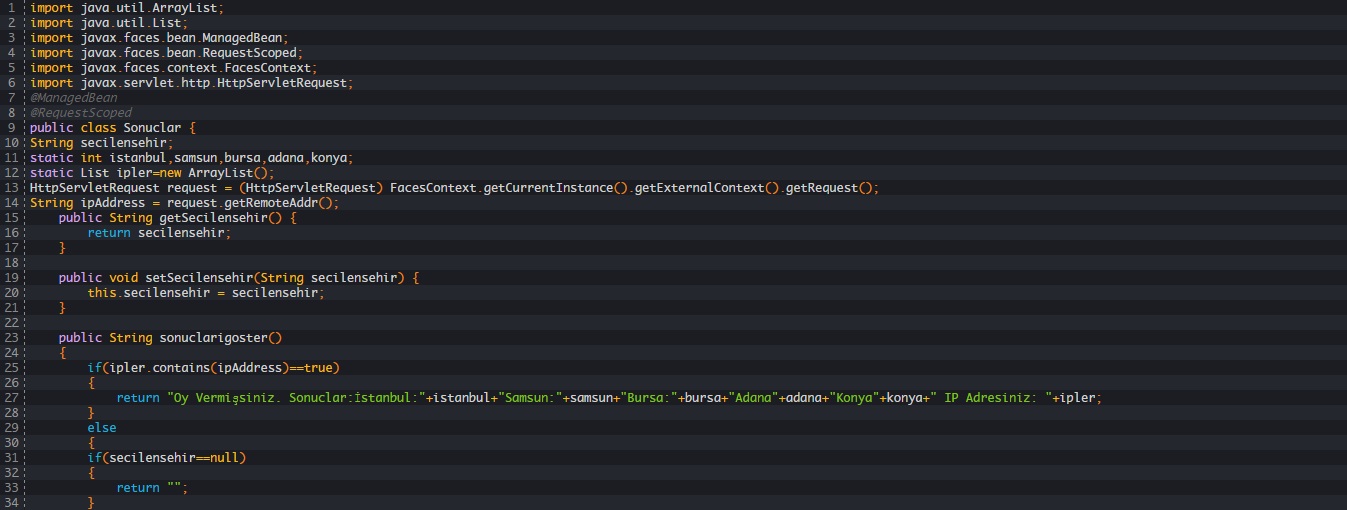
* + AJAX’ın bilinen ve kullanılmakta olan teknolojilerin beraber kullanımından çıktığından ve bu nedenle yeni çıkan teknolojilerden ve web uygulama yaklaşımlarından daha hızlı yayılmıstır açık kaynak kodlu olarak hazırlanan kod kütüphanelerinin artması ve gelişmesi de bu yaklaşımın yayılmasında önemli bir etken olmuştur.
  + Ancak AJAX’ı e popüler hale getiren ve isminin sıkça duyulmasına ve merak uyandırmasına neden olan şey, Google’ın Gmail, Google Suggest ve Google Maps gibi yeni web hizmetlerinde bu teknolojiyi oldukça sağlam ve neredeyse masaüstü deneyimlerle yarışacak kadar etkileşimli ve kullanıcı dostu bir şekilde kullanması olmuştur. Burada kullanılan yeni özellikler hem kullanıcılarda hem web çalışanlarında merak uyandırmış ve benzer özellikleri kendi sitelerinde de kullanmaya yönelerek benzer etkileşimli deneyimleri kendi kullanıcılarına da yaşatma isteği ve çalışmalarıyla AJAX’ın oldukça hızlı yayılmasına katkıda bulunmuştur. Son olarak Yahoo’nun satın aldığı Flickr ve benzeri çok kullanıcılı siteler de AJAX’ın deneyimini kullanıcılarına yaşatarak daha hızlı yayılmasına katkıda bulunmuştur.
  + Bir başka neden de artık hemen hemen tüm belli başlı browserların standartlara uygun şekilde üretilmesi ve AJAX’ta kullanılan teknolojileri ve standartları tam olarak destekliyor olmasıdır. Kullanılan teknolojilerin de olgunlaşmış, oturmuş ve standartlaşmış olması da ek olarak avantaj sağlamıştır.
  + webin en büyük eksiklerinden biri olan masaüstü yazılımlardaki etkileşim ve hız, sayfalar yenilenirken/yeniden yüklenirken beklemekten sıkılmış kullanıcılara AJAX uygulamalarıyla sağlanmış ve kullanıcılar da bu tür uygulamalara rahatlıkla alışmış ve adapte olmuşlardır. Hem masaüstü yazılımlarının hızı, etkileşimi ve deneyimi web’in heryerden erişim gibi özellikleri ile buluşunca AJAX kısa sürede çokça talep edilir ve kullanılır hale gelmiştir.

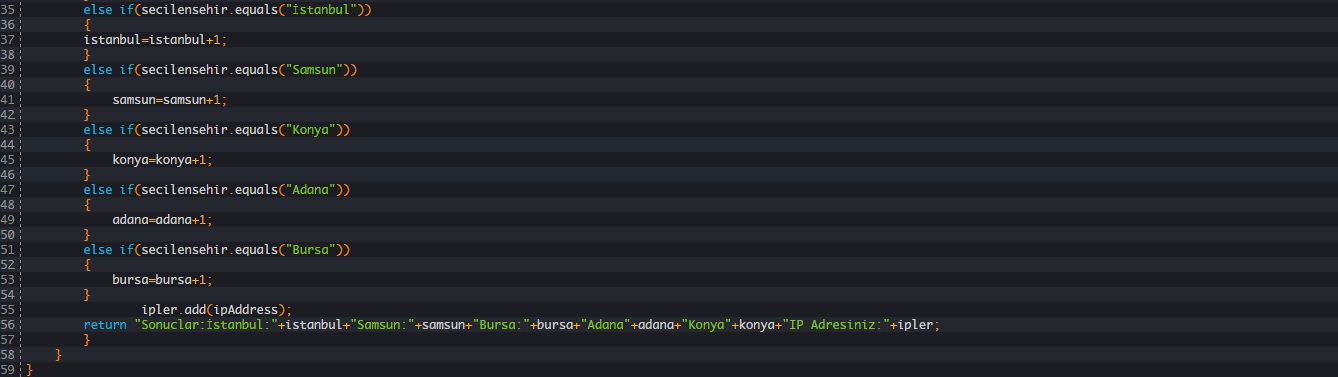
**Ajax ve JSF ile Anket Örneği**

# **index.xhtml Kodu**

# **C:\Users\maer\Desktop\ajx\1.jpg**

**Sonuclar.java Kodu**

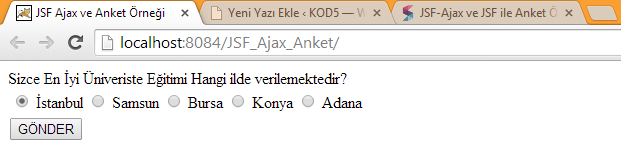


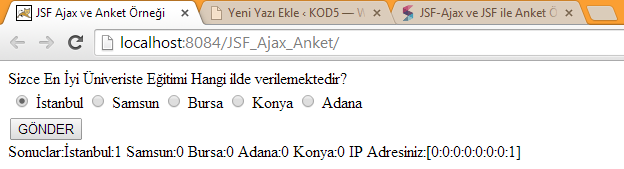


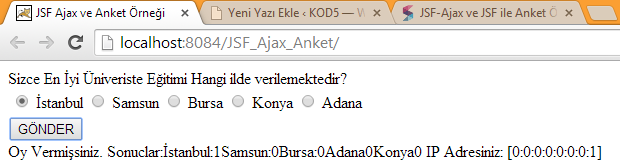
**Java ile sayfayı ziyaret eden kişinin IP’sini şu kodlarla tespit edebiliriz**:

C:\Users\maer\Desktop\ajx\4.jpg

**sonuclarigoster**metodumuz içinde tespit ettiğimiz IP’nin **ipler** listesi içinde olup olmadığını kontrol ediyoruz. Eğer varsa kullanıcıya daha önce oy vermiş olduğunu söyleyip sonuçları ekrana yazdırıyoruz. Yoksa oy verip vermeden butona basıldığını kontrol ediyoruz. Vermeden basmışsa boş bir string döndürüyoruz. Verilmişse oyu işletip IP’sini listeye ekliyoruz ve sonuçları kullanıcının IP adresi ile birlikte ekrana yazdırıyoruz. Tekrar **GÖNDER**butonuna bastığımızda oy vermiş olduğumuzu söyleyip verdiğimiz oyu saymıyor.







**8. Hatalar ve Dikkat Edilmesi Gerekenler**

* + AJAX'ı sadece kullanmış olmak için kullanmak veya kendimizi bir tatmin aracı olarak kullanmak yapılabilecek en büyük yanlışlardandır. Projelerimizde AJAX kullanmayı düşünüyorsak AJAX’ın kullanılacağı yerlere gerçekten uygun olup olmadığını ve kullanıcılara avantajlarını sunacağından emin olmamız gerekiyor. Kullanılan AJAX uygulamaları dezavantaj sunuyorsa ve kullanıcılara kötü tecrübe yaşatıyorsa AJAX kullanmak bize sadece zarar getirecektir.
  + Kullanıcıların gelenekselleşmiş alışkanlıklarını gözardı etmek yapılacak büyük hatalardan bir başkası olacaktır. Kullanıcıların çok büyük çoğunluğunun çok yoğun ve sık kullandığı İLERİ, GERİ, DURDUR butonları AJAX uygulamalarında arka planda giden ve gelen HTTP talepleri nedeniyle işlevselligini yitirmekte ve örnek olarak geri butonuna basıldığında sayfanın bir önceki adımına değil ilk aşamasına ulaşılmaktadır. Şuanda internet ortamında bu sorunlara çözüm sunan javascript kütüphaneleri ortaya çıkmıştır. Bu sorunların çözülmeden uygulamanın kullanıma sunulması bilin ki ziyaretçilerinizde antipati oluşturacaktır. Bunların yanısıra, print, send to friend, ve bookmark gibi fonksiyonları ve diğer yardımcı uygulamalar için de çözüm üretmeyi unutmamak gerekiyor.
  + Sayfa içerikleri AJAX ile çekiliyor ve dinamik olarak sunuluyorsa mutlaka permalink denilen sabit bağlantılarla kullanıcıların bu sayfaları bookmarklarına ekleyebilmeleri ve arama motorlarının da botlarının o sayfalara ulaşabilmesi sağlanmalıdır. Bu gereklilik, dışarıya kapalı, e-ticaret siteleri/alışveriş aşamaları için, anlık ve sürekli değişen türde içerik sağlayan sayfalarla beraber arama botlarınca taranması istenmeyen bölümler için geçerli olmayabilir.
  + Geribildirimlerde bulunmamak: AJAX ile yaptırdığınız işlemler konusunda kullanıcıya bir bilgi veya işaret vermemek bir başka uygulama yanlışıdır. Örnek olarak il seçilince ilçe kutusunun içeriğini AJAX yöntemiyle getiriyorsak, il seçildiğinde mutlaka “Lütfen bekleyiniz! İlçeler çekiliyor” vb. bir mesaj vermemiz gerekmektedir.
  + Eğer uygulamalarınız için yüksek bant genişliği, sağlam ve sorunsuz çalışan bir hattınız veya hosting’iniz yoksa, AJAX kullanmanız kullanıcılarınıza işkence anlamına gelebilir. Normalde bir kaç sayfayı ayrı tab’larda veya pencerelerde açarak birisine bakarken diğerlerinin yüklenmesini bekleyebiliyoruz. Eğer uygulamamız AJAX’ın temel amaçlarından biri olan bu hız talebini ve beklentisini karşılayamıyor ise uygulamamızı AJAX ile yazmamız doğru bir yol olmayacaktır. Çünkü bir kullanıcıyı web sayfası karşısında bekletmek ona yapılacak en büyük işkencelerden biridir.
  + Güvenliği gerektiren uygulamalarda güvenlik kurallarını ihlal etmek! Uygulamaya bağlı olarak SSL kullanmak, önemli verilerin şifrelenmesi, girdi denetimlerinin yapılması gibi önemli güvenlik kurallarını hiçbir AJAX uygulamamızda ihmal etmememiz gerekmektedir. HTTP taleplerinin arkaplanda gidip geliyor olması riskleri bertaraf etmemektedir.
  + AJAX uygulamaları için yeterli test ve denetimlerin yapılmaması! Uygulamalarımızın her browserda ve her platformda aynı şekilde çalıştığının ve aynı sonuçları verdiğininin test edilmesi ve onaylanması gerekmektedir. Bilindiği gibi JS kodları malesef her browserda veya her platformda aynı çalışmayabilmektedir. Bu nedenle bu tür uygulamaların her ortamda aynı davranışlara ve sonuçlara sahip olduğunun yeterli biçimde test edilmesi gerekmektedir. Bu, uygulamanın sorunsuz kılınmasını sağlayıp her tür kullanıcının uygulamanıza erişebilmesine ve uygulamayı sorunsuzca kullanabilmesine imkan tanıyacaktır.
  + AJAX’ın JS tabanlı bir yaklaşım olmasından dolayı istemci tarafındaki JS kodunun oldukça fazla ve ağır olması, tüm işlemlerin kullanıcı tarafında yapılıyor olması veya çok büyük JS/DOM kütüphanelerin kullanılması bir başka planlama ve tasarım yanlışı olacaktır. Çözüm olarak, bir kütüphanede kullanmayacağınız özelliklerin veya kodların ve yorumların çıkarılarak dosyanın boyutlarının küçültülmesi ile bu dosyaların download süresini kısaltabilirsiniz. Bunun yanısıra, sunucu tarafından alacağınız veri cinsine göre çıktıyı XML veya metin formatında sunucu tarafında işlenmiş olarak alabilirsiniz. Verinin sunucu tarafında

işlenmesi ve istemci tarafına hazır gelmesi hem kullanıcı tarafındaki kodu hafifleterek gerekli kaynak tüketimini azaltacak ve hem de güçlü sunucularda çok daha hızlı işleneceği için ek hız avantajı sağlayacaktır. Ek olarak, kodların en kısa ve en doğru şekilde yazılmış olduğundan emin olun. Zira kötü yazılmış basit bir loop, browser'ın tıkanmasına veya kilitlenmesine neden bile olabilmektedir. JS kodu ne kadar kötü veya uzunsa ziyaretçinin bilgisayarında browser bir o kadar fazla kaynak tüketeceğini ve böylece uygulamanızın kötü bir deneyim yaşatacağını unutmayın. Unutmayın herkesin hızlı ve güçlü bir bilgisayarı olmayabilir.

* + JS desteği olmayan veya JS kullanmak istemeyen kullanıcıları gözardı ederek erişilebilirlik kurallarını ve standartları hiçe saymak! Halen bir çok kurumsal web sitesinde Text Version/Metin Versiyonu şeklinde sadece metin tabanlı site sürümleri bulunmaktadır. Aynı şekilde AJAX uygulamanızın da normal ve klasik yollarla çalışan sürümlerini barındırmak ve hazırda tutmak hem standartlara uymuş olmayı hem de %11’e yaklaşan bu kullanıcı kitlesini yoksaymamış olmanızı sağlayacaktır. Örnek olarak, sonradan eklemiş olsa da, Google Gmail’e klasik HTML sürümünü ekleyerek Gmail’i klasik yollarla kullanmak isteyen kullanıcıları kaybetmemeyi garanti altına almış oldu. Klasik sürüm hazırlanmazsa bile mutlaka bu kullanıcılara uygulamanızın neden JS gerektirdiğini ve uygulamanızın güvenilir olduğunu uygun ve anlaşılır bir dille anlatmanız gerekmektedir.
  + Kullanıcıların sahip olduğu alışkanlıklarının dışına çıkmak veya kullanıcıların yeni bir şeyler öğrenmesini zorunlu kılmak! Günümüzde web kullanıcıları hergün yeni birşeyler öğrenmek istemiyor. Herkes genellikle alıştığı ve kendisine kolay gelen ve kendisini anlayan web sayfalarını ve uygulamalarını kullanıyor ve seviyor. Bu nedenle bu klasikleşmiş alışkanlıkları takip edin. Uygulamalarınızda AJAX’ı, bu alışkanlıkları tamamen yoketmeden ufak çaplı ve keskin olmamak şartıyla kullanır.
  + Kullanıcının odak noktasını kırmak! Uygulamalarınızın kullanıcının odaklandığı alan dışında başka alanları değiştirmemesi gerekmektedir. Kullanıcının odaklandığı noktanın kırılması veya birden fazla yerdeki değişikliği görememesi ve algılayamaması uygulamanızla kullanıcı arasındaki etkileşimi bozabilir. Bu değişikliklerin uygun efektlerle ve insanın takip edebileceği ve farkedebileceği bir biçimde yapılması hem uygulamanın güzelleşmesini hem de etkileşimin bozulmamasını sağlayacaktır.
  + Aynı anda birden fazla işlem için HTTP talebi göndermek ve işlemek! Çok mecbur kalmadıkça belli aralıklarla işlemleri topluca göndermek ve topluca işlemek daha mantıklı bir yol olacaktır. Arka arkaya gönderilen HTTP talepleri sayfada sıkça değişikliklere neden olacağı için bu durum kullanıcının o anda yaptığı veya yeni geçtiği işlemin bölünmesi anlamına gelecektir.

# 9. Ajax da Hazır Kütüphaneler

* + **PHP**
    - HTML\_Ajax
    - XOAD
    - xajax
    - sajax
    - pajax
    - Class Ajax
    - Pajaj
    - symfony
    - CakePHP
  + **JAVA**
    - AjaxAnywhere
    - AjaxTags
    - DWR
    - Echo2
    - Guise
    - jquery
    - jWic
    - JSON-RPC
    - JRP
    - ICEfaces
    - Pushlets
  + **ASP/.NET**
    - Ajax.NET Professional
    - Aspects of AJAX
    - Atlas (Microsoft ASP.NET 2.0 AJAX platform)
    - Bitkraft
    - BorgWorX Ajax.NET
    - AutoSuggestBox
    - PowerWEB LiveControls for ASP.NET
    - MonoRail
    - ComfortASP.NET
    - OutPost
    - DotNetRemoting

# 10. Ajax da Güvenlik

**Yaygin Ajax Güvenlik Açıklari ve Çözümleri**

Ajax, ya da tam adıyla asenkronize JavaScrıpt+XML, web tasarımcılarının ve geliştiricilerin websiteleri için uygulama geliştirmek için kullandıkları bir web teknolojileri koleksiyonudur. Birçok insan Ajax’ı bir teknoloji platformları koleksiyonu olarak görmek yerine kendi başına bir teknoloji olarak görme hatasına düşer. Ajax\ uygulaması kolay ve bir çok ozellige sahiptir. Onu web dünyasının en popüler araçlarından biri haline getiren de budur. Ajax uygulamalarını kullanmak sitenizdeki kullanıcı etkileşimini ve website hızını büyük bir oranda iyileştirir, bu da arama motoru optımızasyonu (SEO) konusunda Ajax’ı bir adım öne çıkarır. Yine de, diğer tüm uygulamalar gibi, Ajax uygulamaları da saldırılara karşı korumasızlardır. Ajax kullanmanız sitenizi siber suçlara ve saldırılara karşı daha güvenli hale getirmez. Ajax uygulamalarınızı saldırılara karşı korumada en kilit nokta var olan açıkların farkında olmak ve bunlara karşı önlemler almaktır.

Peki, en yaygın saldırılar hangileridir ve bunlara karşı ne gibi önlemler alınabilir? Yazının devamında bunları sizler için sıraladık.

**Tarayıcı-tabanlı saldırılar**

Bu saldırı siber suçlu ya da hacker websitesinin JavaScript kodlarına erişmeyi başarıp içerde bir takım kodlar çalıştırdığında meydana gelir. tarayıcı-tabanlı saldırılar bir çok şekilde yapılabilir, sisteme herkesin erişebileceği bir virüs koyma ya da ziyaretçileri (genellikle anasayfa) başka bir sayfaya yönlendirme gibi. En kötü tarayıcı-tabanlı Ajax saldırılarıysa yerleştirdiği zararlı yazılımı erişime kapayanlar, yani bulaştırmanın yanında çözümüne de engel olanlar. Buna karşı en basit koruma yöntemi JavaScript kullanmayi birakmak ancak bunu yaptiginizda binlerce JS uygulamasindan da mahrum kalacaksiniz. Bunun yerine, kullandığınız güvenlik duvarının ve tüm yazılımların güncellenmiş olduğundan emin olun, yeni çıkan tüm düzeltmeleri ve güvenlik güncellemelerini zaman kaybetmeden yükleyin.

**Cross-site scripting (XSS)**

XSS, bir siteye zararlı kod enjekte etmenin bir yoludur. Bu kod daha sonrasında siteyi ziyaret eden tarayıcıya aktarılır ve tarayıcının bundan haberi olmaz. Siber suçlular ve hackerlar XSS’i kimlik bilgileri ve veri hırsızlığı ve bir firmanın ticari kayıtlarını ya da kullanıcıların internette ziyaret ettiği sayfaları takip etme gibi dolandırıcılık etkinliklerinde kullanabilirler. Bu yüzden haliyle tüm web sitesi sahipleinin uzak durmaya çalıştığı bir saldırı türüdür zira ünlerinin zedelenmesine ve maddi kayıplara neden olabilir. Örneğin, Twitter 2010 yılında XSS saldırısıyla vurulmuştu. Ajax geliştiricileri aşağıdaki adımları uygularsa XSS riskini azaltabilirler:

-Tersbölü (backslash) kodlamayı kullanmama

-JavaScript hex ve unicode kullanma

-JSON’u ayrıştırmak için JSONçparse ya da json2çjs kütüphanelerini kullanma

**Ajax köprüleme**

Ajax köprüleme aslında başlı başına bir sorun değil ancak önlem alınmazsa sonucu çok ağır açıklara gebe. Sorun, sitesinde üçüncü parti uygulamalarını barındıran, yani “Ajax köprüsü” oluşturan sitelerde. XSS de dahil olmak üzere saldırılar bu uygulamalar aracılığıyla meydana geliyor. Diğer bir deyişle, eğer uygulamasına ev sahipliği yaptığınız site saldırıya uğramışsa sizin siteniz de saldırıya uğrayabilir! Hacklerlar ve siber suçlular bir siteyi exploit etmeyi başaramazlarsa genelde ikinci yöntem olarak Ajax’lı ya da başka uygulamalarla bu yönteme başvuruyorlar. Bariz çözüm Ajax köprülemeyi kullanmama gibi gözükse de siteniz için bu gerekli olabilir. Bu durumda, sitenize erişimi olan diğer siteleri denetimden geçirdiğinizden emin olun ve bununla ilgili diğer gerekli adımlar atın. Siteyi taramak için bir tarama yazılımı kullanın ve bu sitenin güvenilir olduğundan emin olun.

### Ajax Güvenlik Açıkları ile başa çıkmak

Hatırlamanız gereken en önemli şey Ajax açıkları Ajax’a özel açıklar değil, bu yüzden baş etmesi görece daha kolaylar. SEO tarzı şeylerin önemini de göz önüne aldığımızda diyebiliriz ki Ajax açıklarıyla baş etmek tek seferlik bir şey değil süreklilik gerektiren bir şey. Ajandanızda bu konuyu sürekli güncel tutmanız siteniz büyümeye başladığında baş etmenizi çok daha kolaylaştıracaktır.