

# **İstanbul Üniversitesi**

## **Açık ve Uzaktan Eğitim Fakültesi**



**Web Tasarımı**

**Dr. Öğr. Üyesi Fatma Önay Koçoğlu**

**DR. ÖĞR. ÜYESİ FATMA ÖNAY KOÇOĞLU**

# İÇİNDEKİLER

- 1. WEB VE İNTERNET TEKNOLOJİLERİNE GİRİŞ**
- 2. WEB TASARIMIN TEMEL İLKELERİ**
- 3. WEB TASARIMA GİRİŞ**
- 4. HTML METİN VE GÖRSEL DÜZENLEME ETİKETLERİ**

# 1. WEB VE İNTERNET TEKNOLOJİLERİNE GİRİŞ

## Birlikte Düşünelim

Sizce bilgisayarımızda bir internet sayfasına erişirken, adresi girdikten sonra kaç farklı adımdan sonra o sayfadaki bilgilere erişebiliyoruzdur?

Web üzerinde herhangi bir iletişim protokolünden yararlanılıyor mudur? İletişimin bir standardı var mıdır?

İnternet başta hangi amaçla geliştirilmiş olabilir?

## Başlamadan Önce

Web tasarım aşamasına geçmeden önce web tasarım ile ilgili teknolojilere, tasarımını yapacağımız bir web sayfasının nasıl yayınlanacağına ve bir kullanıcının sayfamıza erişim sürecine aşina olmamız gerekir. Bu bölüm kapsamında temel bilgileri öğrenmiş olacaksınız.

## Giriş

Web, internet tabanlı önemli bir iletişim ortamı bir bilgi sistemidir. Web sayesinde günümüzün en önemli değerlerinden bilgiye hızlıca erişim sağlayabiliyor, bankacılık, eğitim, vatandaşlık ile ilgili birçok işlemi hızlı ve zahmetsizce gerçekleştirebiliyoruz. Keza bireysel kullanıcıların yapacağı işlemlerin dışında, bu kullanıcılardan çeşitli verinin toplanması noktasında da web üzerinden yayınlanan formlar kullanılmaktadır. Tüm bu karşılıklı veri-bilgi akışı düşünüldüğünde e-ticaret ve e-iş açısından da web ortamının önemi kuşkusuz ortadadır.

Tüm bu işlemleri gerçekleştirebilmek için gerekli olan web sitelerini tasarlayabilmek ve işe koymak için çeşitli seçenekler mevcuttur. Örneğin, birçok şirket web sayfaları oluşturmak için araçlar ve sayfaları yayınlamak için de alanlar sunmaktadır. Bu tarz hazır tasarımlar ile web siteleri oluşturmak genellikle basit kişisel web siteleri üretmek için kullanılmaktadır. Daha özgün tasarımlar ortaya koyabilmek, web sitelerine dinamik yapılar entegre edebilmek, web sitesini daha kapsamlı hale getirebilmek ve daha profesyonel işlere yönelik web sitesi geliştirmek için daha iyi yapılandırılmış ve tasarlanmış web sitelerine ihtiyaç duyulacağından bu hazır araçlar yeterli gelememektedir. Bu noktada herhangi hazır bir araca bağımlı olmadan web sayfaları tasarlayıp gerçekleştirebilen web tasarımcılara ihtiyaç duyulmaktadır. Bu bölüm kapsamında web tasarım ilkeleri ve süreçlerine geçmeden önce web ile ilgili temel kavram ve teknolojilerine yönelik bilgi vermek amaçlanmıştır.

## 1.1. İnternet

Yukarıda bahsi geçtiği üzere kullanıcılar arasındaki iletişim ve her türlü veri-bilgi alışverişi kullanıcıların bilgisayarları ile bağlandığı büyük bir ağ üzerinde gerçekleşmektedir. Bu ağ **internet** olarak tanımlanmaktadır. İnternet kelimesi “**interconnected network**” kelimelerinin kısaltılmış halidir. Buradan da anlaşılacağı üzere internet, elektronik, kablosuz ve optik ağ teknolojileri kullanılarak, çeşitli protokoller ile iletişim kurmak amacıyla birbirine bağlı bilgisayarlar ağının tamamı, bilgisayar ağlarının ağıdır. Kamu ve özel bilgisayar ağları akademik ve ticari ağlar ile birbirine bağlanmaktadır. Bu noktada belirtmek gerekirse bilgisayar ağları içerisinde bağlanan cihazlar sadece bilgisayarlar olarak düşünülmemelidir. Bu cihazlar; televizyon, cep telefonu, elektronik ev aletleri, otomasyon sistemleri kısacası internet bağlantısı kurulabilen yani ağa dahil olabilen tüm cihazlar şeklinde ifade edilebilir. Dahası mobil teknolojilerinin gelişimi internetin daha yaygın kullanımını sağlamıştır.

ABD Savunma Bakanlığı'na bağlı İleri Düzey Araştırma Projeleri Kurumu (*Advanced Research Project Agency-ARPA*) askeri merkezlerin bir kısmını herhangi bir saldırı anında birbiriyle iletişim kurabileceği güvenli bir ağ kurmayı amaçlamıştır. Bilgisayar bilimci ve psikolog Joseph Carl Robnett Licklider, İntergalaktik Bilgisayar Ağı (*Intergalactic Computer Network-IGCN*) kavramını öne sürmüştü, bu tezi ile birlikte küresel çapta bilgisayar kullanıcılarının birbiriyle bağlantı kurarak veri alışverişi yapabileceği bir ağ yapısı öne sürmüştür. Yine aynı kurumda, bilgisayar ağları kavramı üzerine çalışan Larry Roberts önderliğindeki çalışmalar sonucunda geliştirilen tanımlar ARPANET başlığı ile yayınlamıştır. ARPANET'in spesifikasyonları ve mimari tasarımı netleştikten sonra, Eylül 1969'da UCLA'da ilk bilgisayar bu ağa bağlanmıştır. ARPANET'in ortaya çıkmasında görev alan ilk merkezler University of California at Los Angeles (*UCLA*) ve Stanford Research Institute (*SRI*) olmuştur (Keefer and Baiget, 2001). Bu merkezlerdeki bilgisayarların ARPANET'e bağlanması ile birlikte gerçek anlamda bir bilgisayar ağı ve bu ağ üzerinden veri akışı elde edilmiş oldu. Ardından, öncelikle University of California Santa Barbara ve University of Utah ve sonrasında ABD'deki diğer üniversite ve araştırma kuruluşlarında yer alan çeşitli bilgisayarların dahil olmasıyla birlikte ağ büyümeğe devam etmiş, teknolojik gelişmeler açısından ise farklı iletişim kanallarını kullanan yeni ağlar ortaya çıkmıştır. İki ana makine arasında temel iletişime izin veren nispeten basit bir prosedüre sahip TELNET, ilk kamu paket veri hizmeti ve ARPANET'in ticari versiyonu olmuştur (Cohen-Almagor, 2013). 1970'li yıllarda ağ uluslararası zemine taşınmış, tanımlanan çeşitli protokoller ile birlikte eposta gönderme (*Simple Mail Transfer Protocol-SMTP*), dosya transferi (*File Transfer Protocol-FTP*) gibi çeşitli uygulamalar kullanılabilir hale gelmiştir. Ağdaki trafiğin artması sebebiyle yeni protokoller geliştirilmeye devam edilmiştir. Ancak farklı ağlarda kullanılan protokoller ve standartlar sebebiyle standardizasyona gidilmiştir. Bu doğrultuda, özellikle açık mimarili ağları destekleyecek şekilde İletim Kontrol Protokolü (*Transmission Control Protocol-TCP*) ve İnternet Protokolü (*Internet Protocol-IP*) devreye girmiş, internet için standart haberleşme protokolü oluşturulmuştur. 1980'li yıllarda Paul Mockapetris tarafından Alan Adı Sistemi (*Domain Name System- DNS*) ortaya konulmuş ve ağda yer alan sistemlerin iletişim için kullandıkları internet protokol adresi (*Internet Protocol Address – IP Adress*) okunabilir internet adresleriyle eşleştirilmiş, günümüzdeki haliyle kullanılır olmuştur. Avrupa Nükleer Araştırma Merkezi (CERN)'nde görev yapan Tim Berners-Lee internet üzerinden veri paylaşımını kolaylaştırmak adına çalışmalar yapmış, Hipermetin Transfer Protokolü (*Hypertext Transfer Protocol – HTTP*), Hipermetin İşaretleme Dili (*Hypertext Markup Language – HTML*) ve Benzersiz Kaynak Tanımlayıcı (*Uniform Resource Identifier – URI*) üç önemli web teknolojilerini geliştirmiştir. 1993 yılında CERN'deki çalışmaların sonucunda, internet (web protokolleri) www ön eki ile herkesin ulaşımına açılmıştır. Bu sistemdeki amaç Webte tutulacak dokümanların daha sade bir şekilde etiketlenmesini sağlamak, küresel çapta erişilebilirlik ve standardizasyon ile birlikte farklı sistemler arasındaki uyumluluğun sağlanmasıdır. Yine aynı yıl Tim Berners-Lee öncülüğünde W3C (*World Wide Web Consortium*) kurulmuştur. Bu konsorsiyumun amacı web'in uzun vadede gelişmesini ve büyümesini sağlamak üzere açık web standartları geliştirmektir. İnternetin tarihçesi ile ilgili olarak kronolojik gelişmelerin ayrıntılarına URL1 (2022)'den erişilebilir. Türkiye'de ise internet ilk kez 12 Nisan 1993 yılında 64 Kbps kapasiteli kiralık hat ile, ODTÜ Bilgi İşlem Daire Başkanlığı sistem salonundaki yönlendiriciler kullanılarak, ABD'de NSFNet (*National Science Foundation Network*)'e TCP/IP protokolü üzerinden gerçekleştirilmiştir (ODTÜ, 2022).

Bu noktada belirtilmelidir ki İnternet dünyanın farklı yerlerinden bilgisayar sistemlerinin iletişim kurabildiği bir ağ olmakla birlikte bazı ağlar bu özellikte değildir. Örneğin bir kurum içerisindeki bilgisayarların kendi aralarında veri-alışverişinde bulunabildikleri ağlar oluşturulabilir. Ancak bu ağlar yerelde oluşturulmuş ve kurum dışından ulaşım sağlanamayan ağlardır. Bu ağlara intranet adı verilmektedir. Fiziksel olarak yakın bulunan bilgisayarların kullandıkları aynı ağlar Lokal (Yerel) Ağ (*Local Area Network-LAN*), farklı yerel ağların birbirine bağlanması ile oluşan ağlar ise Geniş Alan Ağı (*Wide Area Network-WAN*) olarak tanımlanır.

## 1.2. Temel Kavramlar

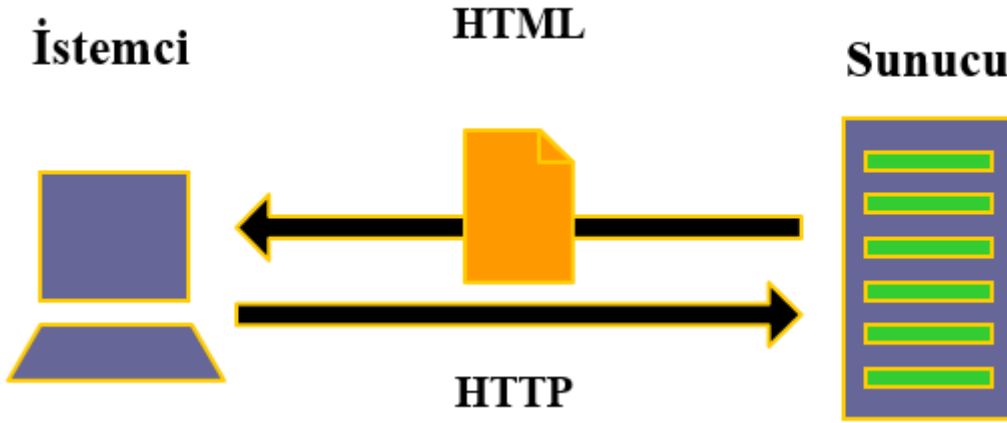
Görüldüğü üzere bilgisayar sistemlerinin birbirine bağlı olduğu ağ üzerinden kullanıcılar gereksinim duyduğu veri ve bilgiyi alırken, kendisi ile ilgili veri ve bilgileri de yine bu ağ üzerinden iletebilmektedir. Bu iletişim çeşitli protokoller ve iletişim standartları ile gerçekleştirilmektedir. Diğer taraftan çeşitli web teknolojilerinin isimleri de internet tarihi içerisinde önemli gelişmeler olarak geçmektedir. Bu kavramlara ilişkin bilgiler bu başlık altında toplanmıştır.

**Protokoller:** Bu bölümün başından itibaren bu iletişimin belirli protokollerle gerçekleştiği ve internetin gelişimi süresince bu protokollerde de bazı değişiklikler olduğu vurgulanmıştır. Farklı protokollerin

kullanılması ihtiyaca bağlı olarak ortaya çıkmıştır. Ağdaki iletişimde gönderilen ve alınan veridir ve verinin farklı tipleri söz konusudur. Bu nedenle web sayfalarını oluştururken kullandığımız dosyalar belirli kodlamalara göre hazırlandığından bunlara ilişkin protokoller ayrı, bir eposta verisini farklı yapıda olduğundan buna ilişkin protokoller ayrı, ya da farklı tipte veri içeren dosya paylaşımı için kullanılması gereken protokoller ayrıdır. Bu farklı protokol yapılarından bazıları şu şekilde açıklanabilir. Öncelikle İletim Kontrol Protokolü (*Transmission Control Protocol-TCP*) farklı ağların birbiri ile bağlantısında iletişimin belirli bir standarda göre oluşturulması için öne sürülmüştür. TCP'den ayrılan bir parça ise IP olarak tanımlanmış, böylece TCP/IP olarak ifade edilmeye başlanmıştır. İletişim sırasında herhangi bir taraftan gönderilecek veri paketlenerek iletilmektedir. Bu doğrultuda TCP/IP temel olarak verinin iletim sırasında bu ağda nasıl paketleneceğini ve içerisinde hata olup olmadığının ne şekilde denetleneceğini belirleyen kurallardır. Cihazlar istemci ya da sunucu olarak ağda iletişim için IP (Internet Protocol) kullanır. Bu protokol ağdaki iletişimi sağlayan temel protokollerdendir. Ağda yer alan cihazlar ise kendilerini tanımlayan IP adresi adı verilen belirli bir sistematığa göre üretilmiş birer numara kullanırlar. Bu adres, 0 ile 255 arasındaki dört sayıdan meydana gelen ve bu sayılar arasında noktalar bulunan kimlik numaralarıdır, yani benzersizdir. Örneğin, 194.27.128.98 bir IP numarasıdır. IP adresleri, ait oldukları bilgisayar sistemlerinin bulunduğu ülkeden şehrine kadar detaylı olarak adres bilgisi ve internet servis sağlayıcısına kadar önemli bilgileri sağlamaktadır. Bir başka protokol Hiper Metin Transfer Protokolü (*Hyper Text Transfer Protokol-HTTP*)'dir. HTTP internette web sayfalarının düzgün görüntülenebilmesi için kullanılır. Şekil 1'de iki bilgisayar sistemi arasındaki ağ iletişiminin http üzerinden yürütüldüğü görülmektedir. İnternette bir bilgiyi araştırmak istediğinizde bunu tarayıcılar (browser) vasıtasıyla gerçekleştirmekteyiz. Tarayıcılar; internet kullanıcılarının, web sitelerine ilişkin dosyaların tutulduğu ortamlara bağlanmak, hazırlanan sayfa içeriklerine erişerek içerikleri görüntülemek ve gerekirse etkileşim kurmak üzere kullandığı yazılımlardır. Google Chrome, Mozilla, Yandex, Opera, Internet Explorer günümüz bilinen tarayıcıları arasında yer almaktadır. Tarayıcılar, kullanıcının isteği üzerine öncelikle HTTP'yi getirmektedirler. Yani herhangi bir web sitesinin adresini doğrudan tarayıcı adres bölmesine yazsanız bile erişim isteğinizde bulunduğunuz an girmiş olduğunuz adresin başına http:// ifadesinin eklendiğini göreceksiniz. Çünkü sunucular gelen erişim isteklerini HTTP çerçevesinde işlemektedir. Buna göre HTTP ile birlikte gelen erişim isteğinin güvenli bir şekilde iletilmesini sağlar. İletişim esnasında çeşitli durum kodları üretilerek veri alışverişi hakkında genel bir sonuca varılabilmektedir. Bu kodlardan bir kısmı 1\*\*-bilgi, 2\*\*-başarı, 3\*\*-yönlendirme, 4\*\*-tarayıcı hatası, 5\*\*-sunucu hatası şeklinde sıralanabilir. Dosya aktarımı için kullanılan protokol Dosya Transfer Protokolü (File Transfer Protocol-FTP)'dir. Protokol ile sağlanan bir dizi komutlar yardımıyla iki bilgisayar sistemi arasındaki dosya alışverişi gerçekleştirilir. FTP vasıtasıyla dosya aktarımı ortamına erişim için bir kullanıcı adı ve parola gereksinimi söz konusuysa, bu bilgiler giriş yapıp bağlantı yapıldıktan sonra dosya transferleri gerçekleştirilebilir. Eposta gönderimi için Basit Eposta Transfer Protokolü (Simple Mail Transfer Protocol-SMTP), alımı için ise Postane Protokolü 3 (Post Office Protocol-POP3) ve İnternet Mesaj Erişim Protokolü (Internet Message Access Protocol-IMAP) kullanılmaktadır.

**WWW (Web, W3):** Veriler ses, görüntü (kayan görüntü), metin gibi farklı yapılarda olabilir. Örneğin bir kamera vasıtasıyla toplanan görüntüler de veri, excel dosyasında yer alan satırlar da veri ya da bir müzik dosyasında yer alan kayıtlar da veridir. En basit şekilde www, bu farklı veri tiplerine internet ağı üzerinden etkileşimli olarak ulaşılmasını sağlayan web sayfaları topluluğu olarak tanımlanabilir. Örneğin bu ders notu metin yapısında bir veri topluluğudur ve bu topluluğa bir web sayfası vasıtasıyla erişebilmiş durumdasınız.

**Sunucu (Server) ve İstemci (Client):** Ağdaki paylaşımda bilgisayar sistemlerinden birisi istemci diğeri ise sunucu konumunda yer almaktadır. İstemci bilgisayar sistemleri erişilmek istenen veriye ilişkin talepte bulunurken, sunucu bilgisayarlar ise bu talebe karşılık veren taraftır. Sunucular kesintisiz olarak ağa bağlıdır, istemcilerin talep edebilecekleri yukarıda bahsi geçen çeşitli veri dosyalarını bünyelerinde barındırır ve talep anında bu dosyaları istemci ile paylaşırlar. Sunucu-istemci ilişkisi Şekil 1'de görülebilir.



Şekil 1. Sunucu-istemci ilişkisi.

Ağın büyüklüğü ve buna bağlı kullanıcı sayısının oldukça fazla olacağı göz önünde bulundurulursa sunucu olarak görev yapacak bilgisayar sistemlerinin donanımsal olarak da bu hizmeti sağlayabilecek kapasitede olması gerekmektedir. Bu nedenle işletim sistemi, işlemci tipi ve hızı, bellek kapasitesi, aşırı ısınmayı engelleyecek şekilde soğutma sistemleri gibi özellikler bu noktada önem kazanmaktadır. Fiziksel olanakların yanı sıra güvenlik tedbirlerinin de ciddi anlamda alınmış olması gerekmektedir. Çünkü barındırılan dosyalar içerisinde kişisel veya değerli veriler yer alabilir. Örneğin; bir bankacılık işleminde çevrimiçi bankacılık hizmetlerini kullanan müşteriler birer istemci iken, istemcinin sistemden yapacağı işlemler için veri tabanından veri çekerek istemci ile paylaşacak olan taraf banka yani sunuculardır. Ancak tüm finansal işlemler (iletişim) süresince veriler belirli güvenlik önlemleri ile korunmalıdır.

Açıktır ki sunucular farklı hizmetler sağlayabilirler. Bu doğrultuda sunucular amaçlarına göre farklı web teknolojileri ve protokolleri kullanabilirler. Çeşitlerine göre sunucular; web sunucuları, dosya paylaşım sunucuları, alan adı sunucuları (DNS), e-posta sunucuları, veri tabanı sunucuları, proxy sunucuları şeklinde sayılabilir.

*Web sunucuları*, www yapısını (web sayfasını) oluşturan tüm dosyaları barındıran sunuculardır. WWW yapısı içerisinde web sayfaları hipermetin temelli oluşturulmakta, bu istemci tarafında yine bir web sayfası olarak görüntülenebilmesi sağlanmaktadır. Bu yapının inşası için ilerleyen kısımlarda daha detaylı ifade edilecek olan HTML işaretleme dili, URL benzersiz web adresi ve ağ üzerinden bağlantı için HTTP iletişim protokolü kullanılmaktadır. İstemci web tarayıcısı kullanarak bir web adresi girdiğinde istekte bulunur ve web sunucusu ilgili web adresine ilişkin web sayfalarını istemcinin bilgisayar sistemine iletir. Web sunucuları konusu içerisinde barındırma (*hosting*) kavramı karşımıza çıkmaktadır. Barındırma, internet üzerinden erişilebilir olması istenen web sayfalarının (web sayfalarından oluşan web sitelerinin) tüm dosyalarının sunucu depolama birimlerinde saklanması hizmetidir. Web sunucularının temel görevi barındırma ve bu sayfaları istemcinin talebi durumunda erişilebilir kılmaktır. Bir web tasarımcısı için barındırma (*hosting*) hizmeti tasarladığı sayfaların istemcilerin erişimine açılması açısından önem arz etmektedir. Tasarımı tamamlanan sayfalara ilişkin tüm dosyalar, barındırma (*hosting*) hizmeti alınması karşılığında sunucunun veri depolama birimlerine aktarılarak web üzerinden ulaşılabilir hale gelmektedir.

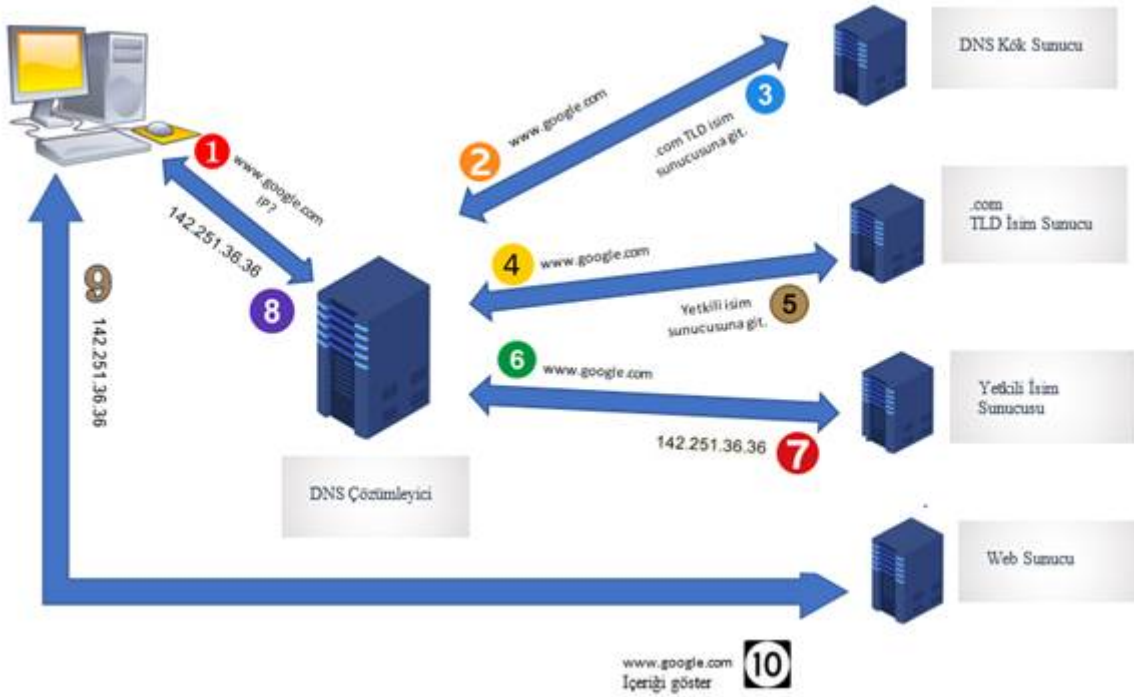
*Dosya paylaşım sunucuları*, ağ üzerinde iletişimde bulunduğu bilgisayar sistemleri ile dosya paylaşımını yapan sunuculardır. Buradaki dosyalar sadece bir web sitesine ilişkin dosyalar olarak düşünülmemelidir. Örneğin bir web sayfasından bağımsız olarak Word, Excel gibi dosyalarının dosya paylaşım sunucusu ve bilgisayar sistemleri arasındaki paylaşımıdır. Dosya paylaşım sunucularına erişim için web tarayıcıları veya özel yazılımlar (Cute FTP, WS\_FTP Pro, Filezilla vb.) kullanılabilir. Dosyalar bilgisayarlarımızda yer aldığı gibi dosya paylaşım sunucularında da klasörler içerisinde bulunabilir. Bu durumda bir dosya indirileceği zaman sunucuya bağlantı sağlandıktan sonra öncelikle dosyanın bulunduğu klasöre girilir ve sonrasında dosya basit bir şekilde indirilebilir. Bu dosya paylaşımı ise FTP protokolü ile gerçekleştirilmektedir. Bu noktada sunucuya bağlantı çoğu zaman sistem yöneticisinin belirlediği kullanıcı adı ve şifre ile gerçekleştirilebilmektedir.

Alan adı sunucularından bahsetmeden önce Alan Adı Sistemini (*Domain Name System-DNS*) ele almak gerekmektedir. Bilindiği gibi ağ üzerinden bir web sayfasına erişim sağlamak istediğimizde o sayfaya ilişkin

adresini tarayıcıya yazmamız ve istek göndermek amacıyla da klavyede Enter tuşuna basmamız gerekir. Kullanıcılar olarak bizler her ne kadar anlamlı ve akılda kalıcı kelimeler bütünü olan bir adres kullansak da bu adres “protokoller” başlığı altında ifade edilen IP adreslerine karşılık gelmektedir. IP adresleri (numaraları) yerine bu günlük dilde, akılda kalması ve yazması kolay olacak şekilde bir isimlendirme sistemi oluşturulmuştur. Örneğin Google arama motorunu açmak istediğimizde tarayıcımızın adres çubuğuna Google’ın IP adresini değil “www.google.com” bağlantı adresini yazmaktayız. Bu bağlantı adresi Tekdüzen Kaynak Gösterici (*Uniform Resource Locator-URL*) olarak tanımlanmaktadır. URL adreslerinin bir sözleşme çerçevesinde adresi talep eden kişi, kurum ya da kuruluşlar tarafından alınması ve tescil edilmesi gerekmektedir. Sözleşme süresi dolduğunda ve sözleşme yenilenmediği takdirde URL adresinin erişimi kapatılır ve başka bir kişi, kurum ya da kuruluşların hizmetine tahsis edilebilir. Aynı örnek üzerinden devam edilecek olursa, URL adresi içerisinde geçen “google” *alan adı*, “.com” ise alan adı uzantısıdır. Bazı özel alan adı uzantıları şu şekildedir.

- edu – Eğitim kurumları bağlantı adresleri (*education*)
- gov – Kamu kurumları bağlantı adresleri (*government*)
- org – Ticari olmayan kuruluşların bağlantı adresleri (*organization*)
- com – Genel kullanım veya ticari kuruluşların bağlantı adresleri (*commercial*)
- mil – Askeri kuruluşların bağlantı adresleri (*military*)
- net – Servis sunucuları bağlantı adresleri (*network*)
- tr, fr, uk – Ülke bağlantı adresleri (*Türkiye, Fransa, Birleşik Krallık vb.*)
- info – Bilgi içerikli web sayfaları bağlantı adresleri (*information*)

DNS, IP adreslerini URL bağlantı adresleri ile eşleştirilmesini sağlayan sistemdir. Tarayıcı adres çubuğuna yazılan bağlantı adreslerinden IP adresi çözümlenerek bu IP için kullanılan web sunucusuna bağlanılmaktadır. Tam bu noktada DNS sunucularının görevi bağlantı adreslerinde bulunan alan adlarını IP adreslerine çevirmeyi gerçekleştirmektir. Bu nedenle DNS sunucuları web adreslerine karşılık gelen IP adreslerini kayıt altında bulundurmaya durumundadır. DNS sunucuları internet ağında dağıtılmış durumdadır. Her bir sunucu konumlandığı bölgedeki IP adreslerini sorgular ve alan adları ile eşleştirme yaparak kayıt altında tutar. Bir web sayfasına erişim sürecinde dört DNS sunucusu görev yapmaktadır. Bunlardan DNS çözümleyici tarayıcıda girilen adresin IP’sinin sorgulandığı ilk basamaktır. DNS sunucularının en tepesi olan kök (*root*) sunucular ve diğer DNS sunucular arasında sürekli veri akışı gerçekleşir. Bu akışta DNS sunucular kök sunucular ile alan adının hangi sunucuda ve hangi IP adresinde yer aldığı bilgisini paylaşmaktadır. Üst Düzey Alan Adı sunucuları (Top Level Domain-TLD) yukarıda da bahsi geçen alan adı uzantılarına (.net, .edu vb.) ilişkin kayıtları tutmaktadır. Dolayısıyla alan adı uzantıları ile ilgili sorgulamalar gerçekleştirildiğinde ilgili veriler bu sunuculardan dönmektedir. İsim sunucuları (Name Server) alan adlarının hangi IP adresinde tutulduğuna ilişkin kayıtların saklandığı asıl sunuculardır. DNS sunucuları aynı zamanda web sunucuları ile birlikte görev yapmaktadırlar. İstemci, DNS sunucuları ve web sunucusu arasındaki akış Şekil 2’de görülmektedir.



Şekil 2. İstemci, DNS sunucuları ve web sunucu arasındaki akış.

*Elektronik posta (e-posta) sunucuları* elektronik iletileri ağda tutan merkezi sunucular olup birer postacı olarak düşünülebilir. E-postalar günümüzün iletişim araçları arasında önemini korumakla birlikte en çok kullanılan uygulamalardan birisidir. Her ne kadar bir e-posta göndermek saniyeler bazında gerçekleşen anlık bir süreç gibi düşünülse de aslında arka planda bir dizi işlemler gerçekleşmektedir. E-posta gönderim sürecini gözümüzle görebildiğimiz hali ile özetleyelim. Kullanıcılar olarak e-posta gönderme işlemi için öncelikle Gmail, Outlook, Hotmail, Thunderbird gibi e-posta yazılımlarına ya da e-posta sunularının web ara yüzlerine ihtiyaç duymaktayız. Bu yazılım ya da web ara yüzlerinde e-posta içeriklerimizi hazırlamakla birlikte yine sunucuların e-postaları doğru bir şekilde iletebilmesi için kullanıcı e-posta adresi gibi bilgileri de yine bu e-posta formu üzerinden iletmemiz gerekmektedir. E-posta adresleri @ işareti ile ayrılan iki ana bölümden oluşmaktadır. “fonaykocoglu@mu.edu.tr” e-posta adresinden de görüleceği üzere fonaykocoglu kullanıcının (alıcının) e-postasının (veya e-posta kutusunun) adı, mu.edu.tr kullanıcının (alıcının) e-posta sunucusunun adıdır. Bu bilgiler kullanılarak aşağıdaki süreç yürütülmektedir.

Gönderilen e-postalar hedef alıcılara ulaşmadan önce çeşitli e-posta sunucularından geçmektedir. E-posta sunucuları giden ve gelen e-posta sunucuları olarak ikiye ayrılmaktadır. Giden posta sunucusu Basit Posta Transfer Protokolü (Simple Mail Transfer Protocol-SMTP) olarak adlandırılmaktadır. Gelen posta sunucuları ise Postane Protokolü (Post Office Protocol-POP3) (versiyon 3) ve İnternet Mesaj Erişim Protokolü (Internet Message Access Protocol-IMAP)’dür. SMTP, e-postaları alıcıya ulaştırabilmek üzere gelen sunucusuna aktararak teslimat kısmını bir diğer ifade ile postalama işlemini gerçekleştirmiş olur. SMTP tek yönlü bir protokoldür ve sadece posta göndermek üzere kullanılır. POP3 ve IMAP arasında bazı farklılıklar söz konusudur. Bunlar arasında; IMAP ile sunucu bağlantısı kurulduğunda istemciye e-postaların sadece başlık bilgilerinin döndürülmesi POP3’ün ise tüm iletileri getirebilmesi, POP3 ile tüm yeni iletiler istemciye aktarılırken bağlantı sona erdirilirken IMAP’te sadece istemcinin talebi üzerine bağlantı kurulması, POP3’ün tek kullanıcı desteği verirken IMAP’in çoklu kullanıcı desteği vermesi, POP3’ün yerel olarak e-postalara erişim sağlayarak çevrimdışı durumda da iletiler üzerinde düzenleme yapılabilmesine olanak sağlaması diğer taraftan e-postaların sunucuda saklanması sebebiyle IMAP ile herhangi bir cihazdan e-postalara erişilebilmesi şeklinde sayılabilir. Gelen ve giden sunucular üzerinden e-posta alıp-gönderme süreci ise şu şekilde işlemektedir:

- İletiyi gönderen kullanıcının e-posta istemcisi SMTP sunucusuna bağlanır.
- E-posta istemcisi SMTP sunucu ile kurduğu iletişimde gönderici ve alıcı e-posta adreslerini, ileti içeriğini ve varsa ekleri paylaşır.

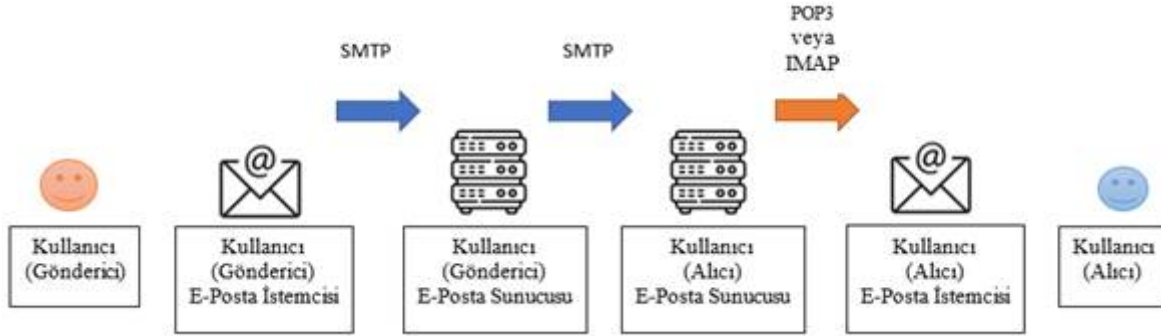


iii. SMTP sunucusu, alıcı e-posta adresini işleyerek alıcının SMTP'si ile iletişime geçer. Gönderici ve alıcı e-posta sunucuları arasında oluşturulan bağlantı için TCP, bu durumda alan adı üzerinden bir erişim sağlanmak isteneceğinden DNS kullanılır. IP adresi belirlenir. Ancak, eğer aynı SMTP sunucusu kullanılıyorsa direkt POP3 ve IMAP sunucuları ile bağlantı kurar.

iv. Alıcı tarafta SMTP ile iletişim kurulduktan sonra alıcı taraf SMTP'si kendisine gönderilen iletiyi kontrol ederek POP3 ve IMAP'e postayı iletir.

v. Alınan e-posta alıcı tarafındaki sunucularda da işlenerek ilgili protokole göre alıcının erişimine sunulur.

Sürece ilişkin akış Şekil 3'te görülmektedir.



Şekil 3. E-posta gönderim süreci

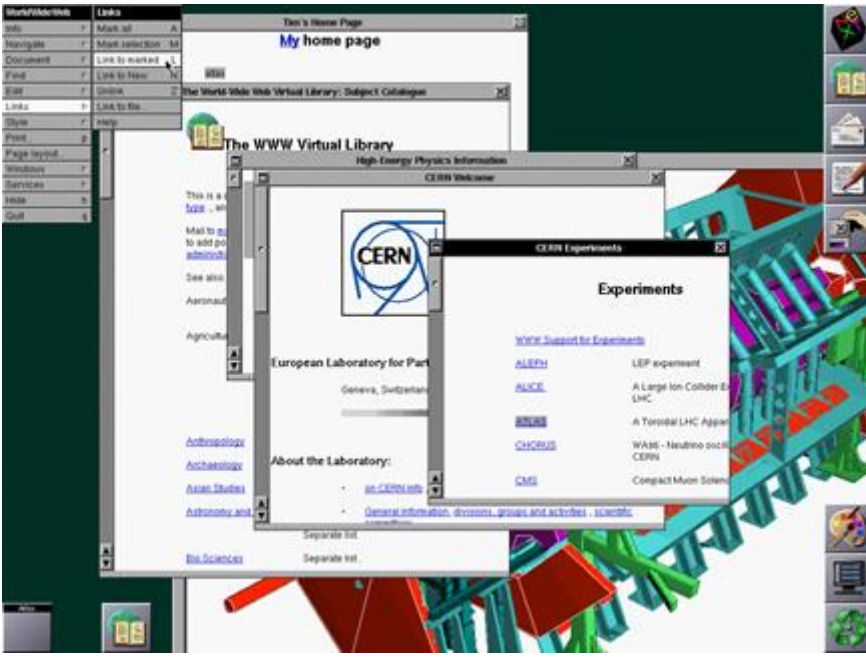
*Veri tabanı sunucuları*, veri tabanı sistemleri ve veri tabanları için hizmet veren sunuculardır. Özellikle kurumsal yapılarda toplanan veriler bu verileri saklayabilecek özelliklere sahip özel sistemlerde saklanmaktadır. Bu sistemler veri tabanı sistemleri ve veri ambarları olarak ifade edilmektedir. Veri tabanı sistemleri veri kümelerinin düzenli bir şekilde tutulduğu ve bu veri kümeleri üzerinde yönetebilme, güncelleyebilme gibi işlemlere olanak sunan ortamlar olarak tanımlanabilir (Koçoğlu, 2012). Fonksiyonel olarak bazı farklılıklar söz konusu olsa da veri ambarları da benzer şekilde tanımlanmaktadır. Kurumsal bir yapının saklaması gereken veri kümelerinin boyutu ve miktarı göz önünde bulundurulursa ihtiyaç duyulacak donanım ve yazılım araçlarının da büyük ölçekli olması gerektiği açıktır. Veri tabanı sunucularının donanımsal bileşen görevi göz önüne alınırsa veri tabanları ve ilişkili mimariler veri tabanı sunucularında tutulabilir. Diğer taraftan aslında veri tabanı sunucuları ağırlıklı olarak bir yazılım bileşenidir. Aynı ya da farklı ağlarda bulunan bilgisayarlar ya da uygulamalara veri tabanı hizmeti sağlamak üzere kullanılmaktadır. Bu doğrultuda, sistemsel yapılandırma, veri ile ilgili erişim ve güvenlik yönetimi, verinin kullanıcı (istemci) isteğine bağlı olarak sorgulanması ve ilgili sonuçların döndürülmesi, verinin güncellenmesi, işlenmesi ve aktarımı gibi görevleri yerine getirmektedir. Günümüzde en bilinen veri tabanı sunucuları arasında SQL Server, MySQL, Oracle sayılabilir.

*Proxy sunucusu* birden fazla kullanıcının yer aldığı bir internet ortamında trafiği düzenlemek ve aynı web sayfalarına erişim için istekte bulunan kullanıcıların oluşturacağı trafiği düzenlemek, hızlı ve verimli erişim sağlamak üzere kullanılmaktadır. Bir diğer ifade ile kullanıcı ile web üzerinde yer alan bir kaynak arasında aracı görevini yürütmektedir. Proxy sunucuları olmadan bir web adresinde yer alan içeriklere erişim isteğinde bulunulduğunda istemcinin bilgisayar, IP adresi, konum gibi farklı özel bilgileri de bu erişim isteği ile birlikte gönderilir. Proxy sunucuları kullanıldığında ise bu doğrudan bağlantı yerine istemcinin belirtilen özel bilgileri filtrelenerek ya da şifrelenerek istek Proxy sunucusu tarafından gönderilir. Diğer taraftan Proxy sunucuları kendisinin kullanan bir kullanıcının erişim isteği gönderdiği web sayfasının kopyasını kendi sabit diskine yedekleyebilir böylece başka kullanıcılar da aynı sayfaya erişmek istediğinde tekrar tekrar sayfayı ana kaynaktan indirmek yerine kendi diskinden kullanıcılara iletebilir. Bu durum daha hızlı erişime olanak sağlamaktadır. Proxy sunucularının temel özellikleri arasında çeşitli web sayfalarına erişimi kısıtlayabilme, tek ağda birden fazla cihazın bağlanabilmesine olanak sağlama, erişim hızının artırılması, kullanıcı verisinin güvenliğini sağlama bir diğer ifade ile anonim gezinme gibi özellikler sayılabilir.

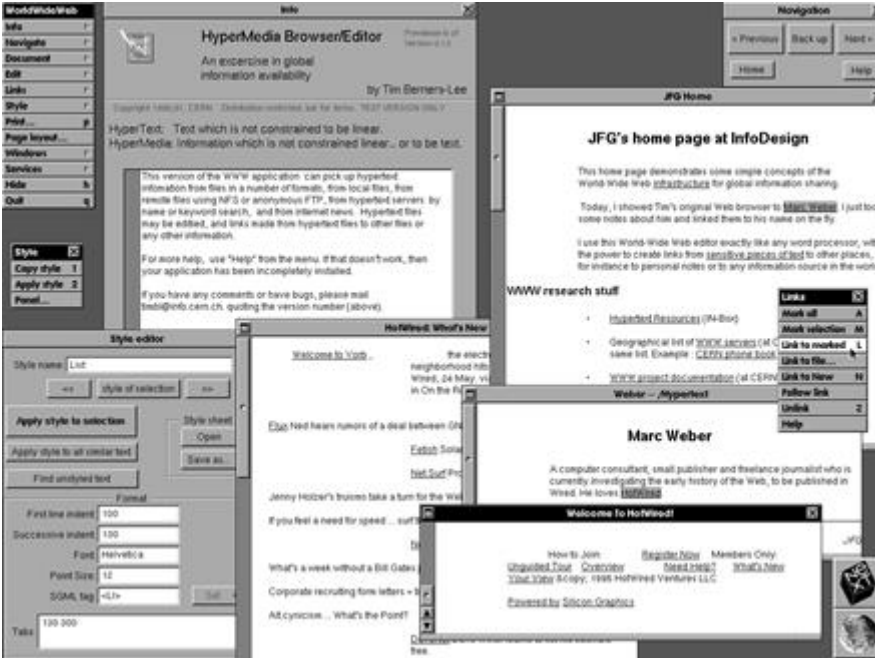
**Web Tarayıcı:** Bilindiği üzere internetten bir web sayfasına erişim sağlamak üzere öncelikle erişilmek istenen sayfanın adres bilgisini yazdığımız bir uygulama yazılımı kullanırız. Bu uygulama yazılımı web tarayıcı (browser) olarak adlandırılır. Tarayıcı, HTTP kullanarak kullanıcı ile kullanıcının erişim isteğinde

bulunduğu sayfa/kaynak arasında köprü görevi görür. Bir diğer ifade ile istemci ve sunucu arasında bağlantının gerçekleştirilmesini sağlamaktadır. Web tarayıcılar WWW’de yer alan bilgi kaynaklarına erişim (dosyalara erişme, dosya indirme, görüntüleme vb.) imkânı sağlayan uygulama yazılımlarıdır. Bu bağlantı sırasında ilgili protokolle birlikte güvenli bir bağlantının oluşturulmasına da olanak sağlamaktadır. Web tarayıcılar erişilmek istenen web sayfalarına ait metin veya animasyon, çizim, görüntü, grafik, metin, ses ve videolar gibi nesneleri içeren çoklu ortam dosyaları üzerinde açma, kaydetme, arama gibi işlemleri gerçekleştirebilir, tüm iletişim protokolleri ve standartlarıyla uygun çalışabilir, ağ üzerinden dosya yükleme ya da indirme işlemlerini gerçekleştirebilir, e-posta ve metin düzenleyicileriyle entegre çalışabilir bunların yanı sıra tarayıcıları kullanırken fark edeceğimiz gibi geçmiş gezinme kaydı tutma, çevrimdışı çalışma, çıktı alma, çeviri ve kaynak yönetimi gibi farklı uygulama yazılımları ile entegre çalışabilme gibi özelliklere de sahiptir. Tarayıcıya URL adresi girildikten sonra ilgili web sayfasının dosyalarının bulunduğu web sunucusu ile bağlantı kurulmaktadır. Bağlantıda kullanılacak protokolün belirlenmesi adres çubuğuna girilen “<http://google.com>” gibi bir adreste yer alan protokole göre belirlenmektedir ki bu adreste de görüldüğü üzere ve de web sunucuları bölümünde de ifade edildiği gibi HTTP iletişim protokolü olarak kullanılmaktadır. Hangi sunucu ile iletişime geçilmesi gerektiğini de yine bu adres belirlemektedir. Sunucu bilgisayarla bağlantı ise IP adresi ile sağlanacağından öncelikle DNS sunucu ile bağlantı gerçekleştirilir ve adrese karşılık gelen IP belirlenerek ilgili web sunucusu ile iletişime geçilmesi sağlanır. Web sunucusuyla kurulan bağlantı sonrasında web tarayıcısı siteye ait eğer başka özel bir sayfa görüntülemesi talep edilmediyse “index.html” isimli sayfayı görüntülenmeye çalışılır ve bunun için de sayfaya ilişkin tüm nesneler sunucudan talep edilir. “index.html” sayfası bir web sitesinin ana sayfası olarak ifade edilebilir ve bu isimlendirme global olarak kabul görmüştür. Web sunucusu sayfaya ilişkin dosyaları gönderirken bu dosyaları ağ üzerinde iletebilecek büyüklükte dosyalar olarak parçalaması gerekir. Gönderim sırasında bu küçük veri paketleri sunucu tarafından gönderilirken, web tarayıcı TCP’yi kullanarak gönderilen bu paketleri birleştirir. Sayfaların gönderilmesi ve birleştirilmesi sırasında arka planda HTML kullanılır. Web sayfaları HTML dili ile kodlanır, yani sayfa bileşenleri olarak tablolar, görüntüler, seslere vb. bileşenlere karşılık gelen HTML kodları ve bu kodların yer aldığı dosyalar mevcuttur. Dolayısıyla veri paketleri bir araya getirilirken içeriğin düzgün bir şekilde yeniden görüntülenebilmesi için sayfalara ait HTML kodları çözümlenmektedir.

Bu bölümün başında paylaşılan internet tarihçesinden de anlaşılacağı üzere sunucu ve istemci bilgisayarlar arasındaki iletişim için HTTP geliştirilmiş, ağ üzerinde yer alan web siteleri ve bu sitelerin içeriğini oluşturan kaynaklar arasında belirli bir standarda uygun olarak iletişim kurulması sağlanmıştır. Yine bu bağlamda web sayfalarının geliştirilebilmesi için HTML dili geliştirilmiştir. Tüm bu gelişmeler ise web tarayıcılarının çıkışına zemin hazırlamıştır. 1990’lı yılların başında Tim Berners-Lee ve ekibi tarafından WorldWideWeb tarayıcısı geliştirilmiş ve ilk web tarayıcısı olmuştur. Bu tarayıcı aynı zamanda düzenleyici olarak da ifade edilmektedir. Belirli bir işletim sistemi (NeXTStep) üzerinden çalışabiliyor olması sebebiyle kısıtlı kullanımı söz konusuydu. Diğer taraftan WWW ile karıştırıldığı için ismi Nexus olarak değiştirilmiştir. Kullanıcı ara yüzüne sahip geliştirilen ilk tarayıcı ise Erwise olmuştur. Sonrasında Marc Andreessen tarafından geliştirilen Mosaic ise web tarayıcıların popülerleşmesinde öncülük eden tarayıcı olarak bilinmektedir. Gelişen web teknolojileri, teknolojik diğer olanaklar ve kullanıcı ihtiyaçlarına göre günümüze kadar birçok farklı özellikte web tarayıcısı geliştirilmiştir. Günümüzde ise bilinen web tarayıcıları arasında Opera, Microsoft Edge, Google Chrome, Mozilla Firefox sayılabilir. Şekil 4’te ve Şekil 5’te ilk web tarayıcısına ait 1993 tarihli görüntüler ve günümüz örnek web tarayıcılarından birisine ait görüntü ise Şekil 6’da verilmiştir.



Şekil 4. İlk web tarayıcı görüntüsü-1

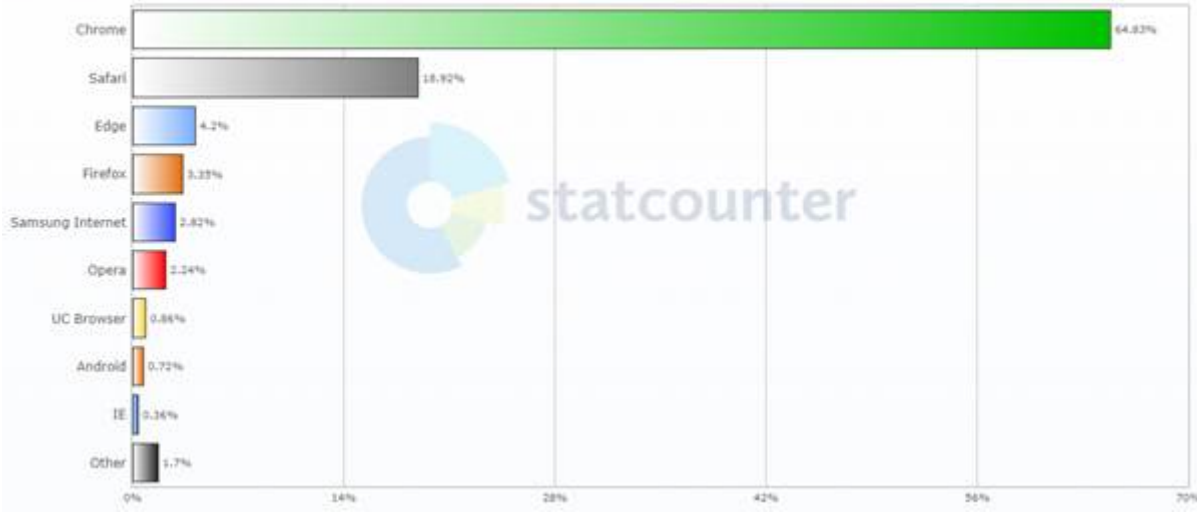


Şekil 5. İlk web tarayıcı görüntüsü-2

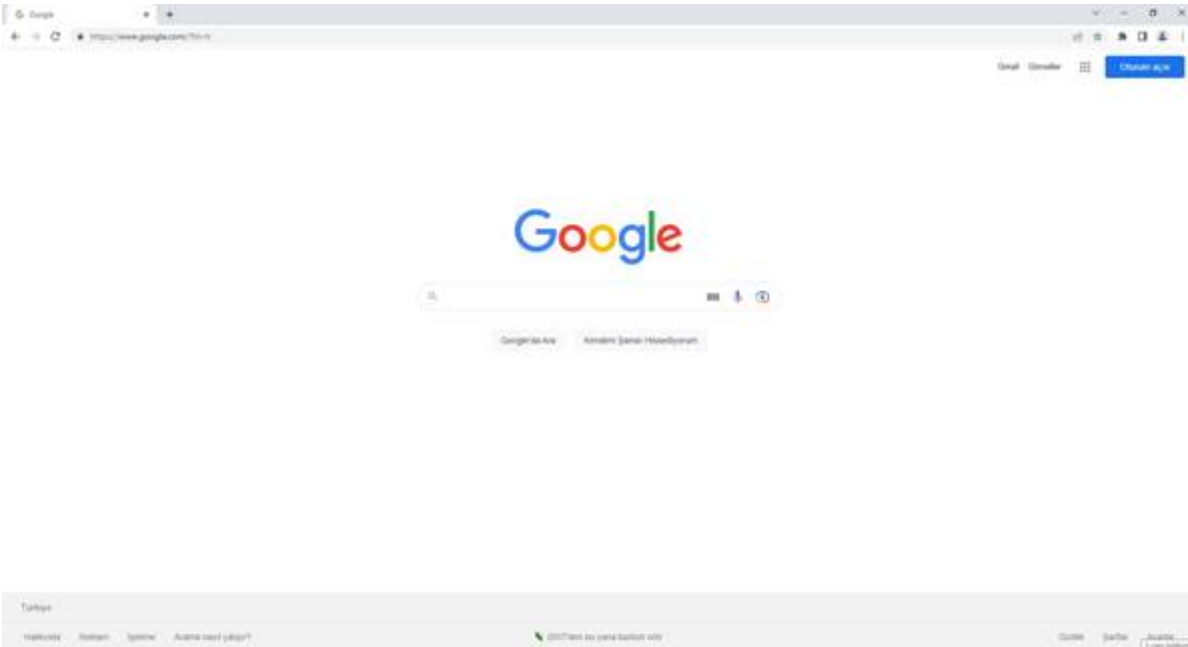
İlk tarayıcıya ait görüntülerde tarayıcının gri tonlamalı olduğu ve satır içi çeşitli imge ve şekillerin ayrı birer pencere olarak açıldığı görülmektedir. Tim Berners-Lee bu durumları kodlamayı gerçekleştirdiği NeXT adlı bilgisayarın sadece gri tonlamaya izin vermesi ve tarayıcının geliştirilmesinden kaynaklı satır içi görüntü yapılamaması şeklinde açıklamaktadır. İlk tarayıcıya ait bu görüntülere ve uygulamaya ait diğer özelliklerin açıklamalarına URL1'den ulaşılabilir.

Günümüzde web tarayıcılarının kullanım oranları görülmektedir (URL2). Oranlar Ocak 2022 ile Ocak 2023 tarihleri arasında masaüstü, mobil, tablet, konsol araçlarındaki tüm kullanımları kapsamaktadır. Buna göre %64.83 oranla açık ara ilk sırada Google Chrome gelirken, %18.92 oranıyla Safari ikinci, %4.2 ile Microsoft Edge üçüncü sırada yer almaktadır. Aynı link üzerinden zaman içerisinde web tarayıcıların kullanım oranlarının değişimi elde edilebilir. Diğer taraftan dünya üzerindeki farklı bölgelerine ve farklı cihazlardaki kullanıma göre de grafiği oluşturmak mümkündür.

## Browser Market Share Worldwide Jan 2022 - Jan 2023

[Edit Chart Data](#)


Şekil 6. Ocak'22 ve Ocak'23 tarihleri arasında web tarayıcılarının kullanım oranları



Şekil 7. Google Chrome web tarayıcısı ana ekranı

Bu web tarayıcısının özelliklerini incelediğimizde Google arama motoru ile ana ekranın geldiğini, Sol üst köşede sayfanın adının yer aldığını, bir altında adres çubuğunun yer aldığını ve erişilmek istenen URL adresinin buraya yazılması gerektiğini, adres çubuğunun sağ tarafında tarayıcıya ait farklı özelliklere ilişkin ve ilgili bağlantılara götüren ikonlar ve butonların yer aldığını ifade edebiliriz.

## Bölüm Özeti

Web tasarım dersine giriş niteliğinde olan bu bölümde web tasarım ilkeleri, tasarım araç ve dillerinin detaylarına geçmeden önce web tasarım ile ilgili çeşitli kavramlar ele alınmıştır. Çünkü bir web sayfasının tasarlanmasının yanında bu sayfanın global iletişim ağında nasıl paylaşılabileceği, hangi önemli teknolojilerin buna olanak sağladığını, web sayfalarına erişim süreçlerinin nasıl gerçekleştiği konusunda da bir web tasarımcının bilgi sahibi olması gerekmektedir. Bu doğrultuda, öncelikle web'in gelişiminden bahsedilmiştir. İnternet günümüzün en önemli araçlarından biridir. İnternet vasıtasıyla dünyanın herhangi bir yeri ile istenilen saatte iletişime geçilebilmektedir. Diğer taraftan kütüphanelerde binlerce raf kaplayacak boyutta bilginin internetten ulaşılabilirliğini biliyoruz. Keza dünyanın farklı yerlerinde üniversitelere de artık bir tuş kadar yakınız. Günlük yaşantımızda zaman kaybı yaşatan, fiziksel yorgunluğa sebep olan birçok işlemi de internet üzerinden gerçekleştirebiliyoruz. Eskiden öğrenci bürolarının önünde ders almak için

kuyruklar oluşturan öğrenciler yerine artık evinden dilediği saatte dersini seçebilen öğrenciler var. Bankacılık işlemleri, online alışverişler, sağlık uygulamaları vb. birçok uygulama sayılabilir. Bunca işimizi yürütürken aslında arka planda çalışan çokça sistem ve teknolojik teknik alt yapı söz konusudur. Kullanıcının daha aşına olduğu iki temel bileşen ise internet ve web tabanlı uygulamalardır. İnternet öncelikle çok küçük bir ağ olarak belirli bir konumdaki belirli bilgisayarların bağlanabildiği bir ortam iken bugün kullanıcı sayısı milyarlarla ifade ediliyor. Elbette bugüne gelene kadar bu iletişim alt yapısını etkin şekilde kurabilmek, iletişimin kesintisiz sağlayabilmek, güvenlik problemlerini aşabilmek için birçok farklı teknoloji kullanılmış, çeşitli standart ve prosedürler belirlenmiştir. Diğer bir bileşen ise web uygulamalarıdır. Web sitelerini görüntülemek için kullandığımız uygulamalar web tarayıcılarıdır. Bu tarayıcılar vasıtasıyla gerek dinamik sistemlere erişim sağlayabiliyor, hem de statik web sitelerinde gezinerek bilgi kaynaklarına erişim sağlayabiliyoruz.

Bu doğrultuda, bu bölümde olmazsa olmaz internetin nasıl doğduğu hangi aşamalardan geçerek günümüze geldiği, bu süreçte geliştirilen teknolojiler, protokoller, standartların neler olduğu ve ne işe yaradığı gibi bilgiler bölüm kapsamında ele alınmıştır. Diğer taraftan web sitesini sunan ve buna erişmek isteyen kullanıcılar arasında bu iletişimin nasıl sağlandığı da ifade edilmiştir. Son olarak web sayfalarına erişimi sağlayan yazılım araçlarına ve gelişim sürecine yer verilmiştir.

### Kaynakça

Cohen-Almagor, R. (2013). Internet history. In *Moral, ethical, and social dilemmas in the age of technology: Theories and practice*, IGI Global, 19-39.

Keefer, A., & Baiget, T. (2001). How it all began: a brief history of the Internet. *Vine*, 31 (3), 90-95.

Koçoğlu, F.Ö. (2012). Veri madenciliği sürecinde veri ayrıklaştırma yöntemlerinin karşılaştırılması ve bir uygulama. İstanbul Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi.

ODTÜ, Erişim Linki: <http://www.internetarsivi.metu.edu.tr/tarihce.php>, Erişim Tarihi: 03.03.2023

Pxhere, 2023, Erişim Linki: <https://pxhere.com/en/photo/1445251>, Erişim Tarihi: 02.03.2023

URL1, Erişim Linki: <https://www.w3.org/2004/Talks/w3c10-HowItAllStarted/?n=1>, Erişim Tarihi: 03.03.2023

URL2 <https://gs.statcounter.com/browser-market-share#monthly-202201-202301-bar>

## Ünite Soruları

### Soru-1 :

Kullanıcılar arasındaki iletişim ve her türlü veri-bilgi alışverişinin kullanıcıların bilgisayarları ile bağlanarak gerçekleştirdiği büyük ağa ne denir?

(Çoktan Seçmeli)

- (•) - Protokol
- (•) - Sunucu
- (•) - İnternet
- (•) - Alan adı
- (•) - Web tarayıcı

**Cevap-1 :**

Protokol

---

**Soru-2 :**

Farklı ağların birbiri ile bağlantısında iletişimin belirli bir standarda göre oluşturulması için öne sürülen protokol hangisidir?

(Çoktan Seçmeli)

(•) - HTTP

(•) - FTP

(•) - TCP

(•) - IP

(•) - STP

**Cevap-2 :**

TCP

---

**Soru-3 :**

İki bilgisayar sistemi arasındaki dosya alışverişi gerçekleştirilmesini düzenleyen protokol hangisidir?

(Çoktan Seçmeli)

(•) - HTTP

(•) - FTP

(•) - TCP

(•) - IP

(•) - STP

**Cevap-3 :**

FTP

---

**Soru-4 :**

Kamu kurumları bağlantı adresleri için kullanılan özel alan adı uzantısı hangisidir?

(Çoktan Seçmeli)

(•) - edu

(•) - gov

(•) - tr

(•) - com

(•) - info

**Cevap-4 :**

gov

---

**Soru-5 :**

“Ağda yer alan cihazlar ise kendilerini tanımlayan \_\_\_\_\_ adı verilen belirli bir sistematığe göre üretilmiş birer numara kullanırlar.” Boşluğu en uygun cevapla doldurunuz.

(Çoktan Seçmeli)

(•) - HTTP

(•) - FTP

(•) - TCP

(•) - IP

(•) - STP

**Cevap-5 :**

IP

---

**Soru-6 :**

IP adreslerini URL bağlantı adresleri ile eşleştirilmesini sağlayan sistem aşağıdakilerden hangisidir?

(Çoktan Seçmeli)

(•) - HTML

(•) - HTTP

(•) - ANS

(•) - DNS

(•) - FTS

**Cevap-6 :**

DNS

---

**Soru-7 :**

Aşağıdakilerden hangisi/hangileri veritabanı sunucularındandır?

I. SQL Server,

II. MySQL,



### III. Oracleqq

(Çoktan Seçmeli)

- (•) - Yalnız I
- (•) - Yalnız II
- (•) - I ve II
- (•) - II ve III
- (•) - I,II ve III

**Cevap-7 :**

I,II ve III

---

**Soru-8 :**

Aşağıdakilerden hangisi günümüz web tarayıcılarından biri değildir?

(Çoktan Seçmeli)

- (•) - Opera,
- (•) - Microsoft Edge,
- (•) - Google Chrome,
- (•) - Mozilla Firefox
- (•) - Windows

**Cevap-8 :**

Windows

---

**Soru-9 :**

Aşağıdakilerden hangisi web sayfasını oluşturan tüm dosyaları barındıran sunuculardır?

(Çoktan Seçmeli)

- (•) - Dosya sunucusu
- (•) - Eposta sunucusu
- (•) - Web sunucusu
- (•) - Proxy sunucusu
- (•) - Bağlantı sunucusu

**Cevap-9 :**

Web sunucusu





## 2. WEB TASARIMIN TEMEL İLKELERİ

### Birlikte Düşünelim

Bir web tasarımcı bütün bir web sitesini geliştirebilir mi? Hangi ölçüdeki bir web sitesini geliştirebilir?

Web geliştirme sürecinde kimler hangi rollerde görev alabilir?

Web tasarımcı ya da web geliştirici olmak için hangi yeteneklere sahip olmayı gerektirir?

Web sitesi tasarlanırken nelere dikkat edilmelidir?

### Başlamadan Önce

Bu bölüm kapsamında web geliştirme süreci ele alınmıştır. Web tasarım bu sürecin bir alt adımı olarak tanımlanmıştır. Süre içerisinde ekip üyelerinin görevleri belirtilmiştir. Web geliştirmeye yönelik gerekli yetkinliklerin neler olabileceği tartışılmıştır. Web tasarım sürecinin temel ilkeleri açıklanmıştır.

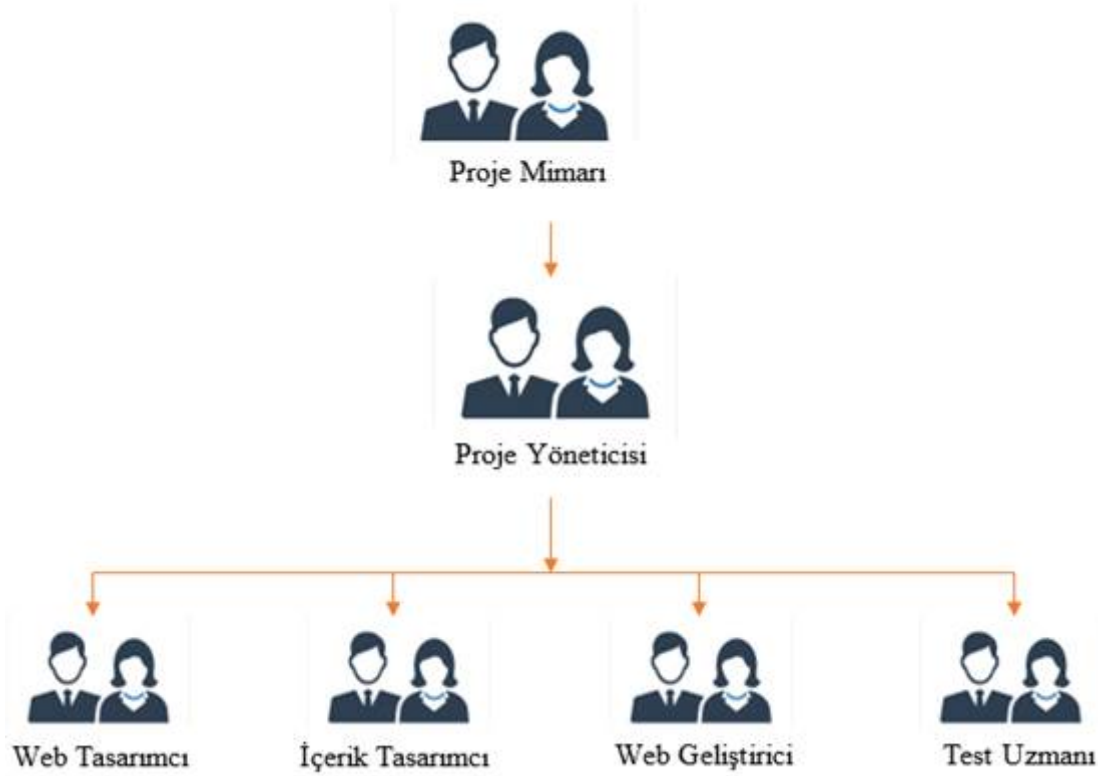
### 2.1. Giriş

İnternet kullanıcıları olarak internet üzerinden çeşitli web sitelerine erişiyor, bu siteler içerisindeki farklı sayfalardan istediğimiz bilgiyi edinebiliyoruz. Bu sayfaların hazırlanmasında web tasarımcılar görev yapmaktadırlar. Bir web sitesi hazırlanmak istendiğinde, bir web tasarımcısı olarak sitenin tasarımını tek ya da birkaç sayfanın tasarlanması olarak düşünemeyiz. Bir web sitesi çokça bazen yüzlerce sayfadan meydana gelen bir yapıdadır. Bu sayfalar arasında bağlantılar söz konusudur. Dolayısıyla tasarım gerçekleştirilirken proje yönetimi bakış açısı ile tasarım süreci ele alınmalıdır. Profesyonel bir yaklaşımla amaca uygun, hedef kullanıcının ilgisini çekecek, kolay kullanıma olanak sağlayacak, var olan sayfalar arasında etkin bir gezintiyi sağlayabilmek adına doğru yapılandırılmış, ihtiyaç duyulan tüm web teknolojilerinin entegre edildiği kısacası hem görsel hem de fonksiyonel anlamda yeterli bir tasarım gerçekleştirilmelidir. Bu amaçla web tasarımı gerçekleştirilirken bazı ilke ve esaslar göz önünde bulundurulmalı, tasarım sistematigi hakkında bilgi sahibi olunmalıdır. Daha açıklayıcı örnek vermek gerekirse, web sitesinin görsel tasarım aşaması ele alınsın. Bir grafik, baskı yapıldığında çok iyi görsel şov sunabilir ancak web sitesinin belirli kısıtları altında bu grafiği kullandığımızda, görüntülemek için kaydırma çubuklarına ihtiyaç duyuluyorsa görseli bir bütün olarak görememek, kullanıcıyı gereksiz kaydırma çubuğu kullanmaya mecbur bırakmak istenen durumlar olmayacağından, web sayfasında aynı grafiği kullanmak pek de etkili olmayabilir. Bu durum hem sayfa yapısının doğru tasarlanmamasından hem de tasarımda kullanılacak görsel materyallerin tasarıma uygun olmamasından kaynaklı olduğu ifade edilebilir. Bir başka örnek ise site içeriğinde gezinme açısından verilebilir. Eğer kullanıcı sayfalar arasında kayboluyor, web sitesinde var olan ve ulaşmak istediği içeriğe daha kısa yol varken daha fazla adımda ulaşıyorsa hatta hiç ulaşamıyorsa web tasarım tarafında yine sorun var demektir. Dolayısıyla web tasarım aşamasında göz önünde bulundurulacak kullanılabilirlik ve sayfa verimliliğine ilişkin temel ilkelerle bu gibi sorunlara çözüm sağlanabilir. Diğer taraftan, bir web sitesinin ortaya çıkışı sadece web tasarımcının görev alması ile gerçekleştirilecek bir süreç olmadığını söylemek mümkündür. Bu noktada, planlama, tasarım, geliştirme ve siteyi hayata geçirme bir bütün olarak ele alındığında bu işin iyi bir ekip işi olması gerektiği de açıktır. Web tasarımcıların çalışacakları ekipte kimlerin yer alacağı hususunda da bilgi sahibi olması gerekmektedir. Peki bu ekip içerisinde kimler yer almalı ve hangi görevleri yerine getirmelidir?

Bu bölüm kapsamında hem bir proje bakış açısıyla web tasarım süreci ele alınacak, hem de tasarımda göz önünde bulundurulması gereken hususlar paylaşılacaktır.

### 2.2. Web Tasarımı Sürecinde Ekip Çalışması

Giriş bölümünde ifade edildiği gibi profesyonel anlamda web sitesi geliştirme işi, iş takvim planlama, bütçe yönetimi, etkin görev dağılımı gibi birçok farklı alt iş ve görevleri barındıran, bu nedenle bu çeşitli görevleri üstlenip yerine getirebilecek bilgi ve yeteneğe sahip ekip elemanlarının yer aldığı takım çalışmasıdır. Bu noktada bir web tasarımcının bu iş süreci içerisinde nasıl konumlandığını, kimlerle birlikte çalışması gerektiğini, ekip elemanlarının görev ve sorumluluklarının neler olduğunu bilmesi gerekmektedir. Şekil 7’de bir web sitesi geliştirme ekibinde yer alan temel ekip üyeleri ve hiyerarşik yapı görülmektedir.



Şekil 8. Web sitesi geliştirme ekibi

- Proje mimarı, müşterinin ihtiyaçlarına, web sitesinin kullanılacak iş alanına ve web sitesinin hedef kullanıcı profiline göre projenin genel konseptini belirleyen, gerekli fizibilite çalışmalarını gerçekleştiren, web sitesinin hizmete girmesi sonrasında kullanıcıların siteye yönlendirilebilmesi için gerekli reklam, organizasyon ve iş birliklerini gerçekleştiren, bir bakıma müşteri, kullanıcı ve geliştirici ekip ile etkileşimde olan ve bu üçü arasında koordinasyonu sağlayan kişidir.

- Proje yöneticisi, proje mimarından gelen müşteri isteklerine göre hayata geçirilecek web sitelerinin planlama ve gerçekleştirme adımlarında kararları alacak, bütçeyi yönetecek, çalışacak ekibin yönetilmesinden, işin belirlenen iş planına uygun gidip gitmediğinin kontrolünden, oluşacak risk faktörlerine göre gerekli adımların atılmasından kısacası işin yürütülmeye başlamasından tamamlanmasına kadar geçen sürede tüm alt süreç ve faaliyetlerinden, proje ekibinin koordinasyonundan sorumlu kişidir. Proje hedefleri, kapsamı, bütçeyi ve ekibi dengede tutabilmesi gerekmektedir.

- Web sitesi tasarımcısı, gerçekleştirilecek sitenin hem görsel tasarımını gerçekleştiren hem de kullanılması gereken fonksiyonları belirleyen kişidir. Bu noktada tasarımcı, görünüm ve gerekli fonksiyonları tasarlarırken müşterinin iş hedefleri ve taleplerine uygun, kullanıcı tarafında da kolay kullanımlı, pratik bir ara yüz tasarımı da gerçekleştirmeli, ihtiyaç duyulacak metinsel, görsel ve işitsel nesneleri ve gezinme öğelerini de tasarımına entegre edebilmelidir. Grafik ve yapısal site yapısı dengesini de bu anlamda korumalıdır. İlgi çekici olmak adına yaratıcı ve mümkün olduğunca benzersiz tasarımlar ortaya koyabilmelidir. Yine site içerisinde sayfalar arasındaki etkili navigasyonu sağlayacak bir site haritası tasarımını yukarıdaki hususları göz önünde bulundurarak gerçekleştirmelidir. Hem ana şablonu hem de ana ve alt sayfa detaylarını tasarlamalıdır. Yine bir web tasarımcısı tasarımın sorunsuz bir şekilde çalışmasından da sorumludur. Bir web tasarımcısından HTML ve CSS hatta JavaScript dillerini bilmesi beklenmektedir. Yine yaratıcı görsel tasarımları gerçekleştirebilmesi adına Photoshop, Adobe Illustrator, FreeHand, Corel Draw, SWiSH Max gibi yazılımlar hakkında da bilgi sahibi olmalıdır.

- İçerik tasarımcılar, sayfalar içerisinde yer alan görsel öğelerin, metinlerin tasarlanmasından sorumludurlar. Metin ve görseller sitenin dikkat çekebilir olmasında önemli unsurlardandır. Bu nedenle kullanıcıyı aktif tutmak için bir kerelik değil bazen yeni tasarımlara da yer vermek gerekebilir.

- Web sitesi geliştiriciler, Java, SQL, PHP, ASP.NET, Python ve C# gibi çeşitli kodlama dilleri vasıtasıyla ürün olarak ortaya çıkarılacak bir web sitesinin temellerini hazırlarlar. Bir diğer ifade ile özellikle dinamik web sitelerinde sitenin sunucu ve veri tabanları tarafında yapılandırılması, site sahiplerinin ya da editörlerinin kullanımına yönelik yönetici panellerinin oluşturulması ile ilgili gerekli olan kodlamaların gerçekleştirilmesinden sorumludurlar.

- Test uzmanları ise geliştirilen sitelerin gereksinimleri ne ölçüde karşıladığını belirleyen, işlevlerinin doğru çalışıp çalışmadığını, herhangi bir güvenlik açığı olup olmadığını inceleyen, son kullanıcıya sunulmadan önce tüm hata ve eksiklikleri ortaya çıkartan kişilerdir. Test uzmanları hataları ortaya çıkartırken bunların nedenlerini de ayrıca tanımlamaktadırlar. Potansiyel hataları ve riskleri de aynı şekilde raporlarlar.

Yukarıda ana hatlarıyla bir ekipte kimlerin yer aldığı listelenmiştir. Elbette çeşitli uzmanlık alanlarına göre ekip içerisindeki pozisyonları arttırmak mümkündür. Örneğin kullanıcı deneyimi son dönemlerde önem kazanan konulardan birisidir. Bir web sitesinin kullanılabilirliğinin yanı sıra, kullanıcının web sitesi içerisinde odaklandığı noktalara göre site tasarımı gerçekleştirilebilir. Örneğin, reklam yerleştirme yapılacak bir web sitesinde kullanıcıların kullanım sırasında odaklandığı bölgeler belirlenebilirse reklamlar bu alanlara yerleştirilebilir ve daha fazla reklam geliri elde edilebilir. Aynı durum ön plana çıkarılmak istenen bir web sitesi içeriği için de geçerli olabilir. Bu noktada karşımıza çıkan uzmanlık alanları kullanıcı ara yüzü (*User Interface-UI*), kullanıcı deneyimi (*User Experience-UX*) tasarımcısı, etkileşim tasarımı (*Interaction Design-ID*) şeklinde sayılabilir. Benzer olarak içerik tasarımcılar da detaylandırılabilir. Görsel öğelerin tasarımında grafik tasarımcılar, metin tasarımında ise metin yazarları görev yapabilirler. Diğer taraftan tasarlanan tüm görsel ve metinsel içeriklerin seçimi konusunda da ayrıca bir içerik editörü görevlendirilebilir. Arama motorlarında sitelerin görüntülenmesi konusunda gerekli çalışmaların yapılması için arama motoru optimizasyonu (*Search Engine Optimization-SEO*) uzmanları da ekip içerisinde görev alabilmektedir. Bu çalışmalar sonucunda web sitesinin arama motorlarında daha üst sıralarda görüntülenmesinin sağlanması ve görüntüleyen kullanıcı sayısının da buna bağlı olarak artırılması amaçlanmaktadır. Web geliştiriciler de çeşitli uzmanlık alanlarına göre farklı görevlerle ekipte yer alabilirler. Bu uzmanlıklar arasında arka-uç (*back-end*), ön-uç (*front-end*) ve tam yığın (*full stack*) geliştirici olarak üç farklı web geliştirici uzmanlığı sayılabilir. Bu ayrım web sitesinin ön yüz (kullanıcı ara yüzü), arka plan temel bileşenlerin tasarlanması ve geliştirilmesine yönelik olarak gerçekleştirilmektedir.

Özellikle web tasarımcı ve web geliştirici kavramları karıştırılmaktadır. Bu iki uzmanlık alanının farklı görevleri ve sorumlulukları söz konusudur. Bu eğitim programı kapsamında web tasarım ve web geliştirme (programlama) derslerinin ayrı ayrı öğrencilere sunulduğunu hatırlayınız. Aradaki farklılıkları detaylandırmadan önce bu iki uzmanlık alanı inşaat mühendisleri ve mimarlar örneğini göz önünde bulundurularak açıklanabilir. Bir binanın inşa edilmesinde mimarlar binanın hem genel hem de kat bazında tasarımını gerçekleştirirler. Kaç katlı olması gerektiği, dış görünüşünün nasıl olacağı, arsa içerisinde nasıl konumlandırılacağı, kat planlarında odaların, havalandırma boşluklarının, yangın merdivenlerinin nerelerde yer alacağını vb. tüm detayları belirler ve de bu detaylara göre bir plan ortaya koyarlar. İnşaat mühendisleri ise bu plana uygun olarak ihtiyaç duyulan çimento, demir vb. malzemelerin belirlenmesi, atılacak temelin tüm detaylarını belirlenmesi, gerekli kolon-kiriş hesaplamalarının yapılmasından sorumludur ve binanın ana yapısının ortaya çıkışını kontrollü bir şekilde inşaat işçilerinin yardımıyla gerçekleştirilmesini sağlamaktadır. Mimar-mühendis ilişkisine bakıldığında birinin tasarım ağırlıklı diğerinin ise gerçekleştirme ağırlıklı çalıştığı görülmektedir. Benzer ilişki web tasarımcı-geliştirici arasında da söz konusudur. Tasarımcı ve geliştiriciler koordineli çalışarak, birbirlerini tamamlamaktadırlar. Tasarımcı ve geliştirici arasındaki farkları şu şekilde sıralayabiliriz:

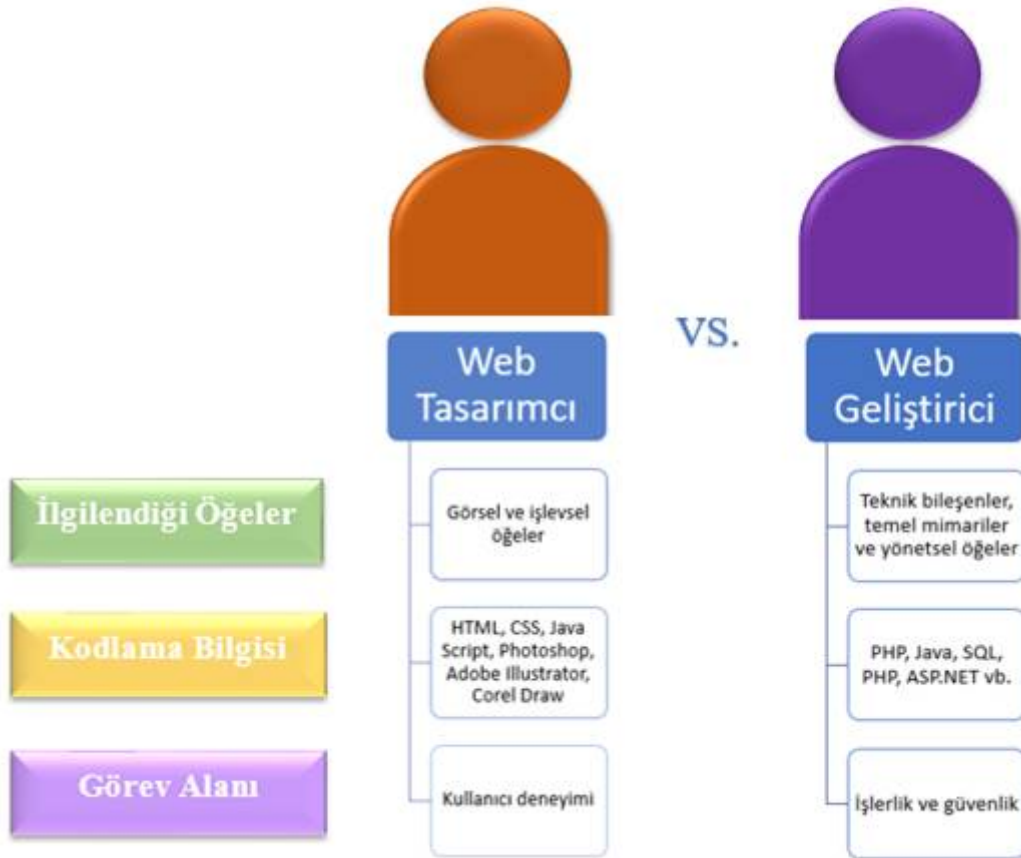
- Tasarımcılar sitelerin görsel ve işlevsel tasarımı ile ilgilenirken, geliştiriciler tasarım ile koordineli olarak tüm yapının mimarisini ortaya koyarak gerçekleştirilmesini sağlar.

- Geliştiricinin görev kapsamında yazılımsal olarak sitenin tüm dinamik yapılarının sürekli olarak bakımını yürütmek, güncellemeleri gerçekleştirmek, oluşan güvenlik açıkları ve risklere karşı önlemler üretmek durumundadır. Tasarımcının sitenin yayına girmesinden sonra site için bir geliştirici kadar güncelleme ve bakım yapmasına gerek kalmaz. Müşterinin zaman içerisinde değişen taleplerine göre yeni düzenlemeler,

tasarımsal eklemeler yapacak çeşitli tarayıcı uyumsuzluklarını düzeltecek olsa da bu süreç yazılım bakım ve güncellemesi kadar sık tekrar etmeyebilir.

- Geliştiriciler web programlama teknolojileri konusunda yani programlama ağırlıklı teknik eğitime gereksinim duyarken, tasarımcılar tasarım teknoloji ve dillerine odaklanır. Geliştirici ve tasarımcıların görevleri doğrultusunda kullandıkları yazılımlar ve programlama dilleri farklılaşmaktadır. Yukarıda yapılan açıklamalarda geliştiriciler ve tasarımcılar için önemli olan yazılım ve diller belirtilmiştir.

- Tasarımcılar son kullanıcının kullanım deneyimini arttırmaya yönelik çeşitli işlevleri belirleyerek site tasarımına entegre etmelidir. Geliştiricinin de bu noktada daha geri planda kaldığını ve tasarımcıdan gelen işlevlere göre site yapısını oluşturduğu ifade edilebilir. Bu noktada geliştiricinin görevi işlevlerin sorunsuz çalışır olmasıdır.

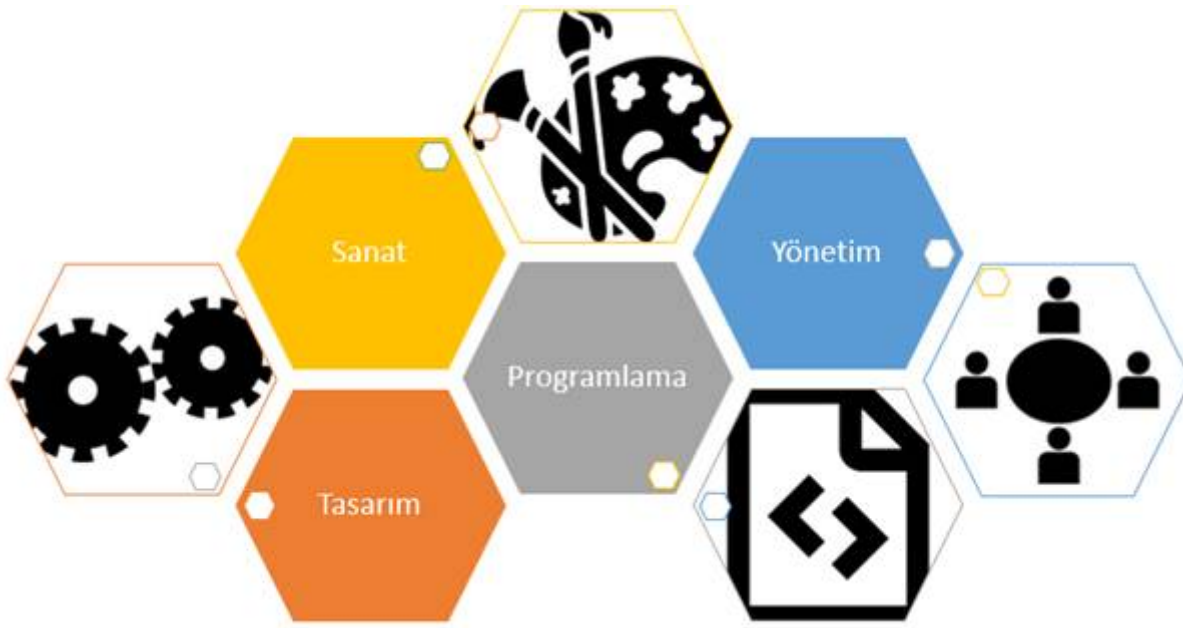


Şekil 9. Web tasarımcı ve geliştirici arasındaki temel farklar

Bir web sitesinin profesyonel anlamda bir ekip ile geliştirilmesi konusunda görev tanımları yukarıda paylaşılmış olsa da örneğin kişisel sayfalar gibi daha ufak çaplı web sitesi geliştirme işlerinde farklı görev ve sorumlulukları aynı kişiler yerine getirebilir, hatta baştan sona bu işi tek bir kişi de yönetebilir. Tasarım dillerinin yanı sıra kodlama becerileri de olan ve farklı programlama dillerine hâkim kişiler de olabilir. Bu durum bilişim ile ilgili farklı iş kollarında da görülebilmektedir. Ancak kullanıcı deneyiminin önemsendiği, büyük çaplı veri tabanları ile çalışılan, güvenlik ile ilgili önemli riskler oluşabilecek, büyük ölçekli firmalar tarafından kullanılacak, sitenin yönetim panelinde fazla sayıda personelin görev alacağı web sitesi geliştirme projesinde elbette ki tek ya da az sayıda kişinin çalışması beklenemez. Çünkü daha önce de belirtildiği üzere web sayfasının sahip olması gereken fonksiyonlar için gerekli tüm teknolojilerin kullanılır olması gerekmektedir. Bu doğrultuda bu alanla ilişkili olarak eğitim alan ve profesyonel anlamda bir meslek olarak bu işi gerçekleştirecek kişilerin de tasarım, yazılım geliştirme, kodlama vb. birçok farklı çalışma alanı içerisinde belirli bir alanda çalışma hedefi ile yola çıkması ve bu alanda da kendini geliştirerek işin gerektirdiği tüm teknik bilgi ve teknolojilere hâkim olması oldukça önemlidir.

## 2.3. Web Geliştirme Süreci ve Web Tasarımın Süreçteki Yeri

Daha önce de bahsedildiği gibi profesyonel anlamda bir web sitesi geliştirmek proje geliştirme ve yönetimi yaklaşımı ile ele alınması gereken bir iştir. Bu noktada, öncelikle hem web sitesi geliştirme projesinde ihtiyaç duyulacak yeteneklerin neler olduğu hem de bu sürecin nasıl işlediği konusunda bilgileri belirtmekte fayda vardır.



Şekil 10. Web sitesi geliştirmek için temel beceriler

Öncelikle bir web sitesi geliştirmek için gerekli olan en önemli yetenekleri ele alalım. Bu yetenekler tasarım, programlama, sanat ve yönetim yetenekleri şeklinde sayılabilir. Tasarım yeteneği, web sitesi içerisindeki sayfaların ve de tüm site içeriğinin organizasyonunu başarıyla gerçekleştirebilmek, kullanıcı deneyimini iyileştirebilmek için oldukça önemlidir. Programlama yeteneği, tasarımın aktif bir web sitesine dönüştürülebilmesini sağlar. Tüm fonksiyonların eksiksiz ve başarılı bir şekilde çalıştırılabilmesi, site-veri tabanı-sunucu-yönetim paneli yapısının hatasız bir şekilde işleyebilmesi, güvenlik açıklarının olmaması, olası risklere karşı alınacak önlemlerin yeterli olması için önem teşkil eder. Sanat yeteneği, kullanıcının ilgisini çekecek görsellerin tasarlanabilmesi, kullanıcı deneyimini iyileştirecek renk uyumlarının belirlenebilmesi, metinlerin büyüklük, yazı biçimi gibi özelliklerini yine uyumlu ve etkin bir şekilde karşılayabilmek için gerekli bakış açısına sahip olmayı gerektirir. Yönetim yeteneği ise, tüm iş sürecinin sorunsuz bir şekilde gerçekleştirilmesini, müşterinin talep ve beklentilerinin doğru bir şekilde alınarak, ekibin de bu doğrultuda aktif bir şekilde idare edilmesini, diğer bir ifade ile müşteri ve ekip arasında köprü kurulması ve doğru bilgi akışının sağlanmasını, altyapı, bütçe ve diğer olanakların web sitesi geliştirme sürecinde etkin bir şekilde kullanılmasının sağlanması, web sitesinin yayına alındıktan sonra da güncel kalabilmesi ve etkin içerik yönetimi gibi durumlar için gereklidir. Tüm bu yeteneklerin elbet tek bir kişide bulunması mümkün değildir. Bölüm başında da ifade edildiği üzere web geliştirme süreci bir ekip işidir. Dolayısıyla web tasarım, web programlama, bilgisayar programlama gibi eğitimler alan insanlar profesyonel iş hayatlarında ilgi duydukları ya da daha yetenekli oldukları konulara göre bahsi geçen bu yetenekler doğrultusunda kendilerini yetiştirebilir, belirli bir alana yönelik olarak web geliştirme projelerinde görev alabilir. Unutulmamalıdır ki, günümüzde bilgi teknolojileri son hızla gelişirken her bir iş için programlama dillerinden tasarım araçlarına kadar birçok farklı seçenek sunulmaktadır. Her bir teknolojiye, programlama diline, tasarım aracına vb. diğer tüm teknolojik gelişmelere tam anlamıyla hâkim olmak mümkün değildir. Dolayısıyla kariyerine web geliştirici olarak devam etmek isteyenlerin web geliştirme altında hangi alt faaliyette görev almak istediğini belirlemesi ve buna göre kariyer planını oluşturması önem arz etmektedir. Dikkat edilirse web tasarımcı ya da web geliştirici olarak görev alırken dahi farklı alt alanlarda uzmanlaşmak mümkün olabilmektedir. Örneğin sadece web sitesinin görünür yüzündeki kullanıcı ekranlarına yönelik olarak tasarım yapmak isteyenler bu süreçte ihtiyaç duyulan HMTL, CSS, JavaScript konularında kodlama yeteneklerini geliştirmeli, tasarlanan ara yüzlerin kullanıcı etkileşimi ile ilgili çalışmak yapmak istediğinde ise insan-bilgisayar etkileşimi alanı altında çalışılan kullanıcı deneyimine yönelik çeşitli testler ve analizler konusunda bilgi sahibi olmalıdır. Dinamik yani üzerinden online ticaret gibi çeşitli faaliyet yürütülen web sitelerinin geliştirilmesi bir diğer ifade ile site- veri tabanı-sunucu üzerinde yürütülen işlemler için sitenin teknik altyapısının oluşturulması, bu yapının kodlanması yani web programlama tarafında çalışacak kişilerin

ise özellikle veri tabanı yönetimi ve programlama konularında olmak üzere SQL, PHP gibi sorgulama ve programlama dillerine hâkim olması gerekmektedir.

Web geliştirme sürecinde ekip ve yetenekler konusunda verilen bu bilgilerden sonra web geliştirme sürecinin adımlarının ne olduğu ve özellikle dersin konusu olan web tasarımın bu süreç içerisinde nerede olduğu ve hangi alt iş faaliyetlerini kapsadığı açıklanmalıdır. Web tasarım, web geliştirme sürecinin şüphesiz ki en önemli adımlarından birisidir. Genel olarak düşünüldüğünde bir iş akışı içerisinde ilk basamaklarda yer alan faaliyetlerin işin doğru bir şekilde ilerlemesi bakımından genellikle daha önemli olduğu söylenebilir. Doğru planlanmamış veya tasarlanmamış bir yapının gerçekleştirilmesi geri dönüşü olmayan sorunlara yol açabilir. Örneğin web tasarımdan bağımsız olarak bir inşaat projesi hazırladığımızı varsayalım. Bu durumda zeminin yapısı, inşaat alanının büyüklüğü, talepte bulunanların istekleri, yapı kullanım esasları projenin gerçekleştirme aşamasına geçmeden önce belirlenen temel unsurlardır. Tüm girdilere göre hem hesaplama tarafında hem de kullanım alanlarının tasarımı tarafında gerekli işlemler yapılır, binalar bu hesaplama ve tasarıma göre inşa edilirler. Dolayısıyla doğru yapılmamış bir planlama ve tasarım detaylarına harfiyen uyularak bir bina inşa edilse bile kullanıma ya da zemine uygun olmama, yeterli dayanıklılıkta olmama gibi sorunlara ve istenmeyen sonuçlara neden olacaktır. Başka bir açıdan yine yanlış yapılan bir planlama ve tasarımı ele alalım. Yanlış tasarım söz konusu ise buna göre yanlış bir geliştirme yapılacağı aşıkardır. Tıpkı bir kartopu gibi tasarım aşamasında gerçekleştirilen bir hata göze gözükmese ya da fark edilip önemsenmese bile bu hatalı tasarıma göre gerçekleştirilen web sitesi de çok sayıda hata barındıracak ve sonuna gelindiğinde fark edilemeyen ya da göz ardı edilen o küçük bir hata ilerleyen adımlarda tüm yanlışların üst üste binmesiyle koca bir kartopuna dönüşecektir. Tasarım aşamasında yapılan hata bir bakıma tüm projenin çöp olması demektir. Çünkü bu aşamada yaptığınız hataları telafi etmek için gerçekleştirme aşamasında yaptıklarını yok sayarak tekrar bu adıma dönmeniz ve tüm tasarımları yeniden şekillendirmeniz gerekmektedir. Bu nedenledir ki hem zaman hem iş gücü hem de bunlara bağlı maddi kayıplar göz önünde bulundurulduğunda proje planı ve web tasarım aşamasının çok büyük bir özenle hazırlanması gerektiği açıktır.

Bu noktada sadece web tasarım adımını değil web geliştirme sürecinin bütünü ile ele almakta fayda vardır. Web geliştirme süreci çeşitli kaynaklarda farklı şekillerde ifade edilebilmektedir. Her ne kadar farklı ifade edilme şekilleri olsa da aslında web geliştirme sürecinin temelde beş adımı olduğu söylenebilir. Eklenecek diğer tüm adımlar genellikle bu ana adımların genişletilmesi, detaylandırılması ile elde edilmektedir. Bu adımlar;

- Amacın belirlenmesi,
- Tasarımın gerçekleştirilmesi,
- Tasarıma uygun olarak web sitesinin geliştirilmesi,
- Geliştirilen web sitesinin test edilmesi,
- Web sitesinin yayına alınması ve gerekli güncelleştirmelerin yapılması

şeklinde sıralanabilir.



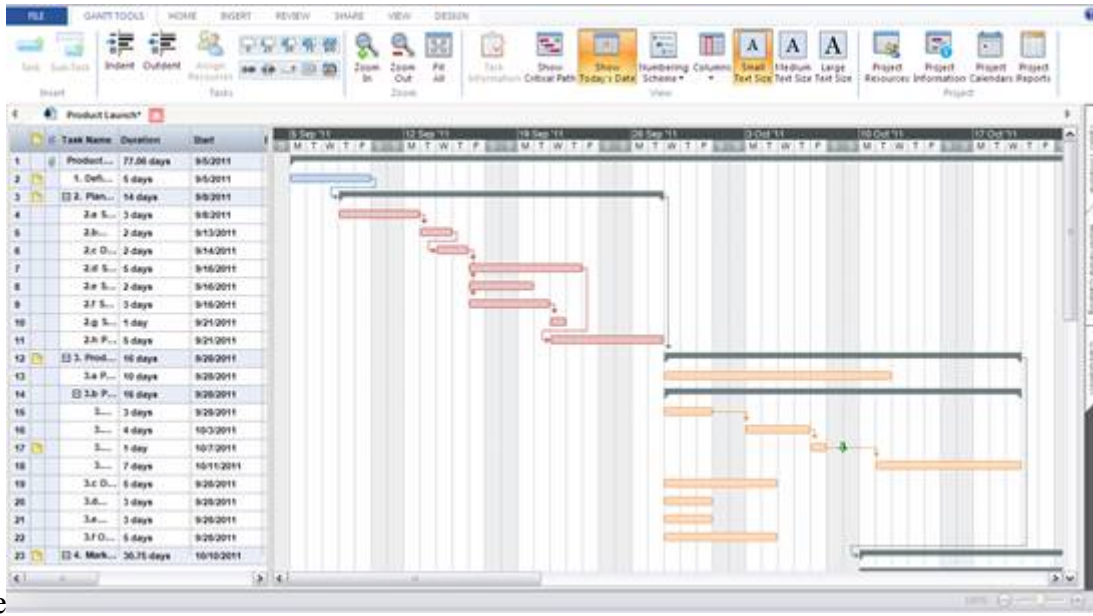
Şekil 11. Web geliştirme süreci adımları

Projeler ana iş paketlerinden ve bu iş paketleri altındaki faaliyetlerden meydana gelmektedir. Dikkat edilirse bu kitap boyunca web geliştirme sürecinin bir proje olarak ele alınması gerektiği birçok yerde belirtilmektedir. Bu nedenle, yine aynı bakış açısıyla Şekil 10'da belirtilen her adım web geliştirme sürecinin bir ana iş paketini belirtmektedir. Bu iş paketleri de kendi alt iş faaliyetlerinden meydana gelir. Her bir iş paketinin açıklamaları aşağıda verilmiştir.



*Amacın belirlenmesi*, görüldüğü üzere bu süreçteki ilk adım, ilk ana iş paketidir. Bu adımda geliştirilecek web sitesinin hangi amaçla kullanılacağı belirlenir. Amacın belirlenmesi bütün geliştirme sürecinin nasıl ilerleyeceği konusunda temeldir. Bu noktada yapılacak hata belki tasarım, geliştirme, test gibi teknik kısımlarda bir problem oluşmasına neden olmasa bile web sitesinin yayınlanması ile birlikte sorunların ortaya çıkacaktır. Geliştirilecek web sitesi öncelikle müşterinin hedeflerini karşılamalıdır. Bu nedenle bu aşamada gerçekleştirilecek fizibilite çalışmalarıyla müşterinin istekleri ve beklentileri alınmalı, neyi hedefledikleri belirlenmelidir. Diğer taraftan müşterinin yer aldığı sektör, bu sektör içerisindeki yeri, sektör özellikleri, müşterinin sektörde sunduğu ürün veya hizmetlerin neler olduğu gibi konular da ayrıca analiz edilmelidir. Tüm bu ön bilgiler ışığında müşterinin hedefleri ile istediği şekilde hazırlanacak bir web sitesinin tam anlamıyla örtüşüp örtüşmediği incelenmelidir. Eğer tutarsızlıklar söz konusu ise müşterinin bu anlamda bilgilendirilmesi, doğru yönlendirilmesi, detaylar konusunda da ortak bir mutabakata bu aşamada varılmalıdır. Aksi halde süre sonunda müşteri ve geliştirici taraf arasında sorunlar yaşanması muhtemeldir. Bu noktada özellikle ifade edilmelidir ki web sitesinin geliştirilmesini talep eden müşterinin her isteği de ana hedef doğrultusunda doğru olmayabilir. Bu nedenle müşterinin her isteği yerine getirilmelidir şeklinde bir algı oluşmamalıdır. Olası problemler müşteri ile paylaşılmalı farklı çözüm önerileri de sunulmalıdır. Yine müşterinin aklında olmayan ama hedefler doğrultusunda web sitesinde yer alması gerektiği düşünülen özellikler ya da içerikler de yine müşteri ile paylaşılmalıdır ki iş tamamlandığında hedeflerin daha yüksek kazanımlar ile gerçekleştirilmesi sağlansın. Belirlenen iş hedefleri ile birlikte hedef kullanıcı kitlesi de iyi analiz edilmelidir. Her kullanıcının bilgi teknolojilerini kullanım becerisi, ilgisi farklı olabilir. Örneğin üniversite öğrencileri için hazırlanacak bir web sitesi ile ileri yaş grubuna yönelik hazırlanacak web sitesinin farklı özelliklerde olması gerektiği açıktır. İleri aş gurubunda daha kolay kullanım amaçlanırken, genç grup için görselliğin ön planda olduğu ilgi çekici bir web sitesi hedeflenebilir. Dolayısıyla farklı hedef grupları farklı tasarımlar, farklı işlevsel özellikler gerektirmektedir. Hedefler ile ilgili detaylardan sonra kullanılması öngörülen teknolojik detayların da netleştirilmesi gerekmektedir. Örneğin dinamik yapıda oluşturulacak bir web sitesi ile statik yapıdaki bir web sitesinin geliştirilmesinde kullanılacak programlama dillerinde farklılıklar oluşabilir. Bu noktada programlama kabiliyeti açısından ekip farklılık göstereceğinden belirlenen iş için kimlerin görevlendirileceği hususunda bu belirleme önemlidir. Yine teknik detayların belirlenmesi işi bitirme süresini de etkileyebilir. Diğer taraftan web sitesi için gerekli olan grafik programları, sayfanın yayınlanabilmesi için destek alınacak alan adı sağlayan firmalar, web sitesinin alan adının ne olabileceği gibi birçok detay da yine bu aşamada ele alınır. Bu basamak hedeflerle birlikte ilerleyen aşamalarda tüm ana iş paketleri ve faaliyetlerinin de detaylarıyla planlandığı adımdır. Dolayısıyla bu adım hem hedeflerin belirlenmesi hem de tüm sürecin planlama adımı şeklinde düşünülmelidir. Proje yönetimi bakış açısı ile ele alarak toparlayacak olursak, hedefler doğrultusunda işi tamamlamak için bütçe oluşturma, iş gücü yani ekibin organizasyonu, zaman çizelgesinin detayları ile çıkarılması yani her iş faaliyetinin başlangıç bitiş zamanları ve esneklik süreleri, gerekli teknik alt yapı ile araçlar ve bunların iç ya da dış kaynaklardan temin edilip edilemeyeceğinin belirlenmesi, dışarıdan temin edilecekse nasıl gerçekleştirileceği, her iş paketinden beklenen çıktılar neler olacağının belirlenmesi gibi tüm bu detaylar bu adımda belirlenmektedir. Projelerde iş-zaman planlaması için Gantt diyagramları kullanılmaktadır. Web geliştirme sürecinde de benzer olarak “hedef” adımı bu diyagramlardan yararlanılabilir. Şekil 11’de örnek bir Gantt diyagramı görülmektedir. Microsoft Office uygulamalarından aşık olacağı üzere üst bölümde kullanılabilecek tüm araçlar, alt kısımda ise ana çalışma alanı yer almaktadır. Bu alan da ikiye ayrılmış sol bölümde iş paketleri ismi, işin başlangıç zamanları, işi tamamlama süreleri yer almaktadır. Bu kısımda ayrıca ekip üyeleri de belirtilebilir. Sağ bölümde ise yatakları renkli çubuklarla girilen sürelerle ve tarihlere göre işlerin hangi sırada gideceği, paralel yürütülecek faaliyetler, işin tamamlanma süreleri gibi bilgiler yer almaktadır. Bu ve benzeri diyagramlar işin akışı hakkında fikir sahibi olmak ve sonrasında da işi takip etmek açısından oldukça önemlidir. Dolayısıyla web geliştirme süreçlerinde çalışacak kişilerin bu tarz araçları tam olarak kullanmayı bilmese bile bu araçlar hakkında genel anlamda fikir sahibi olması gerekmektedir.





e

Şekil 12. Örnek Gantt diyagramı

*Tasarım* aşaması, ilk adımda toplanıp analiz edilen bilgiler doğrultusunda elde edilen sonuçlara, belirlenen müşteri hedefleri ve hedef kullanıcı kitlesine hitap eden, hazırlanan plana bağlı olarak web sitelerinin tasarımının gerçekleştirilmesini içermektedir. Tasarım kısmı da aslında farklı alt tasarım süreçlerine bölünebilir. Örneğin, içerik kısmında kullanılacak nesneler ve görsel çekiciliğe yönelik tasarım çalışmaları bir alt iş paketi olarak gerçekleştirilebilir. Bu noktada, görsel tasarıma yönelik olarak tanıtım alanları (banner), logo, grafikler tasarlanabilir. Hatta site içerisinde yer alacak metinsel (varsa dokümanların/dosyaların (pdf, video, formlar vb.) belirlenmesi) ve işitsel öğeler de hazırlanmaya başlanabilir. Yine renk uyumları ve metinlerin şekilsel yapıları tasarlanabilir. Başka bir iş paketi olarak ise site ana çerçevesinin oluşturulması, buna bağlı olarak öğelerin birbirlerine göre yerleşimleri, sayfa genel yerleşimleri ve gerekli işlevlerin düzenlenmesi, gerek duyulan sayfaların, site haritasının yapılandırılması hem bilgisayar hem mobil cihaz kullanıcıları için gerekli olan ara yüzlerin tasarlanması gerçekleştirilebilir. Tüm bu tasarım süreci elbette kâğıt üzerinde, yukarıda bahsi geçenler hususunda kararlar alıp ve bu kararları notlar halinde tutmayla sınırlı değildir. Tasarımcı(lar) web sitesi ile ilgili belirledikleri detaylar doğrultusunda web sayfalarına ilişkin görüntülerin çizimlerini gerek kâğıt üzerinde gerekse çeşitli çizim ve tasarım araçları vasıtasıyla bilgisayarda gerçekleştirirler. Bu çizimler üzerinde düzenlemeler yapar, ölçeklendirme konusunda da ilgili çalışmaları ortaya koyarlar. Bu noktada yukarıda bahsedilen sanatsal yetenek kısmının devreye girdiği söylenebilir. Çünkü bu çizimler üzerinde de tasarımcılar (özellikle görsel tasarımcılar) sanatsal yeteneklerini konuşturur ve farklı seçenekler de ortaya sunabilirler. Bu doğrultuda, müşterinin beğenisine sunmak üzere birden fazla tasarım gerçekleştirilmektedir. Kullanıcı ara yüzlerinin tasarımı konusu ile birlikte günümüzde gittikçe önemli hale gelen faaliyetleri kapsayan bir alt iş paketinden bahsetmek daha mümkündür. Bu iş paketi tasarımların kullanıcı deneyimi açısından analiz edilmesi ve buna bağlı tasarımların iyileştirilmesini kapsamaktadır. Bu çalışmalar insan-bilgisayar etkileşimi kapsamında gerçekleştirilmektedir. İsimden de anlaşılacağı üzere bu alan insanlar ile bilgisayarlar arasında etkileşime ilişkin tüm çalışmaları kapsamaktadır. Arayüz, kullanıcının bir bilgisayar sisteminde donanım ve yazılımsal birimleri kullanabilmesi ve işlem yapabilmesi için makine dilinde kod ezberlemesine gerek kalmadan kullanıcı ile sistemin birbiri ile iletişime geçtiği yazılım/donanım birimidir. Kullandığımız bilgisayarlar, telefon ve tabletlerde, ATM ekranlarında, toplu taşıma kart dolum istasyonlarında vb birçok yerde bu arayüzler ile etkileşime geçmekteyiz. Bu noktada ekranlar da elbet bir arayüz olarak düşünülebilir ancak bu ekranlardan ziyade web tasarım tarafını da ilgilendiren şekilde bu ekranlar üzerinden görmüş olduğumuz grafiksel ve görsel tasarımlar daha özel olarak grafiksel kullanıcı arayüzler olarak ifade edilmektedir. Fare, klavye, monitör gibi donanımsal bileşenler vasıtasıyla kullanıcının etkileşimde bulunduğu simgeler, pencereler, düğmeler, hipermetinler (*hypertext*) ve panellerin bütünü genel olarak grafiksel kullanıcı arayüzü ( *GUI - Graphical User Interface*) olarak ifade edilmektedir (Sakman, 2017). Kullanıcı ara yüzünde kullanılan bu simgeler, metinler vasıtasıyla kullanıcılar bilgisayarlara çeşitli komutlar göndermekte, bilgisayarların verdiği cevaplar da bu ara yüzler vasıtasıyla anlayacağı şekilde kullanıcıya yansıtılmaktadır. Genel olarak bu ara yüzlerdeki kullanıcıların hareketleri ve tercihleri bilişsel ve kişisel eğilimlere göre değişkenlik göstermektedir. Bu nedenle kişilerin hareket ve tercihleri incelenerek buna bağlı olarak kullanıcıların genelini ilgilendiren bir trend belirlenebiliyorsa kullanıcı grafik ara yüzleri bu sonuçlara göre

tasarlanabilmektedir. Bir web sayfası veya uygulama ara yüzünde bölüm boyunca ifade edildiği gibi metin, logo, grafik, görsel gibi birçok öğe yer almaktadır. Dolayısıyla kullanıcının etkileşimde bulunduğu tüm bu öğeler sebebiyle web sayfaları genel kalitesinin etkilendiği karmaşık bir yapıdan söz edilebilir. Bu doğrultuda tasarım ilkeleri ile birlikte, ara yüzlerin mesaj, hizmet veya bilginin kullanıcıya için kullanıcının dikkatini çekecek şekilde tasarlanması oldukça önemlidir. Dikkat çekmek bir bakıma kullanıcıyı istenilen alana yönlendirme olarak da ifade edilebilir. Bu noktada yönlendirme bir ses efekti ile, bir görsel öğe ile gerçekleştirilebilir. Kullanıcı deneyimi, en genel anlamda bir kişinin bir hizmete veya ürüne karşı verdiği duygusal tepki olarak ifade edilebilir. Bu noktada, tasarımcıların kullanıcı beklentileri karşılayarak, kullanıcı sayısını artıran kullanıcı dostu, akılda kalıcılığı arttıran bir başka ifade ile kullanıcı deneyimini iyileştiren tasarımlar üretmelidir (Türkmenoğlu ve Atalar, 2020). Bu noktada kullanıcı deneyimi analizleri web tasarım açısından geliştirilen bir tasarımın kullanıcıların ihtiyaçlarını karşılama kapasitesini, kullanılabilir olup olmadığını çeşitli anketler, görüşmeler hatta cihazlar vasıtasıyla denek kullanıcılardan elde edilen verilerin analiz edilmesini kapsamaktadır. Günümüzde özellikle farklı çeşitlerde göz izleme cihazları ve yine hareketleri kayıt altına alan yazılımlar bu çalışmalarda sıklıkla kullanılmaktadır. Bu noktada geliştirilmemiş bir sitenin kullanıcı deneyiminin nasıl ölçüleceği akıllara gelebilir. Bu noktada, prototip oluşturma bu analizlerde önemli olduğu ve oluşturulan prototipler üzerinden analizlerin yapıldığı ifade edilebilir. Özetle, kullanıcı deneyimi analizi de tasarlama adımıyla gerçekleştirilecek önemli alt iş paketlerinden biridir. Profesyonel web siteleri tasarımlarında artık sıklıkla gerçekleştirilen analizlerdir. Ancak bilinmelidir ki yayına alınmış bir sitenin de sonradan kullanıcı deneyimi analizleri yapılabilir ve mümkün olan düzenlemeler var olan yapı üzerinde gerçekleştirilebilir. Dolayısıyla, bu iş paketi aynı zamanda web tasarım sürecinin yayın, güncelleme ve bakım aşamasında da gerçekleştirilebilir. Bir diğer alt iş paketi ise temel web tasarım dili olan HTML ve CSS kullanılarak tasarlanan site yapısının ve site içerisinde yer alacak tüm alt sayfaların bilgisayar ortamında hazır hale getirilmesidir. Gerçekleştirme adımına geçilmeden önce sayfa tasarımları bu şekilde hazır hale getirilmiş olacaktır. Sayfalarda yer alacak dinamik yapıların entegrasyonu ise geliştirme adımıyla bu tasarımlar üzerinden yapılacaktır.

*Geliştirme* aşaması ise, tasarım adımıyla gelen müşterinin onaylamış olduğu şablon üzerinden sitenin tüm bileşenlerinin, veri tabanının, sunucu bağlantılarının, gerekli tüm yönetim panellerinin belirlenen programlama dili ile yazılmasını kapsamaktadır. Programlama konusunda bilgi sahibi olanlar, kodlar ve kodların çıktılarının ayrı görüntülere sahip olduğunu bileceklerdir. Bu konuda bilgi sahibi olmayanlar için ise şu örnek verilebilir. Tarayıcınız ile bir web sayfasını görüntülerken klavyede F12 tuşuna bastığınızda bazı anlamlandıramadığımız ifadelerin ekrana geldiğini görebilirsiniz. İşte sizin sayfalar içerisinde gördüğünüz resimler, metinler, butonlar, bu öğelerin yerleştiği konumlar, renkler, büyüklükler vb. her şey aslında bu gördüğünüz ifadelerin karşılığıdır. Bilgisayarlara çeşitli görevler verirken onların anlayacağı bir dil ile bunu gerçekleştirmekteyiz. Bilgisayarların bu görevleri gerçekleştirmesi için kullanılan diller programlama dilleri olarak ifade edilirken, günümüzde birçok farklı programlama dilinin varlığı söz konusudur. Web siteleri geliştirilirken de tüm bileşenler işte bu şekilde bilgisayarın anlayacağı bir dil ile kodlanır. Bilgisayarlar gördükleri bu kodları okur, işler ve sonrasında da son kullanıcıların gördüğü şekliyle bir ara yüze dönüştürerek sunar. Web siteleri geliştirilirken iki farklı açıdan durum incelemesi yapmak gerekir. Web siteleri dinamik ve statik olmak üzere iki farklı şekilde olabilir. Dolayısıyla bu iki farklı yapıdaki web sitesinin geliştirme aşamasında değişiklikler söz konusudur. Statik web siteleri herhangi bir veri tabanı ile bağlantısı olmayan, içerisinde barındıracağı bütün görsel, metinsel, işitsel öğelerin HTML kodları vasıtasıyla tasarlanan web sayfaları içerisinde istenilen yerlere konumlandırıldığı, güncelleme gereği duyulmayan ya da daha seyrek duyulan, değişimi gerekli olmadığı sürece yapılmayan, dolayısıyla güncelleme ve değiştirme için bir yönelim paneline ihtiyaç duymayan, bunun yerine kaynak kodların yer aldığı dosyaların FTP ile bir program aracılığı ile sayfaların yayınlandığı sunucuya aktarıldığı sitelerdir. Dinamik web siteleri ise statik web sitelerin tam tersi olarak düşünebiliriz. Bu web siteleri bir PHP, C# gibi programlama dili ile kodlanan, kullanıcı etkileşiminin olduğu yani butonlar ile program kodları içerisinde tanımlanmış çeşitli fonksiyonları çalıştıran, veri girişleri ile veri tabanına veri gönderen ya da tam tersi yönde veri çeken, dolayısıyla veri tabanı bağlantısı olan, farklı kullanıcı rollerine göre yönetim panelinin yer aldığı sitelerdir. Statik web sayfalarındaki gibi sayfalarda yapılacak içerik değişikliklerinde webde yayınlanmış kaynak kod dosyalarını FTP ile bir program vasıtasıyla güncellemek yerine, yönetim paneli üzerinden herhangi bir kodlama bilgisi olmadan yapılabilmektedir. Bu site yapılarına örnek olarak e-ticaret siteleri verilebilir. Tüm bu farklılıklar göz önünde bulundurulduğunda geliştirme aşamasında her iki tipte web sitesi için yapılması gerekenlerin farklılaştığı açıktır. Statik web sitelerinin geliştirme aşaması dinamik sayfaların geliştirme aşamasına göre daha az faaliyet içermektedir. Statik web sitelerinde HTML kodları ile hazırlanmış ana ve diğer sayfa tasarımı, yine tasarlanan içeriğe yönelik diğer tüm nesneler kullanılarak web sitesine ilişkin tüm ana ve alt sayfalar hazır hale getirilir. Sayfalar arası bağlantılar gerçekleştirilir. Dinamik bir site için ise yine HTML

kodları ile hazırlanmış sayfa tasarımları kullanılır ancak dinamik yapının işleyişi için gerekli olan program kodlarının yer aldığı diğer dosyalar da hazırlanır. Web sitesinin ihtiyaç duyduğu veri tabanı tasarımı gerçekleştirilir. Veri tabanı-site-sunucu bağlantılarına yönelik kod dosyaları da ayrıca oluşturulur. Dinamik web sitelerinin geliştirilmesi hususu web programlama derslerinde detaylı olarak açıklanmaktadır. Bu ders kapsamında ise sayfa tasarımları ve buna ilişkin gerekli olan HTML konusu detaylı ele alınmaktadır.

*Test* aşamasında ise hazırlanan web sitesinin oluşturulan yapıya göre doğru bir şekilde işleyip işlemediği incelenmekte, son kontroller yapılmaktadır. Test aşaması testin planlanması, tasarımı, gerçekleştirilmesi, hataların belirlenmesi ve raporlanması şeklinde alt adımlardan meydana gelmektedir. Örneğin sıklıkla karşılaşılan problemlerden birisi ara yüzlerin uyum problemi olabilmektedir. Geliştirilen ara yüzler mobil ara yüzlerde hata verebilir ve web sitesi mobil araçlarda doğru bir şekilde görüntülenemeyebilir. Bu hatalar farklı tarayıcılarda görüntüleme sorunu olarak da karşımıza çıkabilir. Ya da sayfalar arası gezintide yanlış bağlantılar oluşturulmuş olabilir, bu da kırık linkler ya da site haritasında istenmeyen sayfalara yönlendirilmeye sebep olabilir. Site içerisinde farklı sayfalarda yer verilen tüm nesnelerde de hatalar söz konusu olabilir. Logoların istenilen büyüklük ve konumda görüntüleyememe, görsel nesnelere hatalı bağlantı verilmesi nedeniyle bu nesnelerin görüntülenememesi, metinlerde başlık ve metin stillerindeki hatalar, kaydırma çubuklarının oluşmaması ya da istenmeyen kaydırma çubuklarına sebebiyet veren ölçülendirmenin kullanılması olarak içerikteki hatalar da sayılabilir. Test aşaması sonucunda bu ve benzeri tüm hatalar belirlenir ve raporlanır. Sonrasında ise tüm düzeltmelerin yapılması gerekmektedir.

Yayına alma, güncelleme ve bakım ise bu süreçteki son adımdır. Hazırlanan web siteleri öncelikle test yayınına alınır. Test aşamasında geliştiriciler tarafından fark edilemeyen ancak kullanıcıların deneyimlerine bağlı olarak iletecekleri hatalar nedeniyle öncelikle bu test yayınına ihtiyaç duyulmaktadır. Bu aşamada da iletilen hatalar yine düzeltilmektedir. Yayına alma sırasında elbette ki sunucu hizmeti alınan firma ile ortak hareket edilmektedir. Web sitesinin yayınlanması için ayrılan alana web sitesine ilişkin tüm dosyalar aktarılır. Dosya aktarımı sırasında FTP kullanılır. Dosyalardan kasıt hem web sitesi için hazırlanan sayfalara ait .html uzantılı dosyalar ve sayfalar içerisinde kullanılan tüm materyaller bu alana yüklenir. Site yapısının düzgün işlemesi için sistematik bir dosyalama yapısı kullanılmaktadır. Web sitelerinin yayına alınarak müşteriye temsil edilmesi ile büyük bir süreç tamamlansa bile siteye ilişkin yaşanacak problemlere karşı bakım hizmeti verilmektedir. Elbette ki çoğu zaman ömür boyu bakım ve güncelleme desteği vermek mümkün olamamaktadır. Keza müşterilerin talep ve ihtiyaçlarının sonsuz olabileceği düşünülürse bakım ve güncelleme esasları müşteri ve web sitesi geliştiricisi arasında imzalanacak bir sözleşmeye göre düzenlenir. Bakım ve güncelleme işlemleri için ilk yapılan anlaşmada geçen süre sona erdikten sonra müşteri talep ederse bakım ve güncelleme için sözleşme süresi uzatılabilir. Bu süre zarfında kullanıcı talepleri ve deneyimine, gelişen ihtiyaçlara, değişen içeriklere göre güncelleştirmeler gerçekleştirilebilir.

Tüm süreç incelendiğinde bazı hususlara dikkat çekilmelidir. Örneğin statik web sitelerinin tasarımı ve geliştirilmesi aşaması arasındaki farklar muğlak olabilir. Statik web sitesinde yer alan tüm sayfalar elbette ki birbirinden farklı içeriklerde olacaktır. Sayfa içerisinde yer alan metinler ve görseller bu duruma örnek verilebilir. Ancak sayfa ana yapıları, yani menünün nerde olacağı, metinlerin nerede yer alacağı, tanıtım alanının nerede konumlandırılacağı belirli ve aynıdır. Ana sayfa biraz farklılık gösterebileceğinden bir tasarım, farklı içeriklerin yer aldığı alt sayfalar bir tasarım olarak düşünülebilir ve tasarım aşamasında ana sayfa yapısı için ve alt sayfa yapısı için iki farklı tasarım hazırlanır (alt sayfalarda da farklı yapılar kullanılacağı zaman, örneğin iletişim formunun yer alacağı bir sayfa, bu tasarım sayıları artabilir). Gerçekleştirme aşamasında ise belirlenen site haritasına göre web sitesinde yer alması, sunucuya yüklenmesi gereken tüm web sayfaları tasarım sürecinde oluşturulan sayfa tasarımları kullanılarak hazırlanır. Bir diğer ifade ile tüm site sunucuya yükleme ve yayına almaya hazır hale getirilir. Dinamik sitelerde ise geliştirme sürecinin nasıl olduğu ilgili kısımda ifade edilmiştir. Bir diğer dikkat edilecek husus test ve yayına alma aşamalarına ilişkindir. Test yayını da zaman zaman test aşaması altında ifade edilebilir. Ancak bir yayına alma ve kullanıcıya açma durumu söz konusu olduğundan bu bölümde test yayını yayına alma aşamasında ifade edilmiştir. Kullanıcı deneyimi testleri ilgili kısımda da belirtildiği üzere yayına alma sürecinden sonra da gerçekleştirilebilir. Ancak elde edilen sonuçlar site tasarımını ilgilendirmektedir. İlgili iş paketi tasarım ya da yayın aşamalarında da konumlandırılabilir. Diğer taraftan statik sayfalarda kullanıcı-bilgisayar etkileşimi yoktur. Ancak sayfayı kullanırken kullanıcının hangi noktalara odaklandığı, site içerisinde arama yaparken talep ettiği içeriğe kolay ulaşım ulaşmadığı gibi konular yine bu bağlamda inceleme konusudur.

## 2.4. Web Tasarımın İlkeleri

Bu kısma kadar olan bölümde web tasarım ile ilgili teknolojiler, web sitesi geliştirme aşamasında görev alan ekip üyeleri ve web sitesi geliştirme süreci hakkında bilgi verildi. Bu temel bilgilerden sonra artık web tasarım detaylarına geçilebilir. Bu aşamada ilk akla gelecek sorular “Web sitesi tasarımına nereden başlamalıyız?”, “Web sitesi içeriği nasıl oluşturulmalıdır?”, “Ara yüzde yer alacak öğeler nasıl seçilmeli ve yerleşimi nasıl planlanmalıdır?”, “Tasarım çerçevesi oluşturulan web sitesinin bu detayları bir tarayıcıda görüntülenecek şekilde nasıl kodlanır ve kodlama için hangi diller kullanılmalıdır?” şeklinde sıralanabilir. Bu sorular arttırılabilir. Bu sorulara tek tek cevap vermek mümkündür. Ancak bilinmelidir ki web tasarımın bazı temel ilkeleri söz konusudur ve eğer bu ilkeler doğru bir şekilde anlaşılır ve uygulanabilirse zaten sorular ve sorunların çoğunluğu kendiliğinden çözüme kavuşma eğilimi gösterecektir. Web sitesi tasarlanırken dikkat edilecek temel tasarım ilkeleri bir bakıma herhangi bir görsel öğe tasarlanırken kullanılan sanatsal ve tasarım ilkeleri ile paralellik göstermektedir. Yani her ne kadar yapısal olarak farklı noktalarda yer alsalar da bir görsel öğe de bir web tasarımı da bir insanın gözüne hitap etme noktasında ortak noktada buluşmaktadır. Bu nedenle bahsi geçen bu ilkeler aslında çok da yabancı olduğumuz konular değildir.

*İçerik*, tasarım açısından en önemli ilkelerden birisidir. İçeriğe bağlı olarak tasarımda farklılıklara gitmek muhtemeldir. Web sitesi oluşturulduğunda aslında temel amaç yayınlanan içeriklerin kullanıcıların erişimine sunmaktır. Dolayısıyla yola çıkarken öncelikli olarak iş hedefi ve de buna bağlı hedef kitlesine uygun içeriklerin neler olduğu belirlenmelidir. İçeriği bilmek buna uygun görsel ve metinsel öğeleri tasarlayabilmek ve üretmektir. Web geliştirme sürecinde de hatırlanırsa web tasarımın bir alt iş paketi olarak bu ifade edilmişti. Görsel ve yaratıcılık anlamında oldukça dikkat çekici bir web sitesini tasarlamak ve geliştirmek ilk anda beğeni kazansa bile zaman içerisinde yetersiz içerikler kullanıcının kaybına neden olacaktır. Metinsel içeriklerin anlaşılır, uzun ve gereksiz cümlelerden arındırılmış olması gerekmektedir. Kopyalanmış ve orijinal olmayan metinler yerine web sitesine özgü farklı konularda insanların ihtiyaç duyduğu içerikler hazırlanmalıdır. Vurgulama yapılırken olumsuz etkiler bırakmayacak ölçüde sadelik esasına bağlı kalınmalıdır. Görsel öğeler ilgi çekici bile olsa gözü yormayacak renk tonlamalarına sahip olmalıdır. İçerikler arasında kullanıcının kaybolmaması ve yorulmaması gerekir. Yoğun öğelerden bu nedenle kaçınılmalıdır.

*Sayfa tasarımları*, web sayfalarında yer alacak nesnelerin konumlarının nerede olacağının belirlenmesini, site haritasının oluşturulmasını, ölçeklendirmeyi kapsamaktadır. Bu ilke doğrultusunda sayfa tasarlanırken sayfaların bütünlüğüne dikkat edilmelidir. Yani kullanıcı web sitesini gezdiği zaman bu gezintide sayfa tasarımları açısından bir bütün içerisinde hareket ettiğini hissetmeli, farklı web sitelerine yönlendiğini düşünmemelidir. Dolayısıyla hazırlanan sayfa tasarım yapısından çıkılmamaya özen gösterilmelidir. Elbette ki ana sayfa ve alt sayfa yapılarında değişikliklerin olması muhtemeldir. Ancak burada bahsedilen konu için şu örneği verelim. Üniversitede öğrenci olduğunuz bölümün web sitesine girdiğinizde, ana sayfanın kendine özgü bir yerleşimi olacaktır. Bu yerleşim düzeni alt sayfalara geçildikçe ufak farklılıklar gösterebilir. Ancak alt sayfalara geçtikçe renk düzenlerindeki farklılıklar, sayfa içerisindeki yerleşim düzenindeki farklılıklar, tanıtım alanlarındaki ya da logolardaki farklılıklar istenmeyen durumlardır. Bir menü düzeni söz konusu ise bu menü düzeni diğer sayfalarda da aynı olmalıdır ki kullanıcı ilk siteye giriş yaptığı anda aşinalık kazandığı görünüm her sayfa yapısında değişikliğe uğramasın. Örneğin İstanbul Üniversitesi ana sayfasına açtığınızda farklı bir yerleşim düzeni, haberler ve duyurular sayfalarını açtığınızda birbirine benzer ancak ana sayfadan daha farklı bir sayfa düzeni söz konusudur. Tüm sayfalardaki düzeni incelediğinizde renk tonlamalarının, logo, alt ve üst tanıtım alanlarının belirli yerlerde olduğunu göreceksiniz. *Şekil 12*'de İstanbul Üniversitesi web sitesinin ana sayfası görülmektedir. Sayfa incelendiğinde, sayfanın ilk açılışında üniversite logosunun etrafına iki yanlı dizilmiş menü bileşenlerini, hemen altında manşet haberlere ilişkin geniş bir alanı ve yine bu alanın altında alt menü görülmektedir. Bu aşamada tasarım, menü ve manşet haberlerin vurgulanmasına odaklanmıştır. Sayfanın geri kalan kısmı gözükmemekle birlikte, bu kısma erişebilmek için tarayıcının sağ tarafında yer alan kaydırma çubuğunu kullanmak gerekmektedir. Sayfada daha aşağıya indiğinizde *Şekil 13*'ten görüleceği üzere haberler ve duyurular alanına yer verilmiştir.



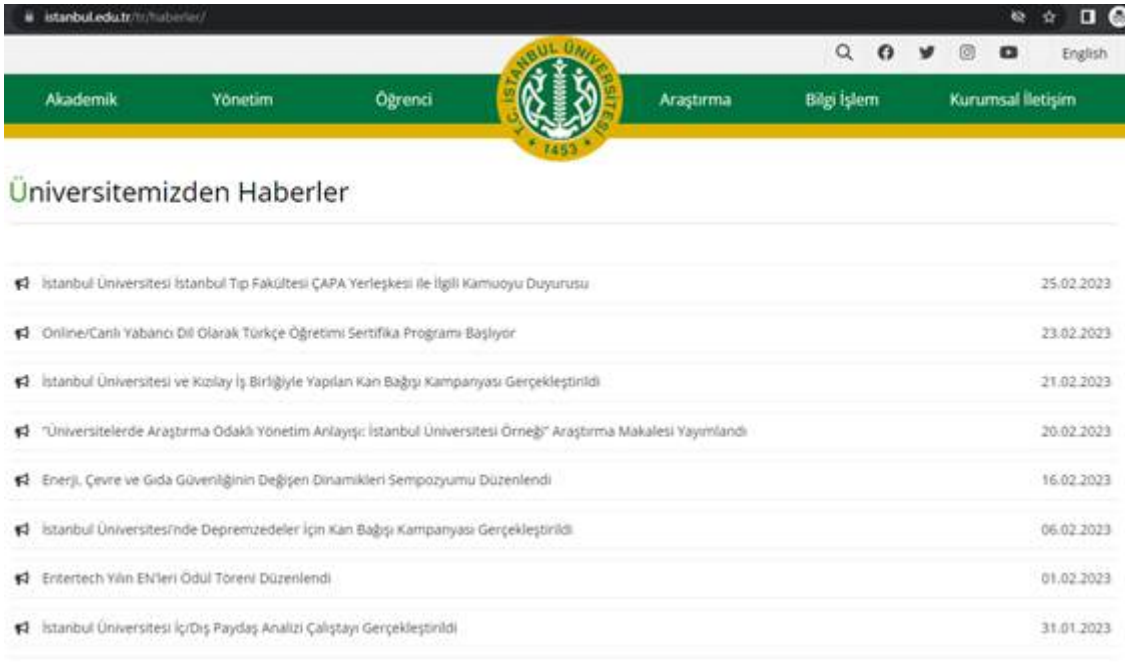
Şekil 13. İstanbul Üniversitesi web sitesinin ana sayfası-1



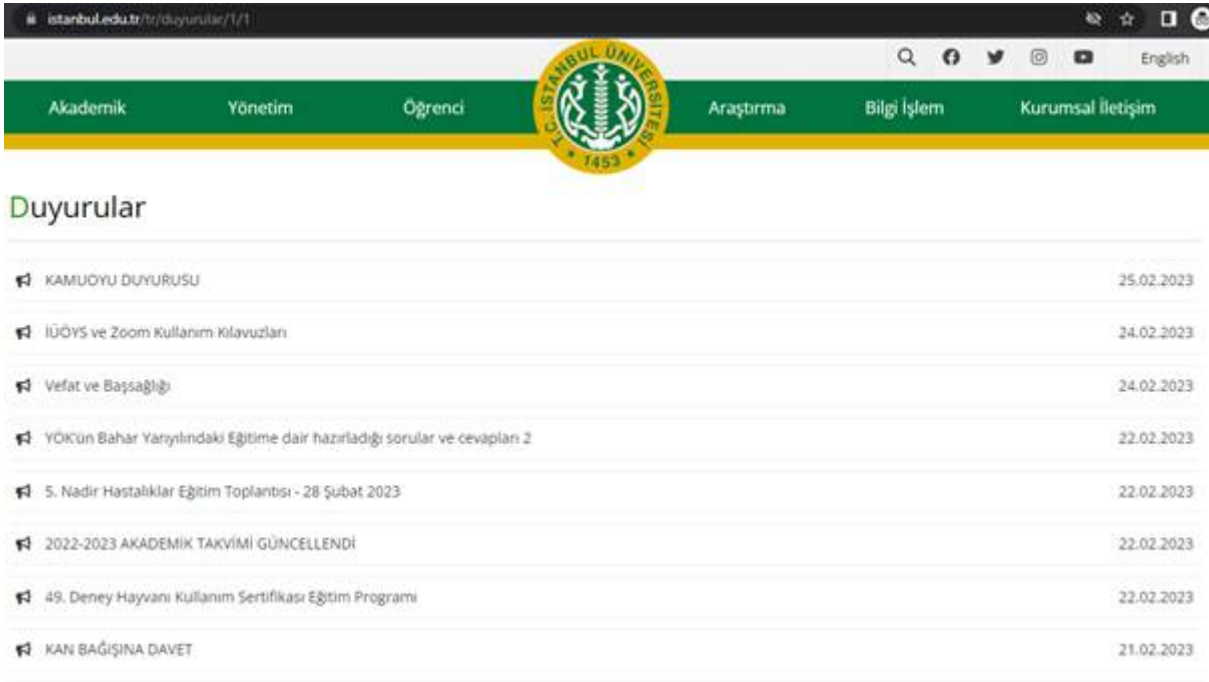
Şekil 14. İstanbul Üniversitesi web sitesinin ana sayfası-2

Bu noktada, tüm haberler ve tüm duyurular için hazırlanan alt sayfa tasarımları ise Şekil 14 ve Şekil 15'te yer almaktadır. Görüldüğü üzere benzer amaca hizmet eden sayfa tasarımları ana sayfa görsel tasarımından ve düzeninden tamamen koparılmadan devamlılık sağlanacak şekilde benzer olarak tasarlanmıştır. Genel yapı anlaşıldığında farklı sayfalarda da dolaşmak kullanıcı adına sorun yaratmamaktadır.



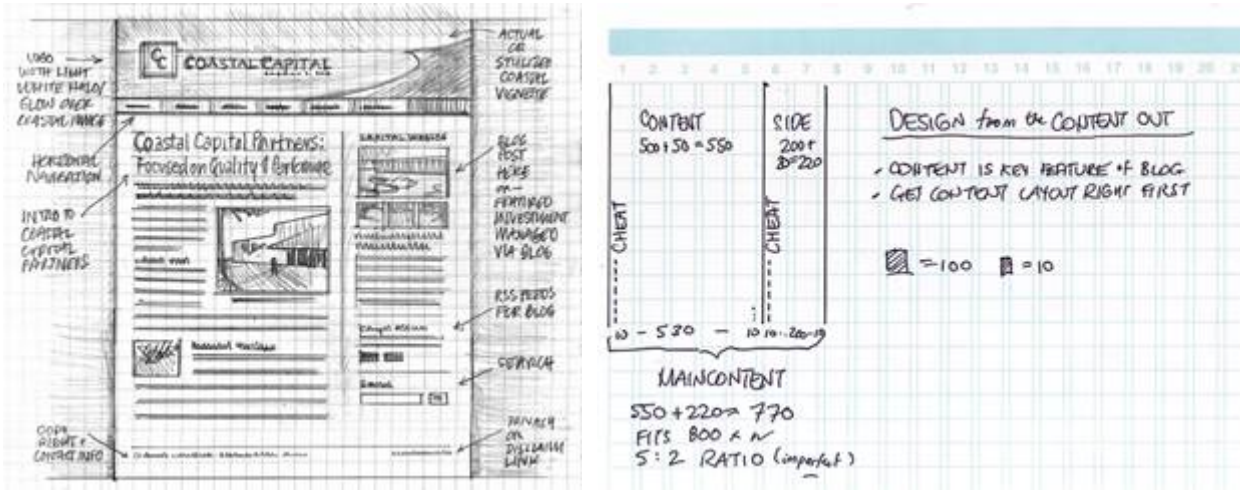


Şekil 15. İstanbul Üniversitesi web sitesinin tüm haberler sayfası



Şekil 16. İstanbul Üniversitesi web sitesinin tüm haberler sayfası

Sayfa tasarımlarında denge unsuru göz önünde bulundurulmalıdır. Yani her öge sayfanın aynı bölgesine yığılmamalı sayfa içerisinde belirli bir oran gözetilmelidir. Eğer bu denge kurulmaz ise kullanıcıda görsel açıdan rahatsızlık oluşabilir. Merkez, yatay ve dikey eksenler belirlenmeli buna göre yerleşim planlanmalıdır. Yine görseller arasında hiyerarşi ya da vurgulanmak istenen yere özgü bazı detaylar da bu tasarım ilkesi kapsamında değerlendirilebilir ve kullanıcının ilgisi yönlendirilebilir. Tasarım sırasında özellikle yerleşim planları çıkartılırken ana taslağın ister bir kâğıda ister bilgisayar ortamında bir uygulama vasıtasıyla hazırlanması oldukça önemlidir. Bu taslak üzerinde ana çerçeveyi görebilmek, gerekli düzenlemeleri ve tasarım önerilerini ortaya çıkarmak tasarımcı için oldukça önemlidir. Şekil 16'da web sayfaları için taslak olarak hazırlanan çizimler görülmektedir.



Şekil 17. Taslak web sayfası tasarımları

**İşlevsellik** özellikle site haritası ve sayfalar arası bağlantı konusunda tasarım ilkelerini kapsamaktadır. Bu doğrultuda site haritasının çok iyi bir şekilde yapılandırılmış olması gerekmektedir. Kullanıcı mümkün olan en kısa sürede ve en kolay şekilde aradığı içeriğe ulaşacak şekilde sayfa bağlantıları yapılandırılmış olmalıdır. Sayfalar arasında kopukluk olmamalıdır. Mümkün oldukça her sayfadan diğer tüm sayfalara erişilebilecek şekilde bağlantılar sağlanmalıdır. Bir sayfadan başka sayfaya geçebilmek için geri gelmeye ihtiyaç duyulmamalıdır. Sayfalarda yer alacak menülerde özellikle en sık erişilmeye çalışılan ve daha önem arz eden sayfalara erişim verilmelidir. Bu noktada fazla sayıda sayfa linki menüde konumlandırılarak karmaşıya sebebiyet verilmemelidir. Tüm bu nedenlerden ötürü site haritası da sayfa tasarımları gibi bir kâğıt üzerinde veya bilgisayar ortamında planlanmalı, site haritası üzerinde olası yollar ve olması gereken bağlantılar belirlenmelidir.

**Kullanılabilirlik** aslında web geliştirme sürecinin anlatıldığı kısımdan itibaren vurgulanan önemli bir konudur. Tasarım ilkeleri arasında da yer almaktadır. Daha önce de ifade edildiği gibi hedef kitleler farklı yaş gruplarına hitap ediyorsa bu kitlelerin bilgisayar kullanım alışkanlık ve yetenekleri göz önünde bulundurulmalıdır. Gereği olmadığı sürece kaydırma çubuklarının aktif hale gelmemesine dikkat edilmelidir. Hazırlanan tasarım, kullanıcı gözü ile değerlendirilmelidir. Yazı büyüklüklerinden, metin renklerine okunabilirliği arttıracak renk kombinasyonları tercih edilmelidir. Başlık büyüklükleri doğru bir şekilde ayarlanmalı, ana ve alt başlıklar arasındaki ayrımlar belirgin olmalıdır. Menüün sayfa içerisindeki konumu, menü kullanımını en iyileyecek şekilde belirlenmelidir. Sunucu taraflı olarak kullanıcının hızlı erişimine olanak sağlayacak paketler tercih edilmelidir. Bu durumda sayfaların açılması/yüklenmesi için geçirilen süre uzadıkça kullanıcı deneyimi kötüye gitmektedir. Belirli bir içeriğe ulaşmak isteyen kullanıcı göz önünde bulundurularak bir arama fonksiyonu (alanı) oluşturulup kullanıcının sayfalar arasında kaybolması engellenmelidir. Benzer şekilde kullanıcının web sitesini kullanımını iyileştirecek farklı birçok durum düşünülebilir.

**Tarayıcı uyumu** kullanıcıların sorun yaşadığı önemli tasarım problemlerinin başında gelmektedir. Bu nedenle farklı tarayıcılar ve farklı ekran tiplerinde görüntülenecek şekilde web sayfalarının hazırlanması gerekmektedir. Tarayıcılar farklı görüntüleme özelliklerini destekleyebileceğinden, tarayıcı açısından tüm tarayıcılarla uyumlu sayfalar tasarlamak önemlidir.

Bu ilkeler göz önünde bulundurulduğunda bu bölüm başlığı altında yer alan soruların çoğunluğunun yanıtlandığı görülecektir. Web geliştirme süreci takip edildiği ve bu ilkeler benimsendiği takdirde başarı ile hazırlanmış web siteleri hazırlamak çok da zor olmayacaktır.

## Bölüm Özeti

Bu bölüm kapsamında web geliştirme süreci ve bu süreç içerisinde web tasarıma ilişkin bazı detaylar ele alınmıştır. Web tasarımcısı olarak profesyonel anlamda görev yapacak kişiler aslında bu işi bir ekip içerisinde yürütecektir. Bu nedenle bu süreçte yer alan ekip üyelerini tanımak, görev dağılımlarının farkında olmak ekip çalışmasında adaptasyon sürecini hızlandıracak diğer taraftan ise bir kariyer planlaması söz konusu ise bu ekip içerisinde nerede konumlandırılmak istendiği keşfedilecektir. Çünkü ekipler incelendiğinde aslında

birbiri ile bağlantılı çok fazla iş gerçekleştirildiği, kazanılan deneyime bağlı olarak farklı pozisyonlarda da görev alınabileceği görülmektedir. Bu görev yapısı ile birlikte bölüm kapsamında web geliştirme süreci detayları ile ele alınmıştır. Bu doğrultuda ekip ve yapılan iş arasında bağ kurulabilmesi oldukça önemlidir. Her bir ana iş paketi aslında çeşitli yetkinlikler gerektirecek birçok alt iş paketlerinden meydana gelmektedir. Genel olarak akışa bakıldığında hedef belirleme, tasarım, geliştirme, test ve yayın, güncelleme ve bakım olmak üzere beş temel aşamanın olduğu ifade edilmektedir. Bu ana aşamalar farklı kaynaklarda detaylandırılarak ifade edilebilmektedir. İş hedefleri ve hedef kitlenin belirlenmesi bu sürecin ana ayaklarını oluşturmaktadır. Çünkü geliştirecek bu web sitesine iş ve kullanıcıyı bir araya getirmek üzere ihtiyaç duyulmaktadır. Bu üçlü arasındaki bağın kopuk olmaması gerekmektedir. Tüm isterler alınıp değerlendirme ve projelendirme yapıldıktan sonra gerçek anlamda bir tasarım süreci başlamaktadır. Bu noktada web sitesi içerisinde yer alacak her türlü nesnenin ve de sayfa ana yapılarının tasarımı gerçekleştirilmektedir. Tasarım aşamasında çeşitli tasarım dilleri ve araçlarını bilmek ile birlikte sanatsal yeteneklere de ihtiyaç duyulacağı açıktır. Tasarımlar tamamlandıktan sonra gerçekleştirme aşamasına geçilmektedir. Gerçekleştirme aşamasında ortaya konacak web sitesinin dinamik ve statik yapıda olmasına göre alt iş paketleri ve ihtiyaç duyulan programlama yetenekleri değişmektedir. Statik bir yapıdaki web sitesi için hazırlanan ana tasarımlar kullanılarak yayına çıkacak tüm sayfalar ve site içeriğinde kullanılacak tüm öğeler hazır edilir. Test aşamasında hazırlanan sayfalar açısından kullanıma engel teşkil eden ya da probleme sebep olan hataların olup olmadığı belirlenmekte, raporlanmakta ve sorunlar çözüme kavuşturulmak üzere ilgili ekiplere iletilmektedir. Yayın hayatı önce test yayını ile başlamakta ardından herhangi bir sorun gözlemlenmediği taktirde web sitesi kullanıma sunulmaktadır. Bu aşamadan sonra müşteri ve web geliştiriciler arasında yapılan sözleşmeye esasen bakım ve geliştirme işlemlerine devam edilir. Web geliştirme sürecinden sonra bu işe başlayanların kafasında öncelikle beliren soru işaretlerine ışık tutulmuştur. Bu doğrultuda web tasarım ilkeleri açıklanmış, her bir ilke doğrultusunda amaçlar belirtilmiştir. İlkeler arasında içerikler ile ilgili ilkeler, sayfa yerleşim yapılandırılmasına ilişkin ilkeler, kullanılabilirlik ve işlerlik ilkeleri, tarayıcılar ve farklı cihazlarla uyumlu çalışabilirlik gibi ilkeler ele alınmıştır.

## Kaynakça

Hippopx, 2023, Erişim Linki: <https://i0.hippopx.com/photos/661/477/153/design-internet-www-web-design-e1364153f3f02f527341164b1e01c0ba.jpg>, Erişim Tarihi: 02.03.2023.

Sakman, S. (2017). Arayüz Tasarımları İçin Prototip Kullanımı. *Sanat ve Tasarım Dergisi*, 20, 225-237.

Türkmenoğlu, H. ve Atalar, D. (2020). Arayüz Tasarımında Kullanıcı Deneyimi Amaçlı Prototip Tasarımı. *İdil Dergisi*, 70 (2020 Haziran), 1041-1046.

# Ünite Soruları

## Soru-1 :

Müşterinin ihtiyaçlarına, web sitesinin kullanılacak iş alanına ve web sitesinin hedef kullanıcı profiline göre projenin genel konseptini belirleyen, gerekli fizibilite çalışmalarını gerçekleştiren, web sitesinin hizmete girmesi sonrasında kullanıcıların siteye yönlendirilebilmesi için gerekli reklam, organizasyon ve iş birliklerini gerçekleştiren, bir bakıma müşteri, kullanıcı ve geliştirici ekip ile etkileşimde olan ve bu üçü arasında koordinasyonu sağlayan kişi kimdir?

(Çoktan Seçmeli)

(•) - Web tasarımcı

(•) - Kullanıcı

(•) - Proje mimarı



(•) - Web geliştirici

(•) - Test elemanı

**Cevap-1 :**

Proje mimarı

---

**Soru-2 :**

Gerçekleştirilecek sitenin hem görsel tasarımını gerçekleştiren hem de kullanılması gereken fonksiyonları belirleyen kişi kimdir?

(Çoktan Seçmeli)

(•) - Proje yöneticisi

(•) - Web tasarımcı

(•) - Web geliştirici

(•) - Analist

(•) - Test elemanı

**Cevap-2 :**

Web tasarımcı

---

**Soru-3 :**

Aşağıdakilerden hangisi görsel tasarım için kullanılan araçlardan değildir?

(Çoktan Seçmeli)

(•) - Photoshop,

(•) - Adobe Illustrator,

(•) - FreeHand,

(•) - Corel Draw,

(•) - Notepad++

**Cevap-3 :**

Notepad++

---

**Soru-4 :**

Aşağıdakilerden hangisi/hangileri web tasarımcının görev alanı içerisinde?

I. Görsel öğeler tasarlamak

II. Web sitesinin güvenlik testlerini yapmak

III. Web sitesinin kullanılabilirlik analizlerini yapmak

IV. HTML, CSS bilmek

V. Java, PHP, MySql bilmek

(Çoktan Seçmeli)

(•) - I ve II

(•) - I, III ve IV

(•) - IV ve V

(•) - II ve III

(•) - Yalnız V

**Cevap-4 :**

I, III ve IV

---

**Soru-5 :**

Aşağıdakilerden hangisi web geliştirme sürecinde ihtiyaç duyulan temel yeteneklerden değildir?

(Çoktan Seçmeli)

(•) - Yönetim

(•) - İstatistik

(•) - Sanat

(•) - Tasarım

(•) - Programlama

**Cevap-5 :**

İstatistik

---

**Soru-6 :**

I. Geliştirme

II. Test

III. Hedef Belirleme

IV. Tasarım

V. Yayın

Yukarıdaki maddeleri web geliştirme sürecini göz önünde bulundurarak doğru bir şekilde sıraladığınızda hangisi bastan 3. sırada olur?

(Çoktan Seçmeli)

(•) - I

(•) - II

(•) - III

(•) - IV

(•) - V

**Cevap-6 :**

I

---

**Soru-7 :**

Web geliştirme sürecinde müşterinin yer aldığı sektör, bu sektör içerisindeki yeri, sektör özellikleri, müşterinin sektörde sunduğu ürün veya hizmetlerin neler olduğu gibi konuların analiz edildiği adım hangisidir?

(Çoktan Seçmeli)

(•) - Geliştirme

(•) - Test

(•) - Hedef Belirleme

(•) - Tasarım

(•) - Yayın

**Cevap-7 :**

Hedef Belirleme

---

**Soru-8 :**

Projelerde iş-zaman planlaması için kullanılan diyagram hangisidir?

(Çoktan Seçmeli)

(•) - CRISP-DM

(•) - PowerChart

(•) - Gantt

(•) - MSProject

(•) - Venn

**Cevap-8 :**

Gantt

**Soru-9 :**

Aşağıdakilerden hangisi web tasarım ilkeleri arasında sayılamaz?

(Çoktan Seçmeli)

- (•) - Sayfa tasarlanırken sayfaların bütünlüğüne dikkat edilmelidir.
- (•) - Web sitesindeki alt sayfalara sadece ana sayfadan erişilecek şekilde planlama yapılmalıdır.
- (•) - Site haritasının çok iyi bir şekilde yapılandırılmış olması gerekir.
- (•) - Metinsel içeriklerin anlaşılır, uzun ve gereksiz cümlelerden arındırılmış olması gerekmektedir.
- (•) - Web sayfaları için yerleşim planları çıkartılırken ana taslağın ister bir kâğıda ister bilgisayar ortamında bir uygulama vasıtasıyla hazırlanması gerekmektedir.

**Cevap-9 :**

Web sitesindeki alt sayfalara sadece ana sayfadan erişilecek şekilde planlama yapılmalıdır.

---

## 3. WEB TASARIMA GİRİŞ

### Birlikte Düşünelim

Metin ve görsel öge içerikli, statik bir web sitesi oluşturmak için hangi dili bilmek gerekir?

HTML ile bir web sitesi tasarımında neler yapılabilir?

HTML kodu yazmak için diğer programlama dilleri gibi bir editöre ihtiyaç var mıdır?

### Başlamadan Önce

Bu bölüm kapsamında web tasarım için en temel dil olan HTML'nin temelleri açıklanmıştır. İşaretleyici-biçimlendirici diller hakkında bilgi verilmiştir. Bir web sitesi geliştirmek için oluşturulması gereken dosyalar ifade edilmiştir. Temel işaretleme etiketleri açıklanmıştır.

### 3.1. Giriş

Kitabın ilk iki bölümünde web tasarım ile ilişkili olarak temel bilgilere yer verilmiştir. Bu doğrultuda web tasarımı sonucunda ortaya koyulacak ürünlerin yayınlaması süreci ile bağlantılı olarak web teknolojilerine ilişkin bilgiler paylaşılmış, web tasarımının da bir parçası olduğu web geliştirme sürecinin detayları ele alınmıştır. Son olarak ise web siteleri tasarlanırken dikkat edilmesi gereken temel ilkeler belirtilmiştir. Bu kısma kadar olan bilgiler ışığında artık tasarımın nasıl gerçekleştirildiğine ve gerekli kodların nasıl yazılacağına dair teknik bilgiler, kullanılacak araçlar ve uygulamaya yönelik olarak örneklerle yer verilebilir. Tekrar belirtmekte fayda vardır ki, bu ders kapsamında genel anlamda statik bir web sitesinin tasarımı ele alınacaktır. Ancak ilerleyen bölümlerde dinamik parçalar yerleştirmeye yönelik olarak da bazı konulara değinilecektir.

Bir web sayfasını görüntülemek istediğimizde web tarayıcılarını kullanmaktayız. Web tarayıcıları çeşitli internet protokollerini de kullanarak sunucu tarafından gönderilen dosyaları işleyerek istemciye gösterilir. Web sitelerini istemciler olarak çeşitli görseller ve metinlerin yer aldığı belirli bir düzene göre yapılandırılmış çıktılar olarak görmekteyiz. Ancak sunucu tarafından web tarayıcıya iletilen dosyalar tam olarak bu şekilde değildir. Bu durumu en net şekilde herhangi bir web sitesi açık olduğu durumda klavyeden F12 tuşuna basıldığında görmek mümkündür. F12 tuşuna basıldığı taktirde karşımıza tam olarak anlamlandıramadığımız birçok karakter toplulukları ve ifadelerin yer aldığı kod satırları gelecektir. İşte bu kodlar sunucunun web tarayıcı ile paylaştığı dosyanın içeriğidir. Bu dosya içerisinde yazılmış kodlar tarayıcı tarafından okunup derlenerek istemcilerin görüntülediği hali ile web sayfalarına dönüştürülmektedir. Bu kodlar içerisinde web sayfasına ilişkin tüm detaylar yer almaktadır. Öyle ki, sayfa içerisindeki öğelerin yerleşimi, kullanılan görsel ve metinsel öğelere ilişkin tüm detaylar, renk düzenlemeleri, stil düzenlemeleri, sayfa içerisindeki bağlantılar bu detaylar arasında sayılabilir. Web tasarımı kapsamında hem sayfaların ana tasarımları hem de statik bir site hazırladığımızı varsayarak, site içeriğini oluşturacak tüm ana ve alt sayfaların oluşturulması yani geliştirme aşaması ile ilgilenmiş olacağız. Web tarayıcıları tarafından çözümlenen bu dosyalar hazırlanırken biçimlendirme-işaretleme dilleri kullanılmaktadır. Bu diller programlama dilleri ile karıştırılmamalıdır. Çünkü programlama dilleri bilgisayarların komutları anlayabilmesi için bilgisayar diline çevrilmek üzere yazılan kod bloklarını kapsamaktadır. İçeriğinde bir görevi yerine getirtmek için komut verme mantığı söz konusudur. Biçimlendirme dilleri ise platform bağımsız olarak bir makinenin yani bilgisayarın bir eylemi gerçekleştirmesini talep etmekten ziyade belirlenen konumdaki biçimsel düzenlemelere yönelik olarak kullanılmaktadır. Biçimlendirme-işaretleme dilleri çalıştırılmaz yani derlenmez, bunun yerine okunur ve yorumlanır. Bir diğer ifade ile biçimlendirme-işaretleme dili ile 4+8 gibi basit bir toplama işlemini bilgisayara yaptıramazsınız. Halbuki programlama dillerini kullanarak farklı hesaplama operatörleri hatta kendinizin yazacağı fonksiyonlarla çeşitli hesaplamaları bilgisayarlara yaptırabilmek mümkündür. Bu diller ile işaretlenen özellikler belirlenerek, ilgili alanda metnin ya da diğer öğelerin işlenmesi, tanımlanması istemci tarafına sunulması gerçekleştirilir. Bu

noktada web tarayıcısı bir metnin nerede başlaması gerektiğini, hangi büyüklükte görüntüleneceğini, bir görsel öğe mevcutsa web sayfası içerisinde bu görseli nerede konumlandırması gerektiğini okuduğu kodlar içerisinde ayırt edebilmekte ve bu farklılıklar ile web sayfası istenildiği şekilde düzenlenebilmektedir. Bir diğer ifade ile bir web sitesi web tarayıcıda görüntülenirken o noktada çalışan program aslında web tarayıcılardır. Biçimlendirme-işaretleme dili ile hazırlanan dosyalar bu programa girdi olarak verilir ve program bu girdiye göre çalışır.

Biçimlendirme-işaretleme dilleri arasında;

- Hiper Metin İşaretleme Dili (*HTML-Hyper Text Markup Language*)
- Genişletilebilir İşaretleme Dili (*XML-Extensible Markup Language*)
- Genişletilebilir Hiper Metni İşaretleme Dili (*XHTML- Extensible HyperText Markup Language*)
- Genelleştirilmiş İşaretleme Dili (*GML-Generalized Markup Language*)
- Standart Genelleştirilmiş İşaretleme Dili (*SGML-Standard Generalized Markup Language*)
- LaTeX
- TeX
- GenCode
- Scribe

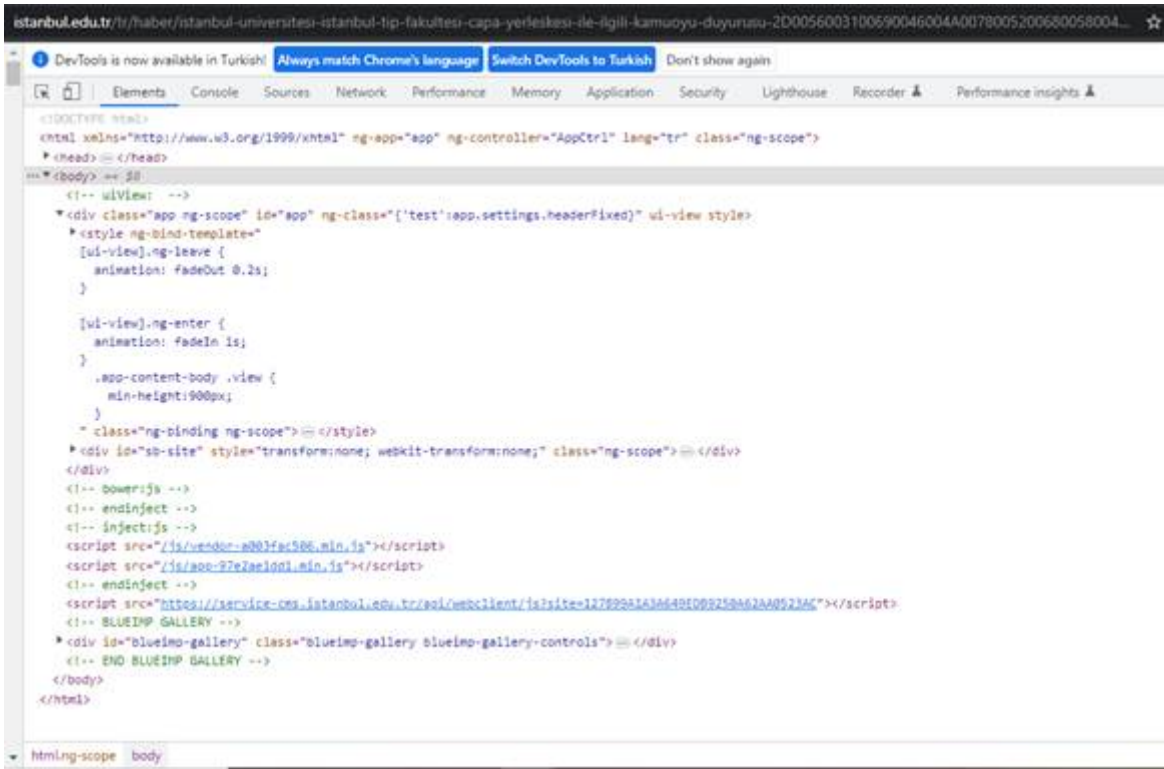
sayılabilir. Genellikle ML yani işaretleme dili (*Markup Language*) uzantılı birçok farklı seçenek de ayrıca mevcuttur. Bu noktada XML de önemli işaretleme dillerindendir. XML veri ve bilgi paylaşımı/transferi noktasında ön plana çıkmakta olup, verilerin paylaşılabilir şekilde tanımlanıp saklanmasını sağlar. Günümüz yazılımları da veri transferi için xml dosyalarından yararlanmaktadır. Uzun ve diğ. (2008) HTML ve XML farklarını şu açılardan ele almıştır:

- Her iki dil birbirine benzemekle birlikte, HTML verinin nasıl görüntüleneceği XML verinin içeriği ve verinin transferi (değişimi) ile ilgilidir.
- XML işaretleri önceden tanımlı olmayıp, kişiye özel düzenlenebilir.
- XML'in hiyerarşik yapısından ötürü veriye erişim HTML'ye göre daha hızlıdır. Buna bağlı arama işlemi daha hızlı gerçekleştirilebilir.
- Etiket seçimi ve hiyerarşik yapının uygulaması kişiye özgü olduğundan işlemci performansı ve bant genişliği açısından sorun oluşma ihtimali söz konusudur.

Bu kitap kapsamında HTML esas olarak alınmış ve örnekler buna göre düzenlenmiştir. İfade etmek gerekir ki HTML ile web sayfası içeriğine yönelik olarak her türlü işlemi gerçekleştirmek de mümkün olamamaktadır. Sayfa içerisinde metinlerin konumlarını belirleme gerek farklı sayfalara gerek çeşitli dosya ve dokümanlara bağlantılar vermek, yukarıdaki gibi bir liste yapısı oluşturmak, tablolar ekleyebilmek gibi işlemleri sadece HTML ile tamamen yapabilmek mümkündür. Ancak üyelik sistemi oluşturmak, forum sayfaları açmak, online ödeme gerçekleştirmek, mesaj bırakılabilecek ziyaretçi defterleri oluşturmak gibi kullanıcının web sayfası üzerinden işlem yapabilmesi, veri girebilmesi gibi etkileşim olarak adlandırılan aksiyonları gerçekleştirebileceği web sayfalarını oluşturmak HTML ile mümkün değildir.



Şekil 18. İstanbul Üniversitesi web sitesinde bir haberin web tarayıcıdaki görüntüsü



Şekil 19. Şekil 17’de yer alan tarayıcı görüntüsünün biçimlendirme-işaretleme dili ile yazılmış kaynak kod dosyası içeriği

Şekil 18 ve Şekil 19’de aynı web sayfasının istemci tarafından web tarayıcıda görüntülediği hali ve F12 tuşuna basılmak suretiyle elde edilen kaynak kod içeriği görülmektedir. Kaynak kod dosyasının tipi ilk satırda yazmakta olup html olduğu görülmektedir. HTML en çok kullanılan biçimlendirme-işaretleme dilidir. Elbette bir biçimlendirme-işaretleme dili kullanmadan da web sitesi tasarlamak mümkündür. Bu noktada hem hazır web sitesi tasarımları sunan ve içerik girmek için bir yönetim paneli bulunan çevrimiçi sistemler mümkündür. Bu sistemler üzerinden hem ücretsiz hem de daha ileri tasarımlar ve gelişmiş araçlar için ücretli olacak şekilde kendi sitenizi tasarlamaz ve yayına alabilmeniz mümkündür. Bu çevrimiçi sistemler arasında sites.google.com, wordpress.com, wix.com, isimtescil.net sayılabilir. Üyelik işlemleriniz ve alan adının belirlenmesinin ardından seçilen hazır bir web sitesi tasarımı için sayfa içeriklerini hazırlayabilir ve web sitenizi yayımlayabilirsiniz. Diğer taraftan bu hazır sistemler yerine kendi sitenizi kodlama bilmeden tasarlayabilir ve sitenizde yer alacak sayfalarınızın dosyalarını hazırlayarak alacağınız bir sunucu hizmetiyle dosyalarınızı belirlediğiniz alan adınız ve size ayrılan dosya alanına yükleyerek de web sitenizi

yayınlayabilirsiniz. Bu durumda Frontpage ya da Dreamweaver gibi programlar kullanılması gerekmektedir. Bu programların menülerinde yer alan tasarım araçlarını kullanarak sayfada olmasını istediğiniz öğeleri yerleştirebilir, sayfa yapısını kolaylıkla düzenleyebilirsiniz. Tasarımı tamamladığınızda biçimlendirme-ışaretleme dili ile hazırlanmış kod dosyalarını kaydedip yayına alabilirsiniz. Tüm bunlar elbette kişisel, basit bir web sitesinin tasarlanması ve yayınlanması açısından yeterli olabilir ancak profesyonel anlamda bir iş hedefi ile web sitesi hazırlanacaksa mutlaka bir biçimlendirme-ışaretleme dili bilinmeli hatta dinamik yapılar için programlama dillerine hâkim olunması gerekmektedir.

## 3.2. Hiper Metin İşaretleme Dili (HTML)

Hiper Metin İşaretleme Dili daha önce de ifade edildiği gibi web siteleri oluşturmak için yaygın olarak kullanılan ve evrensel olarak kabul edilmiş bir işaretleme dilidir. HTML bir programlama dili karmaşıklığına sahip değildir. İşaretleme dili bir başka ifade ile etiketleme dili olduğundan kod bloğu içerisinde “<>” ile başlatılan ve bitirilen etiketler ile ifade edilmektedir. Her bir etiketin olduğu noktada o alanda yer alacak öğeler için biçimlendirme yapılmaktadır. Biçimlendirme etiket içerisinde belirtilen detaylar doğrultusunda gerçekleştirilir. İlerleyen kısımlarda daha detayları ile ele alınacak olmak ile birlikte örneğin <h1> bir başlık biçimlendirmesini ifade etmektedir. Bu etikete gelindiğinde etiketin açılış ve kapanışı arasına kalan metnin bir başlık olacağı ve bu başlığın büyüklüğünün ne olacağı anlaşılmaktadır. HTML ile yazılan kodlar .htm ya da .html uzantılı dosyalarda tutulmaktadır. Web tarayıcılarda kullanıcılar tarafından görüntülenen içerikler bu dosyalardır. HTML ile genel olarak bir web sitesi için aşağıdaki işlemler gerçekleştirilebilir:

- Çeşitli özelliklere göre düzenlenmiş metinler, tablolar, listeler, fotoğraflar ve logo gibi öğeleri barındıran statik bir web sitesi hazırlamak üzere gerekli dosyaların hazırlanmasını sağlar.
- Web sitesi dosyalarında çeşitli köprü ve bağlantıların eklenmesine yardımcı olarak gerek web sitesi içerisindeki diğer sayfalara gerek başka web sitelerine erişim sağlanabilir.
- Çeşitli işlemlerde kullanılmak üzere form tasarımları gerçekleştirilebilir.
- Ses ve görüntü içerikleri ile farklı uygulamaları web sayfalarına eklenmesini sağlayabilir.

HTML sadece statik web sitelerinin geliştirilmesi için değil aynı zamanda web uygulamalarda ara yüz geliştirmek üzere tasarım amaçlı da kullanılmaktadır. Aşağıda basit bir HTML kod bloğu görülmektedir.

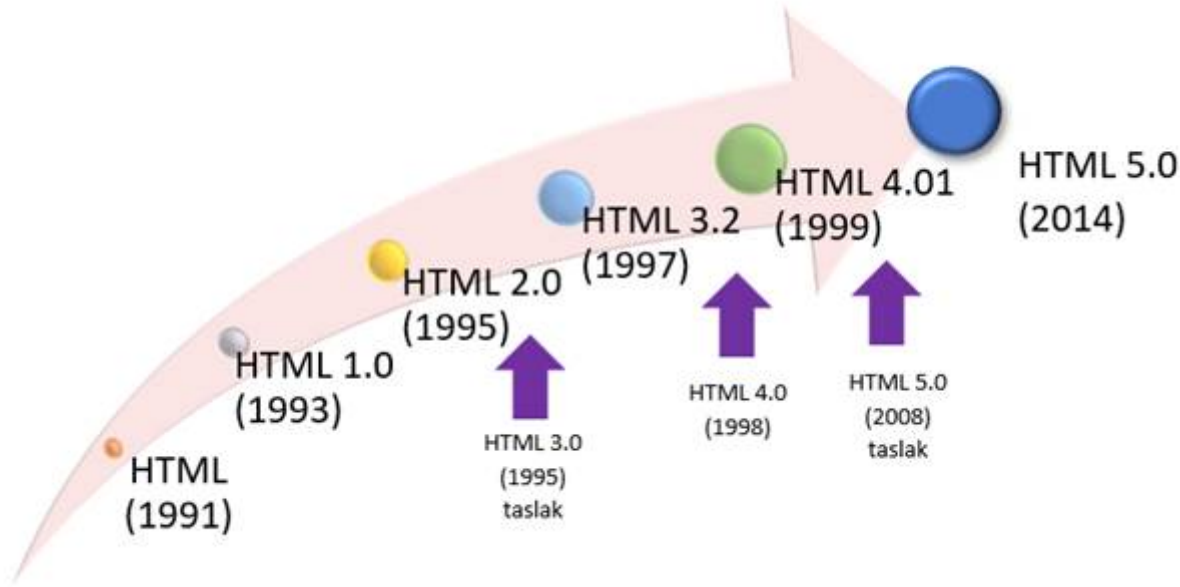
```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv='content-language' content="TR">
    <title>Web Notları</title>
  </body>
</html>
```

Şekil 20. HTML kod bloğu

Yine web tarayıcı tarafındaki işleyişe dönecek olursak, her bir .html uzantılı dosya web sitesi altındaki bir web sayfasını temsil etmektedir. Web sitesi tasarlayıp, geliştirdiğinizde site haritanızda yer alan tüm sayfaların bir .html uzantılı dosyası olacaktır. Bu dosyalarda yer alan etiketler tarayıcının web sayfasını nasıl görüntüleyeceğini belirlemektedir. Bu noktada belirtmek gerekir ki doğru bir klasörleme ve dizinleme çalışması da yapılması önem arz etmektedir. Web sitenizin tüm dosyalarını içerecek bir ana klasör olmalı ve sayfalarınıza ait .html dosyaları ile içeriklerde yer alacak öğeler ve web sitenize ilişkin tüm diğer dosya ve öğeler (.pdf, .jpeg, .mov uzantılı dosyalar ve öğeler) bu klasör altında ya da karmaşıklığı daha azalmak adına oluşturulan alt klasör yapılarında yer almalıdır. Web sitesi yayınlanacağı zaman ana klasör tümüyle sunucuya yüklenmelidir. Eğer tüm sayfalar ve öğeler arası bağlantılar doğru yapılandırıldıysa web sitesinin görüntülenmesinde sorun yaşanmayacaktır. Vericiden alıcıya tek yönlü bağlantıda istemde bulunan kullanıcı web sayfasını görüntülemek üzere talepte bulunduğu ilgili dosya kullanıcının bilgisayarında önbelleklenir ve bu işlem caching olarak ifade edilir (Ceylan, 2020). Programlama dillerinde olduğu gibi herhangi bir derleme söz konusu olmadığından HTML ile yazılan kodların çalıştırılması oldukça hızlıdır.



HTML 1990'lı yılların başında yine web'in öncüsü Tim Berners Lee öncülüğünde ortaya konmuştur. Lee, CERN'deki çalışanların birbirleri ile bilgi, doküman ve çalışmalarını paylaşabilmesini ve bu paylaşım sonucunda paylaşılan öğelerin tüm çalışanların bilgisayarlarında aynı şekilde görüntülenebilmesi amaçlamıştır. WWW'nin sunumundan sonra da temel etiketleri ile birlikte HTML Leenin bu amacı doğrultusunda duyuruldu. Sonraki yıllarda gazete ve dergilerdeki görünümünün web ortamında da benzer olarak oluşturulabilmesi, burada yayınlanan metinlerin benzer bir formatta Webte de yayınlanabilmesi ile başlayan geliştirme çalışmaları ile yeni HTML versiyonları sunulmuştur. Günümüz ile karşılaştırıldığında ilk HTML'nin oldukça kısıtlı ve basit olduğunu ifade etmek yanlış olmayacaktır. HTML en büyük değişimini HTML 4.0 versiyonu ile yaşamıştır. İnternet teknolojilerindeki gelişmelerle birlikte ise HTML 5.0 olarak son halini almıştır. Şekil 21'de HTML'in gelişim süreci grafiksel olarak verilmiştir.



Şekil 21. HTML'in gelişim süreci

Web sitesi geliştirirken ihtiyaç duyulan temel gereksinimler:

- Sayfa tasarımı için:

☐ HTML editörü/editörleri

o Grafik tasarım ve gerekirse animasyon programı/programları

- Sayfaları Web'te yayınlamak için:

☐ Sayfaların barındırılacağı bir web alanı

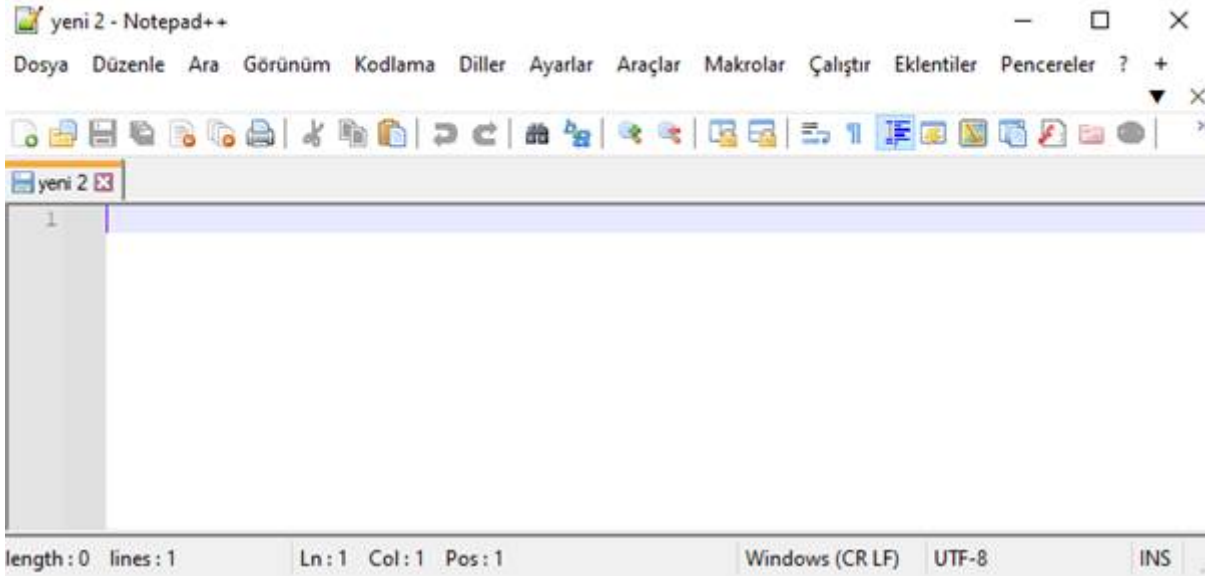
o Sorunsuz bir İnternet bağlantısı

☐ Dosya transfer (FTP) programı

o Sayfaların farklı çözünürlükler ve farklı web tarayıcılarda test edilmesi

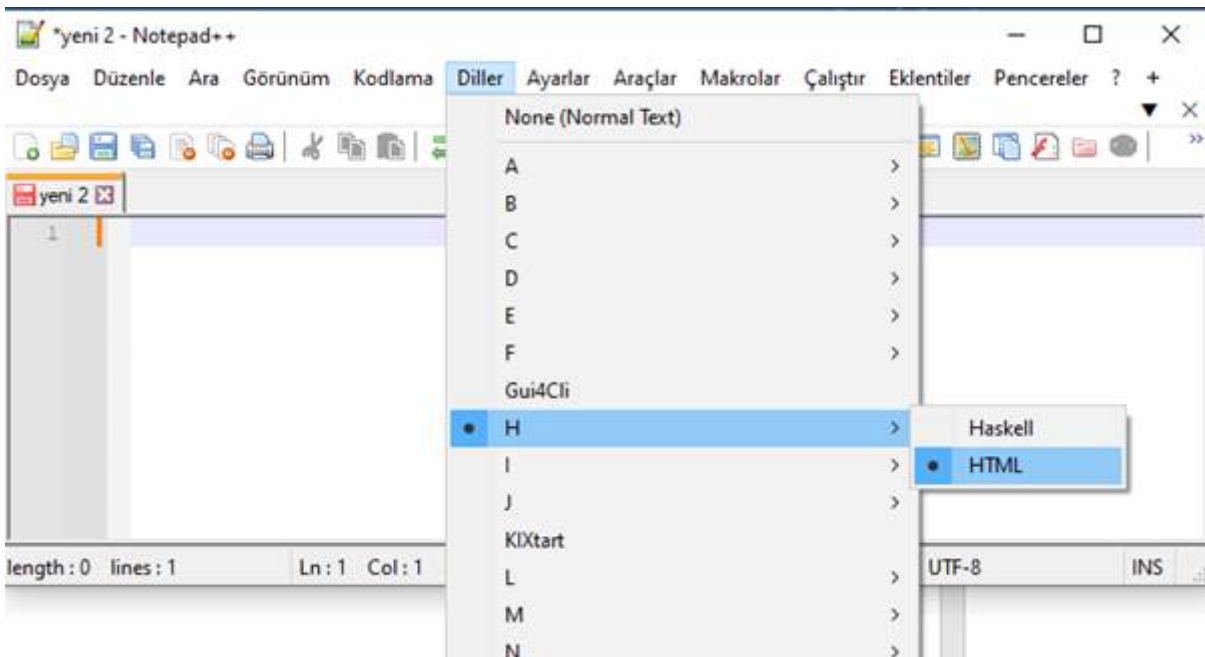
Öncelikle sayfa tasarımına odaklandığımızdan ötürü HTML editörleri konusu açıklanmalıdır. Bölüm içerisinde sunucuya HTML kodlarının yazıldığı .html uzantılı dosyaların eklendiği ifade edilmiştir. Bir programlama dili ile yazılım geliştirmek üzere yazılacak kodlar bir editör diğer ifadeyle geliştirme ortamları vasıtasıyla yazılır ve çalıştırılır. Editör kodların makine diline çevrilmesine ve makinenin verilen komutları yerine getirilmesi sağlanmaktadır. HTML kodlarının yazılması için ise bir metin editörüne ihtiyaç vardır ve farklı editörler kullanılabilir. Bilgisayarlarımızda yer alan “not defteri” uygulaması bir metin editörüdür. Dolayısıyla bu uygulama kullanılarak html dosyaları hazırlanabilir. Ancak daha gelişmiş editör kullanmanın bu noktadaki avantajları daha sistematik kod yazımına yardımcı olması, kod tamamlama işlevinin olması, daha hızlı geliştirmeye yardımcı olması, arama motoru optimizasyonuna yardımcı olabilmesi şeklinde ifade

edilebilir. Bilinen HTML editörleri arasında Notepad++, Atom, Sublime Text, CoffeCup, Adobe Dreamweaver CC sayılabilir. Bu ders kapsamında ücretsiz olan Notepad++ yazılımı kullanılmıştır. Notepad++, <https://notepad-plus-plus.org/downloads/> adresinden uygun sürüm seçilerek yüklenebilir. Uygulama için Türkçe dil desteği de mevcuttur. Şekil 22’de Notepad++ uygulamasına ilişkin ara yüz görülmektedir. Dosya sekmesinden yeni bir dosya oluşturulur ve kodlar bu dosyada yazılarak kaydedilir. Dosyalar kaydedilirken .htm veya .html dosya uzantısı ile, örneğin anasayfa.html şeklinde, kaydedilmesine dikkat edilmelidir.



Şekil 22. Notepad++ ara yüz görüntüsü

“Diller” sekmesinden hangi işaretleme diline özgü kodlar yazılacaksa o dil seçilmelidir (Şekil 23). Bu seçim kod yazmaya başlandığında kod tamamlama özelliği ile yardımcı olacaktır. Diğer taraftan kodlama yapılırken kullanılacak karakterler özelleştirilebilir. Programlamada çeşitli karakter setleri ve karakter kodlamaları kullanılabilir. ANSI ve Unicode karakter setleri karakterler ve bu karakterlere karşılık gelen kodları içermektedir. Kullanılan bit farklılıkları bu karakter set ve kodlamalarında bazı değişikliklerin oluşmasına sebep olmaktadır. Notepad++’da hangi karakter seti ya da kodlaması kullanılacaksa “Kodlama” sekmesinden seçim yapılmalıdır. Türkçe içerik hazırlanacağı zaman Türkçe karakterlerde gösterim sorunu yaşanmaması için UTF-8 ya da BOM Olmadan UTF-8 karakter kodlamasını seçmek bu sorunu ortadan kaldıracaktır. Bu seçim ilgili sekmeden yapılabileceği gibi aşağıda gösterildiği şekilde kod satırının HTML kod bloğunun başına eklenmesi ile de gerçekleştirilebilir.

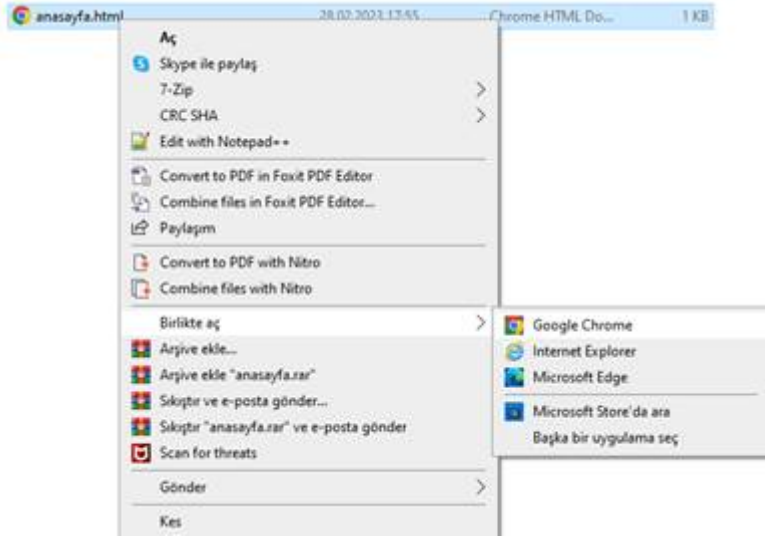


Şekil 23. Notepad++ kodlama dili seçimi

```
1 <meta charset="utf-8">
```

Şekil 24. Türkçe için karakter kodlaması belirlemek üzere kullanılan HTML kod satırı

Belirtildiği şekilde bir HTML dosyası oluşturulup kaydedildikten sonra bu dosyanın olduğu klasöre gidilir. İlgili dosyaya sağ tıklayarak “birlikte aç” seçeneği seçildiğinde gelen web tarayıcı seçeneklerinden birisi seçilir. Seçilen web tarayıcıya göre dosyanın açıldığını ve belirtilen etiketler doğrultusunda yapılan tüm biçimlendirmelerin bir web sayfası şeklinde görüntülendiğini görmüş olursunuz. Buna ilişkin ekran görüntüsü Şekil 25’te görülmektedir.



Şekil 25. HTML dosyalarını çalıştırma

### 3.3. Hiper Metin İşaretleme Dili (HTML) ile Kodlama-Genel Yapıya İlişkin Bilgiler

HTML ile kodlar yazılacağı zaman her bir etiket “< >” şeklinde iki köşeli parantez içerisinde belirtilir. Etiketler aynı zamanda tag olarak da ifade edilebilir. Biçimlendirmeye ilişkin hangi işlemin yapılacağını bu köşeli parantezler içerisine yazılan anahtar sözcükler belirler. Bunu genel olarak ifade etmek gerekirse etiketler <etiket\_adi> olarak kullanılmaktadır. Hiçbir anahtar sözcük Türkçe karakter içeremez. Diğer taraftan her bir etiket önce açılmalı sonra kapatılmalıdır. Tarayıcı html dosyasını okuduğu zaman etiketin başlangıcını gördüğünde o etikete ilişkin biçimlendirmeyi aynı etiketin kapanışına kadar uygulamaya devam eder. Genellikle kodlama sırasında etiketler açılırken kapanışlar unutulmaktadır. Etiketın kapandığı / işareti ile anlaşılır, bir diğer ifade ile kapanış </> şeklinde gösterilir. Her açılan etiket mutlaka kapatılmalıdır. Etiket nasıl açıldıysa aynı anahtar kelimeyle kapatılır, bir anahtar kelimenin eşdeğeri yoktur. Kapatılmayan istisnai bazı etiketler söz konusudur, ilerleyen bölümlerde bunlara ilişkin örnekler de paylaşılacaktır. Bir HTML dosyası <html> etiketi ile başlatılır ve </html> ile kapatılır. Görüldüğü üzere açılış ve kapanışta html anahtar sözcüğü kullanılmıştır. Diğer tüm etiketler bu etiketin açılış ve kapanışı arasında yazılır. Etiketler iç içe yazılabilir. Önce hangi etiket açılmışsa o etiket en son kapanmalıdır. Bazen etiketler için parametreler kullanılır. Bu parametre değerlerine göre biçimlendirme gerçekleştirilir. Programlamada da benzer bir mantık söz konusudur. Kullanılan bazı fonksiyonlar için parametre değeri istenir ve bu değere göre işlem yapılır. Bu açıdan kodlama ve işaretleme dili benzerlik göstermektedir. Parametre değerleri belirtilirken açılış etiketi içerisinde önce parametre belirtilir sonrasında bu parametrenin alması gereken değer tırnak işareti (“”) içerisinde belirtilir.

Şekil 26’te hem etiketlerin açılış ve kapanışında olması gereken sıralama hem de parametre değerlerinin nasıl gösterilmesi gerektiği verilmiştir.

```

1 <etiket_Adil parametre_Adi="parametre_Degeri">
2   <etiket_Adi2>
3     <etiket_Adi3>
4       ...
5     </etiket_Adi3>
6   </etiket_Adi2>
7 </etiket_Adil>

```

Şekil 26. Etiket açılış ve kapanış sıralaması

HTML kod dosyalarında ana yapıda bulunması gereken standart etiketler söz konusudur. Bu etiketler aşağıdaki tabloda verilmiştir.

Tablo 1. Temel HTML etiketleri

Etiket	Açıklama
<code>&lt;html&gt;...&lt;/html&gt;</code>	Dosyanın HTML dosyası olduğunu belirtir. Web tarayıcı bu kodları gördüğünde içerisindeki etiketleri <u>HTML</u> 'e göre yorumlar. Kodlar bu etiket ile başlar, tüm diğer etiket ve içerikler bu etiketin başlangıç ve kapanışı arasında belirtilir.
<code>&lt;head&gt;...&lt;/head&gt;</code>	Sayfaya ilişkin bilgilerin belirtildiği etikettir. Az önce ifade edildiği gibi örneğin karakter setleri ya da kodlamalarına ilişkin bilgiler bu etiket açılış ve kapanışı arasında belirtilir.
<code>&lt;title&gt;...&lt;/title&gt;</code>	Bu etiket <code>&lt;head&gt;</code> etiketinin açılış ve kapanışı arasında belirtilir ve web sayfasının başlığını belirlemeye yarar. Başlıktan kasıt sayfa içerisindeki herhangi başlık değil, tarayıcıda adres çubuğu üzerinde yer alan kısımda sayfaya ilişkin başlıktır.
<code>&lt;body&gt;...&lt;/body&gt;</code>	Bu etiket web sayfası içerisinde kullanıcı tarafından görüntülenen her detayın ilgili etiketlerinin yazıldığı bölümdür.

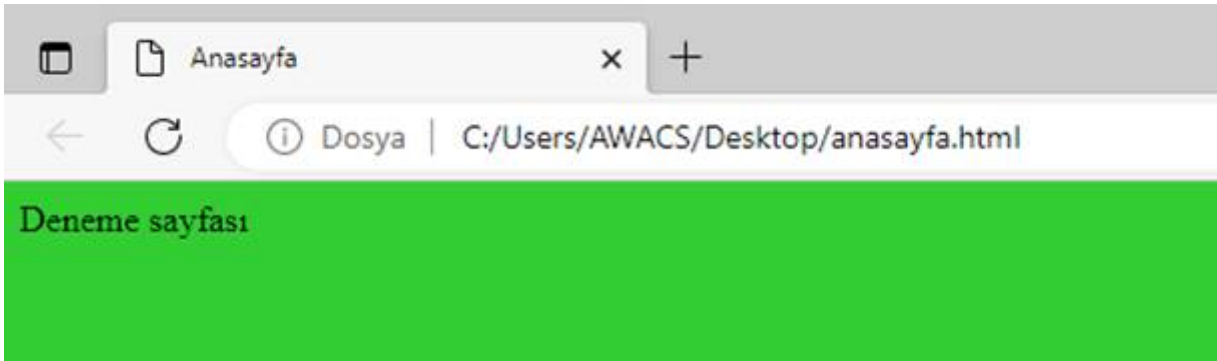
Bu temel kodların açıklamalarından sonra bir HTML dosyası içerisinde bu kodların nasıl gösterildiği incelenir.

```

1  <html>
2      <head>
3          <meta charset="utf-8">
4          <title>
5              Anasayfa
6          </title>
7      </head>
8      <body bgcolor="Limegreen">
9
10         Deneme sayfası
11
12     </body>
13 </html>

```

Şekil 27. HTML dosyası içerisindeki temel kodlar



Şekil 28. Temel kodlara karşılık gelen web sayfası görüntüsü

Şekil 27’te görüldüğü üzere tüm kodlar <html> etiketinin açılış ve kapanışı arasında yazılmıştır. Bu noktada iki ana etiketin varlığını görmek mümkündür. Bunlar <head> ve <body> etiketleridir. <head> etiketinin etkin olduğu kısımda (etiketinin açılış ve kapanışı arası) sayfanın içeriğinden ziyade sayfaya ilişkin meta ve başlık özellikleri belirlenir. Meta özellikleri sayfaya ilişkin açıklamalar, anahtar kelimeler, yazar bilgileri, karakter kodlama stilleri gibi özellikleri kapsamaktadır. Örneğin arama motorunda bir anahtar kelime ile arama yapıldığında sayfa için meta özellikleri arasında anahtar kelimeler tanımlanmışsa ve arama motorunda girilen anahtar kelime ile eşleşirse sayfanın motorda görüntülenme oranı artacaktır. Yine bazı kaynak yönetim sistemlerinde sayfaların meta verileri kullanılarak kayıt yapılmaktadır. Meta özellikler bölümünde sayfaya ilişkin bilgiler girildiyse bu bilgiler kaynak yönetim sistemi tarafından kullanılır. Meta özelliklerini belirlemek için <meta> etiketi kullanılır. Meta etiketi içerisinde kullanılacak özelliklerin nasıl HTML ile kodlanacağı Şekil 29’de verilmiştir. Şekilden de görüleceği üzere meta etiketi içerisinde parametreler ve bu parametrelerin aldığı değerler şeklinde sayfaya ilişkin özellikler tanımlanmaktadır. Buna göre, açıklama, anahtar kelimeler ve yazar ile ilgili bilgi tanımlanacaksa “name” parametresi, parametre değeri olarak ise neye ilişkin bilgi tanımlanacaksa o başlık seçilmektedir. Başlık seçildikten sonra, seçilen bu başlığa ilişkin bilginin ne olduğu da tanımlanmalıdır. Bu bilgiler “content” yani içerik parametresine verilen değerler ile tanımlanır. Her bir girilecek bilgi için yeni meta etiketi açılmalıdır. Şekilden görüldüğü üzere değeri “açıklama” olan “name” parametresi için değeri “Web Tasarım” olan “content” parametresi tanımlanmıştır. Bunun insan dilinde açıklaması bu web sitesinin içeriği “Web Tasarım” ile ilgilidir şeklindedir. Aynı mantıkla anahtar kelime (keyword) ve yazar (author) bilgileri de sırasıyla “HTML, meta” ve “anonim” şeklinde tanımlanmıştır. Buna göre sayfanın anahtar kelimeleri “HTML, meta” ve yazarı “anonim” olarak açıklanmaktadır. Meta etiketi içerisinde “http-equiv” parametresi http ile ilgili tarayıcıya gönderilecek ek bilgilerin ifadesinde kullanılır. Aldığı “refresh” değeri sayfanın yenilenmesini, “content” parametresinin aldığı değer de bu yenilemenin kaç saniyede bir olacağını belirtir. eğer “url” parametresine de değer atandıysa bu değer sayfanın yenilenme anında hangi adrese yönlendirileceğini göstermektedir.



```

1  <head>
2
3      <meta charset="UTF-8">
4      <meta name="description" content="Web Tasarım">
5      <meta name="keyword" content="HTML, meta">
6      <meta name="author" content="anonim">
7
8  </head>

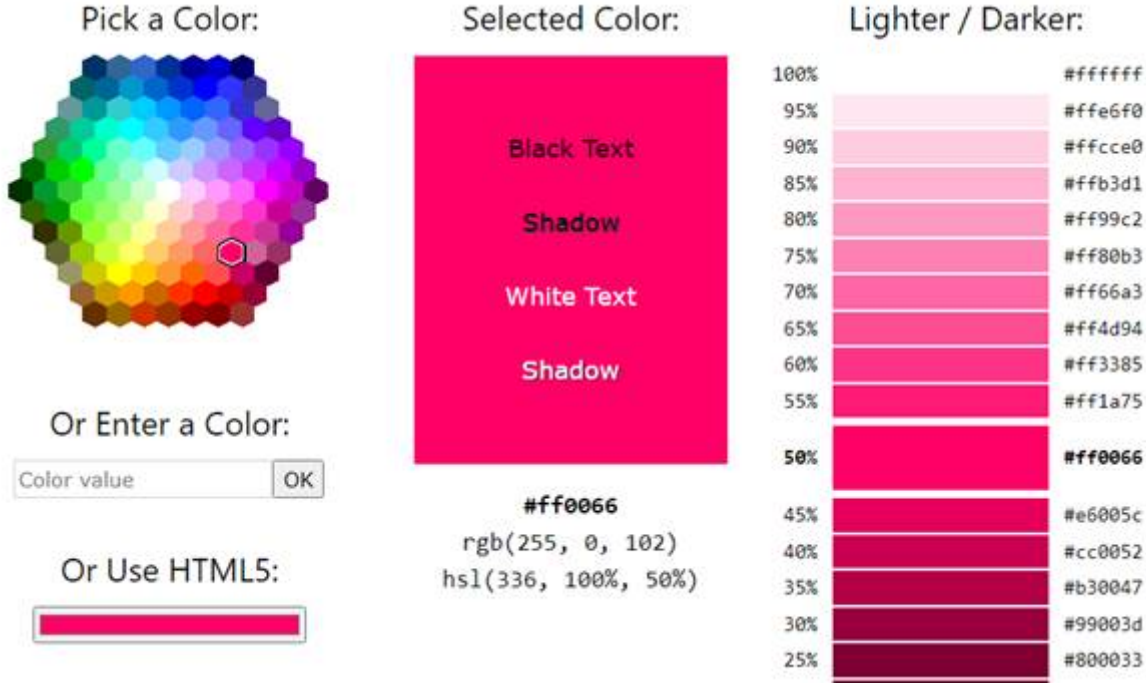
```

Şekil 29. Meta etiketi özellikleri

Meta etiketinden sonra yeniden Şekil 27’teki HTML kodlarına dönüş yapılacaktır. head etiketi altında başlık (title) etiketi olduğu görülmektedir. <title> etiketinin açılış ve kapanışı arasında tarayıcıda sayfaya ait başlıkta ne yazması talep ediyorsa o başlık tanımlanır. Burada başlık “Anasayfa” olarak tanımlanmıştır. Bu noktada sayfa başlığı ve metin başlığı karıştırılmamalıdır. Sayfa başlığı Şekil 28’de da görüldüğü gibi web tarayıcıda adres çubuğunun üzerinde sayfayı tanımlayan bir başlık olarak kullanıcının ekranına yansımaktadır. <head> etiketinin açılış ve kapanışı arasında <meta> ve <title> etiketleri dışında kullanılan etiketler de mevcuttur. Bunlar; stil bilgilerinin tanımlandığı <style>, sitenin temel URL adresinin belirtmeye yarayan <base>, bulunulan HTML dosyası ile dış bir kaynak arasında bağlantıyı (ilişkiyi) belirtmek için <link>, Javascript ile yazılan script kodlarını sayfaya entegre etmek için <script>, desteklemeyen bir başka sayfaya bağlanmak veya gerekiyorsa kullanıcıya mesaj vermek için kullanılan <noscript> etiketleri şeklinde sıralanabilir.

<head> etiketi alanı içerisindeki diğer etiketler ile ilgili bilgilerden sonra <body> etiketinin de yine Şekil 27’teki görüldüğü üzere <html> etiketi etki alanı içerisindeki ikinci ana etiket olduğu görülmektedir. Bu etiketin açılış ve kapanışı arasında tanımlanacak tüm etiketler artık sayfanın içeriğine yönelik olarak düzenlemelerin yapılmasını sağlayacaktır. Yani safada içeriğinde yer alan görseller, metinler, renk düzenlemeleri, tablolar, yerleşim düzeni ile ilgili tüm etiketler bu kısımda yazılmaktadır. Şekil 27’te yazıldığı haliyle bu sayfada iki biçimlendirme yapılmıştır. Birincisi sayfa arka plan rengi belirlenmiş, ikincisi ise sayfa içerisinde yer alacak metin yazılmıştır. Sayfada yer alacak metin bu kısımda yazılmışsa da metin ile ilgili herhangi bir düzenleme yapılmadığına dikkat edilmelidir. Bu kısımda başında yapılacak biçimlendirmeler ya da tanımlamalar için < > köşeli parantezler içerisindeki sözcüklerin aslında ne yapılması gerektiğine ya da neyi tanımladığına dair ipucu verdiği ifade edilmiştir. <body> etiketi göz önünde bulundurulduğunda, bu etiketin açılış ve kapanışı arasındaki bölümün aslında sayfanın “ana gövdesi”ni temsil ettiği kolaylıkla anlaşılmaktadır. Dolayısıyla bu gövde üzerinde yapılacak her türlü biçimlendirmeye ilişkin etiketlerin burada verilmesi gerektiği de açıktır. İlgili HTML kodlarında bu ana gövdenin tamamını ilgilendiren bir biçimlendirmenin yapıldığı görülmektedir. <body> etiketinde bir parametre ve parametre değerinin belirtilmesi bizi bu sonuca götürmektedir. bgcolor="Limegreen" incelendiğinde bgcolor parametresini ve “ ” işareti içerisinde yer alan Limegreen’in ise bu parametrenin aldığı değer olduğu önceki parametre ve parametre değer gösterimi açıklamalarından elde edilmektedir. “bgcolor” karşımıza yine bir anahtar kelime olarak çıkmaktadır. Bu ifade aslında “background color” kelimesinin bir bakıma kısaltılmış versiyonudur ve Türkçe çevirisi “arka plan rengi”dir. “Limegreen” ise sayfanın hangi renkte görüneceğini ifade eden HTML için hazırlanan renk paletinde yer alan bir renktir. Renk değerleri belirtilirken tanımlı renklerin isimleri buradaki örnekte olduğu gibi yazılabilirken, HTML renk paleti içerisinde seçilen renge ait renk kodu da yazılabilir. Bir arama motoru aracılığı ile HTML renk paletine kolay şekilde erişilebilir. Şekil 30’de HTML için renk paleti görülmektedir (URL1). “Pink a Color” altındaki karteladan bir renk seçilip üzerine tıklanıldığında “Selected Color” başlığı altına seçilen renk gelmektedir. Bu alanın altında # işareti ile başlayan bir kod görülmektedir. Bu kod “bgcolor” parametresinin değeri olarak HTML kod satırında yazılabilir. Örneğin seçilen pembe rengin kodu #ff0066’dır. Dolayısıyla ilgili kod satırı <body bgcolor="FF0066"> olarak yazılmalıdır (Renk kodunda küçük ya da büyük harf kullanımı bir hataya sebep olmaz, iki türlü kullanılabilir). Bu kod Hex Kodu olarak da ifade edilmektedir. Hex, heksadesimalden, yani 16 tabanlı sayı sisteminden gelmektedir. 0-15 arası sayılardan 0-9 sayılar aynı rakamlar ile, 10-15 arası sayılar sırasıyla A, B, C, D, E, F harfleri ile temsil edilir. Hex kodu altı haneden meydana gelmekte olup, # işaretinden sonra her iki basamak bir ana rengin tonunu temsil etmektedir. Soldan sağa ilk iki basamak kırmızı (R-Red), ikinci iki basamak yeşil (G-Green), son iki basamak ise mavi (B-Blue) ana rengini temsil eder. Bu üç ana rengin tonları karıştırıldığında da (altı basamak bir arada ele alındığında) ara ton renklerin kodu elde edilir. Her ana renk için renk tonları 00-FF arasında değerler alır. Hex kodunun dışında bu renkler RGB kodları ile de ifade edilir.

Her bir ana renk değeri 0-255 arasında değişir. Örneğin kırmızı ana rengini belirtmek için (255, 0, 0) RGB kodu kullanılır. RGB kodları Hex konuda dönüştürülerek (sayı değeri 16'lık sistemde karşılığına çevrilir) ilgili HTML kodunda parametre değeri olarak kullanılır.



Şekil 30. HTML renk paleti

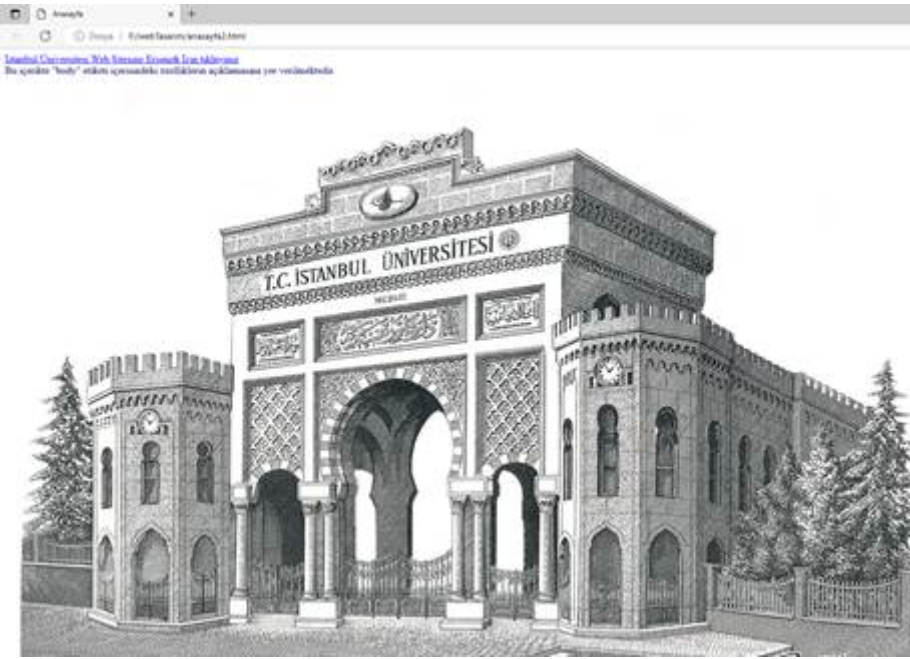
<body> etiketi içerisinde kullanılabilecek “bgcolor” dışında farklı parametreler de söz konusudur. Bunlar ile ilgili açıklamalara yer vermeden önce bilinmelidir ki, bu özellikler HTML5 tarafından desteklenmemektedir. Bu parametreler yerine ilerleyen bölümlerde yer verilen CSS konusunda anlatılan biçimsel düzenlemeler kullanılmaktadır. Şekil 30’da <body> etiketi içerisinde yer alan “background”, “text”, “link”, “alink”, “vlink” parametreler görülmektedir. Bu parametreler sırasıyla arka plan resmini belirleme, metin rengini belirleme, sayfada yer alan linklerin rengini belirleme, linke tıkladığınız anda linkin rengini belirleme, daha önce ziyaret edilmiş linklerin rengini belirleme amacıyla kullanılmaktadır. Şekil 31 ve Şekil 32 birlikte incelendiğinde <body> etiketinde belirtilen tüm özelliklerin tarayıcıda okunup, ekrana yansıtıldığı görülecektir. Ekrandaki görüntüde link mavi görünmektedir. Ancak kod bloğunu içeren HTML dosyasını bir web tarayıcı aracılığıyla çalıştırırsanız öncelikle linkin mavi gözüktüğünü linke tıklarken ve de tıkladıktan sonra renklerde değişiklik olduğunu görebilirsiniz. Bu noktada şu hususa dikkat çekilmesi gerekmektedir. Kod satırından görüleceği üzere background=”iuKapi.jpg” şeklinde tanımlanmış ve sayfa tarayıcıda çalıştırıldığında da arka plan görselinde İstanbul Üniversitesi’nin ana giriş kapısı yer almaktadır. İlgili görsel üniversitenin kurumsal iletişim sayfasından indirilmiştir. Bu noktada eğer resim dosyası ile .html uzantılı dosya aynı klasörde yer almıyorsa arka plan görselinde bu resim yer almayacaktır. Resim yolunda sadece resim adı ve uzantısı verilmişse web tarayıcı görselin .html uzantılı dosya ile aynı klasörde olduğunu düşünüp, klasör içerisinde ilgili görsel dosyayı arayacaktır. Eğer kullanılacak görsel farklı bir klasör altında ise dosya yolunun açık olarak verilmesi gerekmektedir.

```

1 <html>
2 <head>
3 <title>Anasayfa</title>
4 </head>
5 <body background="iuKapi.jpg" text="#000066" link="blue" alink="red" vlink="green" >
6
7 <a href="https://www.istanbul.edu.tr/tr/">İstanbul Üniversitesi Web Sitesine Erişmek İçin tıklayınız</a> <br>
8
9 Bu içerikte "body" etiketi içerisindeki özelliklerin açıklamasına yer verilmektedir.
10
11 </body>
12 </html>

```

Şekil 31. <body> etiketindeki parametreler



Şekil 32. Şekil 30'daki HTML kodlarının web tarayıcıda görüntüsü

Bu bölümün başında etiketlerin açıldığı gibi kapatılması gerektiği ve açılış ile kapanış arasında hazırlanan web sayfasında gösterilecek verinin yazılması, eklenmesi gerektiği ifade edilmiştir. Bu noktada bazı etiketlerin kapanışı olmadığı da ayrıca belirtilmiştir. Bu etiketlerden birisi `<br />` etiketidir. Bu etiketin bulunduğu yerde bir alt satıra geçiş sağlanmaktadır. Yani .html dosyası içerisinde `<br />` etiketini koyduktan sonra aynı satırda metin yazmaya devam edilse bile bu sayfa web tarayıcıda çalıştırıldığı durumda `<br />` etiketinin tam bulunduğu noktadan itibaren metnin bir alt satırdan devam ettiği görülecektir. Bir diğer kapanışı olmayan etiket `<hr />` etiketidir. Bu etiket de bulunduğu yatay çizgi oluşmasını sağlamaktadır. Dikkat edilecek olursa `<meta>` etiketinin de kapanışı olmadığı görülecektir. Benzer olarak; bir web sayfasına grafik görüntü eklemek üzere kullanılan `<img>`, belge ile dış kaynak arasında bağlantının sağlanması için kullanılan `<link>`, etkileşimli içeriklerin eklenmesi için kullanılan `<Embed>` şeklinde örneklenebilir. Bu şekilde kapanışı olmayan ya da tekil olarak adlandırılan etiketler gelecek örneklerde karşılaşıldıkça açıklanacaktır. Şekil 33'te yer alan HTML kodları ve Şekil 34'te bu kodlara ilişkin tarayıcı görüntüsü incelendiğinde `<br/>`, `<hr/>` ve `<img>` etiketlerinin kapanışı olmadığı, bu etiketlerin bulunduğu yerlere karşılık gelen tarayıcı görüntüsünde sırasıyla metnin alt satıra geçtiği, metnin altına yatay çizgi ve çizginin de altına belirtilen genişlik ve yükseklikte görsel eklendiği görülmektedir. Bu noktada hatırlatmak gerekir ki, benzer kodlar üzerinde yapılan değişiklikler ve bu değişiklikler ile birlikte tarayıcıya gelen ekran görüntüleri arasındaki farkları da inceleyiniz (Örneğin Şekil 32 ve Şekil 34)

```

1 <html>
2 <head>
3 <title>Anasayfa</title>
4 </head>
5 <body background="iuKapi.jpg" text="#000066" link="blue" alink="red" vlink="green" >
6
7 <a href="https://www.istanbul.edu.tr/tr/">İstanbul Üniversitesi Web Sitesine Erişmek İçin tıklayınız</a> <br/>
8
9 Bu içerikte "body" etiketi içerisindeki <br/> özelliklerin açıklamasına yer verilmektedir
10 <hr/>.
11 
12 </body>
13 </html>

```

Şekil 33. Kapanışı olmayan tekil etiketler





Şekil 34. Şekil 33'ün tarayıcıdaki görüntüsü

Diğer taraftan hatırlanacak olursa Şekil 20'de verildiği üzere HTML kodlarının yazıldığı dosyanın başına `<!DOCTYPE HTML>` etiketi de eklenmektedir. Bu etiket de kapanışı olmayan bir etikettir ve web tarayıcı bu satırı okuduğunda dosyanın bir HTML dosyası olduğunu anlamaktadır. `<html>` etiketinin varlığı sebebiyle bu etiketin gereksiz olduğu düşünülebilir. Ancak bazı web tarayıcıları dosya tipini otomatik olarak algılayabilirken bazılarında bu özellik mevcut değildir. Bu nedenle uyumsuzluk sorunu oluşmaması adına `<!DOCTYPE HTML>` etiketinin dosya başına eklenmesi uygundur.

Program kodları yazılırken, daha sonra kodu başka bir programcı okursa diye yazılan kodları açıklayıcı açıklama satırları eklenmektedir. Ancak bilgisayarların bu açıklama satırlarını, kodlar derlenip çalıştırılırken bir kod satırı/bloğu olarak görmemesi için farklı bir şekilde ifade edilmesi gerekir. Farklı programlama dillerinde farklı karakter ya da karakterler toplulukları kullanılabilir. HTML kodları yazılırken de aynı mantık söz konusudur. Kod satırları derlenip çalıştırılmasa da okunduğundan, yazılan bu açıklama satırlarının içerikte yer alan metinlerle karıştırılmaması için yine açıklama satırı olarak özel gösterim biçimi ile ekleme yapılır.

HTML dosyası içerisinde bir metnin ve metnin yer aldığı sayfanın biçimlendirilmesi/tasarımına ilişkin HTML kodlarının yazıldığı varsayalım. Dosya web tarayıcıda açıldığında metin içerisinde kullanılabilecek bazı özel karakterlerin görüntülenmeyeceği fark edilecektir. Bu özel karakterler belirli HTML kodları ile dosya içerisinde belirtilmesi gerekmektedir. Bu özel karakterler ve ilişkin HTML kodları aşağıdaki tabloda verilmiştir (Ceylan, 2020).

Tablo 2. Özel karakterlere ilişkin HTML kodları

Karakter	Açıklama	HTML Kodu	Karakter	Açıklama	HTML Kodu
	Boşluk	&nbsp;	\$	Dolar işareti	&dollar;
<	Küçüktür işareti	&lt;	€	Euro işareti	&euro;
>	Büyüktür işareti	&gt;	©	Telif hakkı işareti	&copy;
&	Ampersan işareti	&amp;	®	Tescilli marka işareti	&reg;
"	Çift tırnak	&quot;	%	Yüzde işareti	&percnt;
'	Tek tırnak	&apos;	(	Sol parantez	&lpar;
£	Pound işareti	&pound;	)	Sağ parantez	&rpar;
¥	Yen işareti	&yen;			

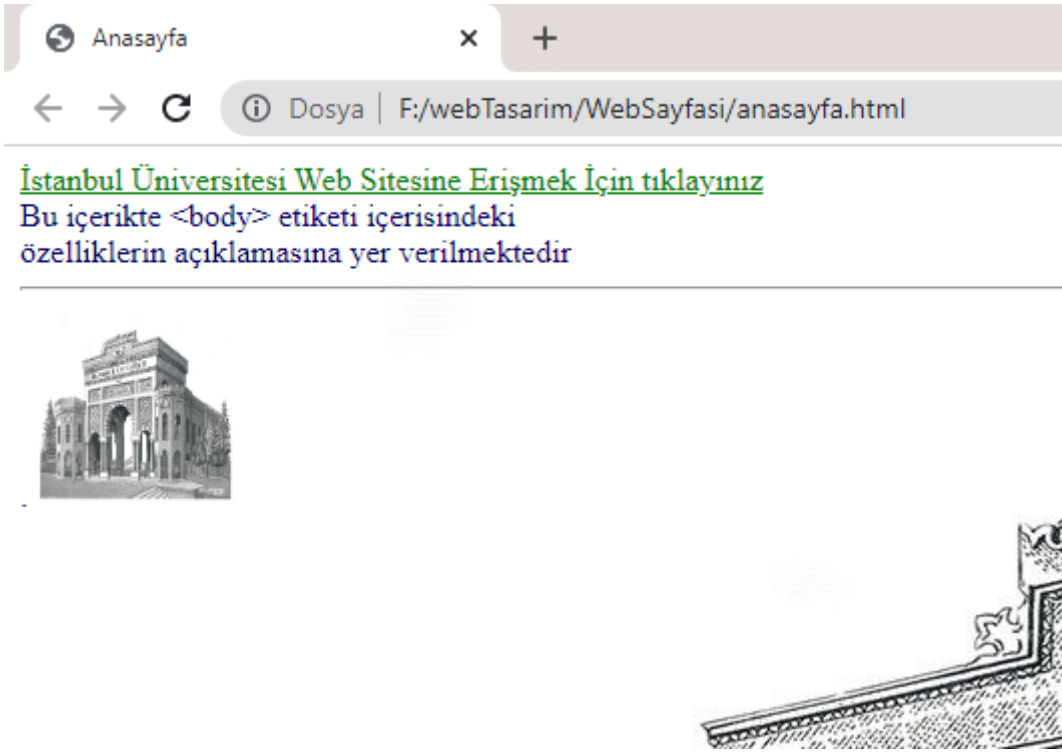
Verilen bilgiler ışığında yeniden düzenlenen HTML kodları ile ilgili tarayıcı görüntüsü Şekil 35 ve Şekil 36'da görülmektedir.

```

1 <html>
2 <head>
3 <title>Anasayfa</title>
4 </head>
5 <body background="iuKapi.jpg" text="#000066" link="blue" alink="red" vlink="green" >
6
7 <a href="https://www.istanbul.edu.tr/tr/">İstanbul Üniversitesi Web Sitesine Erişmek İçin tıklayınız</a> <br/>
8
9 <!--Bu içerikte "body" etiketi içerisindeki <br/> özelliklerin açıklamasına yer verilmektedir-->
10 Bu içerikte <lt;body> etiketi içerisindeki <br/> özelliklerin açıklamasına yer verilmektedir
11
12 <br/>,
13 
14 </body>
15 </html>

```

Şekil 35. Yorum satırı ve özel karakterlere ilişkin HTML kod örneği



Şekil 36. Şekil 35’te yer alan HTML kodlarının web tarayıcıdaki görüntüsü

## Bölüm Özeti

Bölüm kapsamında web tasarıma ilişkin detaylar verilmeye başlanmıştır. Web tasarım için kullanılabilecek en temel biçimlendirme dili HTML’dir. Dolayısıyla statik ya da dinamik bir site tasarlama/geliştirmeyi amaçlayan kişilerin öncelikle HTML diline hâkim olması gerekmektedir. HTML daha önce de ifade edildiği gibi web siteleri oluşturmak için yaygın olarak kullanılan ve evrensel olarak kabul edilmiş bir işaretleme dilidir. İşaretleme dili bir başka ifade ile etiketleme dili olduğundan kod bloğu içerisinde “< >” ile başlatılan ve bitirilen etiketler ile ifade edilmektedir. HTML dili kullanılarak web tasarım gerçekleştirmek üzere metin editörlerine ihtiyaç bulunmaktadır. Bölüm kapsamında farklı metin editörleri belirtilmek ile birlikte verilen örneklerde Notepad++ kullanılmıştır. Web sitelerinde her bir sayfa HTML dosyaları olarak hazırlanmaktadır. Bu dosyaların uzantıları .html veya .htm’dir. Tüm dosyalar hazırlandıktan sonra alan adı ve sunucu hizmeti vasıtasıyla bu dosyalar web’e yüklenir ve web tarayıcılarda dosyalar okunarak, dosyalardaki HTML kodları görsel arayüzlere çevrilir.

Bölüm kapsamında bir HTML dosyasındaki kod yapısının ne olması gerektiği ve bu yapı içerisinde yer alan temel etiketlerden bahsedilmiştir. Bu temel etiketler <html>, <head> ve <body>’dir. Her bir etiket açıldıktan sonra ayrıca kapatılmalıdır. Kapatılmaya ihtiyaç duyulmayan tekil etiketler de yine bölüm kapsamında ele alınmıştır. <head> etiketinin açılış ve kapanışı arasında kullanılan diğer etiketlere de değinilmiş olup, bu etiketlerden en önemli olanı <meta> etiketidir. Bu etiket sayfada yer alacak öğelerle ilgili olmayıp sadece sayfaya ilişkin bilgilerin belirlendiği bir etikettir. Bu etikette belirtilen hususlar web tarayıcıda görüntülenen sayfada doğrudan gözükmez. Diğer taraftan <body> etiketi sayfa içerisinde tarayıcıda görüntülenecek kısımla ilgili her türlü biçimlendirmenin yapıldığı alandır. Başka etiketlerde de kullanılabileceği gibi bu etiketin kendine özgü bazı parametre ve parametre değerleri söz konusudur. Bu değerlere ilişkin detaylar da bölüm kapsamında ele alınmıştır. Bir kod satırında açıklama satırının nasıl gösterilmesi gerektiği, HTML kodları okunurken tarayıcıda görüntülenemeyen karakterler ve bunların görüntülenebilmesi için karakterlerden oluşan gerekli HTML kodları da bölüm sonunda okuyucuya sunulmuştur. Verilen tüm etiketlere ilişkin HTML kodlarının Notepad++ uygulamasındaki yazılışı ve ilişkili kodun tarayıcıdaki görüntüleri bölüm boyunca verilmiştir. Her adımda eklenen kod satırları ve bu satırların ne gibi değişikliklere sebep olduğu şekiller karşılaştırılarak incelenmelidir.

## Kaynakça

Ceylan, T.B. (2020). Web Tasarım Klavuzu. İnkılap Kitabevi, İstanbul.

UR1, Erişim Linki: [https://www.w3schools.com/colors/colors\\_picker.asp](https://www.w3schools.com/colors/colors_picker.asp), Erişim Tarihi:03.03.2023

Hippopx, 2023, Erişim Linki: <https://i0.hippopx.com/photos/386/749/243/programming-html-code-coding-0a0f6b5e0a18ac5441d48e3dedec5161.jpg>, Erişim Tarihi: 02.03.2023.

---

## Ünite Soruları

### Soru-1 :

Aşağıdakilerden hangisi işaretleme dilleri arasında değildir?

(Çoktan Seçmeli)

(•) - HTML

(•) - XML

(•) - PHP

(•) - LaTeX

(•) - SGML

### Cevap-1 :

PHP

---

### Soru-2 :

Bir web sayfasının kaynak kodlarını görüntülemek için klavyeden hangi tuş/tuşlara basılması gerekmektedir?

(Çoktan Seçmeli)

(•) - F5

(•) - F12

(•) - F10

(•) - F8

(•) - Ctrl+Enter

### Cevap-2 :

F12

---

### Soru-3 :

Aşağıdakilerden hangisi HTML versiyonlarından biri değildir?

(Çoktan Seçmeli)

- (•) - HTML 3.2
- (•) - HTML 4.01
- (•) - HTML 5.0
- (•) - HTML 3.5
- (•) - HTML 1.0

**Cevap-3 :**

HTML 3.5

---

**Soru-4 :**

Aşağıdakilerden hangisi HTML editörüdür?

(Çoktan Seçmeli)

- (•) - Notepad++,
- (•) - Atom,
- (•) - Sublime Text,
- (•) - CoffeCup,
- (•) - Hepsi

**Cevap-4 :**

Hepsi

---

**Soru-5 :**

Bir web sayfası ile ilgili yazar, anahtar kelimeler, karakter seti gibi bilgilerin belirtilmesi için kullanılan etiket aşağıdakilerden hangisidir?

(Çoktan Seçmeli)

- (•) - <body>
- (•) - <meta>
- (•) - <img>
- (•) - <head>
- (•) - <link>

**Cevap-5 :**

<meta>

---

**Soru-6 :**

<body bgcolor="#FF0059"> etiketini incelediğinizde buna göre aşağıdakilerden hangisi bir parametre değeridir?

(Çoktan Seçmeli)

(•) - bgcolor

(•) - FF0059

(•) - body

(•) - <>

(•) - =

**Cevap-6 :**

FF0059

---

**Soru-7 :**

Aşağıdakilerden hangisi bir web sayfasının arka planını belirlememizi sağlayan parametredir?

(Çoktan Seçmeli)

(•) - Bgcolor

(•) - vlink

(•) - alink

(•) - text

(•) - line

**Cevap-7 :**

Bgcolor

---

**Soru-8 :**

<meta charset="utf-8"> kod satırı aşağıdakilerden hangisi için kullanılır?

(Çoktan Seçmeli)

(•) - utf-8 sayfasına bağlantı tanımlamak için

(•) - sayfa arka plan rengini belirlemek için

(•) - metin boyutunu belirlemek için

(•) - Türkçe karakter uyumsuzluğunu çözmek için

(•) - paragraf oluşturmak için

**Cevap-8 :**

Türkçe karakter uyumsuzluğunu çözmek için

---

**Soru-9 :**

&nbsp; HTML kodu ne için kullanılır?

(Çoktan Seçmeli)

- (•) - Tırnak işareti için
- (•) - Satır başı oluşturmak için
- (•) - Alt satıra geçmek için
- (•) - Yeni paragraf oluşturmak için
- (•) - Boşluk oluşturmak için

**Cevap-9 :**

Boşluk oluşturmak için

---



# 4. HTML METİN VE GÖRSEL DÜZENLEME ETİKETLERİ

## Birlikte Düşünelim

Web sayfalarını görüntülediğimizde metin ve görsellerin bir arada kullanıldığını, ancak metin ya da görsellerin kendi aralarında farklı biçimsel özelliklere sahip olduğunu görebiliyoruz.

İçeriğe yönelik tüm bu biçimlendirmeler HTML ile nasıl gerçekleştirilmektedir?

## Başlamadan Önce

Bölüm kapsamında HTML etiketlerinin açıklanmasına devam edilmiştir. Bu bölümde ele alınan etiketler sayfa içeriğinde yer alan metin ve görsellerin düzenlenmesine ilişkin önemli etiketlerdir. Zaman içerisinde ilerleyen bölümlerde öğrenilecek yapıların içerisinde bu etiketler kullanılacaktır.

### 4.1. Giriş

Bir önceki bölümde HTML ile ilgili açıklayıcı bilgiler ve gelişim sürecinden bahsedilmiş, temel HTML kodlarından bahsedilmiştir. HTML dosyalarında içeriğe yönelik biçimlendirmeler çeşitli şekillerde yapılabilir. Bunlar şu şekilde açıklanabilir:

- Önceki bölümde de belirtildiği üzere HTML bir işaretleme dilidir ve web tasarım için oluşturulacak sayfalarda biçimlendirmeler için etiketlerden yararlanır. HTML ile ilgili olarak konunun başından bu yana metin üzerindeki tüm düzenlemeler için etiketlerin kullanıldığı ifade edilmektedir. Bu düzenlemenin nerede yapılacağı ise etiketin konulduğu (açıldığı) kısım ile başlayıp kapanışa kadar devam edeceği de ayrıca belirtilmiştir. Hatta bu nedenle bu durum işaretleme olarak belirtilir ki düzenlemenin başlatılacağı yeri işaret eder.
- Önceki bölümde <html> etiketinin temel etiket olduğu ve diğer bütün etiketlerin bu etiketin açılış ve kapanışı arasında yazılması gerektiği belirtilmiştir. Yine <html> etiketinin açılış ve kapanışı altında <head> ve <body> olmak üzere iki ana etiketin varlığından bahsedilmiş ve bunlara ilişkin temel bilgiler verilmiştir. <head> etiketinin açılış ve kapanışı arasında stiller belirtilebilir ve bu stillere <body> etiket açılış ve kapanışı arasından atıfta bulunulur ve stillerde belirlenen biçimlendirmelere göre düzenleme yapılır.
- Son seçenek ise HTML dosyaları dışında CSS dosyaları oluşturulur. Bu dosyalar sadece stillerin belirlendiği bir dosyadır. HTML dosyalarında bu CSS (*Cascading Style Sheet*) dosyalarına atıfta bulunarak gerekli biçimlendirmeler yapılabilir. Bu dosyaların kullanılması tüm stillerin tek bir dosyada toplanabilmesi, aynı stil biçimlerinin tekrar tekrar yazılmamasıdır.

İlerleyen bölümlerde stil tanımlama ve CSS konusu açıklanacaktır ancak şu aşamada HTML'nin temeli bağlamında biçimlendirmelerin etiketlerle nasıl gerçekleştirilebileceği üzerinde durulmaktadır. Bu bölümde <body> etiketi içerisinde metin ve görsellerin biçimlendirilmesi için kullanılabilecek etiketler açıklanmış ve bunlara ilişkin örnekler verilmiştir. Bir web sayfası web tarayıcıda görüntülendiğinde içerikte yer alan her şey aslında belirli bir düzene göre sayfa içerisine eklenmiştir. Metinlerin ya da görsellerin sayfanın içerisindeki konumları, metinlerdeki şekilsel düzenlemeler, sayfa renk düzenlemesi, sayfa içerisindeki gerek bir web sayfasına gerek bir dosyaya bağlantılar ve daha birçok düzenleme her bir özelliğe ilişkin etiketlerle gerçekleştirilmektedir.

### 4.2. Sayfa İçeriği Düzenlemeye İlişkin Etiketler

Yukarıda da ifade edildiği gibi ana etiketlerden olan <body> etiketi altında metinsel ve görsel öğelere ilişkin düzenlemeler yapılmakta ve bunun için de çeşitli etiketler kullanılmaktadır. Bu etiketlerden birisi “başlık” etiketidir. Ancak bu başlık <head> etiket içerisindeki <title> etiketi ile karıştırılmamalıdır. <title> etiketi tarayıcı uygulamasının üst bölümünde gözüken sayfa başlığını düzenlemek/yerini etiketlemek için kullanılmaktadır. Burada bahsedeceğimiz başlık etiketi ise artık sayfa içerisinde yer alacak metinlere ait başlıkların düzenlenmesi/yerinin etiketlenmesi için kullanılmaktadır. Buradaki başlık etiketi site içi başlık etiketi olarak da isimlendirilebilir. Başlık etiketi <h1> ile gösterilmektedir. “h” bu etiketin açılış ve kapanışı arasında belirtilecek metnin başlık olduğunu “1” ise başlık büyüklüğünü belirtmek için kullanılır. Başlıklar için <h2>, <h3>, ... <h6> etiketleri de kullanılabilir. Bu etiketler, etiket açılış ve kapanışı arasındaki ifadenin başlık olacağını ancak bu başlığın farklı büyüklükleri temsil eder. En büyük başlık <h1> en küçük başlık ise <h6>’dır. Bu büyüklükler sırasıyla 8, 10, 12, 14, 18, 24 punto büyüklükleri temsil etmektedir. Bu noktada başlıklar arama motorlarının web sayfalarının içeriğe yönelik olarak veri aldığı önemli noktalardan birisi olduğundan başlıklar doğru belirlenmelidir. Diğer taraftan başlık büyüklükleri ile ilgili biçimlendirmeler belirli bir düzene göre yapılmalıdır. Örneğin bu kitapta başlıkların ana ve alt başlıkları için belirlenen bir büyüklük skalası vardır, başlıklar bu skala ve belirlenen kurallara göre yapılandırılır. Web sayfalarında da başlık büyüklükleri rastgele değil, tasarım taslağında belirlenen ölçeklendirmeye göre hazırlanmalıdır. Şekil 37’de Şekil 38’de site içi başlık etiketine ilişkin kod yazımı ve web tarayıcıdaki görüntüsü yer almaktadır. Bu şekiller başlık etiketlerinin 1’den 6’ya kadar “Tarihten Geleceğe Bilim Köprüsü” metninin başlık olarak işaretlenmesine örnek kodlar ve görsellerdir. Görüldüğü üzere başlıklar gittikçe küçülmüştür.

```

1 <html>
2   <head>
3     <title>Anasayfa</title>
4   </head>
5   <body>
6     <h1>Tarihten Geleceğe Bilim Köprüsü</h1>
7     <h2>Tarihten Geleceğe Bilim Köprüsü</h2>
8     <h3>Tarihten Geleceğe Bilim Köprüsü</h3>
9     <h4>Tarihten Geleceğe Bilim Köprüsü</h4>
10    <h5>Tarihten Geleceğe Bilim Köprüsü</h5>
11    <h6>Tarihten Geleceğe Bilim Köprüsü</h6>
12  </body>
13 </html>

```

Şekil 37. Site içi başlık etiketi kullanımı



Şekil 38. Şekil 37’nin web tarayıcı görüntüsü

Sayfa içi başlık etiketinde başlığın sayfanın neresinde görüntülenmesi talep ediliyorsa bunun etiket içerisinde belirtilebilmesi için “align” parametresinden yararlanılmaktadır. Bu parametrenin değerleri “left”, “right” ve “center” şeklinde olup sırasıyla başlığın sol, sağ ve ortada görüntülenmesi sağlanır. Yerleşim belirtilmediği sürece web tarayıcı sol yerleşimi ana değer olarak kabul etmektedir. Başlık etiketi kapanışından sonra satır başına otomatik geçiş yapılır. Bu nedenle <br/> etiketinin kullanılmasına gerek yoktur.

Aynı bu kitapta olduğu gibi web sayfalarında yayınlanan metinlerde de belirli yerlerde paragraflar bulunmaktadır. İlgili metinlerin hangi kısımlarının paragraf olacağını belirtmek üzere <p> etiketinden yararlanılmaktadır. Bu etiketin konulduğu yerde web tarayıcı kodları okurken o noktadan itibaren kapanışa kadar ilgili metnin bir paragraf olduğunu anlayacaktır. Bir diğer metinsel düzenleme yazı stili ile ilgilidir. Normal metin dosyalarında da bu tür özelliklerin olduğunu hatırlayınız. Bir metinde kalın, italik ya da altı çizili yazma gibi yazı stiline ilişkin düzenlemeler MS Word gibi metin dosyalarında bu özellikler söz konusudur. Özellikle metin üzerinde bazı yerlere dikkat çekmek, ya da alıntı yapılan bir bölümü belirtmek üzere bu özelliklerden yararlanıldığı bilinmektedir. Bu özelliklerin web sayfalarında yayınlanacak metinlerde de kullanılabilmesi için çeşitli etiketler mevcuttur. Bu etiketler <b>, <i> ve <u> olmak üzere açılış ve kapanışları arasında yer alan metinlerin sırasıyla kalın (*bold*), italik (*italic*) ve altı çizili (*underline*) yazı tipleri ile biçimlendirilmesini sağlamaktadır. <b> etiketine alternatif olarak <strong> etiketi de kullanılabilir. <strong> etiketi kapatılması gereken etiketler arasında yer almaktadır. Şekil 39, Şekil 40 ve Şekil 41’de bu etiketlere ilişkin HTML kodları ve web tarayıcıda oluşan ekran görüntüleri görülmektedir. Bu üç şekil bir arada incelenirse ilkinde yukarıda açıklamaları verilmiş paragraf ve temel yazı stili etiketlerinin kullanıldığı HTML kodları görülmektedir. Eğer bu kodlar kullanılmamış olsaydı web tarayıcıda oluşacak ekran görüntüsü Şekil 40’ta yer almaktadır. Şekil 41 ile karşılaştırıldığında metin üzerinde ilgili etiketlerin yarattığı farklılıklar net bir şekilde görülmektedir. Başlığın ayrıldığı, paragrafın oluşturulduğu, üniversite isminin kalınlaştırıldığı, altı çizili ve italik bölümlerin oluştuğu bu farklılıklar arasında sayılabilir.

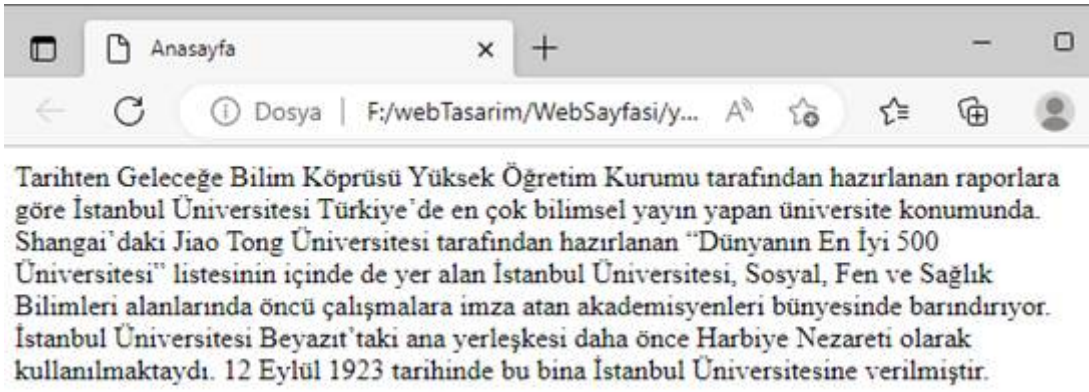
Burada paragraf kullanımına ilişkin belirtilmesi gereken bazı hususlar söz konusudur. Paragraf etiketinin kapanışından sonra yer alan metnin de bir paragraf şeklinde görüntülediği görülmektedir. Bu durum etiket kullanılsa da paragraf oluşacağı algısı oluşturmamalıdır. Burada ilgili metin kısa olduğundan başka bir paragraf gibi görülmektedir. İlgili kod metin editöründe yazılıp metin uzatılmaya devam ederse, paragraf etiketi kullanılmadığı sürece metnin düz bir şekilde ekranın sonuna kadar gittiği, sonrasında alt satıra indiği ve metin bitişine kadar da benzer mantıkla durumun devam ettiği görülecektir. Burada ikinci paragrafın oluşma sebebi ilk paragrafın olmasıdır. Çünkü paragraf etiketi ile o etiketin etki alanı içerisinde yazılan metin paragraf olarak tüm metinden ayrılmıştır. Bir diğer durum ise alt satıra geçme durumudur. Bir önceki bölümde alt satıra geçmek için kullanılan hatta kapanışı olmayan etiket olarak açıklanan <br/> etiketi kullanılmadığı sürece alt satıra geçiş söz konusu değildir. Buradaki alt satıra geçişin sebebi tarayıcı ekranının küçültülmesidir. Ekran büyüdüğünde alt satıra geçiş farklı bir noktadan olacaktır. Yani tarayıcı genişliği boyunca metin devam edecek, tarayıcı ara yüzünün sağ kenarına geldiği zaman bir alt satıra geçiş olacaktır. Burada şekillerdeki metinlerin okunabilmesi için tarayıcı ekranı küçültülerek ekran görüntüsü alınmıştır. Aynı kodlar metin editöründe yazılıp tarayıcıda açıldığında, yukarıda açıklandığı şekilde tarayıcı ara yüzünün genişliği değiştirilerek bahsi geçen durumun detayları incelenebilir. Diğer taraftan <strong> etiketi örnek üzerinde gösterilmemiştir. Kendi hazırlayacağınız HTML dosyalarında buna ilişkin örnekler oluşturarak işe yararlığını test edebilirsiniz.

```

1 <html>
2 <head>
3   <title>Anasayfa</title>
4 </head>
5 <body>
6   <h3>Tarihten Geleceğe Bilim Köprüsü</h3>
7   <p>Yüksek Öğretim Kurumu tarafından hazırlanan raporlara göre
8   <b>İstanbul Üniversitesi</b> Türkiye’de <u>en çok bilimsel yayın yapan
9   üniversite</u> konumunda. Shanghai’daki Jiao Tong Üniversitesi
10  tarafından hazırlanan “<i>Dünyanın En İyi 500 Üniversitesi</i>”
11  listesinin içinde de yer alan İstanbul Üniversitesi, Sosyal,
12  Fen ve Sağlık Bilimleri alanlarında öncü çalışmalara imza atan
13  akademisyenleri bünyesinde barındırıyor.</p> İstanbul Üniversitesi
14  Beyazıt’taki ana yerleşkesi daha önce Harbiye Nezareti olarak
15  kullanılmaktaydı. 12 Eylül 1923 tarihinde bu bina İstanbul Üniversitesine verilmiştir.
16  </body>
17 </html>

```

Şekil 39. Yazı stiline ilişkin kullanılan etiketler



Şekil 40. Şekil 39’da kullanılan etiketler olmasa web tarayıcıda oluşacak görüntü



Şekil 41. Şekil 39’da yer alan HTML kodlarının web tarayıcıdaki görüntüsü

Bir diğer metin düzenlemesi liste yapısının metin içerisinde kullanılmasına yöneliktir. Liste yapısı bilindiği üzere bazı ifadelerin maddeler halinde belirtilmesi için kullanılmaktadır. Burada bazı detaylar söz konusudur. Bazı liste yapıları sırasız olup sadece nokta, yıldız gibi çeşitli karakterler ile belirtilirken, bazı yapılar sıralı olduğundan rakamlar, roma rakamları, harfler gibi belirli sırayı takip eden karakterler ile ifade edilir. Diğer taraftan liste yapısının içerisinde alt liste yapısı da kullanılabilir. Dolayısıyla burada da hizalamaya dikkat edilmesi gerekmektedir. Sırasız (*unordered*) liste yapısı için `<ul>` ve sıralı (*ordered listed*) liste yapısı için `<ol>` etiketleri kullanılmaktadır. Her iki etiket de kapanışı olan etiketlerdir. Metnin bir liste yapısı içerisinde yazılacağı `<ul>` veya `<ol>` etiketleri ile belirtildikten sonra, liste içerisinde yer alacak her bir madde için `<li>` etiketi ayrı ayrı açılıp kapatılmalıdır. Sırasız ya da sıralı liste yapısının bazı farklı karakterler ile ifade edildiği belirtilmişti. Burada hangi karakterlerin kullanılacağı “type” parametresi ile belirlenmektedir. Bu parametrenin sırasız liste yapısı için alacağı ve sıklıkla kullanılan parametre değerleri “circle”, “square” ve “disc” şeklinde, sıralı liste yapısı için alacağı parametre değerleri ise “A”, “a”, “i” şeklinde sıralanabilir. Eğer sıralı liste yapısı için herhangi bir parametre ve değeri belirtilmez ise 1,2,3 ve devamı şeklinde liste numaralandırılır. Liste yapısı herhangi bir karakter ile göstermeksizin oluşturulmak isteniyorsa öncelikle liste yapısının başladığını gösteren `<dl>` etiketinin açılıp kapanmasını, bu açılış ve kapanış arasında ise listenin her bir maddesi için `<dd>` etiketinin açılıp kapatılması gerekmektedir. Şekil 42 ve Şekil 43’te ilgili etiketlerin kullanımına yönelik HTML kodları ve bu kodlara karşılık web tarayıcıda elde edilen ekran görüntüleri yer almaktadır. Herhangi bir karakter kullanmaksızın liste yapısının oluşturulmasına örnek teşkil etmesine ilişkin listenin bir kısmı buna göre düzenlenmiştir. Ancak gerçek tasarımlarda tasarım ilkeleri bölümünde de belirtildiği üzere genel akış ve bütünlüğün bozulmaması adına böyle bir hata yapılmaması gerektiği hususu ayrıca vurgulanmalıdır. İç içe liste yapısında ise `<ol>`, `<ul>` veya `<dl>` ile yeni bir alt liste oluşturulur ve ilgili yani ana liste yapısında `<li>` ile belirtilen bir liste maddesinin altına eklenir. Burada herhangi bir karışıklığa mahal bırakmamak ve de sonra başka biri tarafından kodların okunurken



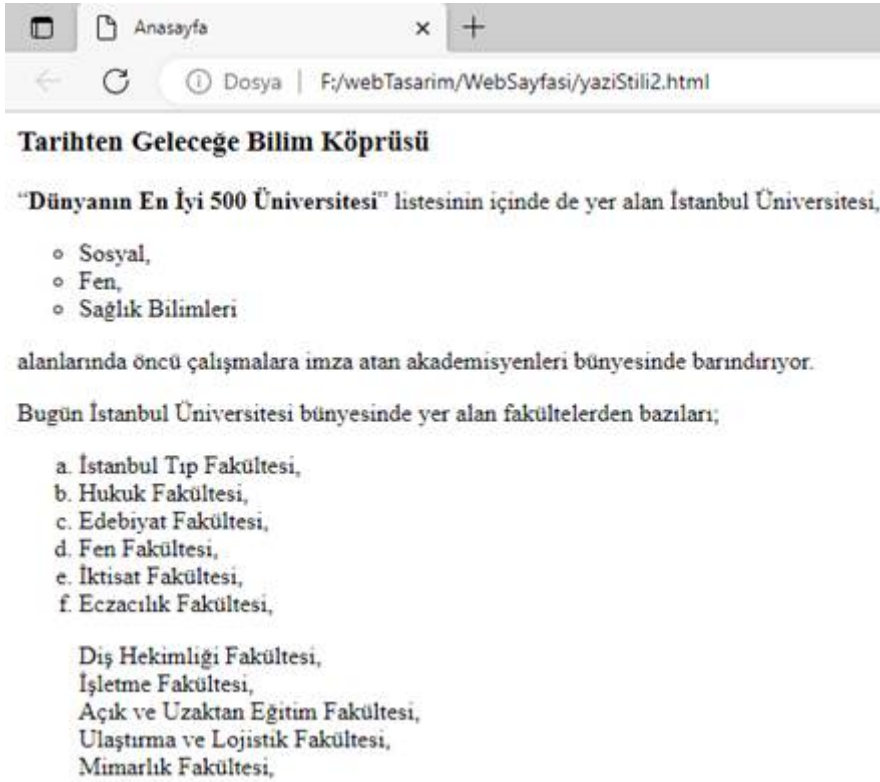
anlaşılır olması adına girintilemelerin doğru yapılması önemlidir. Bu noktada daha önce de belirtildiği üzere HTML’ye özgü metin editörlerinin önemi bir kez daha ortaya çıkmaktadır.

```

1 <html>
2 <head>
3   <title>Anasayfa</title>
4 </head>
5 <body>
6   <h3>Tarihten Geleceğe Bilim Köprüsü</h3>
7   <p><strong>Dünyanın En İyi 500 Üniversitesi</strong>" listesinin içinde de yer alan İstanbul Üniversitesi,
8   <ul type="circle">
9     <li>Sosyal,</li>
10    <li>Fen,</li>
11    <li>Sağlık Bilimleri</li>
12  </ul>
13  alanlarında öncü çalışmalara imza atan akademisyenleri bünyesinde barındırıyor.</p>
14  Bugün İstanbul Üniversitesi bünyesinde yer alan fakültelerden bazıları:
15  <ol type="a">
16    <li>İstanbul Tıp Fakültesi,</li>
17    <li>Hukuk Fakültesi,</li>
18    <li>Edebiyat Fakültesi,</li>
19    <li>Fen Fakültesi,</li>
20    <li>İktisat Fakültesi, </li>
21    <li>Eczacılık Fakültesi, </li>
22  </ol>
23  <dl>
24    <dd>Dış Hekimliği Fakültesi, </dd>
25    <dd>İşletme Fakültesi, </dd>
26    <dd>Açık ve Uzaktan Eğitim Fakültesi,</dd>
27    <dd>Ulaştırma ve Lojistik Fakültesi, </dd>
28    <dd>Mimarlık Fakültesi, </dd>
29  </dl>
30 </body>
31 </html>

```

Şekil 42. Liste yapısı kullanımına ilişkin ilgili etiketlerin kullanımı.



Şekil 43. Şekil 42’de yer alan kodun web tarayıcıdaki görüntüsü

İç iç listesi yapısının kullanımına ilişkin örnek ve web tarayıcıda elde edilen ekran görüntüsü ise Şekil 44ve Şekil 45’te görülmektedir.

```

1 <html>
2 <head>
3   <title>Anasayfa</title>
4 </head>
5 <body>
6   <h3>Fakülteler ve Bölümleri</h3>
7   <ul type="circle">
8     <li>Mühendislik Fakültesi
9       <ol>
10        <li>Bilgisayar Mühendisliği</li>
11        <li>Elektronik Mühendisliği</li>
12        <li>Yazılım Mühendisliği</li>
13      </ol>
14    </li>
15    <li>Fen Fakültesi
16      <ol>
17        <li>Biyoloji</li>
18        <li>Fizik</li>
19        <li>Matematik</li>
20      </ol>
21    </li>
22  </ul>
23 </body>
24 </html>

```

Şekil 44. İç içe liste yapısında etiketlerin kullanımı



Şekil 45. Şekil 44'te yer alan örneğe ilişkin web tarayıcıdaki ekran görüntüsü

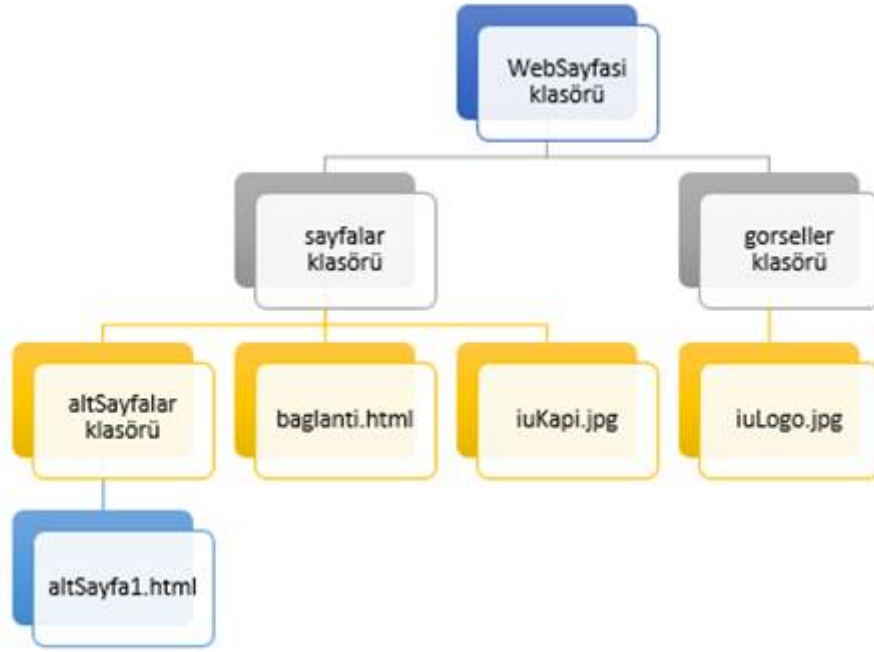
Sayfa içeriğinde kullanılan bir diğer düzenleme etiketi ise bağlantı etiketleridir. Daha önceki bölümde renk paleti kullanımına örnek olarak kısaca değinilen bir örnek verilmişti. Bu kısımda ise etikete ilişkin diğer detaylar da açıklanmıştır. Bağlantılar web siteleri içerisinde önemli unsurlardır. Çünkü bir web sitesi içerisinde alt birçok sayfa mevcuttur. Bunun yanı sıra çeşitli dosyalar da paylaşılması gerekebilir. Hem web sitesi içerisinde sayfalar arası bağlantı kurabilmek hem başka bir siteye erişim için bağlantı adresi vererek yönlendirme yapabilmek hem de paylaşılacak istenen dosyalar için dosya indirme bağlantısı oluşturmak için bağlantı etiketlerine ihtiyaç duyulmaktadır. Hatta eğer web sitesi kullanıcısının site içerisindeki bir görselin orijinal görseline erişimini sağlamak için de yine bu etiketlerden yararlanılmaktadır. Kullanılabilecek etiketlerden birisi <a> etiketidir.

Bu etiket içerisinde çeşitli parametreler ve parametre değerleri söz konusudur. İlgili yönlendirmeleri istenildiği şekilde yapabilmek için bu parametre ve değerlerinin etiket içerisinde belirtilmesi gerekmektedir. Etiket kullanımına yönelik kod örneklerine geçilmeden önce bu parametrelere ve değişkenlerine ilişkin açıklamalara ihtiyaç vardır. Bu parametrelerden ilki hiper referans (*hyper ref*) yani "href"tir. Bu parametreye değer olarak bağlantının gerçekleştirileceği adresin yazılması gerekmektedir. Etiket parametresi olarak ilgili bağlantı adresi verilirken, etiketin açılış ve kapanışı arasında web tarayıcıda gözüken ara yüzde bu linki ifade eden metin yazılmalıdır. Bir diğer ifade ile <a href="...">Buraya Tıklayınız</a> şeklinde bir kullanım ile kullanıcıyı bağlantıya yönlendirmek için bir metin paylaşılmalıdır. Kullanıcı ekranda sadece "Buraya Tıklayınız" ifadesini görecektir. Web tarayıcı ara yüzünde metin, altı çizgili şekilde ekrana yansıtacaktır. Bu

da o metne bağlantı özelliği kazandırıldığının göstergesidir. Ancak, web tarayıcı kullanıcı tarafında metinleri <u> etiketi ile de altı çizgili şekilde yazdırabildiğimizden her altı çizili metnin bir bağlantı özelliği taşıdığı söylenemez. Diğer taraftan, parametre değeri olarak elbette herhangi bir web sayfasının adresi yazılabileceği gibi kullanıcı, eposta gönderimi için de yönlendirilebilir. Bunun için parametre değerinin başında “mail to:” ifadesi konmalı ve değer devamında da mail adresi belirtilmelidir. Bu ifade belirtilen eposta adresine eposta gönder demektir. Bir dosya transferi için ilgili bağlantıya yönlendirme yapılmak isteniyorsa bu sefer parametre değerinin başına “ftp:” eklenip devamında dosya transfer bağlantısı belirtilmelidir.

Parametre ve değerine ilişkin bir diğer husus ise belgeye/görsele ya da aynı web sitesi içerisindeki başka bir sayfaya yönlendirecek bağlantı adresleri ile ilgilidir. Artık bağlantı ile ilgili etiketler üzerinden işlem yapılabileceğinden bu noktada izinleme ile ilgili bazı bilgilerin paylaşılması gerekmektedir. Web siteleri yayınlandığında içeriğine ilişkin tüm HTML dosyaları ile birlikte kullanılan tüm görseller ve diğer ilişkili dosyalar da yine sunucuya yüklenmektedir. Örneğin web sayfalarından birinde görüntülenecek bir görsel sunucuya yüklenmediği takdirde o sayfada ilgili görselin bulunduğu yerde hata alınacaktır. Dolayısıyla web sitesi içerisinde kullandığımız ya da bağlantı verdiğimiz dosyalar, web sitemizin alan adına (adresine) karşılık ayrılan dosya alanı içerisinde yüklü olmalıdır. Örneğin “www.fkocoglu.com” şeklinde adres (fkocoglu şeklinde alan adı ve .com şeklinde alan adı uzantısı) ve sunucu hizmeti aldığımızı varsayalım. Diğer taraftan, kitabın önceki bölümlerinde sayfa arka planına eklemek üzere kullanılan “iuKapi.jpg” dosyasını hatırlayınız. Bu dosya dahil web sitesine ilişkin tüm dosyaları “icerik” adlı bir klasör altında topladığımızda ve bu klasörü sunucuya yüklediğimizde ilgili resim dosyasının adresi “www.fkocoglu.com/icerik/iu.Kapi.jpg” olacaktır. Yine ana sayfaya ilişkin HTML ile yazılmış anasayfa.html dosyasının adresi de “www.fonaykocoglu.com/icerik/anasayfa.html” olacaktır. Tekrar bağlantı verme etiketine dönülürse, kendi sitemiz dışında başka web sayfaları ile ilgili adresleri tam olarak “href=” parametresinde değer olarak giriyorken, kendi sitemize ilişkin hem .html uzantılı dosyalar hem de bu görsel dosya sunucuda aynı klasörde (icerik klasörü) yer aldığından, bahsi geçen durumda adresin tamamen etiket içerisinde parametre değeri olarak yazılmasına gerek yoktur. Yine “iuKapi.jpg” görseli örneğinden devam edelim. Kullanıcının hazırladığımız web sayfası içerisinde vereceğimiz bağlantıya tıkladığında bu görseli tarayıcıda açıp görüntülemesini isteyelim. Bunun için kullanacağımız bağlantı etiketinde “href=” parametresine değer olarak “iuKapi.jpg” girdiğimizi varsayalım. Kullanıcı ilgili bağlantıya tıkladığında, etiket ve parametre değeri olarak verilen “iuKapi.jpg” değeri web tarayıcı tarafından okunacak, başka bir dizin (klasör adı) belirtilmediğinden okuma yaptığı .html uzantılı dosyanın bulunduğu klasörde bu .jpg uzantılı dosyayı arayacaktır. Belirtilen görsel dosya, tarayıcının o an okuduğu .html uzantılı dosya ile aynı klasör içerisinde yer aldığından web tarayıcı bu dosyayı bulacak ve web tarayıcı ara yüzünde kullanıcının ekranına ilgili görsel yansıyacaktır. Ancak varsayalım ki sayfalara ilişkin HTML ile yazılmış dosyaları “icerik” klasöründe, resimleri “gorseller” klasöründe, dokümanları “dokuman” klasörü altında sunucuya yükledik. Yani görselimiz ve sayfamıza ait .html dosyamız farklı dizinlerde (klasörlerde) yer alıyor olsun. Bu durumda, “href” parametresine klasör adını (yani dosya dizinini) belirtmeden sadece “iuKapi.jpg” şeklinde görsele ilişkin dosya ismini yazarsak web tarayıcı .html uzantılı dosyanın olduğu klasörde arama yapacak, ilgili dosyayı bulamayacak ve kullanıcıya sunulan bağlantı çalışmıyor olacaktır. Bu noktada alt ve üst dizinler (yani alt ve üst klasörler) kullanılarak “href” parametresine girilecek değerler özel şekilde belirtilir. Tarayıcı bu değeri okuduğunda alt ya da üst dizine geçmesi gerektiğini anlamaktadır. Dolayısıyla okuduğu .html uzantılı dosyanın bulunduğu klasörde (dizinde) değil, ilgili dizinlerde belirtilen dosyayı aramaya başlar. Bu açıklamalara ilişkin izinleme (klasörlendirme) hiyerarşisi, bilgisayardaki görüntüsü, bağlantı etiketine ilişkin kodlar, bağlantı etiketi kullanılan sayfanın web tarayıcıdaki görüntüsü ve her bir bağlantıya tıkladığında gelen ekran görüntüsü sırasıyla Şekil 46, Şekil 47, Şekil 48, Şekil 49 ve Şekil 50’de verilmiştir.



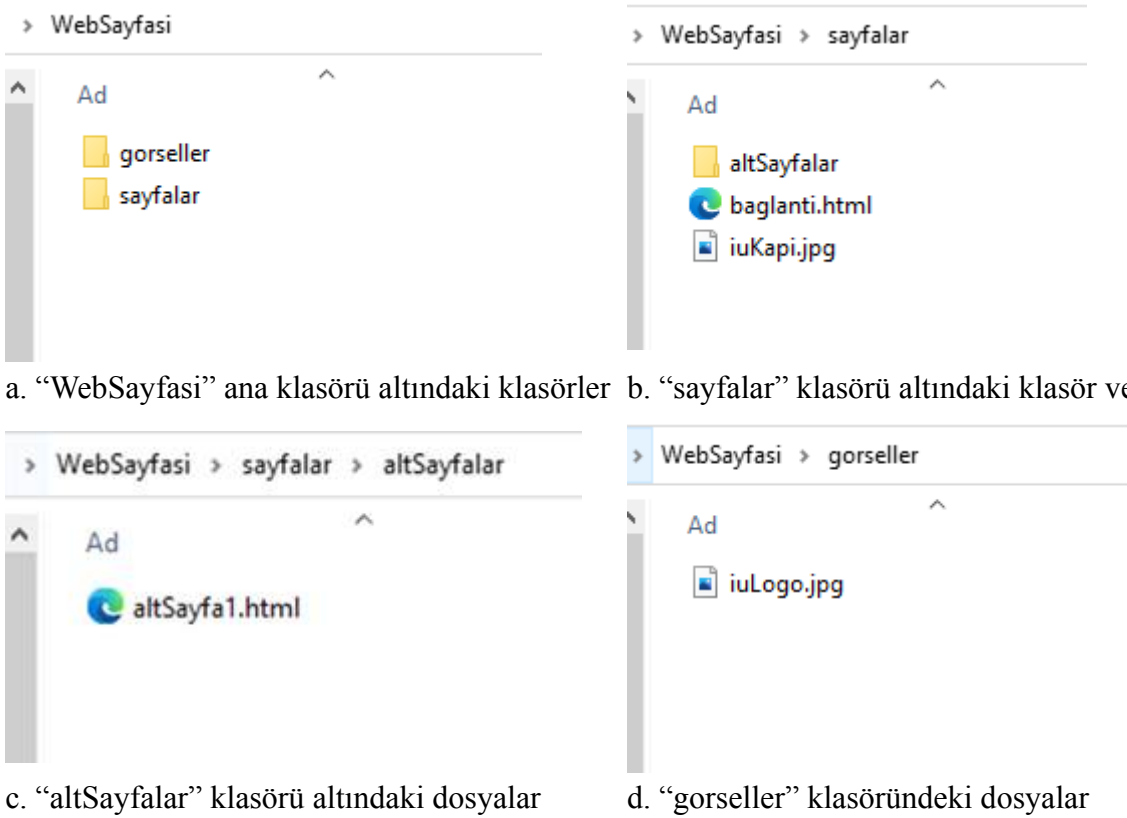


Şekil 46. Örnekte kullanılan dizinleme hiyerarşisi

Dizinleme (ya da klasör yapısının oluşturulması) şu şekilde yapılmıştır:

- Web sitesinin tüm dosyalarının yer alacağı “WebSayfasi” adı verilen bir klasör oluşturulmuştur.
- “WebSayfasi” ana klasörünün altında ise web sitesi içeriğinde kullanılacak tüm görsellerin yer aldığı “gorseller” alt klasörü ve web sitesinde yer alacak tüm sayfalara ait .html uzantılı dosyaların yer aldığı “sayfalar” klasörü oluşturulmuştur.
- “sayfalar” klasörü içerisinde ana sayfa olarak kabul edilecek bir “baglanti.html” dosyası oluşturulmuştur. Bu dosyada yazılan HTML kodları Şekil 48’de görülmektedir. Bu noktada belirtmek gerekir ki, normal bir web sitesi oluşturulacağı durumda ana sayfa için isimlendirmenin “anasayfa.html” şeklinde yapılması daha uygundur. Ancak burada bağlantı etiketi konusu anlatıldığı için dosyanın adı “baglanti.html” şeklinde düzenlenmiştir. Aynı klasörde yer alan görsele bağlantı verme örneğini gösterebilmek üzere bu klasöre “iuKapi.jpg” dosyası da eklenmiştir. Normal şartlar altında bu dosya da “gorseller” klasörü altında yer almalıdır.
- “sayfalar” klasörü altında ana sayfa dışındaki diğer alt sayfaları bir arada tutmak için “altSayfalar” klasörü açılmıştır. Bu klasör içinde “altSayfa1.html” dosyası yer almaktadır.
- “gorseller” klasörü içerisinde web sitesinde kullanılacak görseller bir araya getirilmiştir. Şu aşamada “iuLogo.jpg” şeklinde tek bir görsel dosya klasörde yer almaktadır (“iuKapi.jpg” ile ilgili yukarıdaki bilgiyi göz önünde bulundurmaya unutmayınız).

Şekilde yer alan örneği incelerken bilgisayarımızda klasörlerin bulunduğu yerdeki dizin yapısını (klasör adresini; örneğin Şekil 47.b’de WebSayfasi > sayfalar) inceleyip, yukarıdaki hiyerarşi ile karşılaştırınız. Bunun yanı sıra Şekil 49’da ekranda görüntülenen bağlantılara tıklandığında açılan sayfaların web tarayıcı adres çubuğundaki dizinleri de Şekil 50’deki görsellerden inceleyerek (örneğin, c’de yer alan görselin tarayıcının adres çubuğunda yer alan adresi “WebSayfasi/gorseller/iuLogo.jpg”) yine verilen hiyerarşi ile karşılaştırınız.



Şekil 47. Bağlantı etiketi için kullanılan klasörlerin dizinlenmesi

```

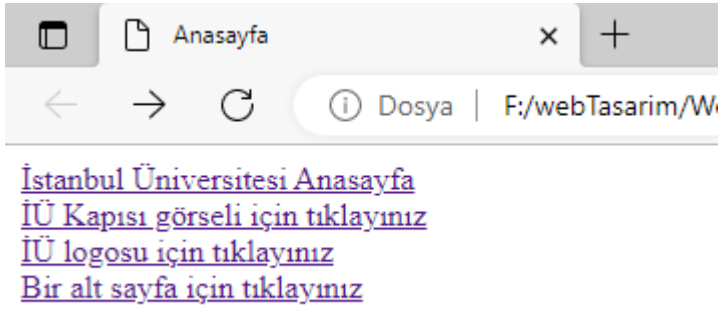
1  <html>
2  <head>
3    <title>Anasayfa</title>
4  </head>
5  <body>
6    <a href="https://www.istanbul.edu.tr/tr/">İstanbul Üniversitesi Anasayfa</a><br/>
7    <a href="iuKapi.jpg">İÜ Kapısı görseli için tıklayınız</a><br/>
8    <a href="../gorseller/iuLogo.jpg">İÜ logosu için tıklayınız</a><br/>
9    <a href="altSayfalar/altSayfa1.html">Bir alt sayfa için tıklayınız</a><br/>
10 </body>
11 </html>

```

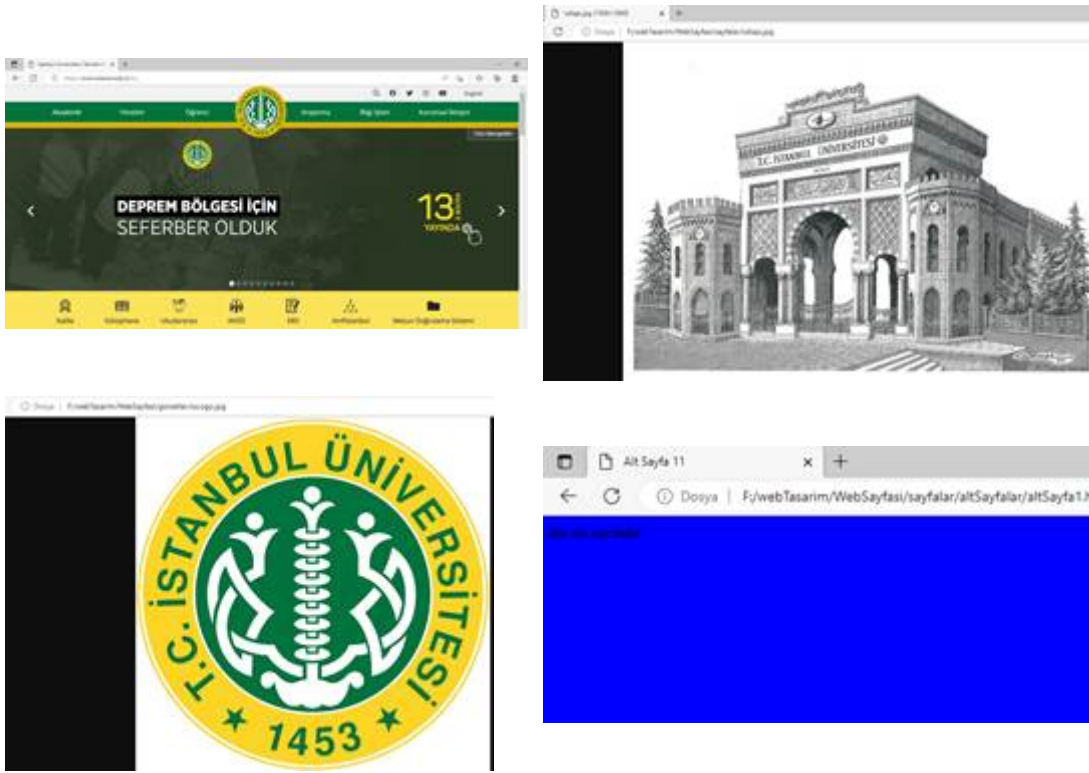
Şekil 48. Bağlantı etiketi kullanımı

Bu kodları özellikle alt ve üst dizinlerin belirtildiği kısımlar açısından açıklayalım. <body> etiketi altında <a> etiketi kullanılarak 4 farklı bağlantı tanımlanmıştır. İlk tanımlanan bağlantı başka bir web sayfasına yönlendirme yapmaktadır ve “href” parametresine yönlendirme yapılan sayfanın bağlantı adresi değer olarak girilmiştir. İkinci açılan <a> etiketi ile, kodların yazılı olduğu “baglanti.html” dosyası ile aynı dizinde yer alan başka bir ifade ile aynı klasör altında bulunan bir görsele bağlantı tanımlanmıştır. Görüldüğü üzere görsel dosya adı dışında bir dizin belirtilmemiştir. Üçüncü <a> etiketi ile ise kodların yazılı olduğu “baglanti.html” dosyası ile farklı dizinde yer alan başka bir ifade ile farklı klasör altında bulunan bir görsele bağlantı tanımlanmıştır. Burada “../gorseller/iuLogo.jpg” href parametre değerinde dikkat edilmesi gereken husus “../” ifadesidir. Bu ifade bir üst dizine çıkılması gerektiğini belirtmektedir. Dizin hiyerarşisinden de görüleceği üzere “baglanti.html” dosyası “sayfalar” klasöründe yer almaktadır. Ancak bağlantı verilmek istenen “iuLogo.jpg” dosyası ise “WebSayfasi” adlı klasörün altında bulunan “gorseller” klasöründedir. Bu nedenle “baglanti.html” dosyası web tarayıcı tarafından okunurken “../” ifadesini gördüğü anda “sayfalar” klasöründen bir üst klasör olan “WebSayfasi” klasörüne çıkması gerektiğini anlar. “../” ifadesinden sonra ise görüldüğü üzere “gorseller/iuLogo.jpg” uzantısı yer almaktadır. Yani bu aşamada “WebSayfasi” klasöründen, bu klasörün altında yer alan “gorseller” klasörüne geçmesi ve ilgili görsele bu klasörden ulaşması gerektiğini anlayacaktır. Dördüncü <a> etiketi ile yine benzer olarak kodların yazılı olduğu “baglanti.html” dosyası ile farklı dizinde yer alan başka bir ifade ile farklı klasör altında bulunan bir dosyaya bağlantı tanımlanmıştır. Ancak buradaki fark sadece görsel yerine bir dosyaya bağlantı kurmak değil, üst klasörde değil bir alt klasörde yer alan dosyaya bağlantı kurulacak olmasıdır. Dizin hiyerarşisinden de görüleceği üzere “baglanti.html” dosyası “sayfalar” klasöründe yer almaktadır. Bağlantı kurulmak istenen “altSayfa1.html” ise

“sayfalar” klasörünün altında yer alan “altSayfalar” klasöründe yer almaktadır. Bu nedenle href parametresi olarak “altSayfalar/altSayfa1.html” ile ilgili dizin işaret edilmiştir. Web tarayıcının ara yüzünde görüntülenen sayfa alanında ise <a> etiketinin açılıp ve kapanışı arasında yazılan metinlerin görüntülediğine dikkat edilmelidir. Bağlantı için verilen href parametre değerleri (bağlantı adresleri) tarayıcıda gözükmemektedir. Diğer taraftan <a> etiketinin her satırda kapanışından sonra <br/> etiketi konulmasaydı ekranda görüntülenen tüm metnin alt alta değil de yan yana olacağı unutulmamalıdır.



Şekil 49. Şekil 48’deki HTML ile yazılan kodların web tarayıcıdaki görüntüsü



Şekil 50. Şekil 49’de yer alan ekran görüntüsünde yer alan bağlantılara tıklandığında elde edilen sonuçlar

Her ne kadar <a> etiketinin açılış ve kapanışı arasında metin kullanılmış olsa da burada bir görsel de kullanılabilir ve görsele tıklayan kullanıcı ilgili bağlantıya gidebilirdi. Daha önce kapanışı olmayan etiketler kapsamında <img> etiketi görülmüştür. Bu etiket bulunduğu noktaya bir görselin eklenmesini sağlamaktadır. Şekil 51’de ilgili örnek görülmektedir. Buna göre yükseklik ve genişliği verilen görsel <a> etiketinin açılış ve kapanışı arasında <img> etiketi ile eklenmiştir. <img> etiketi içerisinde yer alan “src” parametresinin aldığı değer yine bir dizini gösterdiğinden bu değerın yazılma mantığı “href” için ifade edilenlerle aynıdır.

```

1 <html>
2 <head>
3   <title>Anasayfa</title>
4 </head>
5 <body>
6   <a href="https://www.istanbul.edu.tr/tr/" target="_blank">
7     
8   </a><br/>
9   Yukarıdaki görsele basarak İstanbul Üniversitesi <br/>
10  web sitesine erişebilirsiniz.
11 </body>
12 </html>

```

Şekil 51. Bağlantı etiketinin farklı özelliklerinin kullanımı

Aynı HTML kodlarından da görüleceği üzere <a> bağlantı etiketi içerisinde “target” isimli bir parametre daha söz konusudur. Bu parametre kullanıcı web tarayıcısındaki sayfa alanında tanımlanan bağlantıya tıkladığında bağlantının nerede açılacağını belirler. “target” parametresi “\_self”, “\_blank”, “\_parent”, “\_top” değerlerini alabilir. Bu değerler sırasıyla bağlantının bulunulan sekmede, farklı bir sekmede, ana çerçevede, pencerenin tam gövdesinde açılmasını sağlamaktadır. İlgili örnekte “\_target” değeri kullanılmıştır.



Şekil 52. Şekil 51’de yer alan HTML kodlarının web tarayıcısındaki görüntüsü

Sayfa içeriğinde en önemli öğe aslında metinlerdir. Metinler ile ilgili yazı stiline ilişkin özellikler <font> etiketi ile gerçekleştirilmektedir. Yazı stili ile ilgili liste yapısından yukarıda bahsedilmişti. Bu noktada yazı büyüklükleri, yazı tipi ve benzeri diğer özellikler incelenecektir. <font> etiketinin açılış ve kapanışı arasındaki metin bu etiket içerisinde belirtilen parametre ve parametre değerlerine göre biçimlendirilmektedir. Etiket parametrelerinden birisi “face”dir. Bu etiket yazı tipinin ne olması gerektiğini belirler. Yani arial, times new roman, cambria vb. stillerinden birinin belirlenmesini sağlamaktadır. Bir diğer parametre elbette ki metin büyüklüğünü belirlemeye yarayan “size” parametresidir. 1-7 arasında kullanılacak bir rakam ile metnin büyüklüğü belirlenebilir. Punto cinsinden ifade etmek gerekirse bu rakamlar sırasıyla 8, 10, 12, 14, 16, 18, 36 punto büyüklüğünü temsil etmektedir. Herhangi bir font büyüklüğü belirtilmediği takdirde web tarayıcılar otomatik olarak bu değeri 3 olarak yani 12 punto olarak kabul eder. “color” parametresi adından da anlaşılacağı üzere metin rengini belirlemek için kullanılmaktadır. Metin rengi ile ilgili düzenleme <body> içerisinde de yapılabilmektedir. Daha önce gösterilen örneklerde sayfa arka plan renginin <body> etiketi ile nasıl değiştirildiği gösterilmiştir. Metin ile ilgili renk belirlemelerinde ise “text” parametresi kullanılır. Daha önce renk paleti kullanımı ve renk kodları kısmında ifade edilen durum burada da geçerlidir. Belirlenen rengin renk kodu veya ismi “color” ya da “text” parametresine değer olarak atanabilir. Metin için genel olarak aynı font büyüklüğü belirlenip sadece metnin belirli yerlerinde <font> etiketi ile değişiklik yapılmak istendiğinde <basefont> etiketi kullanılabilir. Bu etiket içerisinde kullanılacak “size” parametresi ile büyüklüğün değeri belirlenebilir.



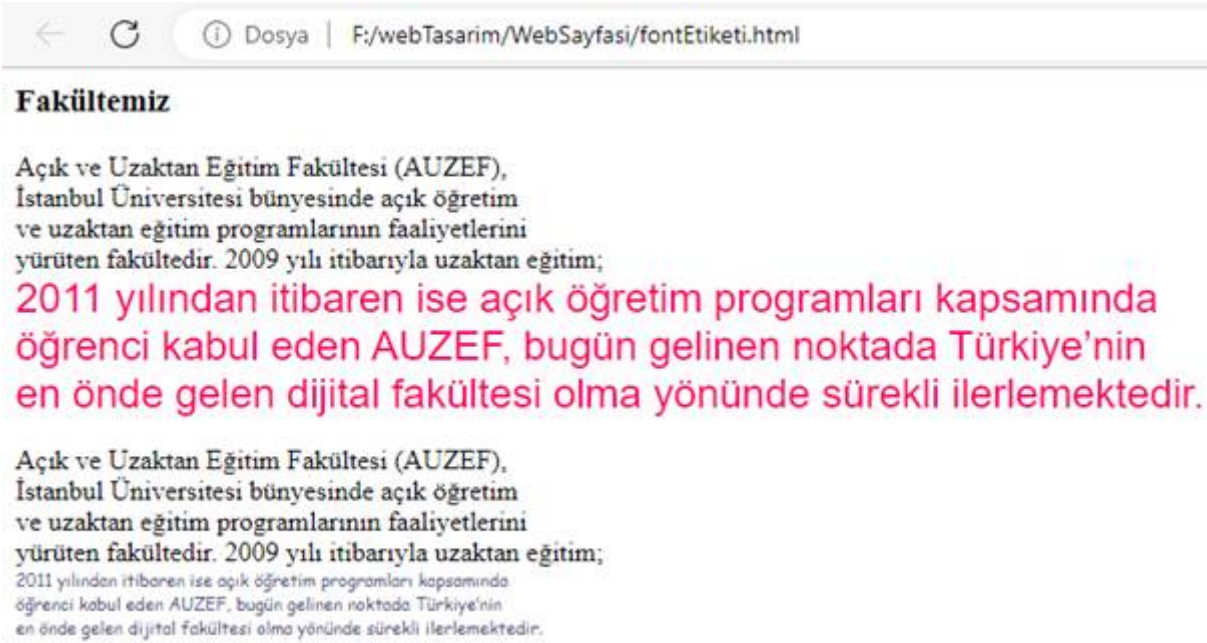
```

1 <html>
2 <head>
3 <title>Anasayfa</title>
4 </head>
5 <body>
6 <h3>Fakültemiz</h3>
7 <basefont size="3">
8
9 <p>Açık ve Uzaktan Eğitim Fakültesi (AUZEF), <br/>
10 İstanbul Üniversitesi bünyesinde açık öğretim <br/>
11 ve uzaktan eğitim programlarının faaliyetlerini <br/>
12 yürüten fakültedir. 2009 yılı itibarıyla uzaktan eğitim: <br/>
13 <font face="Arial" size="3" color="#FF0065">2011 yılından itibaren ise açık öğretim programları kapsamında <br/>
14 öğrenci kabul eden AUZEF, bugün geline nokta Türkiye'nin <br/>
15 en önde gelen dijital fakültesi olma yönünde sürekli ilerlemektedir. </p>
16 </font>
17
18 <p>Açık ve Uzaktan Eğitim Fakültesi (AUZEF), <br/>
19 İstanbul Üniversitesi bünyesinde açık öğretim <br/>
20 ve uzaktan eğitim programlarının faaliyetlerini <br/>
21 yürüten fakültedir. 2009 yılı itibarıyla uzaktan eğitim: <br/>
22 <font face="comic sans ms" size="1" color="#525289">2011 yılından itibaren ise açık öğretim programları kapsamında <br/>
23 öğrenci kabul eden AUZEF, bugün geline nokta Türkiye'nin <br/>
24 en önde gelen dijital fakültesi olma yönünde sürekli ilerlemektedir. </p>
25 </font>
26 </body>
27 </html>

```

Şekil 53. Metin düzenlemesine ilişkin kullanılan etiketler

Yukarıdaki kod bloklarında metnin ana yazı boyutu <basefont size="3"> etiket ile belirlenmiştir. Özellikle 13 ve 23. satırlarda <font> etiketi ile metin içerisinde başta belirtilen font özelliklerinden daha farklı tanımlamalar yapılmıştır. Temel font düzeni ve belirtilen bu satırlardaki yapılan farklı tanımlamalar ile birlikte metinde yazı biçimi, büyüklüğü ve renginden ne gibi değişiklikler olduğunu Şekil 54'te görebilirsiniz. Daha iyi görülebilmesi için aynı metin kullanılarak iki paragraf oluşturulmuştur. Her bir paragraf da metin düzenlemesi ile birlikte kendi içerisinde iki farklı stilde oluşmuştur. Paragrafların ilk bölümündeki metnin stilleri <basefont>, ikinci bölümündeki stilleri ise <font> etiketi ve farklı parametre değerleri ile elde edilmiştir.



Şekil 54. Şekil 53'te verilen HTML kodlarının web tarayıcıdaki görüntüsü

Yine metin, tarayıcı ara yüzünde büyük puntolarla görüntülensin isteniyorsa <big> ya da küçük puntolarla görüntülensin isteniyorsa <small> etiketleri kullanılabilir. Bu etiketlerin açılış ve kapanışları arasındaki metin bu etiket özellikleri ile biçimlendirilmiş olacaktır. Bu etiketler arka arkaya kullanıldığında etiketin sağladığı yazı boyutunu büyültme ya da küçültme özelliği tekrar tekrar uygulanır. Dolayısıyla yazı gittikçe büyür ya da küçülür. İlişkili örnek ve ekran görüntüsü Şekil 55 ve Şekil 56'da verilmiştir.

```

1 <html>
2 <head>
3   <title>Anasayfa</title>
4 </head>
5 <body>
6   <h3>Fakültemiz</h3>
7   <basefont size="3">
8
9   <p><big>Açık ve Uzaktan Eğitim Fakültesi (AUZEF),</big> <br/>
10  <big><big>İstanbul Üniversitesi bünyesinde açık öğretim</big></big> <br/>
11  <big><big>ve uzaktan eğitim programlarının faaliyetlerini</big></big></big> <br/>
12  <small>yürüten fakültedir. 2009 yılı itibarıyla uzaktan eğitim:</small> <br/>
13  <small><small>2011 yılından itibaren ise açık öğretim programları kapsamında</small></small><br/>
14  <small><small><small>öğrenci kabul eden AUZEF, bugün gelinen noktada Türkiye'nin</small></small></small> <br/>
15  en önde gelen dijital fakültesi olma yönünde sürekli ilerlemektedir.</p>
16 </body>
17 </html>

```

Şekil 55. Metin boyutu büyütme ve küçültme



Şekil 56. Şekil 55'te verilen kod yapısının web tarayıcıdaki görüntüsü

Bilindiği gibi metinler sadece düz yazılardan oluşmaz bazen matematiksel bir ifadenin verilmesi ve açıklanması gerekebilir. Bu ifadelerde alt ve üst indis kullanımı oldukça yaygındır. HTML ile alt ve üst indis biçimlendirmesi yapabilmek için kullanılan etiket sırasıyla <sub> ve <sup>'tur. Etiketlerin açılış ve kapanışları arasındaki metin bu etiketlere göre biçimlendirilmiş olur.

```

1 <html>
2 <head>
3   <title>Anasayfa</title>
4 </head>
5 <body>
6   <h3>Matematiksel İfadelerde İndis Kullanımı</h3>
7   <font size="6">
8     <math>x</math><font size="-1"><sub>i</sub></font>+<math>y</math><sub>j</sub></font><sup>2</sup>
9   </font>
10 </body>
11 </html>

```

Şekil 57. İndis etiketlerinin kullanımı



Şekil 58. Şekil 57'de yer alan kodların web tarayıcıdaki ekran görüntüsü

Burada dikkat edilirse  $x$  değişkenine ait  $i$  indisini göstermek için kullanılan `<sub>` etiketinin açılışından önce `<font size="-1">` şeklinde ana font yapısının dışında ek olarak yazı stiline ilişkin bir düzenleme yapılmıştır. Burada yeni font büyüklüğünü belirtmek yerine “-1” yazılarak var olan font büyüklüğünün 1 birim düşürülmesi sağlanmıştır. Şekil 58’de yer alan ekran görüntüsünde de  $i$  indisinin  $y$  değişkenine ait  $j$  indisinden daha küçük olduğu görülecektir.

Bölüm kapsamında son olarak görsel ekleme ve biçimlendirme etiketi olan `<img>` etiketine ilişkin bazı detayları verelim. `<img>` etiketinin aldığı parametre değerleri arasında daha önce de belirtildiği üzere kullanılacak görselin dizinini ve dosya adını belirten “src”, genişliğini piksel cinsinden belirten “width” parametresi, yüksekliğini piksel cinsinden belirten “height”, kullanıcı web tarayıcısındaki görüntülenen sayfada görsel üzerine geldiğinde görsel ile ilgili bir açıklama balonunun oluşması için “alt” parametreleri mevcuttur. Görsel kenarında ayrıca bir kenarlık oluşturulması da mümkündür Bunun için kenarlığa ilişkin piksel değerinin belirtildiği “border” şeklinde bir parametresi daha bulunmaktadır. Metinler ile birlikte görsel kullanıldığında bilindiği üzere metin görselin sağ ya da solunda olacak şekilde konumlandırılabilir. Bu konumlandırmanın web sayfası içerisinde de benzer olarak yapılabilmesi için “align” parametresi kullanılmaktadır. Bu parametrenin alacağı değerler “left” ve “right” şeklindedir. Bu değerler görselin metnin solunda ya da sağında görüntülenmesini sağlar.

```

1 <html>
2 <head>
3   <title>Anasayfa</title>
4 </head>
5 <body>
6   <h3>Görsel Kullanımı</h3>
7   
8   <font size="3">
9     Burada metin ile birlikte görselin nasıl <br/>
10    konumlandırılacağı ve görsele ile ilgili biçimlendirmelerin<br/>
11    ilgili etiketle nasıl gerçekleştirileceğine ilişkin örnek verilmiştir.
12  </font>
13 </img>
14 </body>
15 </html>

```

Şekil 59. Görsel etiketi kullanımı



Şekil 60. Şekil 59’de yer alan kodların web tarayıcısındaki ekran görüntüsü

Verilen örnekler genellikle parametre değerlerinden bazılarının kullanılması ile oluşturulmuştur. Bütün değerleri denemek ve bunlar arasındaki farkları tek tek göstermek anlaşılabileceği üzere pek mümkün değildir. Bu nedenle dilediğiniz metin editörünü kullanarak HTML ile öğrendiğiniz etiketlerle .html uzantılı sayfalar oluşturup bunları tarayıcılarınızda görüntüleyiniz. Bahsi geçen farklı parametre değerlerini deneyiniz ve farklılıkları keşfediniz. Keza etiketler için farklı parametreler de söz konusu olabilir. Bu noktada kitap kapsamında sıklıkla kullanılan ve ileride yapacağımız örneklerde işimize yarayacakları belirtilmiştir. Ancak farklı kaynaklardan mümkün diğer parametreleri ve özelliklerini de ayrıca araştırmanız tavsiye edilmektedir.



## Bölüm Özeti

Bölüm kapsamında bir önceki bölümde başlanan HTML'ye özgü etiketlerin devamı açıklanmaya çalışılmıştır. Bir önceki bölümde temel yapıya ait etiketler söz konusu ilen bu bölümde özellikle sayfa içerisinde kullanılacak metin ve görsellere ilişkin düzenlemelerin gerçekleştirilebileceği etiketlerin kullanımına ilişkin açıklamalar yapılmıştır. Bu düzenlemeler arasında metin içi başlık kullanımları, liste yapılarının kullanımları (sıralı ve sırasız listeler, iç içe listeler), paragraf oluşturma, başka sayfalara, kendi web sitemizde yer alacak diğer sayfalara ya da görsel ve dosyalara bağlantı oluşturma, büyüklük, yazı biçimi, renk, alt ve üst indis gibi çeşitli metin özellikleri ve son olarak da görsel eklemeye yönelik detaylar paylaşılmıştır. Bölüm kapsamında ele alınan konulardan belki de en önemlilerinden birisi ise dizinlemedir. Dizinleme web sitesini doğru yapılandırma ve yayınlama aşamasında oldukça önemli bir konu başlığıdır. Dizinleme belirli bir mantık çerçevesinde yapılması gerekir. Alt öğelere kolay erişim sağlanması için klasörler kullanılmalı ve ilgili öğeler aynı klasörler altında toplanmalıdır. Özellikle kendi web sitelerimiz içerisinde bağlantılar oluştururken doğru dizin gösteriminin ilgili parametreye değer olarak girilmesi web sayfalarımızın düzgün görüntülenebilmesine olanak sağlamaktadır. Daha önce de ifade edildiği gibi tüm parametreleri ve değerleri tek tek ele almak ve bunlara ilişkin örnekleri göstermek mümkün değildir. Dolayısıyla mümkün olduğunca farklı kaynaklardan etiketler ve ilişkili parametreler incelenmelidir.

Burada bahsi geçen etiketler genellikle içeriğe yönelik düzenlemelerde kullanılmaktadır. Ancak ileride CSS konusunda da bazı özelliklerin kullanımını anlama hususunda kolaylaştırmış olacaktır. Yine ilerleyen bölümlerde sayfanın içeriğinin yanı sıra gerçek tasarımlar oluşturmak üzere tablolar, çerçeveler, formlar gibi yapısal özellikler ve bunlara ilişkin etiketler ele alınmaya başlanacaktır. Gerçekleştirilecek örnekler içerisinde bu bölümde açıklanan etiketler kullanılmaya devam edecektir. Ancak her biri yeniden açıklanmayacaktır. Bu nedenle her bir etiket parametreleri ile birlikte doğru bir şekilde anlaşılmalı ve öğrenilmelidir.

### Kaynakça

Seobility wiki (2023). Erişim Linki: [https://www.seobility.net/en/wiki/HTML\\_Special\\_Characters](https://www.seobility.net/en/wiki/HTML_Special_Characters), Erişim Tarihi: 05.03.2023

## Ünite Soruları

### Soru-1 :

Bir web sayfası web tarayıcıda görüntülendiğinde içerikte yer alan metin ve görsellerin biçimlendirilmesi ve düzenlenmesini sağlayan etiketler HTML ile yazılan kodlar içerisinde hangi ana etiket yapısı altında tanımlanır?

(Çoktan Seçmeli)

(•) - <head>

(•) - <body>

(•) - <title>

(•) - <h>

(•) - <meta>

### Cevap-1 :

<body>

**Soru-2 :**

<h1> etiketi ile metin içi başlık kaç punto ile yazılmış olur?

(Çoktan Seçmeli)

(•) - 8

(•) - 10

(•) - 16

(•) - 24

(•) - 36

**Cevap-2 :**

24

---

**Soru-3 :**

Sıralı liste yapısı oluşturmak için aşağıdaki etiketlerden hangisi öncelikle kullanılmalıdır?

(Çoktan Seçmeli)

(•) - <ul>

(•) - <ol>

(•) - <li>

(•) - <list>

(•) - <al>

**Cevap-3 :**

<ol>

---

**Soru-4 :**

~~<a href=""~~www.google.com">Buraya Tıklayınız HTML ile yazılmış kod bloğu ne işe yarar?

(Çoktan Seçmeli)

(•) - Ekranı Google arama motorunun adını yazar

(•) - Kodun yazıldığı web sayfası tarayıcıda görüntülendiğinde ekrana arama çubuğu getirir.

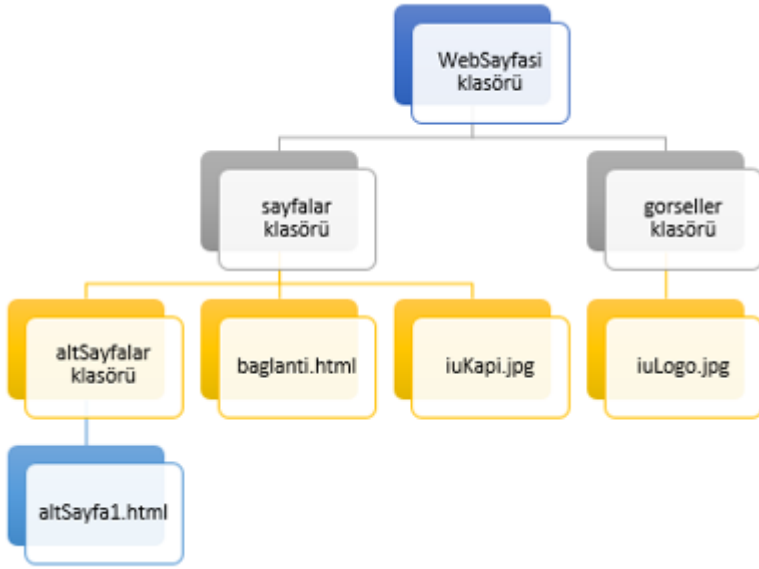
(•) - www.google.com linkine bağlantı oluşturulur.

(•) - www.google.com arama motorunda “Buraya Tıklayınız” kelime grubunu arar.

(•) - Hazırlanan web sayfasının Google arama motorunda görüntülenmesini sağlar.

**Cevap-4 :**

www.google.com linkine bağlantı oluşturulur.

**Soru-5 :**

Yukarıda yer alan klasör yapısına göre aşağıdakilerden hangisi “baglanti.html” dosyasında yazılan ve bu dosyadan iuKapi.jpg görsel dosyasına bağlantı oluşturulmasını sağlayan etiket kullanımıdır?

(Çoktan Seçmeli)

- (•) - `<a href="" altSayfalar/iuKapi.jpg> tıklayınız</a>`
- (•) - `<a href="" sayfalar/iuKapi.jpg> tıklayınız</a>`
- (•) - `<a href="" WebSayfasi/iuKapi.jpg> tıklayınız</a>`
- (•) - `<a href="" ../sayfalar/iuKapi.jpg> tıklayınız</a>`
- (•) - `<a href="" iuKapi.jpg> tıklayınız</a>`

**Cevap-5 :**

`<a href="" iuKapi.jpg> tıklayınız</a>`

**Soru-6 :**

Bağlantı etiketi içerisinde kullanılan “target” etiketinin aşağıdaki hangi parametre değeri, kullanıcı ekranda ilgili bağlantıya tıkladığında bağlantı sağlanacak sayfayı ayrı sekmede açılmasını sağlar?

(Çoktan Seçmeli)

- (•) - parent”
- (•) - new”

(•) - top”

(•) - blank”

(•) - self”

**Cevap-6 :**

blank”

---

**Soru-7 :**

“Metin için genel olarak aynı font büyüklüğü belirlenip sadece metnin belirli yerlerinde <font> etiketi ile değişiklik yapılmak istendiğinde \_\_\_\_\_ etiketi kullanılabilir.

Boşluğa aşağıdakilerden hangisi getirilmelidir?

(Çoktan Seçmeli)

(•) - <basefont>

(•) - <title>

(•) - <big>

(•) - <size>

(•) - <tag>

**Cevap-7 :**

<basefont>

---

**Soru-8 :**

Alt indis gösterimi için hangi etiket kullanılmalıdır?

(Çoktan Seçmeli)

(•) - <h>

(•) - <br/>

(•) - <sub>

(•) - <sup>

(•) - <font>

**Cevap-8 :**

<sub>

---

**Soru-9 :**

Bir görselin sayfa içerisindeki konumunu belirlemek için <img> etiketi içerisinde kullanılan parametre hangisidir?

(Çoktan Seçmeli)

(•) - href”

(•) - size”

(•) - target”

(•) - align”

(•) - font”

**Cevap-9 :**

align”

---