

BOOTSTRAP İLE DUYARLI WEB TASARIMI



İÇİNDEKİLER

- Bootstrap Nedir?
 - Bootstrap Paketi Neler İçerir?
 - Bootstrap Nasıl Çalışır?
- Bootstrap Kurulumu
- Bootstrap ile Duyarlı Web Sayfası Tasarımı
 - Duyarlı Web Tasarımı Kavramı
 - Duyarlı Web Sayfası Şablonu
- Bootstrap ile Web Sitesi İnşası
- Bootstrap Web Sitesi Örnekleri



HEDEFLER

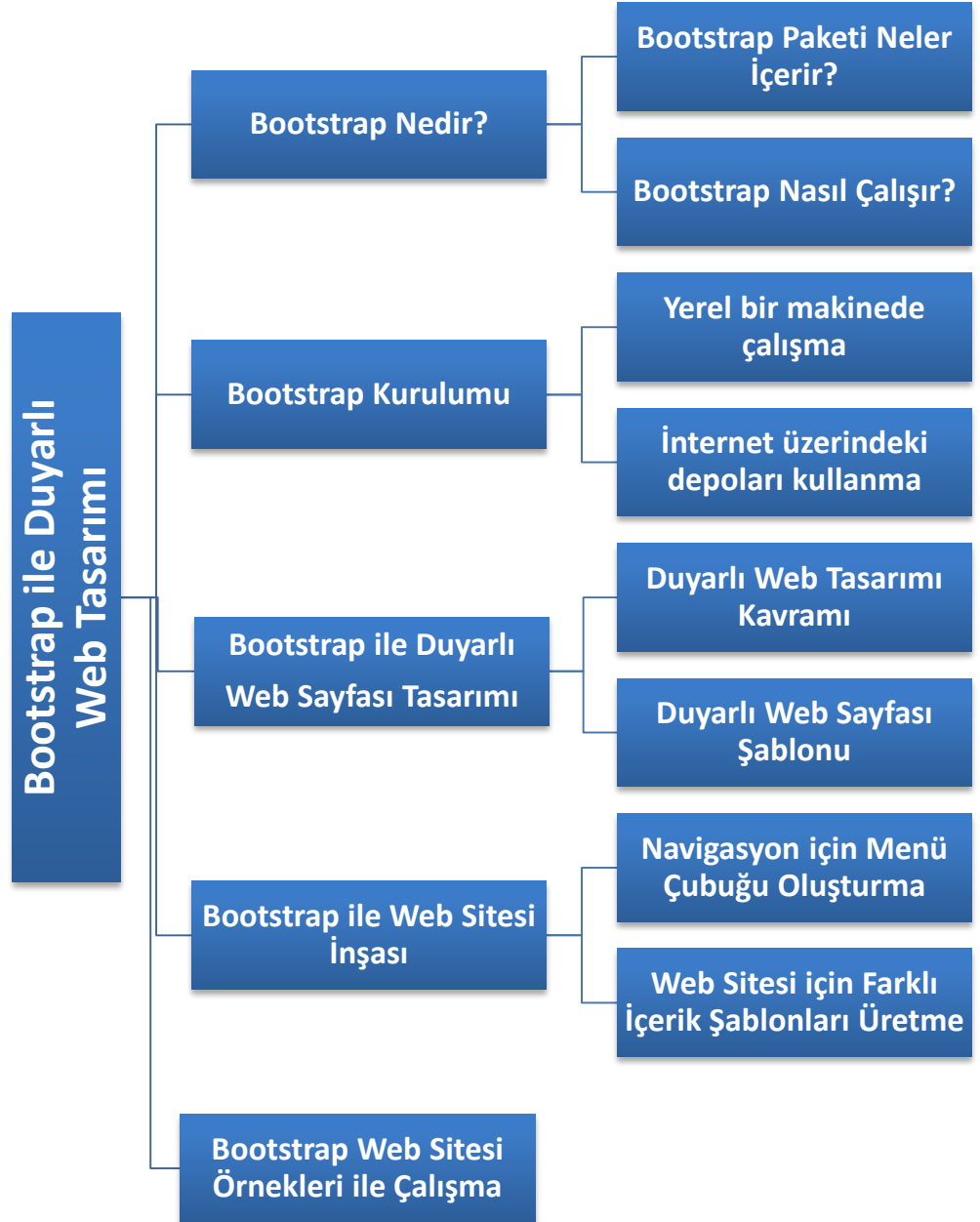
- Bu üniteyi çalıştıktan sonra;
 - Bootstrap çatısının kurulumunu yapabilecek,
 - Duyarlı Web tasarımı ile ilgili bilgi sahibi olacak,
 - Duyarlı Web sayfası şablonu oluşturabilecek,
 - Duyarlı Web sitesi inşa edebilecek,
 - Farklı Bootstrap örneklerini kendi Web sitesi çalışmalarınızda kullanabileceksiniz.



Atatürk Üniversitesi
Açıköğretim Fakültesi

İNTERNET PROGRAMCILIĞI I Öğr. Gör. Rafet Orçun MADRAN

ÜNİTE 14



GİRİŞ

İnternet üzerinden bilgiye erişen cihazların çeşitliliği, Web sitesi geliştiricilerini özellikle arayüz tasarımında farklı çözümler üretmeye sevk etmiştir. Mobil cihazların ilk akıllı sürümleri piyasada yer almaya başladığı zamanlarda Web sitelerinin masaüstü ve mobil sürümlerini iki ayrı site olarak tasarlamak sıkça kullanılan bir yöntemdi. Ancak bu yöntem mobil cihazların da kendi içinde çeşitlenmesi (farklı ekran boyutlarına sahip cep telefonları, tablet bilgisayarlar vb.), masaüstü sistemlerin farklı dizüstü modellerle geniş bir yelpazeye ulaşması ve artık akıllı TV gibi çok daha *büyük ekran boyutuna* sahip cihazların da oyuna girmesi ile ihtiyaçlara cevap veremez hâle geldi.

Çözüm, farklı ekran boyutuna sahip cihazların tamamı ile uyumlu çalışabilecek tek bir Web sitesi tasarlanmasında yatıyordu. *Duyarlı Web Tasarımı* (Responsive Web Design) olarak ifade edilen bu tasarım yaklaşımı, Web sitesinin görüntülediği cihaza göre tasarımın kendini otomatik olarak *ekrana uyumlu* hâle getirmesi prensibine dayanıyordu. Bu sayede cihaz ekranının genişliği İnternet tarayıcısı tarafından algılanmakta ve bu bilgiye göre tasarımı oluşturan bölümler *farklı şekilde ekrana yerleştirilerek* ya da *farklı şekilde boyutlandırılarak* görüntülenmekteydi.



Bootstrap,
Duyarlı Web Siteleri
geliştirebileceğimiz
HTML5 tabanlı bir
çatıdır (framework).

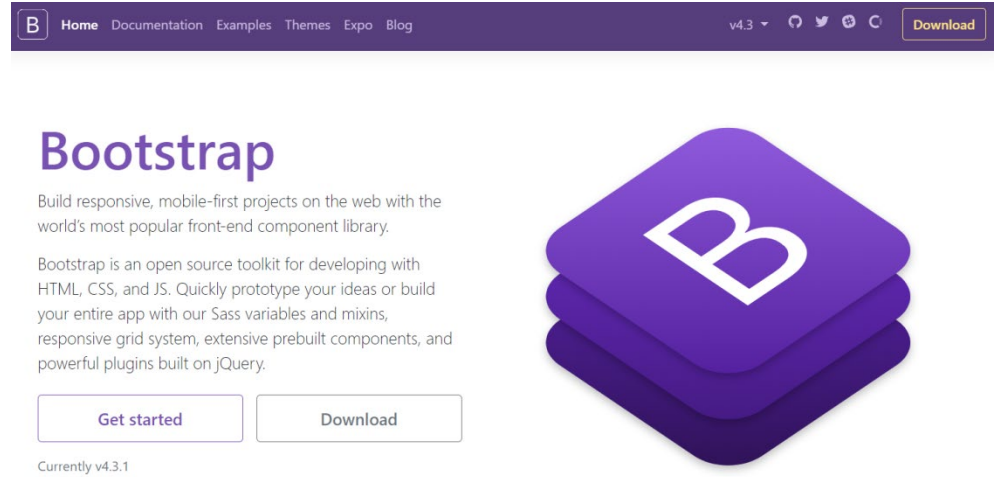
Duyarlı Web Tasarımı, gelişmiş Web tasarımı bileşenlerine ihtiyaç duymaktadır. *HTML5* adını verdiğimiz bu tasarım bileşenleri *HTML*, *CSS* ve *JavaScript*'ten oluşur. Bu üç bileşenin birbiri ile uyumlu ve performanslı çalışabilmesi, geliştiricinin bilgi seviyesine bağlıdır. Aynı zamanda farklı İnternet tarayıcıları ile de uyum içinde çalışmalıdır. Duyarlı Web Tasarımı'nı zorlu bir süreçte dönüştüren bu teknik altyapı gereksinimlerini karşılayabilecek bütünlük sistemlere *çatı* (framework) adı verilir. *Çatılar*, geliştiricinin tasarım sürecinde ihtiyacı olan tüm karmaşık tanımlamaların önceden yapılandırıldığı hazır bir geliştirme ortamı sunar. Bu ortamı kullanan geliştiriciler, teknik detaylarda boğulmayarak tasarımı *çok daha hızlı* şekilde hayata geçirebilirler.

Bu ünite, HTML5 çatıları arasında en çok tercih edilen *Bootstrap* adlı çatı üzerinde durulacaktır. Bootstrap'in temel özellikleri, geliştiriciye sağladığı imkânlar, kurulumdan başlayarak farklı örneklerin kullanımına kadar bu ünite içerisinde detaylandırılacaktır.

BOOTSTRAP NEDİR?

Bootstrap, *mobil cihazlar* için öncelikli ve *duyarlı web siteleri* geliştirmek için arayüz bileşen kütüphanelerine sahip bir çatıdır. Açık kaynak kodlu bir araç olan Bootstrap, projeler için hızlı bir şekilde *prototip* oluşturulabilmesi, ızgara sisteminin kullanımı ve jQuery üzerine inşa edilmiş güçlü bileşenleri ile popüler bir Web çatısıdır.

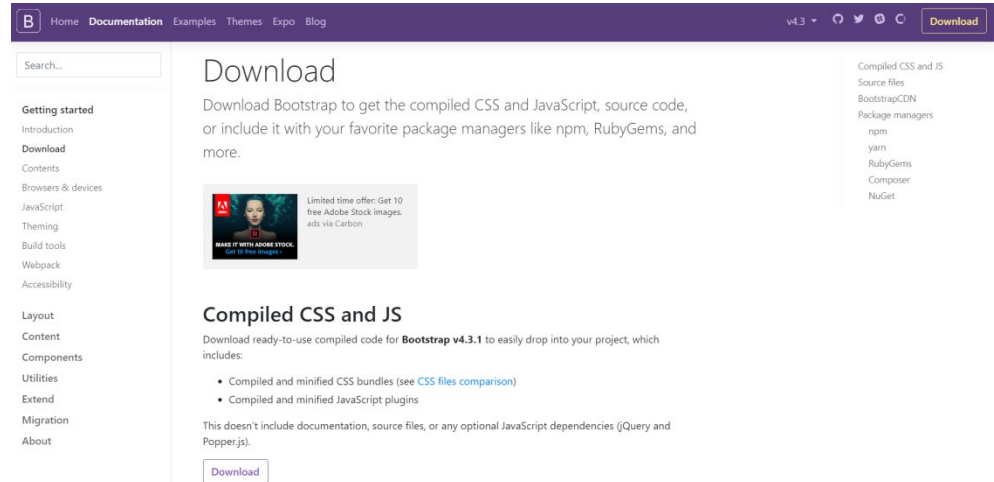
Bootstrap ile ilgili ihtiyaç duyulabilecek tüm kaynak dosyalar ve dokümantasyon <https://getbootstrap.com/> adresinde yer almaktadır (Şekil 14.1.).



14.1. Bootstrap Web sitesi.

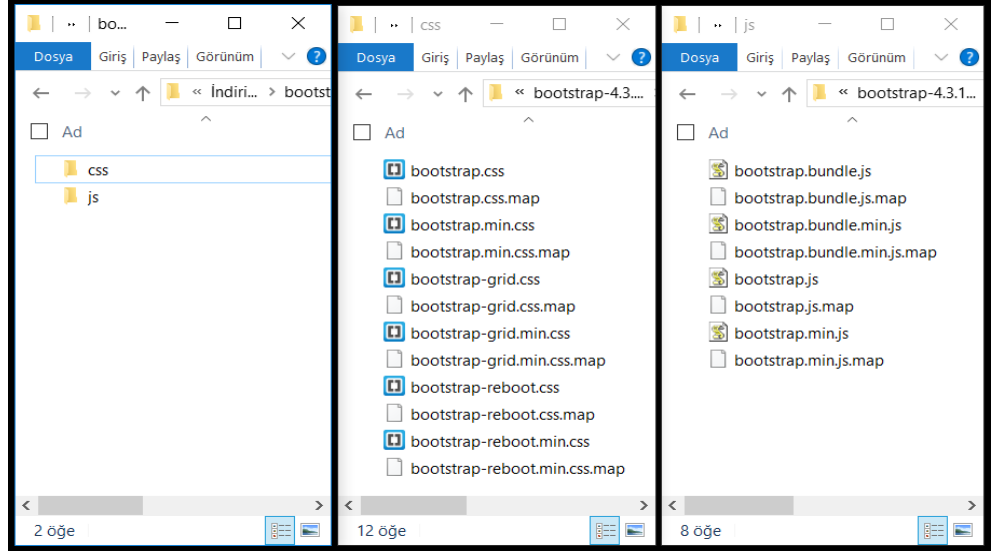
Bootstrap Paketi Neler İçerir?

Bootstrap paketinin içeriğini temel olarak iki farklı dosya türü oluşturur. Bu dosya türleri *CSS tanımlamalarının* yer aldığı **.css* uzantılı dosyalar ve *JavaScript kütüphanelerinin* yer aldığı **.js* uzantılı dosyalardır. Birçok ek bileşenin de kullanılabildiği Bootstrap'in çekirdek dosyalarını Şekil 14.2.'deki indirme sayfasından (<https://getbootstrap.com/docs/4.3/getting-started/download/>) kendi kişisel bilgisayarınıza indirebilirsiniz.



Şekil 14.2. Bootstrap indirme Sayfası.

İndirme işlemi gerçekleştiğinde Bootstrap dağıtım sürümü sıkıştırılmış dosya formatında (zip) bilgisayarınıza kaydedilmiş olur. Zip dosyası açıldığında ana klasörün içinde iki farklı alt klasör yer alır. Bu alt klasörler "*css*" ve "*js*" klasörleridir. Klasörlerin içeriği Şekil 14.3.'te görüntülenmektedir.



Şekil 14.3. Bootstrap Dağıtım Sürümü Klasör İçerikleri.

Daha önceki ünitelerde paket içeriklerinden bahsettiğimizde her bir dosyanın iki farklı sürümü olduğunu belirtmiştik. Bu dosya sürümleri birbirlerinden “min” eki ile ayrılmaktaydı. Dosyaların uzantısında yer alan “min” eki o dosyanın *sıkıştırılmış* bir yapıda hazırlandığını belirtiyordu. Dosya boyutu olarak *daha az yer kaplayan* “min” sürümlerini projelerde kullanmak Web sitesinin optimizasyonu açısından avantaj sağlıyordu.

Bootstrap Nasıl Çalışır?

Bootstrap’in çalışma mantığı Ünite 12’deki jQuery kütüphanesinin ve Ünite 13’teki jQuery Mobile çatısının çalışma mantığı ile paralellik göstermektedir. Çatının kullanılacağı HTML dosyasından ilgili dosyalara (css ve js) *referans verilerek* bağlantı sağlanması Bootstrap çatısının aktif hâle gelmesi için yeterlidir. Referans verilme işlemi gerçekleştirildikten sonra JavaScript kütüphanesindeki fonksiyonları ve CSS tanımlamalarını *kod içinden çağırarak* istenilen yapı inşa edilebilir. Şekil 14.4.’te standart bir Bootstrap sayfa şablonu yer almaktadır.



Bootstrap, diğer Web geliştirme çatılarına benzer bir çalışma mantığına sahiptir.

```

1. <!doctype html>
2. <html lang="tr">
3.   <head>
4.     <meta charset="utf-8">
5.     <meta name="viewport" content="width=device-width,
      initial-scale=1">
6.     <link rel="stylesheet" href="bootstrap.css">
7.     <script src="jquery.js"></script>
8.     <script src="popper.js"></script>
9.     <script src="bootstrap.js"></script>
10.    <title>Bootstrap</title>
11.  </head>
12.  <body>
13.    <h1>Bootstrap Çatı Şablonu</h1>
14.    <p>Web sayfasının içeriği burada yer alacak!</p>
15.  </body>
16. </html>

```

Şekil 14.4. Bootstrap Sayfa Şablonu Örnek Kodları.

BOOTSTRAP KURULUMU



Bootstrap kurulumu iki farklı şekilde gerçekleştirilebilir: internet üzerinde yer alan depolardan, bilgisayarınızın sabit diskine kaydederek.

Bootstrap kurulumu iki farklı şekilde gerçekleştirilebilir. Bunlardan ilki Bootstrap çekirdek dosyalarını *internet üzerinde yer alan depolardan* kullanmaktır. Bu yöntemin avantajı ilgili sürümlerin en güncel hâllerine sürekli olarak erişebilmemizdir. Dezavantajı ise geliştirme ve yayın aşamasında *internet bağlantısına* olan ihtiyaçtır. Standart özelliklere sahip bir Bootstrap projesinde ihtiyaç duyulan çekirdek dosyalar eğer internet üzerinden kullanılacaksa Şekil 14.5.'teki satırların Web sayfasının başlık (<head>) bölümüne yerleştirilmesi yeterli olacaktır.

```
1. <link rel="stylesheet"
  href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
2. <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"></script>
3. <script
  src="https://cdn.jsdelivr.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
4. <script
  src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script>
```

Şekil 14.5. Bootstrap Çekirdek Dosyalarının İnternet Üzerindeki Depolardan Kullanımı.

Diğer yöntem ise Şekil 14.5.'te referans verilen dosyaları *bilgisayarınızın sabit diskine kaydederek* bağlantı kurmaktır. Bu yöntem, İnternet bağlantınız olmadan da geliştirme yapmanıza olanak tanır ancak ilgili sürümlerde bir güncelleme olduğunda bu *güncellemeden faydalanmanızı engeller*.

Kurulumu, ihtiyacımız olan dosyaları sabit diskimizde oluşturduğumuz bir proje klasörünün içine kaydederek tamamlayalım ve bir şablon dosya oluşturarak Bootstrap projemizi test edelim.



Web sitesi tasarımında aynı türdeki dosyaların alt klasörlerde toplanması geliştirme sürecine olumlu katkı sağlar.

Web siteleri yaşayan organizmalara benzediği için zamanla dosya sayılarında çok ciddi bir artış olabilir. Bu açıdan ana proje klasörümüz altındaki dosyaları da alt klasörlere ayırarak daha düzenli bir yapı oluşturabiliriz. Genelde CSS tanımlamalarının yer aldığı *.css uzantılı dosyalar "css" adlı bir klasör içinde, JavaScript kütüphanelerinin yer aldığı *.js uzantılı dosyalar da "js" klasörü içinde yer alır. Proje klasörümüz ve içeriği Tablo 14.1.'deki gibi olmalıdır.

Bootstrap'in *sadece CSS* özelliklerinden faydalanılmak isteniyorsa *JavaScript kütüphaneleri kullanılmayabilir*. Ancak bu durumda sayfa içindeki birçok etkileşim ve JavaScript ile gerçekleştirilecek fonksiyonlar devre dışı kalacaktır.

Tablo 14.1. Bootstrap Örnek Proje Klasör Ve Dosya Yapılanması.

Proje Ana Klasörü	Alt Klasörler	Dosya İsimleri	Dosya Açıklama
bootstrap	css	bootstrap.css	Bootstrap CSS tanımlama dosyası
	js	bootstrap.js	Bootstrap JavaScript Kütüphanesi
		jquery.js	jQuery JavaScript Kütüphanesi
		popper.js	Popper JavaScript Kütüphanesi



Bireysel Etkinlik

- Sabit diskinizde "bootstrap" adlı bir proje klasörü oluşturun.
- Proje klasörü içerisine "css" ve "js" adlarında alt klasörler oluşturun.
- Şekil 14.5.'teki adreslerden CSS ve JS dosyalarını bilgisayarınıza indirerek, Tablo 14.1.'deki gibi isimlendirin ve ilgili alt klasörlere yerleştirin.
- Şekil 14.4.'teki örnek kodlardan faydalanarak bir şablon dokümanı oluşturun ve İnternet tarayıcınızda test edin.
- CSS ve JS dosyalarına referans verirken bulundukları klasörleri dikkate almayı unutmayın.



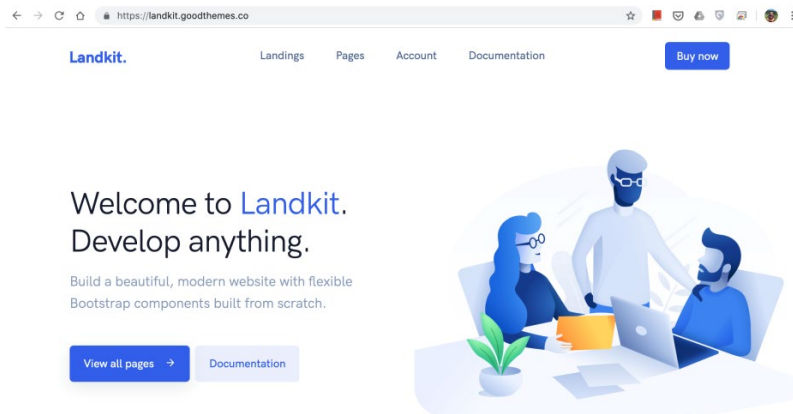
Bootstrap çatısının en önemli özelliği farklı ekran boyutlarına sahip cihazlar için duyarlı tasarımların yapılabilmesidir.

BOOTSTRAP İLE DUYARLI WEB SAYFASI TASARIMI

Bootstrap'in en önemli özelliklerinden birinin duyarlı web sayfaları inşa edebilmek olduğundan bahsetmiştik. Duyarlı bir web sayfasının temel özelliği olan *ekran boyutuna göre sayfa öğelerinin yeniden yerleşimi* ile ilgili bir örnek çalışma yapalım. Bu örnek çalışma aynı zamanda Web sitemizde kullanacağımız temel sayfa yapısının da şablonu olsun.

Duyarlı Web Tasarımı Kavramı

Bir şablon oluşturmaya başlamadan önce duyarlı web sitesinden ne kastettiğimizi bir örnek ile netleştirelim. Şekil 14.6.'da bir masaüstü ya da dizüstü bilgisayar tarafından görüntülenen Bootstrap ile tasarlanmış bir Web sitesinin ekran görüntüsü yer almakta.



Şekil 14.6. Örnek Web Sitesi Masaüstü / Dizüstü Bilgisayar İle Görüntüleme.

Şekil 14.6.'daki ekran görüntüsünü incelediğimizde *sayfa içi navigasyonu* sağlayan menünün sağ üst bölümde konumlandırıldığını görüyoruz. Navigasyondan sonra sayfanın 2 bölüme ayrıldığını, sol bölümde slogan, metin ve

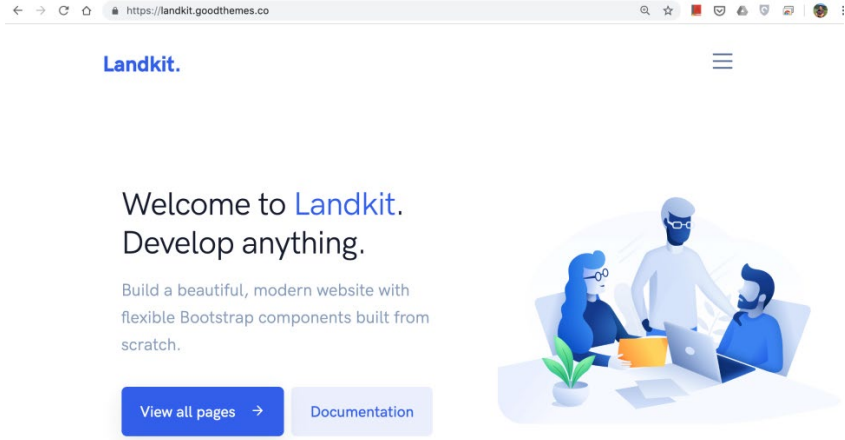


Geliştirme ortamınızda farklı cihaz ekranlarını test edebilmek için İnternet tarayıcınızın yakınlaştırma özelliğinden faydalanabilirsiniz.

butonların, sağ bölümde ise bir görselin yer aldığını görüyoruz.

Web sayfalarının farklı ekran boyutlarına göre test edebilmenin çeşitli yolları vardır. Bunlardan biri İnternet tarayıcınızın *pencere genişliğini daraltarak*, sanki daha küçük bir ekranda görüntüleniyormuş gibi yapmaktır. Diğer bir yöntem ise İnternet tarayıcınızın *yakınlaştırma* (zoom) özelliğinden %100'ün üzerinde bir değer ayarlayarak içeriğin oran olarak büyümesini sağlamaktır. Eğer farklı cihazlarda (cep telefonu, tablet PC vb.) fiziksel olarak test imkânı bulunmuyorsa, yukarıda bahsedilen yollar denenebilir ve çok büyük oranda test ortamı size gerçeğe yakın bir fikir verecektir.

Şekil 14.7.'de İnternet tarayıcımızın pencere genişliğini biraz azaltıyoruz (ya da yakınlaştırma değerini örn. %125 olarak belirliyoruz) ve sayfa tasarımındaki değişiklikleri gözlemliyoruz.



Şekil 14.7. Örnek Web Sitesi Tablet PC (Temsili Olarak) İle Görüntüleme.

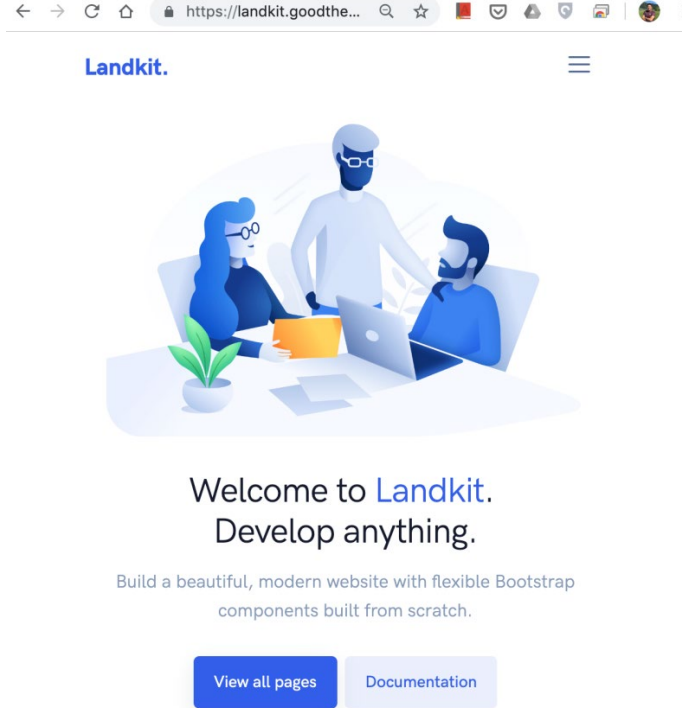


İnternet tarayıcınızın pencere boyutlarını değiştirerek farklı cihaz ekranlarını test edebilirsiniz.

Şekil 14.7.'yi dikkatlice incelediğimizde sağ üstte yer alan navigasyon bölümünün değişikliğe uğradığını görüyoruz. Şekil 14.6.'da menü öğeleri açık bir şekilde duruyorken, Şekil 14.7.'de menü öğeleri artık bir butonun (*alt alta üç çizgi*) altına gizlenmiş durumda. Ekran boyutu azaldıkça sayfa içerisinde yer alan öğelerin daha az yer kaplaması amaçlanıyor.

Bir sonraki aşamayı gözlemleyebilmek için ekran boyutunu iyice küçültmemiz ya da içeriğin oranını yapay bir şekilde (yakınlaştırma kullanarak) daha da büyütmemiz gerekiyor.

Şekil 14.8.'de artık sayfayı bir *cep telefonundan* görüntülüyormuş gibi düşünebiliriz.



Şekil 14.8. Örnek Web Sitesi Cep Telefonu (Temsili Olarak) İle Görüntüleme.

Şekil 14.8.'i incelediğimizde sayfa içerisindeki öğelerin *yeni yerleşimleri* hemen kendini belli etmekte. Ekranın genişliği daraldığı için artık slogan bölümü ve görsel öge *yan yana* iki sütunda duramıyor. Bu öğeler *alt alta* sıralanmış durumda.

Yukarıda 3 farklı ekran boyutuna göre test ettiğimiz Web sayfası duyarlılık ile ilgili kavramı anlayabilmemiz için güzel bir örnek oluşturdu. Web sayfasının geliştiricisi aslında *tek bir sayfa tasarladı* ancak bu sayfa *üç farklı ekran* boyutuna göre kendini adapte edebildi ve uyum sağladı.



Bireysel Etkinlik

- İnternet tarayıcınız ile <https://themes.getbootstrap.com/> adresini ziyaret edin.
- Bu adreste yer alan temaları tarayıcınızda farklı ekran boyutlarına göre görüntüleyin.
- Farklı ekran boyutlarını tarayıcınızın pencere genişliğini daraltarak ya da yakınlaştırmayı arttırarak simüle edebilirsiniz.

Duyarlı Web Sayfası Şablonu

Önceki bölümde “Duyarlı Web Tasarımı” kavramını örneklerle incelediğimize göre artık kendi duyarlı sayfa şablonumuz üzerinde çalışmaya başlayabiliriz. Kurulumda, ihtiyacımız olan çekirdek dosyaları temin etmiştik ve bunları proje klasörümüz içindeki alt klasörlere yerleştirmiştik.

Sayfanın başlık kısmı

Şimdi proje klasörümüz içerisinde “*sablon.html*” adlı bir dosya oluşturalım. Bu dosyanın *başlık* (<head>) bölümü Şekil 14.8.’deki gibi kodlanmalıdır. *Bu bölüm* bizim Bootstrap çalışmalarımızda (ek bir JS kütüphanesi ya da bir CSS tanımlaması kullanılmadıkça) standart olarak *her zaman yer alacaktır*.

```
1. <!doctype html>
2. <html lang="tr">
3.   <head>
4.     <meta charset="utf-8">
5.     <meta name="viewport" content="width=device-width,
      initial-scale=1">
6.     <link rel="stylesheet" href="css/bootstrap.css">
7.     <script src="js/jquery.js"></script>
8.     <script src="js/popper.js"></script>
9.     <script src="js/bootstrap.js"></script>
10.    <title>Bootstrap Şablon Sayfası</title>
11.  </head>
12.  <body>
13.  </body>
14. </html>
```

Şekil 14.8. Bootstrap Şablon Sayfası Başlık Bölümü.

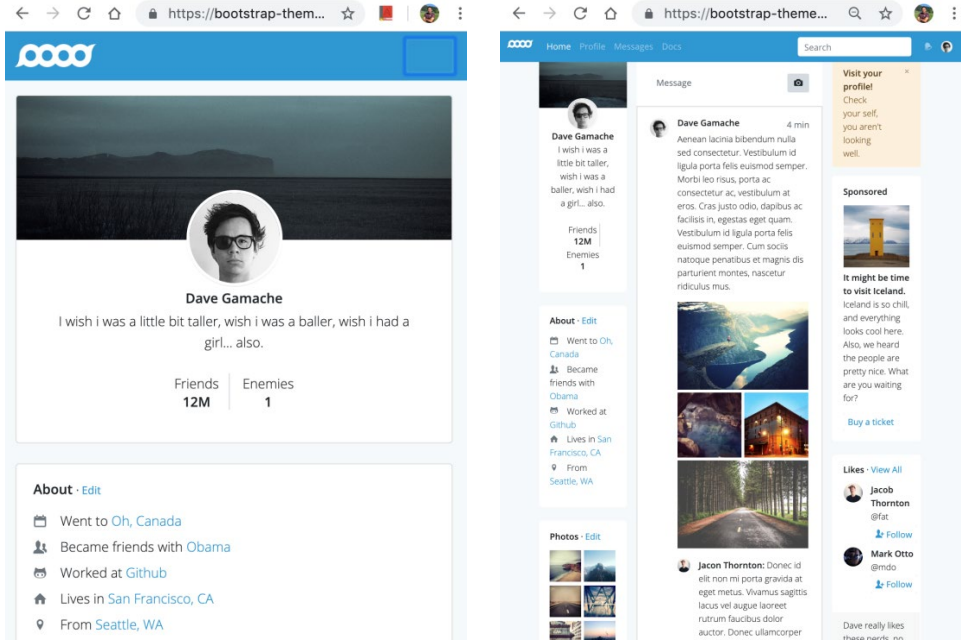
Şekil 14.8.’deki başlık bölümü kodlarında satır 6, 7, 8 ve 9 Bootstrap çatısını oluşturan çekirdek dosyalara verilen *referanslardır*. Bu dosyaların bağlantılarının doğru bir şekilde verilmesi çok önemlidir. Bulundukları klasör adı ve dosya adları tam olarak yazılmalı ve bağlantının kırık olmadığı mutlaka test edilmelidir. Başlık bölümündeki bir diğer önemli satır ise 5. satırdır. 5. satırda görülebilir alanla ilgili bir üst veri tanımlaması (viewport / görüntü alanı) yer almaktadır. Bu tanımlama, Web sayfasının duyarlı olarak tasarlandığını ve görülebilir alanın küçük olmasından dolayı İnternet tarayıcısının *sığdırma işlemi yapmamasını* söyler.

“Viewport” üst veri tanımlamasını alan tarayıcı sığdırma işlemi yapmaz, sayfayı küçülterek okunması zor bir hale sokmaz, duyarlı tasarımın öğeleri yeniden sayfa içinde konumlandırmasına ve yapılandırmasına (yazı tipi boyutu vb.) izin verir.

Şekil 14.9.’da “*viewport*” üst veri tanımlaması *yapılmış* ve *yapılmamış* örnekler yan yana yer almaktadır. Örnekler arasındaki farklılık “viewport” tanımlamasının duyarlı Web tasarımındaki önemini gözler önüne sermektedir.



Duyarlı Web Tasarımı, İnternet tarayıcılarına tasarımın genel özellikleri ile ilgili üst veri tanımlamaları da sağlar.



Şekil 14.9. Aynı Web Sitesinin Temsili Mobil Cihaz Ekranında Görüntülenmesi; “Viewport” Tanımlaması Yapılmış (Solda) Ve Yapılmamış (Sağda) Web Sayfası Örnekleri.

Sayfanın gövde kısmı

Duyarlı sayfa şablonumuzun başlık (<head>) bölümü ile ilgili düzenlemeleri tamamladığımıza göre artık **gövde** (<body>) bölümü ile ilgili olan çalışmalara başlayabiliriz.

Bootstrap sayfa yapısının en önemli bileşeni **bölmelerdir** (<div>). Bölmeler bize **ızgara** (grid) adı verilen **hayali bir şablon** üzerinde çalışmamıza olanak sağlar. Bu sayede HTML içerisinde istediğimiz sayfa yerleşimini gerçekleştirebiliriz.

Sayfamızın gövde bölümünü oluşturacak kodlar Şekil 14.10.’daki gibi olmalıdır.

```

1. <body>
2. <div class="jumbotron text-center">
3. <h1>Bootstrap Sayfa Şablonu</h1>
4. <p>İnternet tarayıcısının pencere genişliğini
   değiştirerek sayfada yaşanan değişiklikleri
   gözlemleyebilirsiniz!</p>
5. </div>
6. <div class="container">
7. <div class="row">
8. <div class="col-sm-4">
9. <h3>Başlık</h3>
10. <p>İçerik</p>
11. </div>
12. <div class="col-sm-4">
13. <h3>Başlık</h3>
14. <p>İçerik</p>
15. </div>
16. <div class="col-sm-4">
17. <h3>Başlık</h3>
18. <p>İçerik</p>
19. </div>

```

```

20.         </div>
21.     </div>
22. </body>

```

Şekil 14.10. Bootstrap Şablon Sayfası Gövde Bölümü.

Şekil 14.10.'daki kod yapısını incelediğimizde gövde bölümünde *iki ana bölmenin* (<div>) yer aldığını görüyoruz. Bu bölmelerden sayfanın üst bölümünde yer alacak karşılama mesajının olduğu bölme satır 2-5 arasında, sütunların olduğu içerik bölmesi ise satır 6-21 arasına kodlanmış durumda. 2. satırda yer alan CSS sınıf tanımlaması (class="jumbotron text-center") bu bölmedeki metinlerin sayfa içinde ortalanmasını sağlamakta. 6. satırdaki sınıf tanımlaması (class="container") ise sütunlar için bir ana taşıyıcı (konteynır) görevi görmektedir.

Bootstrap içerisinde sayfa içi yerleşiminde kullanabileceğimiz *standart* olarak *12 sütunlu* bir yapı bulunmaktadır. Bu sütunlar bölmelerin (<div>) içerisinde *CSS sınıfları* (class) olarak tanımlanabilmekte ve her bölmenin kaç sütundan oluşacağı belirtilmektedir. Şekil 14.10.'da 8., 12. ve 16. satırlarda tanımlanan bölme sınıfı (<div class="col-sm-4">) sayfanın *üç eşit parçaya* bölünmesini sağlamaktadır.

Burada önemli olan nokta toplam sütun sayısının 12'ye tamamlanmış olmasıdır. Örnek olarak *ilk sütun dar*, ikinci ve üçüncü sütunlar daha geniş ve *birbirine eşit* olsun isteniyorsa uygulanacak formül *2-5-5* olabilir. İlk bölme (<div>) 2 sütundan, 2. ve 3. bölmeler ise 5 sütundan oluşacaktır.

Şekil 14.10.'daki kodları "sablon.html" dosyasına eklediğimizde ve İnternet tarayıcımızda dosyayı açtığımızda ekran görüntüsü Şekil 14.11.'deki gibi olacaktır.



Şekil 14.11. Bootstrap Şablon Dosyası Ekran Görüntüsü.

Şekil 14.11.'de ilk duyarlı web sayfası tasarımıımızı gerçekleştirmiş olduk. Bu görünümünden tasarımıımızın standart bir *masaüstü bilgisayar* için iyi bir sayfa yerleşimi sunduğunu anlayabiliyoruz. Şimdi İnternet tarayıcımızın penceresini *genişlik olarak daraltalım* ve sonucu gözlemleyelim (Şekil 14.12.).

HTML5 çatıları, sayfa içinde öğelerin yerleşimi için ızgara sistemi ve boyutlandırılabilir sütunlar sağlar.

← → ↻ 127.0.0.1:55921/sablon.html 🔍 ☆ 👤 ⋮

Bootstrap Sayfa Şablonu

İnternet tarayıcısının pencere genişliğini değiştirerek sayfada yaşanan değişiklikleri gözlemleyebilirsiniz!



Duyarlı Web Sayfaları, görüntülendikleri ekrana göre öğelerin sayfa içi yerleşimlerini yeniden yaparlar.

Bootstrap Nedir?

Bootstrap, mobil cihazlar için öncelikli ve duyarlı web siteleri geliştirmek için arayüz bileşen kütüphanelerine sahip bir çatıdır.

Açık kaynak kodlu bir araç olan Bootstrap, projelerin hızlı bir şekilde prototiplenmesini, izgara sisteminin kullanımı ve jQuery üzerine inşa edilmiş güçlü bileşenleri ile popüler bir Web çatısıdır.

Bootstrap Paketi

Bootstrap paketinin içeriğini temel olarak iki farklı dosya

Şekil 14.13. Bootstrap Şablon Dosyası Mobil Cihaz (Temsili) Ekran Görüntüsü.

Şekil 14.13.'teki ekran görüntüsünü incelediğimizde 3 sütundan oluşan içerik alanının *kendini yeniden yapılandırıldığını* ve *tek sütunlu* bir yapıya dönüştüğünü gözlemliyoruz. Tasarımımız farklı boyutta görüntü alanına (viewport) sahip cihazlar için de iyi bir çözüm üretmiş durumda.



Bireysel Etkinlik

- Proje klasörünüzde "sablon.html" adlı bir dosya oluşturun.
- Şekil 14.8. ve Şekil 14.10.'daki kod yapılarından faydalanarak "sablon.html" içerisinde bir örnek sayfa kodlayın.
- Kodladığınız sayfayı İnternet tarayıcınızda farklı ekran genişliklerinde test edin ve değişiklikleri izleyin.

Bootstrap ile Web Sitesi İnşası

Duyarlı bir sayfa şablonunu başarıyla oluşturduğumuza göre, bu deneyimimizi bir Web sitesinin inşa sürecinde kullanabiliriz. Navigasyonu sağlayacak *menü yapısını da duyarlı bir şekilde oluşturup* bunu mevcut yapıya ekleyerek 4-5 sayfadan oluşan bir Web sitesi hayata geçirelim.

Web sitesinin navigasyonu için tasarlayacağımız menü çubuğumuzda en solda bir *logo* alanı, daha sonra siteyi oluşturan *sayfalara bağlantıların* yer alacağı bir bölüm ve en sağda ise bir *arama bölümü* yer alsın. Bu menü çubuğu ekranın görülebilir alanına göre belirli bir genişliğin altına düşündüğü zaman tek bir menü düğmesi altında toplanacak bir yapıda olsun. Duyarlı menü çubuğumuzun genel



Duyarlı Web Tasarımında, navigasyonu sağlayacak menü ögesinin de duyarlı bir yapıda tasarlanması gereklidir.

görüntüsü şekil 14.14.’teki gibi olacaktır.



Şekil 14.14. Duyarlı Menü Çubuğu Genel Görünümü.

Şekil 14.14.’te menü çubuğunu oluşturan kod yapısı Şekil 14.15.’te yer almaktadır. Menü çubuğunun içinde yer alan öğeler temelinde **HTML liste elemanlarıdır**. Çok farklı şekillerde menü çubukları tasarlamak mümkündür.



Menü çubuğu
oluşturmada kullanılan
en önemli öğe HTML
liste elemanlarıdır.

```

1. <nav class="navbar navbar-expand-lg navbar-dark bg-
   dark" style="margin:24px 0;">
2.   <a class="navbar-brand" href="#">Logo</a>
3.   <button class="navbar-toggler navbar-toggler-right"
   type="button" data-toggle="collapse" data-
   target="#navb">
4.     <span class="navbar-toggler-icon"></span>
5.   </button>
6.   <div class="collapse navbar-collapse" id="navb">
7.     <ul class="navbar-nav mr-auto">
8.       <li class="nav-item">
9.         <a class="nav-link" href="#">Bağlantı 1</a>
10.      </li>
11.      <li class="nav-item">
12.        <a class="nav-link" href="#">Bağlantı 2</a>
13.      </li>
14.      <li class="nav-item">
15.        <a class="nav-link" href="#">Bağlantı 3</a>
16.      </li>
17.      <li class="nav-item">
18.        <a class="nav-link" href="#">Bağlantı 4</a>
19.      </li>
20.    </ul>
21.    <form class="form-inline my-2 my-lg-0"
   method="get" target="_blank"
   action="http://www.google.com/search">
22.      <input class="form-control mr-sm-2" type="text"
   name="q">
23.      <button class="btn btn-success my-2 my-sm-0"
   type="submit">Ara</button>
24.    </form>
25.  </div>
26. </nav>

```

Şekil 14.15. Duyarlı Menü Çubuğu Örnek Kodları.

Şekil 14.15.’teki kod blokunu incelediğimiz zaman menü çubuğunun genel yapısının 1. satırda tanımlandığını görüyoruz. Daha sonra sırasıyla çubuk içinde yer alacak elemanların tanımlamaları yer alıyor. 2. satırda “Logo” olarak kullanabileceğimiz bölüm, 3. ve 4. satırlarda ise görüntü alanı daraldığında menü çubuğunun nasıl bir davranış göstermesi gerektiği tanımlanıyor. Satır 7-20 arasında tanımlanan HTML liste öğeleri, bağlantı sağlanacak diğer sayfa ya da siteler için tıklanabilecek metinler oluşturuyor. Satır 21-24 arasında ise bir arama formu yer almakta. Bu arama formu içerisinde bir metin kutusu ve tetikleme işlemini gerçekleştirecek bir buton yer alıyor.

Şimdi daha önceden hazırlamış olduğumuz duyarlı Web şablonumuza menü çubuğumuzu yerleştirelim. Sayfamızın görünümü Şekil 14.16.'daki gibi olmalıdır.



Bootstrap Nedir?

Bootstrap, mobil cihazlar için öncelikli ve duyarlı web siteleri geliştirmek için arayüz bileşen kütüphanelerine sahip bir çatıdır.

Bootstrap Paketi

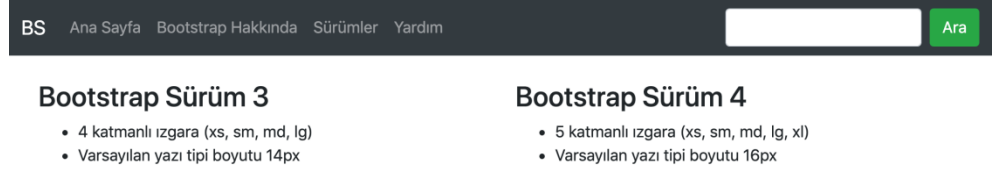
Bootstrap paketinin içeriğini temel olarak iki farklı dosya türü oluşturur. Bu dosya türleri CSS tanımlamalarının yer aldığı *.css uzantılı dosyalar ve

Bootstrap Kurulumu

Bootstrap kurulumu iki farklı şekilde gerçekleştirilebilir. Bunlardan ilki Bootstrap çekirdek dosyalarını İnternet üzerinde yer alan depolardan

Şekil 14.16. Duyarlı Web Sayfasına Menü Çubuğunun Eklenmiş Hâli.

Duyarlı Web sitemizin genel çatısını oluşturduk. Artık bu sayfanın *kopyalarını çıkartarak* ve çıkarttığımız kopyalara *menü çubuğundan bağlantılar* sağlayarak Web sitemizi tamamlayabiliriz. Kopyasını oluşturduğumuz sayfaların genel görünümünün de *aynı olmasına gerek yok*. Menü çubuğunu sabit bırakarak alt bölümde istediğimiz gibi sütunlar üzerinde oynayarak farklı bölmeler oluşturabiliriz. Şekil 14.17.'de Bootstrap'in 3. ve 4. Sürümlerini karşılaştıran bir Web sayfası örneği yer almakta. Bu sayfada *slogan bölümü yok* ve sayfa 6 sütundan 2 bölmeye ayrılmış durumda.



Bootstrap Sürüm 3

- 4 katmanlı ızgara (xs, sm, md, lg)
- Varsayılan yazı tipi boyutu 14px

Bootstrap Sürüm 4

- 5 katmanlı ızgara (xs, sm, md, lg, xl)
- Varsayılan yazı tipi boyutu 16px

Şekil 14.17. Sayfanın İçeriğine Göre Farklı Tasarım Öğelerinin Kullanımı.

Bootstrap, gerek CSS tanımlamaları, gerekse de JavaScript kütüphaneleri ile çok hızlı bir şekilde duyarlı Web siteleri geliştirmemize olanak tanımakta. Her yeni sürüm ile sahip olduğu özellikleri arttıran Bootstrap iyi bir kullanıcı topluluğu desteğine de sahip. İnternet üzerindeki birçok kaynaktan Bootstrap ile ilgili ek özelliklere sahip *farklı çatılar* bulabilir ve bunları *mevcut yapıya entegre* de edebilirsiniz.

Bootstrap ile ilgili ücretli ya da ücretsiz bulabileceğiniz birçok tema da bu çatının limitlerini, yapılabilecek farklı çalışmaları ve görsel bileşenlerin detaylı kullanımını görebilmeniz açısından da faydalı olacaktır.



Sayfa şablonu oluşturulduktan ve test edildikten sonra şablonun kopyaları farklı kaydedilerek Web sitesi içinde yer alacak diğer sayfalar kolaylıkla oluşturulabilir.



Bireysel Etkinlik

- Proje klasörünüzde oluşturduğunuz "sablon.html" adlı bir dosyayı farklı kaydederek adını "index.html" olarak belirleyin.
- Sayfanızın gövde bölümünün en üstüne Şekil 14.15.'teki menü çubuğunu yerleştirin. Sitenizde yer alacak diğer sayfalara göre menü öğelerini yapılandırın.
- "index.html" adlı dosyadan kopyalar oluşturarak sitenizi tamamlayın.

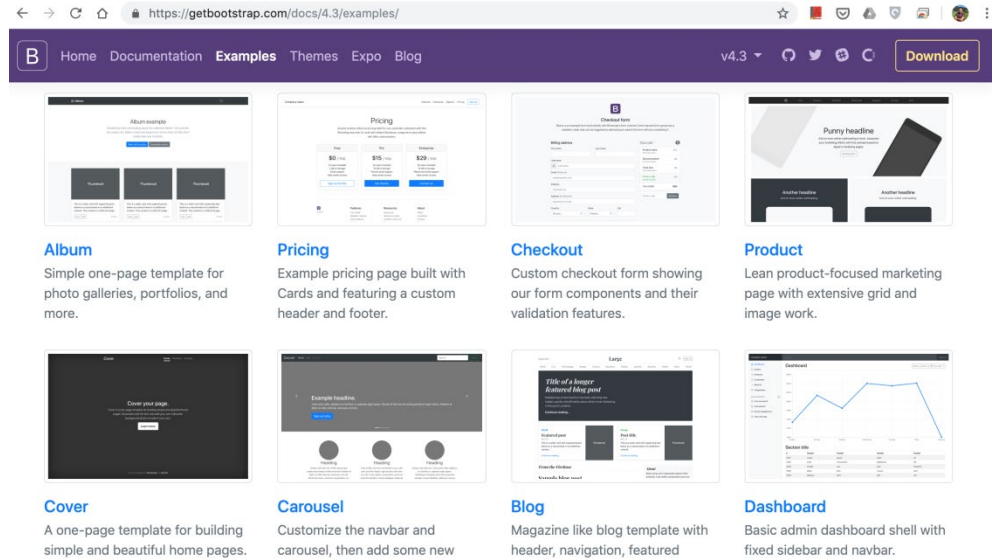
BOOTSTRAP WEB SİTESİ ÖRNEKLERİ

Bootstrap yaygın kullanımından dolayı İnternet üzerinde çok fazla örnek bulabileceğiniz bir çatı. Bu örnekler haricinde *beğendiğiniz herhangi bir Web sitesinin* kodlarına bakarak (ki bu kodları standart bir İnternet tarayıcıda görüntüleyebilmeniz için hiçbir engel yok), bu sitenin altyapısını hangi çatı ya da sistemlerin oluşturduğunu da anlayabilirsiniz.

Bootstrap'ın kendi Web sitesi içerisinde bulunan *örnekler* (examples) bölümü, yeni bir Web sitesinin tasarımına başlamak için ideal bir nokta. Bootstrap'ın örnekler bölümüne <https://getbootstrap.com/> adresinde üst menüde yer alan "Examples" bağlantısı ile ulaşabilir ya da direkt olarak <https://getbootstrap.com/docs/4.3/examples/> adresini ziyaret edebilirsiniz (Şekil 14.18.).



Bootstrap örnek Web sayfaları standart dosyalara ek olarak farklı bileşenler içerebilir.



Şekil 14.18. Bootstrap Örneklerinin Yer Aldığı Web Sayfası.

Şekil 14.18.'de görüntülenen sayfada yer alan örnekler, Bootstrap çatısına ek olarak *farklı CSS ve JS bileşenlerinin* de kullanıldığı örneklerdir. Bu açıdan bu örnekleri bir şablon dosya yapısı hâlinde kullanabilmek için *Web sayfasını tüm bileşenleri* ile birlikte bilgisayarınızın sabit diskine kaydetmek iyi bir başlangıç

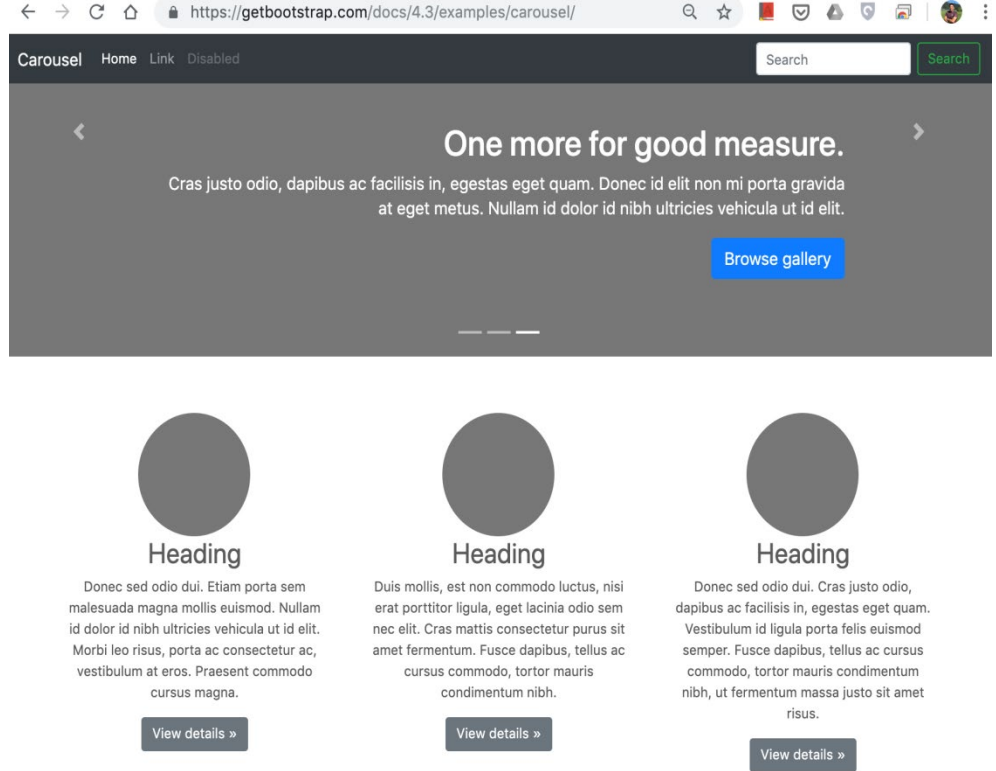


Bootstrap örneklerine <https://getbootstrap.com/docs/4.3/examples/> adresinden erişilebilir.

olacaktır. Bu işlem hemen her İnternet tarayıcısında benzer bir şekilde yapılmaktadır. “*Dosya > Sayfayı farklı kaydet*” menü öğelerini takip ederek örnek Web sayfası tüm bileşenleri ile kaydedilebilir.

Bootstrap örnekleri içinden günümüzde çok kullanılan bir temayı seçelim. Seçtiğimiz tema bir menü çubuğuna, atlı karınca adı verilen (Carousel) bir slayt alanına ve içerik bölümünde sütunlu bir yapıya sahip olsun. Temanın genel görünümü Şekil 14.19.’daki gibidir ve aşağıdaki bağlantıdan erişilebilir:

<https://getbootstrap.com/docs/4.3/examples/carousel/>

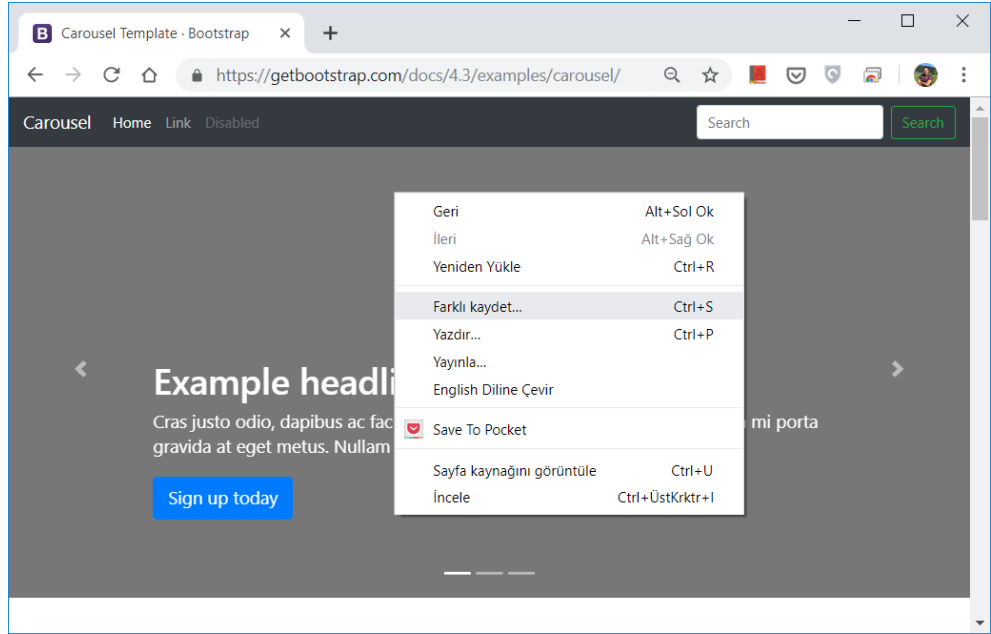


Şekil 14.19. Bootstrap Örnekleri, Atlı Karınca (Carousel) Teması.

Sayfayı İnternet tarayıcınızda görüntüledikten sonra sayfayı sabit diskinize kaydetmek için klavyenizin **CTRL + S** tuş kombinasyonunu kullanabilir (bu özellik *Google Chrome* ve *Mozilla Firefox* tarayıcılarda geçerlidir) ya da sayfanın boş bir noktasında farenize sağ tıklayarak açılan menüden “*Farklı kaydet*” menü öğesine tıklayabilirsiniz (Şekil 14.20.).

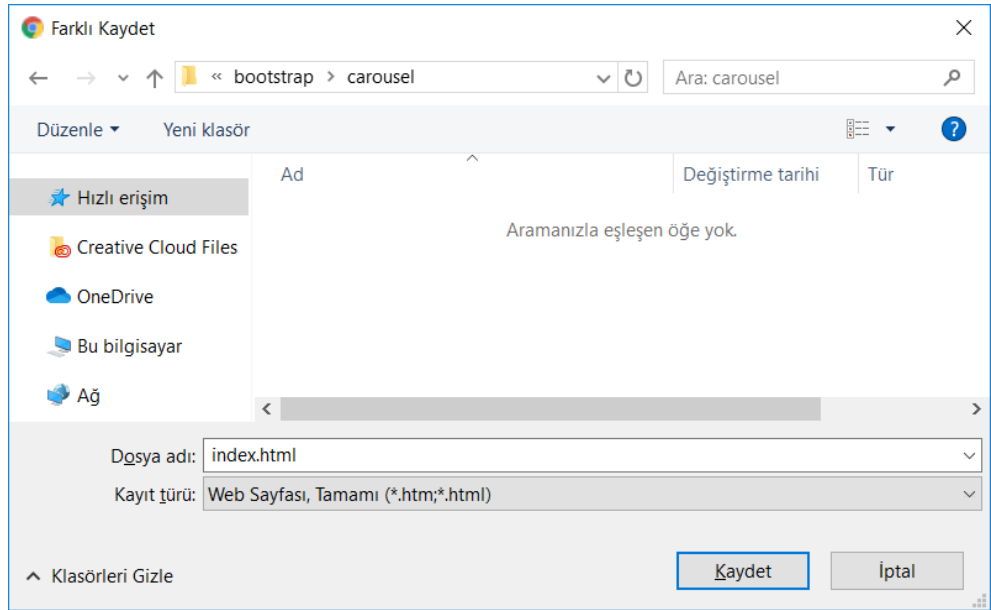


Tasarımını beğendiğiniz herhangi bir Web sayfasını İnternet tarayıcınız aracılığıyla sabit diskinize kaydedip inceleyebilirsiniz.



Şekil 14.20. Farenin Sağ Kliği İle Açılan “Farklı Kaydet” Menü Ögesi.

Sayfayı nereye kaydetmek istediğinizi soran bir *iletişim penceresi* açılacaktır. Bu pencere içerisinde sayfa ve bileşenlerini kaydetmek istediğiniz yeri seçiniz. Lokasyon olarak proje klasörünüz “*bootstrap*” içerisinde “*carousel*” adlı bir alt klasör oluşturabilir ve *tüm dosyaların* bunun içine kaydedilmesini sağlayabilirsiniz (Şekil 14.21.).



Şekil 14.21. Örnek Web Sayfasının Bilgisayar Sabit Diskine Kaydedilmesi.

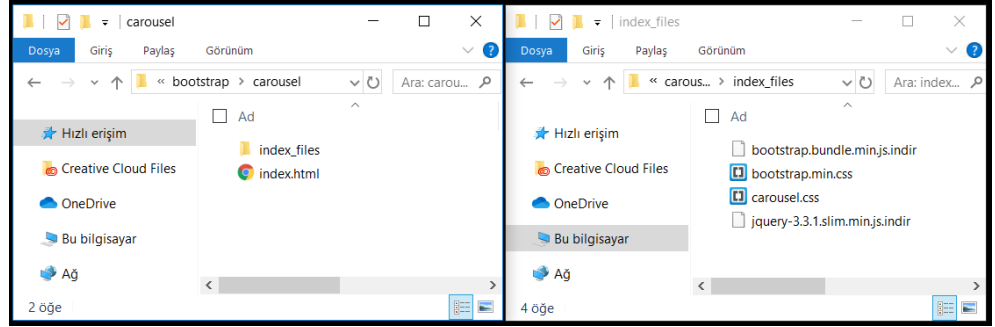
Şekil 14.21.’de dikkat etmemiz gereken iki önemli nokta var. Bunlardan ilki kaydederken Web sayfasına vereceğimiz ad. Bu ad genelde orijinal sayfanın başlık kısmında belirtilen ad olur. Bu adı örneğin “*index.html*” olarak değiştirebilirsiniz. Bu şekilde bileşenlerin kaydedileceği klasör adı Web için daha uyumlu bir adlandırma olur. Bu klasör adı daha sonra da *değiştirilebilir*, ancak o zaman sayfada verilen *referansları* da *güncellemek* gerekir.



Bir Web sayfasını sadece HTML dosyası olarak ya da tüm bileşenleri ile (CSS, JS) bilgisayarınıza indirebilirsiniz.

İkinci nokta ise sayfa kaydedilirken iletişim penceresinin alt bölümünde yer alan biçim (format) kısmında “*Web sayfası, tamamı*” ibaresinin seçili olması. Bu ayar varsayılan olarak “Web sayfası, sadece HTML” şeklindedir. Bu şekilde bırakılırsa sadece HTML dosyası kaydedilir, sayfa içinde referans verilen bileşenler kaydedilmez. Bu durumda ihtiyacımız olan CSS ve JS dosyaları açısından İnternet üzerindeki kaynaklara bağımlı oluruz ve sitenin sonraki aşamalarında çeşitli sorunlarla karşılaşabiliriz.

Örnek Web sayfasını tüm bileşenleri ile sabit diskimize kaydettiğimizde, elimizde bir HTML dosyası ve bileşenlerin yer aldığı bir klasör bulunmalıdır. “carousel” adlı yeni proje klasörümüzün ve bileşen alt klasörünün görünümü Şekil 14.22.’deki gibi olmalıdır.



Şekil 14.22. “Carousel” Projesi Klasör Yapısı Ve Dosyaları.

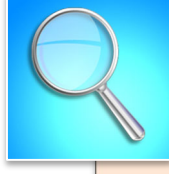
Şekil 24.22.’deki “index_files” adlı klasörün içeriğini incelediğimizde hem Bootstrap ile ilgili dosyaları (*bootstrap.bundle.min.js*, *bootstrap.min.css*), hem jQuery kütüphanesini (*jquery-3.3.1.slim.min.js*), hem de bu projeye özel olarak kullanılan Carousel CSS dosyasını (*carousel.css*) görmekteyiz.

Örnek bir Web sayfasını şablon olarak kaydetmek istediğimizde sayfayı tüm bileşenleri ile kaydetmenin önemini şimdi daha iyi anlamış olmalıyız. Sayfa içinde kullanılan onlarca farklı bileşen olabilir ve bu bileşenler farklı sürümleri ile proje içerisinde yer alabilir.

Örnek projemizin dosyalarını başarıyla kaydettiğimize göre artık proje kalsörümüzde yer alan “*index.html*” adlı dosya üzerinde kendi istediğimiz değişiklikleri yapabiliriz. Menü çubuğundaki bağlantıları kendi Web sitemize göre düzenleyebiliriz. “index.html” dosyasını *şablon olarak* kullanıp diğer sayfalarımızı bu dosyayı temel olarak oluşturabiliriz.



Bir Web sayfası içerisinde birçok farklı bileşen yer alabilir. Bu açıdan Web sayfalarını tüm öğeleri ile birlikte sabit diske kaydetmek önem taşımaktadır.



Özet

- Web sitelerine erişebilen cihazların farklı ekran boyutlarına sahip oluşu, arayüz tasarımı birtakım sorunları da beraberinde getirdi.
- Önceleri masaüstü ve mobil cihazlar için ayrı ayrı yapılan arayüz tasarımları yerini Duyarlı Web Sitelerine bıraktı.
- Duyarlı Web Siteleri, ekran görünür alanına göre sayfadaki öğeleri yeniden konumladır bir yapının kullanılabilmesine olanak sağladı.
- Bu sayede farklı cihazlar için ayrı tasarım yapılma dönemi sona ermiş oldu.
- Bootstrap, mobil öncelikli olarak Duyarlı Web Siteleri geliştirmeyi kolaylaştıran bir HTML5 çatısıdır.
- Bootstrap JavaScript (JS) kütüphanelerinden ve CSS tanımlamalarından oluşur.
- Açık kaynak kodlu bir proje olan Bootstrap, birçok geliştirici tarafından tercih edilen popüler bir çatıdır.
- Bootstrap'in çalışma mantığı diğer HTML5 çatıları ile paralellik gösterir.
- Çalışmanın yapıldığı HTML dosyasından JS ve CSS dosyalarına referans verilerek bağlantı sağlanması yeterlidir.
- Bootstrap çekirdek dosyaları, İnternet üzerindeki depolardan kullanılabileceği gibi, kişisel bilgisayarınızın sabit diskine kaydedilerek de kullanılabilir.
- Dosyalara referans verilmesiyle tamamlanan kurulum işleminden sonra ilk yapılması gereken Duyarlı Web Sayfası Şablonu'nun tasarlanmasıdır.
- Hazırlanacak şablon daha sonra Web sitesinin iskeletini oluşturacak, farklı arayüz şablonlarına zemin teşkil edecektir.
- Gerek şablon, gerekse de şablondan üretilen dosyalar, İnternet tarayıcısı üzerinde test edilir.
- Bu test işlemi sırasında, İnternet tarayıcısının pencere boyutları değiştirilerek farklı cihaz ekranları simüle edilmiş olur.
- Bu işlem İnternet tarayıcısının içeriğini yakınlaştırarak (zoom) da gerçekleştirilebilir.
- Yapılan çalışmaların farklı İnternet tarayıcılarda test edilmesi önemlidir.
- Mobil cihazlardaki İnternet tarayıcıları ile masaüstü sistemlerdeki İnternet tarayıcılarının davranışları kimi zaman farklılıklar içermektedir. Bu farklılıklar sonucunda yaşanabilecek sorunları en aza indirmek için test işlemleri ilgili cihazın kendi içinde de yapılabilir.
- Bootstrap, birçok farklı bileşenin yer aldığı örnek tasarımları da kendi Web sitesinde bulundurur. Bu örnek çalışmalara rahatlıkla erişilebilir ve bir şablon dosya olarak kullanılabilir.
- Örnek Web sayfalarının şablon dosya olarak kullanılabilmesi için İnternet tarayıcılarının yardımıyla ilgili sayfanın kayıt edilmesi gerekir.
- Kayıt işlemi sırasında dikkat edilmesi gereken nokta Web sayfasının tüm bileşenleri ile kayıt edilmesidir.
- Kayıt işlemi gerçekleştirildikten sonra ana HTML dosyası üzerinde istenilen değişiklikler gerçekleştirilebilir ve kopyaları çıkartılarak Web sitesinin diğer sayfaları bu şablon temel alınarak oluşturulabilir.

DEĞERLENDİRME SORULARI

1. Aşağıdakilerden hangisi duyarlı bir Web sitesi tasarımının özellikleri içinde yer almaz?
 - a) HTML5 çatısına sahiptir.
 - b) Farklı Web yapılarına göre farklı bileşenler kullanılabilir.
 - c) Platformlar (İşletim sistemi, İnternet tarayıcı vb.) arası kullanıma uygun bir yapıdadır.
 - d) Ekran boyutu farklı olan cihazlar için ayrı ayrı tasarıma ihtiyaç duyulur.
 - e) Görüntülediği cihaza göre Web sayfasını yeniden yapılandırabilir.
2. Bootstrap kullanım hakları açısından aşağıdakilerden hangisi söylenebilir?
 - a) Açık kaynak kodludur.
 - b) Ticari amaçla kullanılamaz.
 - c) Kaynak kodların üzerinde değişiklik yapılamaz.
 - d) Kaynak kodlar kopyalanamaz.
 - e) Kaynak kodlar dağıtılamaz.
3. Bootstrap JavaScript ve CSS dosyalarının sıkıştırılmış sürümlerinin (min) sağladığı avantaj aşağıdakilerden hangisidir?
 - a) Boyut olarak daha küçük oldukları için Web sitesi optimizasyonu sağlar.
 - b) Dosyaların içeriğini herkes göremez, koruma sağlar.
 - c) Güncelleme işlemleri daha kolay gerçekleştirilir.
 - d) Farklı sunuculara aktarılması kolaydır.
 - e) Daha çok fonksiyonu içinde barındırabilir.
4. Bootstrap'i oluşturan çekirdek dosyaları İnternet üzerindeki depolardan kullanmanın avantajı aşağıdakilerden hangisidir?
 - a) Dosya boyutları daha küçük olur.
 - b) Birden çok kütüphane dosyası ile çalışmaya olanak sağlar.
 - c) Yedekleme işlemlerini hızlandırır.
 - d) Yazılan kodların daha hızlı çalışması sağlanır.
 - e) Dosyaların ilgili sürümlerinin her zaman en güncel hâlleri kullanılmış olur.
5. Bootstrap hangi bileşenlerden oluşur?
 - a) HTML, PHP, JavaScript
 - b) HTML, PHP, ASP
 - c) HTML, JavaScript, CSS
 - d) JavaScript, CSS, XLS
 - e) JavaScript, CSS, XML

6. Bir Bootstrap projesinde dosya tiplerine göre alt klasörler oluşturmanın avantajı nedir?
 - a) Web sitesinin toplam dosya boyutu küçülür.
 - b) Proje ilerledikçe artan dosya sayısının yol açacağı karmaşa önlenir.
 - c) Web sitesi kullanıcıları aradıklarını daha kolay bulur.
 - d) Web sitesinin güvenliğini artırır.
 - e) Web sitesinin daha hızlı açılmasını sağlar.
7. Duyarlı bir Web sitesi tasarımında menü çubuğu ile ilgili aşağıdakilerden hangisi söylenemez?
 - a) İç içe menü öğelerinden oluşabilir.
 - b) Arama kutusu gibi farklı bileşenler içerebilir.
 - c) Ekran görüntülenebilir alanına göre menü çubuğu yeniden yapılanamaz.
 - d) Sabit bir genişliğe sahip olamaz.
 - e) Aynı sayfada birden fazla menü çubuğu kullanılabilir.
8. Bootstrap standart olarak kaç sütunlu bir yapı oluşturmaya olanak verir?
 - a) 3
 - b) 6
 - c) 9
 - d) 12
 - e) 16
9. Bootstrap ile oluşturulacak içerik alanında her biri 3 sütundan toplam kaç bölme oluşturulabilir?
 - a) 3
 - b) 4
 - c) 6
 - d) 8
 - e) 12
10. Aşağıdakilerden hangisi Bootstrap ile gerçekleştirilebilecek işlemlerden değildir?
 - a) Veri toplamak için formların oluşturulması
 - b) Verilerin yayınlanabilmesi için tabloların kullanımı
 - c) Site içi navigasyonun sağlanması
 - d) Prototip oluşturulması
 - e) Veri tabanı bağlantısı

Cevap Anahtarı

1.d, 2.a, 3.a, 4.e, 5.c, 6.b, 7.c, 8.d, 9.b, 10.e

YARARLANILAN KAYNAKLAR

Bilge İş Web Tasarımının Temelleri Ders Notları. 18 Temmuz 2019 tarihinde <https://bilgeis.net/tr/courses/28/web-tasariminin-temelleri-html-ve-css> adresinden erişildi.

Bootstrap Resmî Web Sitesi. 14 Temmuz 2019 tarihinde <https://getbootstrap.com/> adresinden erişildi.

Bootstrap Sürüm 4 Dokümantasyon. 15 Temmuz 2019 tarihinde <https://getbootstrap.com/docs/4.3/getting-started/introduction/> adresinden erişildi.

Bootstrap Web Sayfası Örnekleri. 17 Temmuz 2019 tarihinde <https://getbootstrap.com/docs/4.3/examples/> adresinden erişildi.

Gelişken, U. (2016). Bootstrap 4: Responsive Tasarım Teknikleri ve Front-end Toolkit'leri. İstanbul: Level Yayınevi

Mermerkaya, A. (2015). HTML5 & CSS3. İstanbul: Abaküs Kitap

W3Schools Bootstrap Sürüm 4 Ders Notları. 14 Temmuz 2019 tarihinde <https://www.w3schools.com/bootstrap4/> adresinden erişildi.