

## FARKLI EKLAN ÇÖZÜNÜRLÜKLERİ İÇİN ESNEK WEB ARAYÜZ YAPILARI TASARLANMASI

Veli Özcan Budak <sup>1</sup>, Murat Gezer <sup>2</sup>

### ÖZET

Web'in hızlı gelişimi ve sahip olduğu esnek yapısı, bu platform üzerinde çeşitli uygulamaların ortaya çıkmasına sebep olmaktadır. Bilgiye hızlı erişmek için en çok tercih edilen seçeneklerden biri olan web siteleri bu uygulamalardan sadece biridir. Web siteleri üzerindeki resim, video, yazı, vb. her türlü içerik, görsel açıdan, uygulama içinde bulunan sayfaların arayüz yapıları sayesinde organize edilmektedir. Web'e erişimi bulunan cihazların boyutları düşünüldüğünde, web sayfalarında farklı çözünürlüklere yönelik ayrı tasarımlar geliştirmenin zaman ve emek açısından büyük bir iş yükü oluşturduğu açıktır. Bu sorun, farklı cihaz çözünürlüklerine özel arayüz yapılarının aynı anda pratik bir şekilde tasarlanmasıyla aşılabılır. Bu amaç doğrultusunda bu çalışmada, farklı cihazlar için arayüz yapılarını aynı anda tek ekran üzerinden oluşturabilen ve organize edilebilir bir prototip geliştirilmiştir. Bu sayede, kodlama bilgisine ihtiyaç duymadan kolay kontrol edilebilir bir arayüz yapısı tasarlama olanağı sunulmuştur. Diğer taraftan prototipin web uygulama geliştiricilerin kendi sistemlerinde kullanabilecekleri bir araç olabileceğine de inanılmaktadır.

**Anahtar Kelimeler:** Esnek web tasarımı, Çözünürlük, Bootstrap, PHP, MySQL

## DESIGNING RESPONSIVE WEB INTERFACE STRUCTURES FOR DIFFERENT SCREEN RESOLUTIONS

### ABSTRACT

Rapid growth of Web and it's flexible structure cause that the various applications come out on this platform. Websites which is one of the most preferred options to reach information rapidly is just one of these applications. Every sort of content such as picture, video and article etc. on websites are organized through the interface structures of the pages in these applications visually. When considering of the size of devices which have Web access, it is obvious that developing different designs for different resolutions creates a huge workload in terms of time and effort. This problem can be overcome practically by creating interface structures at the same time for different device resolutions. In accordance with this purpose in this study, a prototype has been developed which can create and organize the interface structures at the same time for different devices on a single screen. Thus, a possibility of designing interface structure is provided which can be controlled easily without any requirement coding knowledge. On the other hand, it is believed that the prototype will be a tool which web application developers will be able to use in their systems.

**Keywords:** Responsive Web Design, Resolution, Bootstrap, PHP, MySQL

---

\* Bu çalışma, Akademik Bilişim 2016'da sunulmuş olan "Esnek Web Ara Yüzü Tasarlanması Üzerine Bootstrap İle Bir Uygulama" başlıklı bildiriden üretilmiştir.

<sup>1</sup> Uzman, Kırklareli Üniversitesi, veliozcanbudak@gmail.com

<sup>2</sup> Öğr. Gör. Dr., İstanbul Üniversitesi, murat.gezer@gmail.com

## Giriş

İlk kez Tim O'Reilly (2005) tarafından dile getirilen Web 2.0 teknolojisinin sürekli gelişmesiyle birlikte İnternet ile birbirine bağlı bulunan kullanıcıların günlük yaşamlarında sıklıkla kullandıkları uygulamaların sayısı gittikçe artmaktadır. Bu uygulamalar, hitap ettiği kullanıcıların ihtiyaçlarına yönelik olarak günlük hayatı daha pratik hale getirmek amacıyla ortaya çıkarılmaktadır. Bu uygulamalar sayesinde, nerede olursak olalım mesafe tanımsızın istediğimiz yere bağlanıp bilgi paylaşımı yapabilme, her konuda sınırsız bilgi havuzuna ulaşabilme, hayranı olduğumuz bir sanatçının albümlerini dinleyebilme, video izleyebilme, banka hesaplarımızı oturduğumuz yerden organize edebilme, vb. bir sürü işlemi gerçekleştirebilmemiz mümkündür. 1989 yılında fizik araştırmacılarının paylaşımlı işlerinin birbirine bağlanabilmesi için Tim Berners Lee'nin bir öneri getirmesiyle (Veen, 2000) birlikte hayatımıza giren Web, 21. yüzyılın en çok kullanılan teknoloji ürünlerinden biri olarak hayatımızda yer etmektedir. Bu platform üzerinde kullanılan ve bilgi paylaşımı için en çok tercih edilen uygulamalardan biri olan web siteleri ise, ilk çıktığı zamanlarda çoğunlukla masaüstü bilgisayarlar sayesinde kullanılabilmesine rağmen zamanla ihtiyaçtan dolayı ortamdaki bağımsız bir şekilde erişilebilir olmaya mecbur kalmıştır. Web siteleriyle beraber İnternet'i temel alarak hizmet veren diğer uygulamaların da bu ihtiyacı daha belirgin hale getirmesi, masaüstü bilgisayarların gitgide yanımızda taşıyabileceğimiz boyutlara düşürülmesine öncülük etmiştir. Bu değişim sayesinde ortamdaki bağımsız olarak uygulamalara gittiğimiz her mekânda ulaşabilir hale gelmiş bulunmaktayız.

Günlük hayatta İnternet oldukça yoğun bir şekilde kullanılmaktadır. 2000-2015 yılları arasında İnternet'ten yararlanan kullanıcı miktarının 8 katlık bir artışla yaklaşık 3.5 milyar seviyesine ulaştığı tahmin edilmektedir (İnternet World Stats, 2015). Türkiye'ye bakıldığında ise, 16-74 yaş arası bireyler temel alınarak yapılan araştırmalarda, 2010 yılında %41,6 olan İnternet kullanımının 2015 yılında yaklaşık olarak %56 seviyesine yükseldiği belirlenmiştir (TUİK, 2015). İnternet'in bilgiye erişme noktasında başvurulan ilk seçeneklerden birisi olması dolayısıyla bu platformu kullanabilen cihaz türlerinin bilinmesi uygulama tasarlama noktasında oldukça önemlidir. Smart Insights (2015) tarafından Amerika'da yetişkinlerin İnternet'e erişimde hangi cihazları kullandıklarıyla ilgili bir araştırma yapılmıştır. İnternet üzerinde geçirilen günlük zaman verilerinden yararlanılarak elde edilen sonuçlarda, 2015 yılı itibarıyla tüm İnternet kullanıcıları içinde %51'lik kısmın mobil cihazları, %42'sinin masaüstü ve dizüstü bilgisayarları, geriye kalan %7'lik kısmın ise oyun konsolları vb. cihazları kullandıkları tespit edilmiştir. Aynı araştırmanın 2010 yılındaki verileri incelendiğinde, masaüstü ve dizüstü bilgisayar kullanımının %75, mobil cihaz kullanımının ise %12,5 olduğu belirlenmiştir (Smart Insights, 2015). Bu sonuçlardan yola çıkarak 5 yıllık süreç içerisinde mobil cihaz kullanımına olan eğilimin oldukça belirgin şekilde artış gösterdiği söylenebilir. Türkiye'deyse, Türkiye İstatistik Kurumu (TUİK) tarafından hanelerde bilişim teknolojileri sahiplenme oranlarının araştırıldığı bir çalışma yapılmış olup elde edilen veriler Tablo 1'de gösterilmektedir.

**Tablo 1:** Hanelerde bilişim teknolojileri bulunma oranı (TUİK, 2015).

Cihaz Türü	010	011	012	013	014	015
Masaüstü Bilgisayar	3,8	4,3	1,8	0,5	7,6	5,2
Taşınabilir Bilgisayar	6,8	2,6	7,1	1,4	0,1	3,2
Tablet				,2		
Cep Telefon (akıllı telefonlar dâhil)	0,5	1,9	3,2	3,7	6,1	6,8
İnternet'e bağlanabilen TV "Smart TV "				,3	2,4	0,9
Oyun Konsolu	,1	,8	,6	,0	,6	,3
Avuçiçi (PDA)	,7	,2	,4			

**Not:** Değerler yüzdeler olarak verilmiştir.

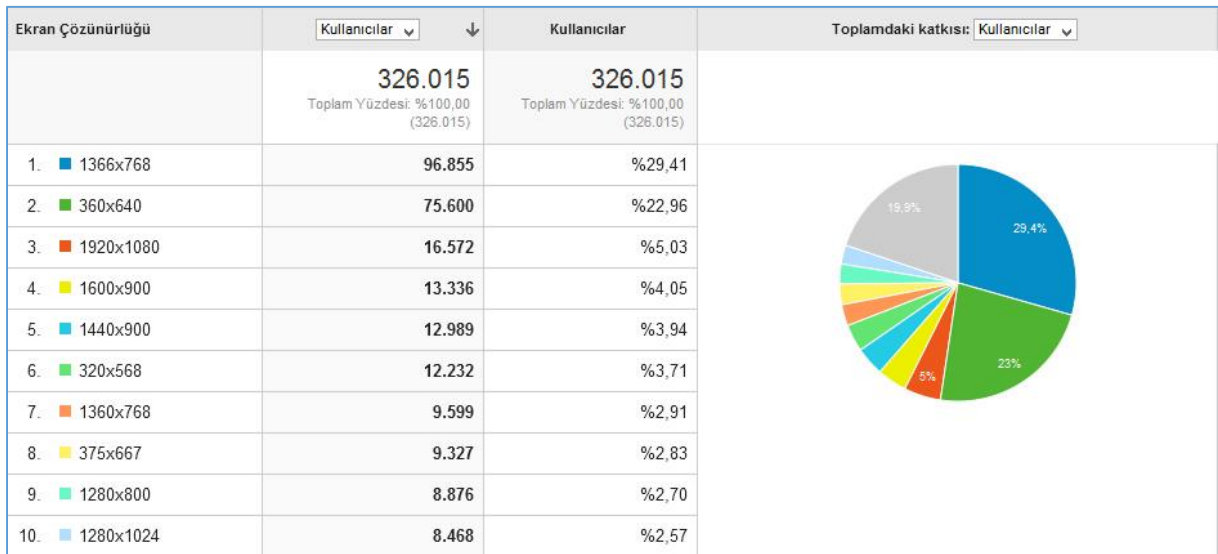
Tablo 1 incelendiğinde, cep telefonu (akıllı telefonlar dâhil) sahiplenme oranı 2010 yılında %90,5 iken, 2015 yılında bu oran %96,8 seviyesine çıkmaktadır. Aynı zaman aralığında masaüstü bilgisayar kullanım oranının %33,8'den %25,2'ye düştüğü, taşınabilir bilgisayar kullanım oranının %16,8'den %43,2'ye yükseldiği de dikkat çekmektedir (TUİK, 2015). Tablo 1 deki cihaz türleri dikkatle incelendiğinde, birbirinden farklı fiziki boyutlara ve İnternet erişim özelliğine sahip oldukları da tahmin edilebilir. İnternet ortamında hizmet veren ve dünya

genelinde en fazla üyeye sahip sosyal medya uygulamalarından biri olan Twitter, 2015 yılının son çeyreğiyle ilgili raporlarında, aylık ortalama toplam aktif kullanıcı miktarının geçmiş yıla oranla %9'luk büyümeyle 320 milyona ulaştığını belirtmiştir (Twitter, 2016). Bu raporda dikkati çeken nokta ise, mobil cihazlar üzerindeki kullanımın, bu miktarın yaklaşık %80'ini oluşturmasıdır (Twitter, 2016). Kırklareli Üniversitesi (KLU) web sitesini 01.01.2016-31.03.2016 tarihleri arasında ilk kez ziyaret eden kullanıcıların sahip oldukları cihazlara ait istatistikler ise Şekil 1'deki gibi ortaya çıkmıştır.



Şekil 1: KLU web sitesine erişimde kullanılan cihaz türleri (Google, 2016).

Genel itibariyle yapılan araştırmalar ve raporlar incelendiğinde, İnternet'e erişimde kullanılan cihazların farklılaştığı dikkat çekmektedir. Bu cihazları birbirinden ayıran temel özellik ise, fiziki boyutlarının ve dolayısıyla ekran çözünürlüklerinin farklı olmasıdır. Örneğin; mobil cihazlar, yanımızda daha rahat taşıyabilmemiz amacıyla tasarlanmış olmalarından dolayı masaüstü ve dizüstü bilgisayarlara oranla daha küçük ekran boyutlarına ve çözünürlüklerine sahiptirler. KLU web sitesinin 01.01.2016-31.03.2016 tarihleri arasındaki ziyaretçilerinin sahip olduğu cihazlara ait çözünürlükler Şekil 2'deki gibi oluşmuştur (Google, 2016).



Şekil 2: KLU web sitesini ziyaret eden kullanıcıların sahip oldukları cihazlara ait çözünürlükler (Google, 2016).

Şekil 2 incelendiğinde, kullanıcıların sahip olduğu cihaz çözünürlüklerinin oldukça fazla çeşitlilik gösterdiği görülmektedir. Şekil 2 üzerinde 2, 6 ve 8. sıradaki çözünürlükler akıllı telefonlara, bunların dışında yer alan diğer çözünürlükler ise akıllı telefonlara oranla daha büyük cihazlara ait değerlerdir. En ünlü web uygulama geliştirme sitelerinden biri olan W3school, kendi log dosyaları üzerinden ziyaretçilerinin kullandıkları cihazların ekran çözünürlüklerini tespit etmiştir (W3Schools, 2015). 2015 yılı için 1024x768'lik ekran boyutu merkeze alındığında, ziyaretçilerinin yaklaşık %97'lik bölümünün bu boyut ya da bu boyuttan büyük ekran kullandıklarını belirlemiştir (W3Schools, 2015). Yine aynı sonuçlarda, aynı yıl için %97'lik dilimin içindeki %33'lük kısmın, ekran boyutu olarak 1366x768'lik ekranı kullandığı tespit edilmiştir (W3Schools, 2015).

İnternet ortamındaki uygulamalara erişirken kullanılan cihazların farklı çözünürlüklere sahip olmaları, bu ortam üzerinde hizmet verecek uygulamaları geliştiren tasarımcılar açısından dikkat edilmesi gereken bir noktadır. Çünkü farklı cihaz çözünürlüklerine özel uygulama geliştirmek ekstra işlem yapmayı gerektirmektedir. Masaüstü veya dizüstü bilgisayar ortamına uyumlu olan web siteleri, telefonlar, tabletler ve hatta bilgisayarlara oranla daha büyük ekran çözünürlüklerine sahip cihazlarda da düzenli bir görünüme sahip olmalıdır. Bu durum, kullanıcı memnuniyeti açısından da büyük önem taşımaktadır. Çünkü farklı türdeki cihazlarla uyumlu olmayan uygulamalar, zamanla kullanım yoğunluğunu kaybetme tehlikesiyle karşı karşıya kalabilir. Bu açıklamayı destekler nitelikte Google market ekibinden Fisch (2012), mobil uygulama deneyiminin gittikçe önemli hale geldiğini ve mobil dostu web uygulama sahibi olmamanın rakiplerin işine geldiğini belirtmektedir. Yine de bu kadar farklı ekran çözünürlüğünün hepsine birden hâkim olabilecek bir uygulamanın varlığından şuan için söz etmek mümkün olmasa da en yoğun kullanılanları kapsayacak şekilde tasarım yapmak bu dezavantajı biraz olsun giderebilir.

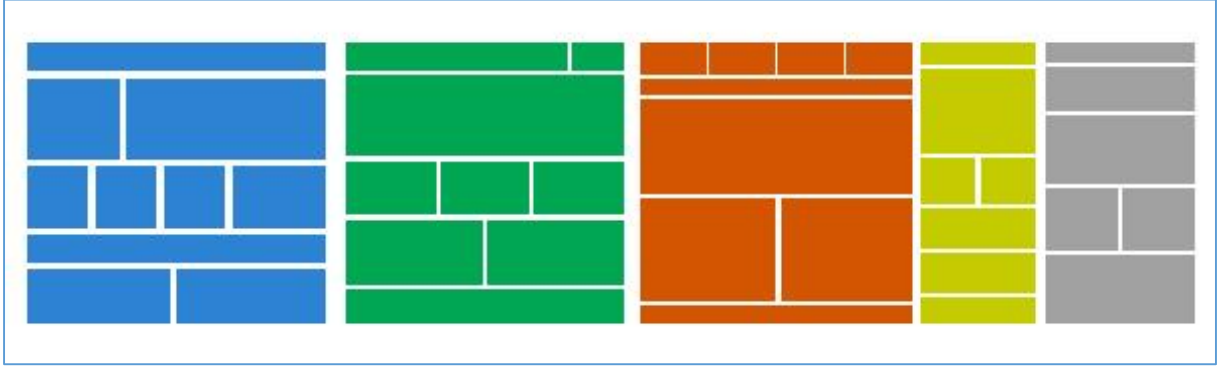
İnternet ortamında hizmet veren ve bu çalışmanın da odağında bulunan web siteleri, insanların bilgiye hızlı erişim için yoğun olarak kullandıkları uygulamalardan sadece birisidir. Kullanıcılarla iyi bir etkileşim oluşturulması amacıyla web sitelerinin piyasadaki İnternet erişimine sahip çoğu cihazla uyumlu olması oldukça önemlidir. Bu noktada Esnek Web Tasarımı (EWT) kavramı hayatımıza girmiştir. EWT ilk olarak, Ethan Marcotte'un 2010 yılında "A List Apart" sitesindeki "Responsive Web Design" başlıklı makalesiyle dile getirilmiştir (Marcotte, 2010). Bu makalesinde Marcotte, değişken ızgara yapılarının, esnek resimlerin ve medya sorgularının bu kavramın teknik malzemesi olduğunu belirtmiş ve bu kavramın aynı zamanda farklı düşünme şekillerine ihtiyaç duyduğunu belirtmiştir (Marcotte, 2010). Türkçe kaynaklarda "Duyarlı Web Tasarımı" olarak da karşımıza çıkan bu kavram Sharkie ve Fisher (2013) tarafından "en pratik olacak şekilde tek bir uygulamanın farklı cihazlara dağıtımına olanak sağlayan bir takım teknik ve teknolojilerin birleşimidir" şeklinde tanımlanmıştır. Bu düşüncenin temelinde yatan mantık, ziyaretçilerin web sitelerine farklı cihazlardan ulaştıkları zamanlarda, sayfaların, kullanılan cihazın boyutuna göre kendisini düzenleyebilmesidir. Bu sayede ziyaretçi kendi cihazına en uygun görünümde web sayfalarını gezerken daha memnun edici bir deneyim yaşayabilir. Bu düşünceyle tasarlanacak web siteleri sayesinde kullanıcılar sayfaları ziyaret ettiklerinde aradıkları bilgiye zaman kaybetmeden en az emekle ulaşabileceklerdir. Kısacası EWT sayesinde web siteleri, tüm cihazlarla uyumlu görünüme sahip olma şansını yakalamıştır.

EWT, HTML ve CSS3 medya sorguları kullanılarak bir web sayfası üzerindeki içerik elemanlarının, üzerinde görüntülediği cihazın çözünürlüğüne göre gizlenmesi, yeniden boyutlandırılması veya farklı bir noktaya taşınması ile gerçekleştirilmektedir. Bu sayede web sayfasındaki içerik elemanları, farklı cihazlarda yapılan ayarlamalara göre değişkenlik göstermektedir. Bu değişkenlik, web sayfası üzerinde kullanılan tüm elemanlara uygulanabilmektedir ve buradaki limit, tasarımcının hayal gücü doğrultusunda değişmektedir. EWT sayesinde örnek bir web sitesi farklı cihazlarda Şekil 3'deki gibi görünebilir:



Şekil 3: Esnek web tasarımı (Switzer Creative, 2015).

Web sitelerinin cihazlara göre kendini şekillendirebilmeleri ise, genel olarak sayfalardaki **arayüz yapılarının** esnekliğiyle ile sağlanmaktadır (Şekil 4). Yani cihazlara özel farklı davranış gösterme eylemi, web sitelerini oluşturan sayfalardaki arayüz yapılarının kendilerini cihazlara uydurmasıyla gerçekleşmektedir.



Şekil 4: Örnek arayüz yapıları.

#### Esnek Web Tasarımı Ve Bootstrap

Web siteleri, genellikle belirli cihazların ekran boyutlarına göre geliştirilmektedir. Tasarımcılar, hangi boyutta uygulama geliştireceklerine, hedef kitlelerinin yoğun olarak kullandığı cihaza ya da dünya genelinde en yoğun olarak kullanılan cihaz boyutuna göre karar vermektedirler. Cihaz boyutuna göre uygulama geliştirme noktasında 1'den fazla ekran boyutundan bahsedildiği zaman ise, işler biraz daha karmaşıklaşmaktadır. EWT tanımı ortaya çıkmadan önce tasarlanan web siteleri, ziyaretçilere, çoğunlukla masaüstü bilgisayar ortamında hizmet verirken, zamanın popüler ekran boyutuna göre tasarlanmaktaydılar. Ancak, zamanla Internet erişim özelliği bulunan farklı cihazların da hayatımıza girmesi bu durumunun değiştirilmesine sebep olmuştur. Bu cümleyi destekler nitelikte Ahmadi ve Kong (2012), mobil cihazların hayatımızda daha fazla yer etmeye başlamalarından beri masaüstü bilgisayarlar için tasarlanan çoğu web sitesinin, mobil cihazlarla uyumlu olmamalarından dolayı ziyaretçilerin ilgilendikleri içeriği bulmaya çalışırken sinir bozucu bir şekilde ekranı sürekli kaydırmalarına sebep olduğunu belirtmişlerdir. Bu sebeptendir ki, farklı cihazların kullanım oranlarının artması ve özellikle de mobil cihazların oldukça yoğun bir şekilde kullanımı, EWT'nın gitgide daha fazla önem kazanmaya başlamasına sebep olmuştur.

Tasarımcıların her cihaz boyutuna uygun bir şekilde web sitesi geliştirmeleri, hem verilecek emek açısından hem de harcanacak zaman açısından bir sorun teşkil etmektedir. EWT sayesinde bu sorun rahatlıkla aşılabilmektedir ama yine de web sayfalarında sunulması düşünülen içerik elemanlarının farklı boyutlarda gösterecekleri davranışları planlamak ve bunu tasarlamak belli bir iş yükü getirmektedir. Bu iş yükü, her bir içerik elemanı için HTML ve CSS3 medya araçlarının ayrı ayrı ayarlanması gerektiğinden gelmektedir.

Kendisini 'farklı niteliklerdeki insanlar, farklı şekillerdeki cihazlar ve farklı boyutlardaki projeler için tasarlanmıştır. Bootstrap web uygulama geliştirme sürecini daha hızlı ve kolay yapar.' şeklinde tanımlayan

Bootstrap (2015), web uygulama geliştirme sürecinin en önemli aşamalarından biri olan farklı cihazlarda uyumlu görünüm, yani EWT konusunda günümüz web uygulama geliştiricilerine çok kullanışlı araçlar sunan bir sistem olarak karşımıza çıkmaktadır. Twitter tasarımcılarının ve geliştiricilerinin ürettiği bu sistem, ilk olarak 2011 yılında kullanıma sunulmuştur. Ücretsiz olarak yararlanılabilen bu sistemin 30'u aşkın sürümü yayınlanmış olup halen geliştirilmeye devam etmektedir (Bootstrap, 2015).

Bootstrap içerik olarak, web uygulamalarındaki renk, yazı boyutu, yazı biçimi, büyüklük vb. görselliğe hitap eden tasarım işlemlerinde kullanılan CSS3 stil dilinin ve statik web uygulamalarını daha dinamik, hareketli ve animasyonlu bir biçimde sunmamıza olanak sağlayan Javascript dilinin birleşimiyle oluşturulmuştur. Bu iki dilin kompozisyonu olarak nitelendirebileceğimiz Bootstrap, farklı boyutlardaki cihazlar için daha uyumlu bir web uygulama geliştirme sürecinin oluşmasına katkıda bulunmaktadır. İçinde barındırdığı araçlar sayesinde aynı anda telefon, tablet, masaüstü bilgisayarlar ve daha büyük ekranlı cihazlar olmak üzere 4 farklı ekran boyutuna sahip cihaz türü için tasarım gerçekleştirmek mümkündür. Bu sayede web uygulama geliştiricileri tasarladıkları uygulamalarda, görünüm uyumluluğu noktasında bu araçlardan yararlanarak hem zamandan tasarruf etmiş hem de daha verimli bir geliştirme sürecini hayata geçirmiş olacaktırlar.

Bootstrap'e benzer biçimde EWT amacıyla ücretsiz olarak kullanılacak çeşitli sistemler de bulunmaktadır. Bunlardan bazıları; Skeleton (Skeleton, 2016), Yaml (Yaml, 2016), Kube (Imperavi, 2016), vb. şeklinde sıralanabilir. Bu sistemlerin de tıpkı Bootstrap gibi farklı cihazlarda arayüz yapılarını değiştirebilen araçları bulunmasına rağmen bu çalışma için son zamanlarda popüler olmaya başlayan Bootstrap kullanılmıştır.

### Prototip Bileşenleri ve Arayüzü

Web sayfalarında bulunan resim, yazı, video, vb. içerik elemanlarını organize etmeye yarayan arayüz yapılarının, pratik bir şekilde oluşturulabilmesi amacıyla geliştirilen prototip, açık kaynak kodlu yazılım dili olan PHP'yi ve veritabanı olarak MySQL'i kullanmaktadır. Prototip sayesinde, aynı anda farklı cihaz boyutları için arayüz yapıları tasarlanabilmektedir.

Prototip üzerinde, Bootstrap'in bazı araçlarının PHP ile programlanmasıyla oluşturulan arayüz yapıları, sürekli kullanılacak şekilde durması için MySQL veritabanına kaydedilmiştir. Bu sayede, hiçbir kodlamaya ihtiyaç duyulmadan hem farklı arayüz yapılarının tasarlanabilmesi hem de var olan arayüz yapılarının değiştirilebilmesi amaçlanmıştır. Bu çalışma için geliştirilen prototipe benzer bir yapı, mobil uyumlu olması amacıyla geliştirilip 2014 Eylül ayında hizmete sunulan KLU web sitelerinin tamamında kullanılmaktadır. KLU bünyesinde 2016 Nisan ayı itibarıyla toplam 134 web sitesinin yayında olduğu düşünüldüğünde, bu çalışmadakine benzer bir prototipin tasarımcılar açısından oldukça pratik bir geliştirme sürecini hayata geçireceği rahatlıkla söylenebilir.

Bootstrap, aynı anda 4 farklı cihaz türüne özel arayüz yapısı tasarlamak için kendi içinde "Izgara Yapısı" ve "Duyarlı Araçlar" olmak üzere 2 araca sahiptir. <http://getbootstrap.com/> adresinden incelenebilecek olan bu 2 aracın PHP ile programlanması sayesinde, arayüz yapıları hızlı ve pratik bir şekilde oluşturulabilir. Bu noktada sadece bu araçların ne işe yaradığının ve nasıl kullanıldığının bilinmesi gereklidir.

### BOOTSTRAP IZGARA YAPISI (BIY)

Bu yapı, farklı cihazlarda esnek bir görünüm oluşturmak ve web sayfalarını satır ve sütunlara bölmek için kullanılan mantıksal bir düzenleme aracıdır. Tüm cihazlar için geçerli olacak şekilde 12 sütuna kadar sayfayı bölebilmektedir. CSS stil dilinin '@media' elementi kullanılarak web sayfalarının farklı boyutlardaki cihazları tanıyıp ona göre davranış göstermesini sağlar. Kendi içinde kullanılabilir ön tanımlı CSS sınıflarını içermektedir. Bu sınıfları kullanarak bir sayfa, satır ve sütunlara bölünebilmektedir.

Her cihaz türüne yönelik olarak içinde barındırdığı CSS sınıfları farklı olup bu aracın genel olarak özellikleri Tablo 2' deki gibidir (Bootstrap, 2015).

**Tablo 2:** Bootstrap ızgara yapısı seçenekleri (Bootstrap, 2015).

<b>Boyut</b>	Çok Küçük	Küçük	Orta	Büyük
<b>Cihaz</b>	Telefonlar	Tabletler	Masaüstü	Masaüstü
<b>Ekran Genişliği</b>	<768px	>=768px	>=992px	>=1200px
<b>Sabit Genişlik</b>	Otomatik	750px	970px	1170px
<b>Stil Sınıfı</b>	.col-xs-	.col-sm-	.col-md-	.col-lg-
<b>Bölünebilir Sütun Miktarı</b>	12			

Tablo 2'ye göre küçük cihazlar için bir sayfa, 6 eşit sütuna bölmeye çalışıldığında aşağıdaki gibi bir kodun yazılması gerekecektir:

```
<div class="row"> → her bir satır için eklenmesi gereken stil sınıfı '.row'
  <div class="col-sm-2"> içerik </div>
  <div class="col-sm-2"> içerik </div>
  <div class="col-sm-2"> içerik </div>
  <div class="col-sm-2"> içerik </div>
  <div class="col-sm-2"> içerik </div>
  <div class="col-sm-2"> içerik </div>
</div>
```

Bu tarz örnekleri aynı anda farklı cihazlar için çoğaltmak da mümkündür.

### BOOTSTRAP DUYARLI ARAÇLAR (BDA)

Duyarlı araçlar, BIY aracı gibi farklı cihazlarda görünümüne etki etmektedir. BIY aracı sayesinde satır ve sütunlara bölünen sayfadaki her bir satırın, sütunun ya da parçacığın hangi cihazda görüneceğinin ve gizleneceğinin ayarlandığı araçtır. Tıpkı BIY gibi bu araç da kendi CSS stil sınıflarına sahiptir. Kullanılabilir CSS sınıfları Tablo 3'de verilmiştir (Bootstrap, 2015):

**Tablo 3:** Bootstrap duyarlı araçlar sınıfları (Bootstrap, 2015).

	Çok Küçük Cihazlar	Küçük Cihazlar	Orta Boyutlu Cihazlar	Büyük Cihazlar
.visible-xs-*	Görünür	Gizli	Gizli	Gizli
.visible-sm-*	Gizli	Görünür	Gizli	Gizli
.visible-md-*	Gizli	Gizli	Görünür	Gizli
.visible-lg-*	Gizli	Gizli	Gizli	Görünür
.hidden-xs	Gizli	Görünür	Görünür	Görünür
.hidden-sm	Görünür	Gizli	Görünür	Görünür
.hidden-md	Görünür	Görünür	Gizli	Görünür
.hidden-lg	Görünür	Görünür	Görünür	Gizli

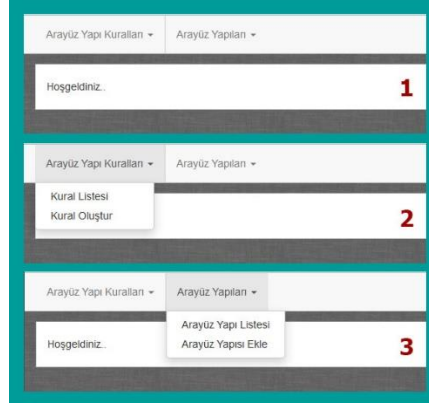
Bu aracın işleyiş mantığını anlamak için orta büyüklükteki cihazlarda 4 sütunlu olan ancak, büyük boyutlu cihazlarda tüm sütunların gizli olduğu kod biçimi aşağıdaki gibi olacaktır:

```
<div class="row"> → her bir satır için eklenmesi gereken stil sınıfı '.row'
  <div class="col-md-3 hidden-lg"> içerik </div>
  <div class="col-md-3 hidden-lg"> içerik </div>
  <div class="col-md-3 hidden-lg"> içerik </div>
  <div class="col-md-3 hidden-lg"> içerik </div>
</div>
```

Yukarıdaki örnekten de anlaşılacağı gibi kodun içerisinde hem BIY hem de BDA aracı aynı anda kullanılabilir. BDA, kendi başına kullanılabileceği gibi BIY ile birlikte kullanıldığında daha etkili bir şekilde arayüz yapısı oluşturmaya yardımcı olmaktadır.

**PROTOTİP ARAYÜZÜ**

Arayüz yapısı oluşturmak amacıyla geliştirilen prototipe ait ilk giriş ekranı ve kullanılacak 2 menü Şekil 5’de gösterilmektedir:



**Şekil 5:** Prototip giriş ekranı.

Şekil 5’de görüldüğü üzere prototip üzerinde arayüz yapısı oluşturmak için 2 tane menü bulunmaktadır. Bu menüler ve sahip oldukları alt menüler şu şekildedir:

1. Arayüz Yapı Kuralları
  - a. Kural Listesi
  - b. Kural Oluştur
2. Arayüz Yapıları
  - a. Arayüz Yapı Listesi
  - b. Arayüz Yapısı Ekle

Bu menülerden ilki olan “Arayüz Yapı Kuralları”, arayüz yapısı tasarlarken kullanılan ve sayfanın satır ve sütunlara bölünmesine yarayacak olan kuralların oluşturulduğu bölümdür. “Arayüz Yapıları” menüsüyse, oluşturulan kuralların kullanılmasıyla 4 farklı cihaz türü için arayüz yapılarının tasarlandığı bölümdür.

**Şekil 6:** Kural oluşturma ekranı.

Şekil 6’da görünen kural ekleme ekranı, arayüz yapılarında sayfayı satır ve sütunlara bölmek için kullanılacak olan kuralların eklendiği formdur. Formda “İsim” ve “Temsili Resim” alanları kuralı temsil edecek isim ve resim olmakla beraber diğer alan olan “Yapı Kodu (YK)” en önemli bölümü oluşturmaktadır. YK alanı doldurulurken dikkat edilecek nokta; BIY’nin bize sunduğu izne göre ayarlanmasının gerektiğidir. Bu kısımda YK yazılırken toplamda 12 olacak şekilde rakamlar aralarında “;” işareti konularak girilmektedir. Örnek olarak; “4;4;4”, “3;6;3”, “4;8”, “6;6” vb. değerler verilebilir. YK’nun toplamda 12 olmasının sebebi BIY sayesinde bir sayfanın en fazla 12 sütuna ayrılabilmesinden gelmektedir. Toplamda 12 olacak şekilde yazılmış olan bir YK “4;4;4” şeklinde ise, tek satırlık ve 3 eşit sütunluk bir kuralı, “6;6” şeklindeyse, tek satırdan oluşan 2 eşit sütunluk bir kuralı ya da “6;3;3” şeklindeyse sol taraftaki sütun, sağda eşit olarak bulunan 2 sütun



genişliğinde olacak şekilde tek satır ve 3 sütunluk bir kuralı oluşturacaktır. Kurallar oluşturulduktan sonra kuralların bulunduğu liste sayfası ise Şekil 7’de görülmektedir:

#	İsim	Temsilli Resim	Yapı Kodu
11	Sol Geniş - Sağ 2 Eşit		6,3,3
10	Sol 2 Eşit - Sağ Geniş		3,3,6
9	2 Sütun - Sağ Geniş		3,9
8	2 Sütun - Sol Geniş		9,3
7	3 Sütun - Orta Geniş		3,6,3
6	4 Eşit Genişlik		3,3,3,3
5	3 Eşit Genişlik		4,4,4
4	2 Eşit Genişlik		6,6
3	Tam Genişlik		12

Şekil 7: Kural listesi ekranı.

Şekil 7’de oluşturulmuş kurallar, arayüz yapılarının tasarlanacağı 2. menü olan, “Arayüz Yapıları” altında kullanılmaktadır. Arayüz yapısının tasarlandığı ekran ise Şekil 8’de verilmiştir:

Arayüz Yapı Kuralları - Arayüz Yapıları

İsim

Yapı için isim yazınız.

Büyük Boyutlu Cihazlar - G >= 1200px Orta Boyutlu Cihazlar - G >= 992px Tablet Benzeri Cihazlar - G >= 768px Telefon Benzeri Cihazlar - G < 768px 2

1

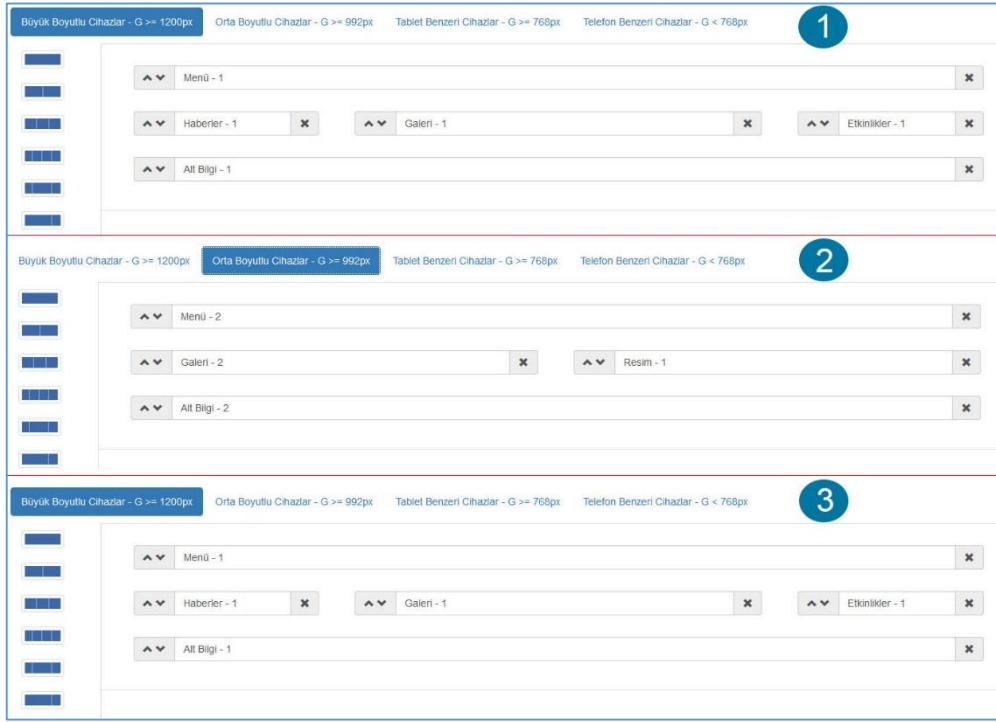
3

Arayüz Yapısı Oluştur

Şekil 8: Arayüz yapısı ekleme ekranı.

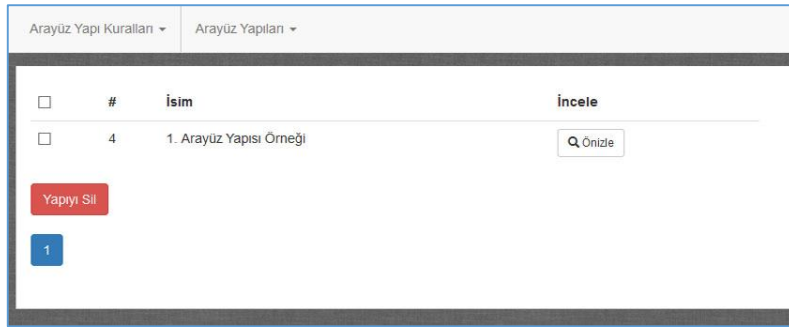
Şekil 8’de görünen “İsim” kısmı, oluşturacak olan yapıyı temsil edecek ismin girildiği bölümdür. Bu bölüm dışında 1 numaralı alan, daha önce oluşturulan kuralların bulunduğu alan, 2 numaralı alan, 4 farklı cihaz türü için tasarım yapılacak olan geçiş sekmelerinin bulunduğu alan ve son olarak 3 numaralı alan, tasarımın gerçekleştirildiği alandır. 1, 2 ve 3 alanları aynı anda kullanılarak tasarım gerçekleştirilmektedir. Örnek senaryo olarak; genişliği 1200px’den büyük olan cihazlar için tasarımın yapıldığı ilk sekme tıklandığında 3 numaralı tasarım alanı boş olarak gelmektedir. Daha sonra 1 numaralı alanda bulunan kurallardan istediğimize tıklayarak 3 numaralı tasarım alanına kuralın koduna göre satır ve sütunlar eklenmektedir (Şekil 9 – No: 1). İstenildiği kadar satır ve sütun eklendikten sonra diğer cihaz türleri için tasarım yapmak amacıyla 2 numaralı alanda farklı bir sekme tıklandığında 3 numaralı alan ilk olarak boş gelmekte ve yine 1 numaralı alanda bulunan kurallarla 3 numaralı tasarlama alanına satır ve sütunlar eklenebilmektedir (Şekil 9 – No: 2). Bir cihaz türü için 3 numaralı

alanda tasarım yapıp daha sonra farklı cihaz için tasarlama yaptıktan sonra tekrar tasarımı yapılmış cihazın olduğu sekmeye döndüğünde de son tasarım düzeni korunmaktadır (Şekil 9 – No: 3).



Şekil 9: Arayüz tasarlama aşamasında sekmeler arası geçişler.

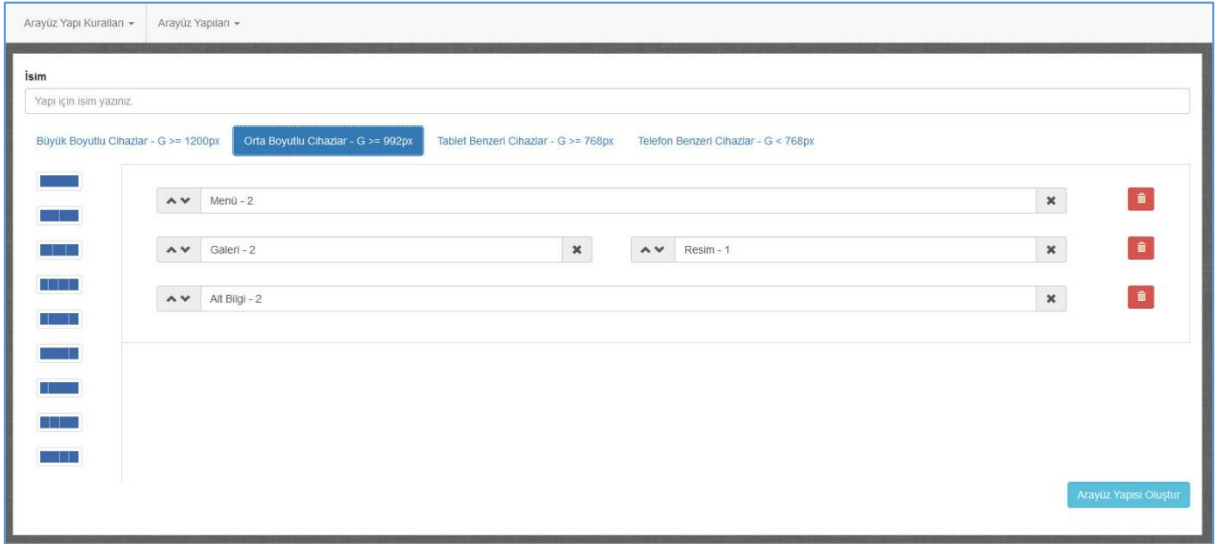
4 farklı türdeki cihaz için tasarımlar tamamlandıktan sonra Şekil 8’de görünen sağ alt köşedeki butonla da kayıt işlemi tamamlanmaktadır. Bu işlem sonucunda eklenen arayüz yapılarının bulunduğu liste Şekil 10’daki gibi görülmektedir:



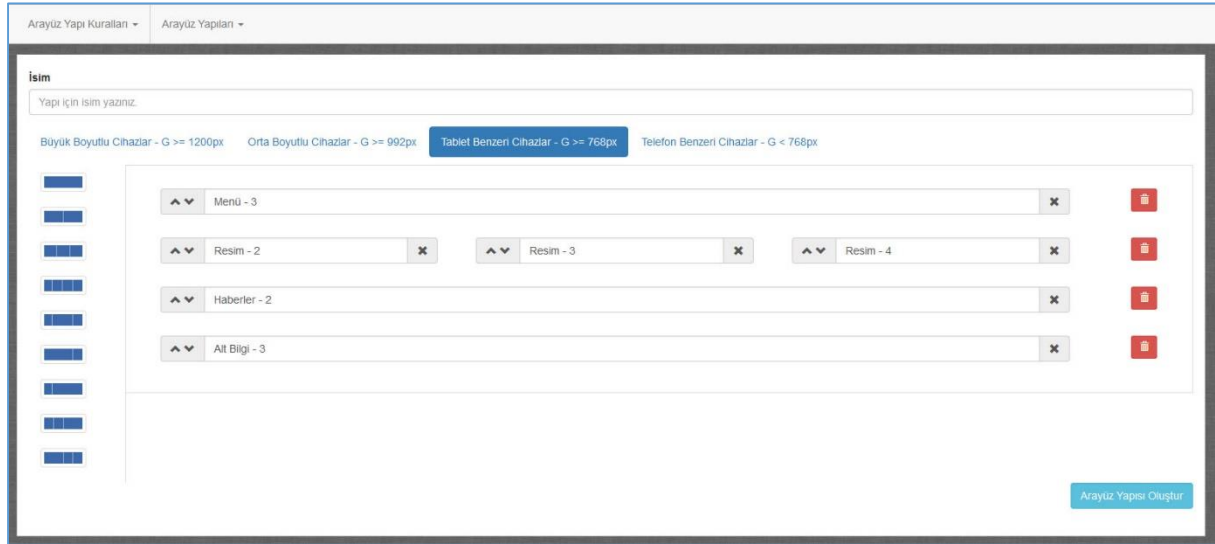
Şekil 10: Arayüz yapıları listesi.

### Uygulama ve Test

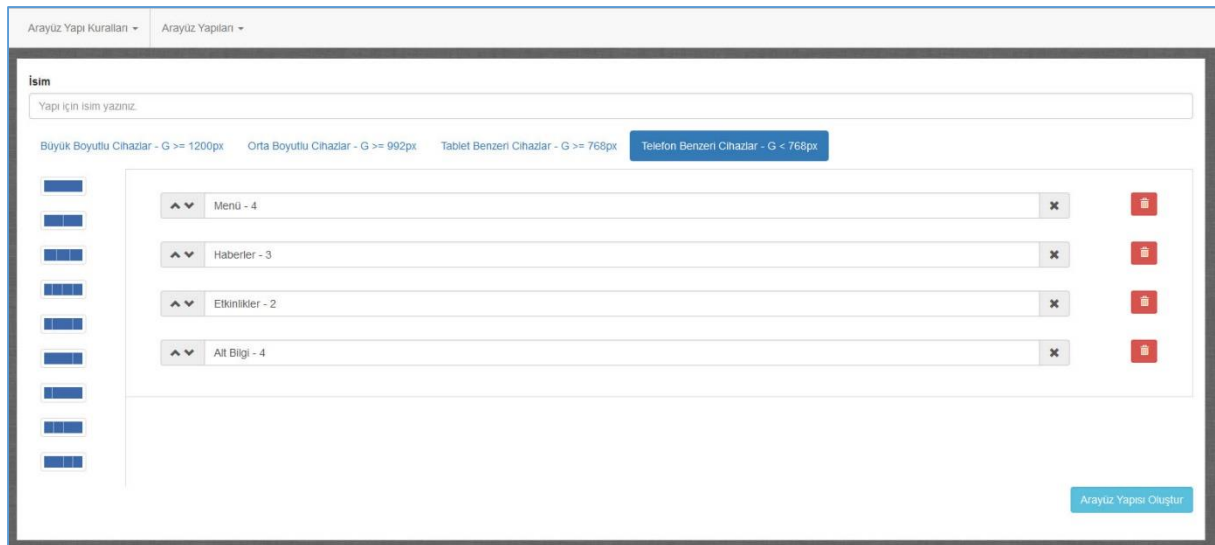
Arayüz yapısı oluşturmak amacıyla prototipte oluşturulan kurallar sayfayı satır ve sütunlara ayırırken kullanılmaktadırlar. Bu kurallar, BIY aracı sayesinde içerik elemanlarının organize bir şekilde sunulmasına yardım etmektedir. 4 farklı cihaz türü için aynı anda arayüz yapısı oluşturulabilen bu prototip, farklı sistemlere de entegre edilerek farklı amaçlar için kullanılabilir. Elbette ki bu prototipin nasıl çalıştığının da örneklenmesi yapılacak olan bu gibi araçların etkililiğini kanıtlaması açısından önemlidir. Buna yönelik olarak prototip üzerinde 4 farklı cihaz türü için 1 tane arayüz yapısı oluşturulmuştur. Bu örneğe ait arayüz yapı oluşturma ekranı üzerinde 4 farklı tasarım sırasıyla; “Büyük Boyutlu Cihazlar”, “Orta Boyutlu Cihazlar”, “Tablet Benzeri Cihazlar” ve “Telefon Benzeri Cihazlar” sekmelerinde düzenlenmiş olup aşağıdaki grafiklerde gösterilmiştir:



Şekil 12: Orta boyutlu cihazlar sekmesi.



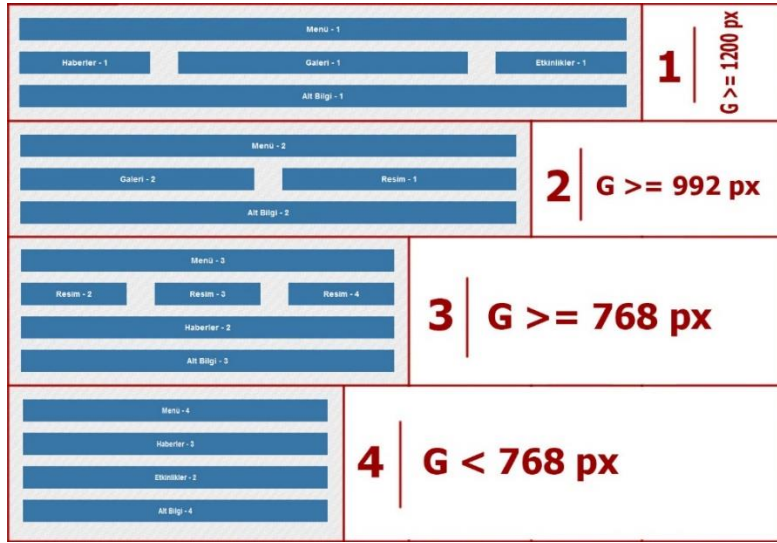
Şekil 13: Tablet benzeri cihazlar sekmesi.



Şekil 14: Telefon benzeri cihazlar sekmesi.

Yukarıdaki grafiklerde her bir cihaz türü için ayrı ayrı sekmelerde tasarımlar, kurallara tıklanarak oluşturulmuştur. Her cihaz türü için eklenen satırların içinde bulunan hücelere *textboxlar* (*hücreyle beraber Javascript sayesinde eklenmektedir*) eklenerek farklı isimler girilmiştir. Burada her bir hücreye farklı isim girilerek farklı içeriklerin hücelere bağlanabileceğini göstermek amaçlanmıştır. Prototipte yapılan örneklerde sadece isim yazılmasına rağmen farklı sistemlerde ihtiyaca göre bu hücelere resim, video, yazı, vb. içerik elemanları da bağlanabilir. Bununla birlikte her hücrenin solunda yer alan ok işaretleri aynı hizada alta ya da üste hücre satır eklemek için kullanılırken sağında bulunan çarpı işareti iste hücreyi silmek için kullanılmaktadır. Her bir satırın en sağında bulunan kırmızı arka plana sahip olan çöp kutulu butonlar ise kurala göre eklenen satırı içinde barındırdığı bütün sütunlarla beraber silmek için kullanılmaktadır.

4 cihaz türü için tasarımlar yapıldıktan sonra oluşturulan arayüz yapısı "1. Arayüz Yapısı Örneği" adıyla kaydedilmiştir (Şekil 10). Arayüz yapısının farklı cihaz boyutları için test edilmesiyle elde edilen sonuçlar ise Şekil 15'de gösterilmektedir:



Şekil 15: Oluşturulan arayüz yapısının ön izlemesi.

Şekil 15'deki arayüz görünümleri, ekran boyutu 1200 pikselden büyük bir bilgisayarda Google Chrome tarayıcısı üzerinden tarayıcı boyutunun değiştirilmesiyle elde edilmiştir. Prototipin, oluşturulan arayüz yapısındaki büyük (Şekil 15 – No: 1) ve orta boyutlu cihazlar (Şekil 15 – No: 2) için üretmiş olduğu kodlar ise Şekil 16'da görülmektedir:

```

1  <!--Orta Boyutlu Cihazlar-->
2  <div class="hidden-lg hidden-sm hidden-xs col-md-12">
3    <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Menü - 2</div>
4  </div>
5  <div class="hidden-lg hidden-sm hidden-xs col-md-6">
6    <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Galeri - 2</div>
7  </div>
8  <div class="hidden-lg hidden-sm hidden-xs col-md-6">
9    <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Resim - 1</div>
10 </div>
11 <div class="hidden-lg hidden-sm hidden-xs col-md-12">
12   <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Alt Bilgi - 2</div>
13 </div>
14 <!--Büyük Boyutlu Cihazlar-->
15 <div class="hidden-md hidden-sm hidden-xs col-lg-12">
16   <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Menü - 1</div>
17 </div>
18 <div class="hidden-md hidden-sm hidden-xs col-lg-3">
19   <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Haberler - 1</div>
20 </div>
21 <div class="hidden-md hidden-sm hidden-xs col-lg-6">
22   <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Galeri - 1</div>
23 </div>
24 <div class="hidden-md hidden-sm hidden-xs col-lg-3">
25   <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Etkinlikler - 1</div>
26 </div>
27 <div class="hidden-md hidden-sm hidden-xs col-lg-12">
28   <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Alt Bilgi - 1</div>
29 </div>

```

Şekil 16: Büyük ve orta boyutlu cihazlara yönelik arayüz yapısı kodları.

Şekil 16'nın devamı olacak şekilde tablet (Şekil 15 – No: 3) ve telefon benzeri cihazlar (Şekil 15 – No: 4) için üretilen kodlar da şekil 17'deki gibidir:

```

30 <!--Tablet Benzeri Cihazlar-->
31 <div class="hidden-lg hidden-md hidden-xs col-sm-12">
32 <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Menü - 3</div>
33 </div>
34 <div class="hidden-lg hidden-md hidden-xs col-sm-4">
35 <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Resim - 2</div>
36 </div>
37 <div class="hidden-lg hidden-md hidden-xs col-sm-4">
38 <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Resim - 3</div>
39 </div>
40 <div class="hidden-lg hidden-md hidden-xs col-sm-4">
41 <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Resim - 4</div>
42 </div>
43 <div class="hidden-lg hidden-md hidden-xs col-sm-12">
44 <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Haberler - 2</div>
45 </div>
46 <div class="hidden-lg hidden-md hidden-xs col-sm-12">
47 <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Alt Bilgi - 3</div>
48 </div>
49 <!--Telefon Benzeri Cihazlar-->
50 <div class="hidden-lg hidden-sm hidden-md col-xs-12">
51 <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Menü - 4</div>
52 </div>
53 <div class="hidden-lg hidden-sm hidden-md col-xs-12">
54 <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Haberler - 3</div>
55 </div>
56 <div class="hidden-lg hidden-sm hidden-md col-xs-12">
57 <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Etkinlikler - 2</div>
58 </div>
59 <div class="hidden-lg hidden-sm hidden-md col-xs-12">
60 <div style="padding:10px; margin:10px; text-align:center; background: #3876a8;color:#ffffff;font-weight:bold">Alt Bilgi - 4</div>
61 </div>

```

Şekil 17: Tablet ve telefon benzeri cihazlara yönelik arayüz yapısı kodları.

Şekil 16 ve Şekil 17 incelendiğinde prototipin dezavantajı olarak değerlendirilebilecek bir durum da söz konusudur. Bu çalışmada her bir hücreye isimlendirme yapılmasına rağmen farklı çalışmalarda hücrelere resim, video, vb. medya elemanlarının eklenmesi muhtemeldir. Bu durum, web sayfasının yavaş çalışmasına sebep olabilir. Bu dezavantajın giderilmesi noktasında her türlü içerik boyutunun küçültülerek arayüz yapılarındaki hücrelere bağlanması önemlidir.

Bu çalışmada Bootstrap altyapısı kullanılarak geliştirilen prototip benzeri Layoutit (Layoutit, 2016) ve Shoelace (Shoelace, 2016) gibi ücretsiz olarak yararlanılabilecek uygulamalar da mevcuttur. Bu uygulamalardan Layoutit, Bootstrap'ın tüm araçlarından yararlanan ve sürükle-bırak yöntemiyle tasarlama yapabilen bir araç sunmaktadır. Aynı zamanda istenilen tasarım yapıldıktan sonra tasarıma ait HTML kodları indirilebilmektedir. Diğer uygulama olan Shoelace ise, bu çalışmada da kullanılan BIY aracı yardımıyla sadece sayfa arayüz yapıları tasarlamaya ve bu yapının HTML kodlarını indirmeye olanak sağlamaktadır. Her iki uygulama, tek bir web uygulaması geliştirirken kullanılacak olan arayüz yapısını tasarlamaya yardımcı olabilir ancak bu yapıların veritabanına kaydedilme durumu söz konusu olmadığı için farklı arayüz yapısı istendiğinde tekrar aynı işlemleri yapmak gerekebilir. Bu çalışmadaki prototipin bu uygulamalarla karşılaştırıldığında en büyük farkı ise, çok sayıda arayüz yapısını tasarlayıp kaydedilebilmesidir. Bu sayede, istendiğinde kısa sürede pratik bir şekilde arayüzü değiştirmeye olanak sağlamaktadır.

## Sonuç ve Öneriler

Bu çalışmada, web siteleri tasarlanırken içerikleri organize etmek için kullanılan arayüz yapılarının, pratik şekilde tasarlanmasına yardım eden bir prototip geliştirilmiştir. Bu prototip üzerindeki kurallar, BIY sayesinde oluşturulurken kurullarla oluşturulan arayüz yapılarının ön izlemesinde ise BDA kullanılmıştır. Böylece BIY sayesinde sayfa, satır ve sütunlara bölerken BDA sayesinde de farklı türdeki cihazlar için farklı davranış gösterme işlemi başarılı bir şekilde gerçekleştirilmiştir. Ön izlemede, BDA'nın bize sunduğu içerikleri gizlemeye yarayan CSS sınıfları kullanılmıştır. Böylece eklenen satır ve sütunlar, sadece ilgili cihazda görünür olurken diğer cihaz türlerinde gizlenmiştir.

Çalışma kapsamında bu prototipin oluşturulmasında PHP dili kullanılmasına rağmen farklı web uygulama geliştirme dilleri de kullanılarak benzer amaçlara hitap eden araçlar geliştirilebilir. Bu sebeple web uygulama geliştiricilerinin bu çalışmada geliştirilmiş olan prototipe benzer bir yapıyı tasarlamaları için PHP dilini bilmelerine gerek yoktur. Bununla birlikte dikkat edilmesi gereken diğer nokta, arayüz tasarlama ekranında sayfayı satır ve sütunlara böldükten sonra yapı üzerindeki her bir hücreye isim verilmesidir. Bu çalışma için isim

verme işlemi tercih edilmesine rağmen daha farklı çalışmalarda ihtiyaca göre hücrelere medya, yazı vb. daha farklı içerik elemanları da bağlanabilir. Aynı zamanda arayüz yapısının tasarlandığı ekrandaki her bir hücre, bu çalışmada sabit tutulmasına rağmen yapılacak olan diğer çalışmalarda Javascript yardımıyla sürükle-bırak özelliği verilip daha esnek olması da sağlanabilir.

Sonuç olarak, bir web uygulamasında kullanıcıların karşısına çıkan arayüz yapılarının nasıl pratik bir şekilde oluşturulabileceği üzerinde çalışılmıştır. Oluşturulan prototip, Bootstrap'in "Izgara Yapısı" ve "Duyarlı Araçlar" araçlarının PHP ile rahatlıkla programlanabildiğini göstermiştir. Bu prototip sayesinde, web arayüz yapıları oluşturma işlemi daha pratik hale getirilmiştir. Farklı amaçlara hitap eden başka sistemlerde, bu prototipe benzer bir yapının kullanılması, içerik elemanlarını organize ederken hem zamandan hem de verilecek emekten tasarruf edilmesine yardımcı olacaktır. Bu çalışmadaki prototipe benzer bir yapı KLU bünyesindeki tüm web sitelerinde kullanılmaktadır. Üniversiteler gibi kurumsal olan ve aynı çatı altında çok fazla web sitesi barındıran yerlerde bu tarz bir uygulamanın kullanılması iş yükü açısından da oldukça önemli avantajlar sağlayabilir. KLU bünyesinde bulunan web siteleri, ilgili oldukları yerdeki yetkili kişilerce tasarlanmaktadır. Çok fazla sayıda web sitesinin tasarlanmasının gerektiği bu gibi durumlar için böyle bir yapının kullanılması tasarım sürecinin hızlanmasına da katkı sağlayabilir. Yani web sitelerini tasarlamak için kullanılan yönetim sistemlerine ya da daha farklı sistemlere dâhil edilecek olan bu çalışmadakine benzer bir prototip yardımıyla, yetkilendirilecek olan kişilerin kodlama bilgisine ihtiyaç duymadan içerikleri organize bir şekilde sunmasına yardım edilebilir.

#### Kaynakça

- Ahmadi, H., ve Kong, J. (2012). User-centric adaptation of Web information for small screens. *Journal of Visual Languages & Computing*, 23(1), 13–28.
- Bootstrap. (2015). *The world's most popular mobile-first and responsive front-end framework*. <http://getbootstrap.com/> adresinden 3 Eylül 2015 tarihinde alınmıştır.
- Fisch, M. (2012). *Mobile-friendly sites turn visitors into customers*. <http://googlemobileads.blogspot.com.tr/2012/09/mobile-friendly-sites-turn-visitors.html> adresinden 20 Kasım 2015 tarihinde alınmıştır.
- Google. (2016). *Kırklareli Üniversitesi Google Analitik Raporları*. Kırklareli: Kırklareli Üniversitesi.
- Imperavi. (2016). *Kube CSS Framework*. <https://imperavi.com/kube/> adresinden 29 Nisan 2016 tarihinde alınmıştır.
- Internet World Stats. (2015). *Internet Usage and World Population Statistics*. <http://www.internetworldstats.com/stats.htm> adresinden 15 Aralık 2015 tarihinde alınmıştır.
- Layoutit. (2016). *Bootstrap Interface Builder*. <http://www.layoutit.com/> adresinden 29 Nisan 2016 tarihinde alınmıştır.
- Marcotte, E. (2010). *Responsive Web Design*. <http://alistapart.com/article/responsive-web-design> adresinden 17 Aralık 2015 tarihinde alınmıştır.
- O'Reilly, T. (2005). *Web 2.0: Compact Definition?*. <http://radar.oreilly.com/2005/10/web-20-compact-definition.html> adresinden 12 Aralık 2015 tarihinde alınmıştır.
- Sharkie, C., ve Fisher, A. (2013). *Jump Start Responsive Web Design*. Australia: SitePoint Ltd.
- Shoelace. (2016). *Visual Bootstrap 3 Grid Builder*. <http://shoelace.io/> adresinden 29 Nisan 2016 tarihinde alınmıştır.
- Skeleton. (2016). *Skeleton: Responsive CSS Boilerplate*. <http://getskeleton.com/> adresinden 29 Nisan 2016 tarihinde alınmıştır.
- Smart Insights. (2015). *Mobile Internet Trends*. <http://www.smartinsights.com/internet-marketing-statistics/insights-from-kpcb-us-and-global-internet-trends-2015-report/attachment/mobile-internet-trends-mary-meeker-2015-1/> adresinden 6 Kasım 2015 tarihinde alınmıştır.
- Switzer Creative. (2015). *Solve your mobile woes with responsive web design*. <http://switzercreative.com/responsive-web-design-solve-mobile-woes/> adresinden 14 Ocak 2016 tarihinde alınmıştır.
- TÜİK. (2015). *Hanelerde Bilişim Teknolojileri Bulunma Oranı*. Türkiye İstatistik Kurumu.
- TÜİK. (2015). *Hanelerde Bilişim Teknolojileri Kullanımı*. Türkiye İstatistik Kurumu.
- Twitter. (2016). *Twitter Q4 and Fiscal Year 2015 Shareholder Letter*. San Francisco: Twitter .

Veen, J. (2000). *The Art & Science of Web Design*. San Francisco: New Riders.

W3Schools. (2015). *Screen Resolution Statistics*. [http://www.w3schools.com/browsers/browsers\\_display.asp](http://www.w3schools.com/browsers/browsers_display.asp) adresinden 2 Ocak 2016 tarinide alınmıştır.

Yaml. (2016). *YAML CSS Framework*. <http://www.yaml.de/> adresinden 29 Nisan 2016 tarihinde alınmıştır.