

Beykent Üniversitesi
Yazılım Mühendisliği
Yazılım Mühendisliği Tasarım Projesi
2020 – Bahar

Rapor 1-2

EVA Kişisel Sağlık Asistanı (EKSA)

Grup Elemanları

ARDA KEYİŞOĞLU 160313008

<https://github.com/canptleon>

FATİH GÖKTUĞ YILDIRIM 160313049

MEHMET UĞUR ATMACA 160313043

MERT EFE 160313005

SEMİH EFE 160313009

İçindekiler

1 EKSA	3
1 a. Projenin Açıklaması.....	3
2 b. Proje Ekibi.....	3
2 c. Projenin Aşamaları.....	5
2 Dönem İçinde Yapılan İşler(Yapılan işlerin yarısı ile hazırlanmıştır).....	7
2 a. Web Sitesi Menü Tasarımı(Arda Keyişoğlu).....	7
2 b. Web Sitesi Footer Tasarımı(Arda Keyişoğlu).....	8
2 c. Web Sitesi JavaScript Kod Bloğu(Arda Keyişoğlu).....	10
2 d. Web Sitesi Video Gösterim Paneli(Arda Keyişoğlu).....	11
2 e. Web Sitesi Vücut Kitle Endeksi Tasarımı(Arda Keyişoğlu).....	12
2 f. Web Sitesi İlaç Hatırlatma Tasarımı(Arda Keyişoğlu).....	13
2 g. Web Sitesi GoogleMaps Entegrasyonu(Arda Keyişoğlu).....	14
2 h. Mobil Uygulama İlaç Hatırlatma Tasarımı(Fatih Göktuğ Yıldırım).....	14
2 ı. Mobil Uygulama Navigator Bloğu(Fatih Göktuğ Yıldırım).....	15
2 j. Mobil Uygulama Component Listeleme(Fatih Göktuğ Yıldırım).....	17
2 k. Mobil Uygulama Kitle Endeksi Tasarımı(Fatih Göktuğ Yıldırım).....	18
2 l. Mobil Uygulama Kitle Endeksi Hesaplanması(Fatih Göktuğ Yıldırım).....	19
2 m. Mobx State Yönetimi(Fatih Göktuğ Yıldırım).....	20
2 n. Mobil Uygulama Navigasyon Tasarımı(Mehmet Uğur Atmaca,Fatih Göktuğ Yıldırım)...	22
2 o. Mobil Uygulama Randevu Ekranı Tasarımı(Mehmet Uğur Atmaca).....	23
2 p. Mobil Uygulama Giriş Ekranı ve Kullanıcı Kayıt Ekranı Tasarımı (Mehmet Uğur Atmaca).....	28
2 q. Mobil Uygulama Kullanıcı Profil Ekranı (Mehmet Uğur Atmaca).....	33
2 r. Mobil Uygulama Sayfa Yönlendirme Ekranı-Drawer Navigation(Mehmet Uğur Atmaca).....	34
2 s. Web Sitesi MVC Structure Tasarımı[Controller] (Mert EFE).....	35
2 t. Web Sitesi MVC Structure Tasarımı [Views] (Mert EFE).....	37
2 u. Web Sitesi Model Katmanı(Mert Efe).....	38
2 v. Web Sitesi Kullanıcı Ekranı Tasarımı(Mert Efe).....	40
2 w. Web Sitesi Randevu Ekranı Tasarımı(Mert Efe).....	41
2 x. Web sitesi Login[Sign Up] Tasarımı(Semih EFE).....	42
2 y. Web site Login [Sign in] Tasarımı(Semih EFE).....	43
2 z. Web Sitesi Haberler Ekran Tasarımı(Semih Efe).....	44
2 aa. Web Sitesi Blog Ekran Tasarımı(Semih Efe).....	45
2 ab. Web Sitesi Kullanıcı Giriş Tasarımı (Semih Efe).....	46
2 ac. Web Sitesi Kullanıcı Kayıt Ol Tasarımı (Semih Efe).....	47
3 Sonuç	47
3 a. Başarı ile tamamlanan işler.....	47
3 b. Başarı ile tamamlanamayan işler.....	48

3 c. Kişilerin görevlerini yerine getirme durumu.....	48
3 d. İlerlemeler	50

1 EVA Kişisel Sağlık Asistanı

1 a. Projenin Açıklaması

İnternet bazı konularda gerekli bilgiye ulaşmada yeterli değildir. İnternet'ten alınacak bilginin içeriği doğru olsa da bazı durumlarda bu bilgiye kısa sürede erişmek mümkün olmamaktadır.

Bu çalışmada gerçekleştirilen proje, bir canlı için en önemli şey olan sağlık sektörüne hizmet etmeyi amaçlamaktadır. Geliştirilen uygulama, kişinin her an erişebildiği telefonundan indirilebilecek ve aynı zamanda arzu eder ise web sitesine de erişim sağlayabilecektir. Bu sayede kişi istediği zaman saatine bakar gibi EVA ile sesli olarak iletişime geçip, ondan gerekli bilgileri alabilecektir. Bu durumda da EVA'nın önerilerine uyar ise yaşamı daha kaliteli bir hal alacaktır.

Gereksiz ilaç kullanımı ve hastane ziyaretlerinin sayısı azaldıkça, hem insanlar daha uzun ve sağlıklı yaşamlarına kavuşabilecek hem de hastane çalışanları, bu kişilerden daha çok yardıma muhtaç insanlara yardım edebileceklerdir.

1 b. Proje Ekibi

Tablo – 1: Proje Ekibi

	Numara	Ad Soyad	Projedeki Görev Tanımı
1	160313008	Arda Keyişoğlu	<ul style="list-style-type: none"> Proje konusuna uygun web sitesi için uygun sayfa-başlık ve bileşenlerin araştırılması.(Web tasarım standartları) HTML CSS JavaScript, Ana sayfa-Hakkımızda-Bize ulaşın-Doktorlar ekran tasarımı ve kodlaması HTML CSS JavaScript, Randevu sistemi, İlaç hatırlatıcı, Vücut Kitle Endeksi veTahlil sorgulama ekran tasarımı MSSQL, İlaç hatırlatıcı, Vücut kitle endeksi ve Tahlil sorgulama ekranı veritabanı kodlaması Google Maps navigasyon entegrasyonu ve ekran tasarımı Adobe Photoshop, logo ve web sitesi içerisinde kullanılan projeye uygun görsellerin hazırlanması MVC İlaç hatırlatıcı, Tahlil sorgulama, Vücut Kitle Endeksi kodlaması. Veri Tabanı oluşturulmasında bölünen kısma veri eklenmesi/tablo oluşturulması/tablo düzenlenmesi. MVC OCR teknolojisine uygun tasarımlar ve kontrol testleri.
2	160313049	Fatih Göktuğ Yıldırım	<ul style="list-style-type: none"> React-Native ilaç hatırlatma ekran tasarımı ve kodlaması. React-Native Alternatif Giriş ekran tasarımı ve

			<p>kodlaması.</p> <ul style="list-style-type: none"> • React-Native ilaç ekleme ekran tasarımı ve kodlaması. • React-Native Ekran navigasyon mimarisi tasarımı ve kodlaması. • React-Native mobx uygunluğu araştırma ve uygunsa entegre edilmesi. • React-Native Profil ve Ayarlar ekran tasarımı ve kodlaması. • React-Native OCR teknoloji entegre çalışması. • Veri Tabanı oluşturulmasında bölünen kısmı Create etme. • MSSQL de Kullanıcı ve rol aşamalarını create edilmesi. • MSSQL de Kullanıcı ve İlaç ekran aşamalarını create edilmesi. • MSSQL de Kullanıcı ve rol aşamalarını create edilmesi. • MSSQL de Doktor,Hastane ve hasta ilişkisinin create edilmesi. • MSSQL de Kullanıcı ve rol aşamalarını create edilmesi. • MSSQL Ocr dışında güncellenecek işlemler. • Api' kodlarının doğruluğunu test etme ve hayata geçirme.
3	160313043	Mehmet Uğur Atmaca	<ul style="list-style-type: none"> • React-Native Giriş ekran tasarımı ve kodlaması. • React-Native Randevu Alma ekran tasarımı ve kodlaması. • React-Native Kayıt ekran tasarımı ve kodlaması. • React-Native Ekran navigasyon mimarisi tasarımı ve kodlaması. • React-Native Profil ve Ayarlar ekran tasarımı ve kodlaması. • Veri Tabanı oluşturulmasında bölünen kısmı Create etme. • Api' kodlarının doğruluğunu test etme ve hayata geçirme. • MVC OCR teknolojisi entegre çalışması. • React Native Anasayfa ekran tasarımı. • React-Native OCR teknolojisi entegre çalışması. • OCR ile ilgili kaynakların ve 3.parti yazılımların sisteme entegre edilmesi. • OCR verilerinin okunması, yorumlanması, veri tabanına aktarılması işlemleri. • OCR verilerinin tutulacağı MSSQL tablolarının oluşturulması. • MSSQL View ve Procedurelerin oluşturulması. • Tahili bilgilerinin toplanması, referans aralılarının tespiti, parametrelerin tanımlanması. • Veri tabanının oluşturulması, veri tabanı

			<p>yedeklemelerinin yapılması, kullanıcı tanımlamalarının yapılması.</p> <ul style="list-style-type: none"> • React Native Tahillerin karşılaştırılması ekranın hazırlanması. • React Native Tahlil Yükleme ekranı tasarımı.
4	160313005	Mert Efe	<ul style="list-style-type: none"> • HTML CSS JavaScript, Alternatif Randevu sistemi, İlaç hatırlatıcı ve Tahlil sorgulama ekran tasarımı ve kodlaması • MVC Randevu sistemi, İlaç hatırlatıcı ve Tahlil sorgulama kodlaması. • Veri Tabanı oluşturulmasında bölünen kısmı Create etme. • MVC Controller ve Model tasarımı kodlaması ve dizaynı • MVC EntityFrameWork tool eklentisi ve entegre edilmesi • MVC Structure'ı ile websitenin entegre edilmesi ve tekrar oluşturulması • MVC Structure'ı ile veritabanı bağlantısının sağlanması
5	160313009	Semih Efe	<ul style="list-style-type: none"> • HTML CSS JavaScript, Alternatif Randevu sistemi, İlaç hatırlatıcı ve Tahlil sorgulama ekran tasarımı ve kodlaması • MVC Randevu sistemi, İlaç hatırlatıcı ve Tahlil sorgulama kodlaması. • Veri Tabanı oluşturulmasında bölünen kısmı Create etme. • Webstie ve MVC için özel JQuery,Bootstrap kodlaması ve entegresi • Website için özel Login ekranlarının dizaynı ve entegresi • Website için özelleştirilmiş CSS kodlarının entegresi

1 c. Projenin Aşamaları

Tablo – 2: İş paketleri

No	İş Paketi Adı	Dönem	İçeriği
1	Analiz	01.15.2020 – 11.03.2020	<p>Yapım aşamı için gereksinimler belirlenmesi.</p> <p>Projenin maliyet, fayda ve sorunlarının belirlenmesi.</p> <p>Projenin kısa ve uzun dönemli amaçları.</p> <p>Yeterlilik ve güncel olup olmadığının belirlenmesi.</p>

			Güvenilirlik şartlarının sağlanması. Gereksinimlere göre görevlere ayrılıp proje bazlı geliştirmeler yapılması. Rakiplerin arayüzleri ve kullanılabilirliği analiz edilmesi.
2	Tasarım	01.15.2020 – 11.03.2020	Anasayfa ve diğer ekranların geliştirilmesi. Hatırlatma ve Tahlil sorgulama ekranlarının tasarlanması. Tasarım şablonları oluşturularak en uygun arayüz belirlenmesi. Web site tasarımı için bootstrap framework, jquery, css, html, javascript kullanılması. Web sayfası için Mvc'ye entegre edilip Mvc mimarisinin kullanılması. Mobil-App için React Native kullanılması. Arayüzlerin entegre edilmesi. Web ve Mobil için responsive olması.
3	Ön Uç Geliştirme	10.02.2020 – 30.03.2020	Mobil programlama hybrid bir platform olduğu için arka uç geliştirmeyi de kapsamaktadır bu durum yapılacak tasarımların React ile bütünleştirilmesi yapılır. Web programlama kısmında güncel bootstrap.js kütüphaneleri ile belirlenen ara yüzler geliştirilir.
4	Arka Uç Geliştirme	10.02.2020 – 30.03.2020	Ön yüzü geliştirilen arayüzler her iki platformda da fonksiyonellikleri çalışır hale getirilmelidir.
5	Tasarım Testleri	31.03.2020 – 03.04.2020	Fonksiyonlar çalıştığı zaman tasarımda problemlerin çıkıp çıkmadığı, farklı ekran boyutlarında şekil bozuklukları var mı bunlar kontrol edilir.
6	Veri Tabanı Tasarımı	20.03.2020 – 23.03.2020	Raporda tasarlanan veritabanı tablo modelleri ve özellikleri sonucu da Create edilir ve yayına alınır.
7	Api Geliştirilmesi	31.03.2020 – 05.04.2020	Veritabanı modelinin verilerini erişme ve veri kaydetme seçeneklerini hayata geçirebilmek ve güvenliği sağlamak için API geliştirilir.
8	Test	20.04.2020	Apiden gelen verilen ile tasarım da ve arka uç kısmında verilerin doğru işlenmesi kontrol edilir.Bu kontrolün ardından platformlardan apiye girilen veriler veritabanına doğru bir şekilde ulaşmış mı kontrol edilir.Verilerin

			doğru bir şekilde gelip gelmeyeceği ve yanlış verilen gelmesi durumunda neler olacağı gözlemlenir.
--	--	--	--

2 Dönem İçinde Yapılan İşler

2 a. Web Sitesi Menü Tasarımı(Arda Keyişoğlu)

Öncelikle web sitesine eklenecek olan ana başlıkları seçilip, sağlık alanına uygun bir tasarım ile genel format oluşturulmuştur(HTML CSS JsScript). Tasarıma uygun seçilen fontlar kullanılmıştır. Gerekli görseller düzenlenip uygun bölümlere yerleştirilmiştir.

Aşağıdaki kod bloğu, sitenin en üstünde yer alan menü bölümüne aittir. Bu bölümde W3C ve Open Stand standartlarından yararlanılmıştır. Şu ana kadar herhangi bir sorun ile karşılaşmamıştır.”Header part” olarak ayrılıp tüm ekranlarda kullanılmıştır.

```
<!--:header part start:-->
<header class="main_menu home_menu">
  <div class="container">
    <div class="row align-items-center">
      <div class="col-lg-12">
        <nav class="navbar navbar-expand-lg navbar-light">
          <a class="navbar-brand" href="index.html">  </a>
          <button class="navbar-toggler" type="button" data-toggle="collapse"
            data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
            aria-expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
          </button>
          <div class="collapse navbar-collapse main-menu-item justify-content-end"
            id="navbarSupportedContent">
            <ul class="navbar-nav align-items-center">
              <li class="nav-item active">
                <a class="nav-link" href="anasayfa.html">Ana Sayfa</a>
              </li>
              <li class="nav-item">
                <a class="nav-link" href="hakkımızda.html">Hakkımızda</a>
              </li>
              <li class="nav-item">
                <a class="nav-link" href="randevuSistemi.html">Randevu Sistemi</a>
              </li>
              <li class="nav-item dropdown">
                <a class="nav-link dropdown-toggle" href="blog.html" id="navbarDropdown"
                  role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
                  Ek Hizmetler
                </a>
              </li>
            </ul>
          </div>
        </nav>
      </div>
    </div>
  </div>
</header>
```

```

        </a>
        <div class="dropdown-menu" aria-labelledby="navbarDropdown">
            <a class="dropdown-item" href="ilacHatirlatici.html">İlaç Hatırlatıcı</a>
            <a class="dropdown-item" href="tahlilSorgulama.html">Tahlil Sorgulama</a>
        </div>
    </li>
    <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="blog.html" id="navbarDropdown_1"
            role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
            Boş Başlık
        </a>
        <div class="dropdown-menu" aria-labelledby="navbarDropdown_1">
            <a class="dropdown-item" href="bos1.html">Alt Başlık 1</a>
            <a class="dropdown-item" href="bos2.html">Alt Başlık 2</a>
        </div>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="bizeUlas.html">Bize Ulaşın!</a>
    </li>
    <li class="d-none d-lg-block">
        <a class="btn_1" href="#">Kullanıcı Girişi</a>
    </li>
</ul>
</div>
</nav>
</div>
</div>
</div>
</header>
<!-- Header part end-->

```

Kod Örneği – 1: Web Sitesi Menü Tasarımı(Arda Keyişoğlu)

2 b. Web Sitesi Footer Tasarımı(Arda Keyişoğlu)

Aşağıdaki kod bloğu, sitenin en altında yer alan iletişim, kısa yollar gibi bölümlere aittir. Bu bölümde W3C ve Open Stand standartlarından yararlanılmıştır. Şu ana kadar herhangi bir sorun ile karşılaşılmamıştır.”Footer part” olarak ayrılıp tüm ekranlarda kullanılmıştır.

```

<!-- footer part start-->
<footer class="footer-area">
    <div class="footer section_padding">
        <div class="container">
            <div class="row">
                <div class="col-xl-2 col-sm-4 mb-4 mb-xl-0 single-footer-widgit">
                    <h4>Ek Hizmetler</h4>

```



```

<ul>
  <li><a href="#">İlaç Sorgulama</a></li>
  <li><a href="#">Tahlil Sorgulama</a></li>
  <li><a href="#">Boş 1</a></li>
  <li><a href="#">Boş 2</a></li>
  <li><a href="#">Boş 3</a></li>
</ul>
</div>
<div class="col-xl-2 col-sm-4 mb-4 mb-xl-0 single-footer-widget">
  <h4>Kısayollar</h4>
  <ul>
    <li><a href="#">Hakkımızda</a></li>
    <li><a href="#">Randevu Sistemi</a></li>
    <li><a href="#">Bize Ulaşın</a></li>
    <li><a href="#">Kullanıcı Kayıt Ol</a></li>
    <li><a href="#">Kullanıcı Giriş</a></li>
  </ul>
</div>
<div class="col-xl-2 col-sm-4 mb-4 mb-xl-0 single-footer-widget">
  <h4>Keşfet</h4>
  <ul>
    <li><a href="#">Boş 1</a></li>
    <li><a href="#">Boş 2</a></li>
    <li><a href="#">Boş 3</a></li>
    <li><a href="#">Boş 4</a></li>
    <li><a href="#">Boş 5</a></li>
  </ul>
</div>
<div class="col-xl-2 col-sm-4 mb-4 mb-xl-0 single-footer-widget">
  <h4>Kaynaklar</h4>
  <ul>
    <li><a href="#">Boş 1</a></li>
    <li><a href="#">Boş 2</a></li>
    <li><a href="#">Boş 3</a></li>
    <li><a href="#">Boş 4</a></li>
    <li><a href="#">Boş 5</a></li>
  </ul>
</div>
<div class="col-xl-4 col-sm-8 col-md-8 mb-4 mb-xl-0 single-footer-widget">
  <h4>Haberler</h4>
  <p>EVA'daki son gelişmelerden anında haberdar olmak için email adresinizi girmeniz yeterli.</p>
  <div class="form-wrap" id="mc_embed_signup">
    <form target="_blank"
      action="https://spondonit.us12.list-manage.com/subscribe/post?u=1462626880ade1ac87bd9c93a&amp;id=92a4423d01"
      method="get" class="form-inline">

```

```

<input class="form-control" name="EMAIL" placeholder="Email Adresinizi Giriniz"
onfocus="this.placeholder = ''" onblur="this.placeholder = 'Email Adresinizi Giriniz'"
required="" type="email">
<button class="click-btn btn btn-default text-uppercase">Abone ol</button>
<div style="position: absolute; left: -5000px;">
<input name="b_36c4fd991d266f23781ded980_aefe40901a" tabindex="-1" value=""
type="text">
</div>
<div class="info"></div>
</form>
</div>
</div>
</div>
<div class="copyright_part">
<div class="container">
<div class="row align-items-center">
<p class="footer-text m-0 col-lg-8 col-md-12">
Copyright &copy;<script>document.write(new Date().getFullYear());</script> Tüm haklar saklıdır. | Bu tasarımı EVA Sağlık Asistanı ekibi
tarafından yapılmıştır.</p>
<div class="col-lg-4 col-md-12 text-center text-lg-right footer-social">
<a href="#"><i class="ti-facebook"></i></a>
<a href="#"><i class="ti-twitter"></i></a>
<a href="#"><i class="ti-instagram"></i></a>
<a href="#"><i class="ti-skype"></i></a>
</div>
</div>
</div>
</div>
</footer>
<!-- footer part end-->

```

Kod Örneği – 2: Web Sitesi Footer Tasarımı(Arda Keyişoğlu)

2 c. Web Sitesi JavaScript Kod Bloğu(Arda Keyişoğlu)

Aşağıdaki kod bloğu, sitenin tüm hareketliliğini sağlayan kısımdır, farklı bileşenlerin kendilerine has kodlarının referanslarını vermektedir.Bölüm ” jquery plugins” olarak ayrılıp tüm ekranlarda kullanılmıştır.

```

<!-- jquery plugins-->
<script src="js/jquery-1.12.1.min.js"></script>
<!-- popper js -->
<script src="js/popper.min.js"></script>
<!-- bootstrap js -->
<script src="js/bootstrap.min.js"></script>

```

```

<!-- easing js -->
<script src="js/jquery.magnific-popup.js"></script>
<!-- swiper js -->
<script src="js/swiper.min.js"></script>
<!-- swiper js -->
<script src="js/masonry.pkgd.js"></script>
<!-- particles js -->
<script src="js/owl.carousel.min.js"></script>
<script src="js/jquery.nice-select.min.js"></script>
<!-- swiper js -->
<script src="js/slick.min.js"></script>
<script src="js/jquery.counterup.min.js"></script>
<script src="js/waypoints.min.js"></script>
<!-- contact js -->
<script src="js/jquery.ajaxchimp.min.js"></script>
<script src="js/jquery.form.js"></script>
<script src="js/jquery.validate.min.js"></script>
<script src="js/mail-script.js"></script>
<script src="js/contact.js"></script>
<!-- custom js -->
<script src="js/custom.js"></script>

```

Kod Örneği – 3: Web Sitesi JavaScript Kod Bloğu(Arda Keyişoğlu)

2 d. Web Sitesi Video Gösterim Paneli(Arda Keyişoğlu)

Aşağıdaki kod bloğu, örnek olarak eklenen ekstra bir bileşendir. Bu kısımda ilerleyen zamanlarda çekilecek olan proje videosu eklenecektir. Şu an için herhangi bir video eklenmemiştir.Sitenin her sayfasında bu tarz bileşenlerden faydalanılmıştır.

```

<!-- intro_video_bg-->
<section class="intro_video_bg">
  <div class="container">
    <div class="row">
      <div class="col-lg-12">
        <div class="intro_video_iner text-center">
          <h2>Sisteme Giriş Yap</h2>
          <div class="intro_video_icon">
            <a id="play-video_1" class="video-play-button popup-youtube"
              href="https://www.youtube.com/watch?v=pBFQdXA-apI">
              <span class="ti-control-play"></span>
            </a>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>

```

```
</div>
</section>
<!-- intro_video_bg-->
```

Kod Örneği – 4: Web Sitesi Video Gösterim Paneli(Arda Keyişoğlu)

2 e. Web Sitesi Vücut Kitle Endeksi Tasarımı(Arda Keyişoğlu)

Birinci rapordan sonra tamamlanmamış ekranlar tamamlanmış, yeni ekranlar ve elementler eklenmiştir.Kod örneklerinde bazıları verilmiştir.

Aşağıdaki kod bloğu, sitenin vücut kitle endeksi hesaplama bölümüne aittir.Bu bölümde kullanıcılar kendi boy ve kilolarını ekleyip cinsiyetleri ile hesaplanan oranlar ile karşılaşacaklar. Şu ana kadar herhangi bir sorun ile karşılaşılmamıştır.Bir sonraki raporda veri tabanına bağlanıp özel olarak kayıt edileceklerdir..

```
<section class="breadcrumb breadcrumb_bg" style="margin-top:100px">
<div class="container">
<div class="row">
<div class="col-lg-12">
<div class="breadcrumb_iner">
<div class="breadcrumb_iner_item">
<h2>Vücut Kitle Endeksi Hesapla</h2>
<p>Ana Sayfa<span>/</span>Vücut Kitle Endeksi Hesapla</p>
</div>
</div>
</div>
</div>
</div>
</section>
<br><br><br><br><br><br><br>
<section class="regervation_part">
<div class="container">
<div class="row align-items-center regervation_content">
<div class="col-lg-7 col-md-6">
<div class="regervation_part_iner">
<form>
<h2>Vücut Kitle Endeksi Hesapla</h2>
<div class="form-row">
<div class="form-group col-md-6">
<select class="form-control" id="Select">
<option value="1" selected>Kadın</option>
<option value="2">Erkek</option>
</select>
</div>
<div class="form-group col-md-6">
</div>
<div class="form-group col-md-6">
<input class="form-control" id="inputBoy" placeholder="Boyunuz">
</div>
<div class="form-group col-md-6">
<input class="form-control" id="inputKilo"
placeholder="Kilonuz">
</div>
</div>
<div class="regerv_btn">
<a href="#" class="regerv_btn_iner">Hesapla</a>
</div>
</form>
</div>
</div>
<div class="col-lg-5 col-md-6">
<div class="reservation_img">

</div>
</div>
</div>
</div>
```

```
</div>
</section>
```

Kod Örneği – 5: Web Sitesi Vücut Kitle Endeksi Tasarımı(Arda Keyişoğlu)

2 f. Web Sitesi İlaç Hatırlatma Tasarımı(Arda Keyişoğlu)

Aşağıdaki kod bloğu, sitenin ilaç hatırlatma bölümüne aittir. Bu bölümde kullanıcıların kullandıkları ilaçları kayıt edip, belirli aralıklar ile onlara bildirim gönderecek ekran tasarlanmıştır. Şu ana kadar herhangi bir sorun ile karşılaşılmamıştır. Bir sonraki raporda kodlar tamamlanıp tam fonksiyonel hale getirilecektir.

```
<section class="breadcrumb breadcrumb_bg" style="margin-top:100px">
  <div class="container">
    <div class="row">
      <div class="col-lg-12">
        <div class="breadcrumb_iner">
          <div class="breadcrumb_iner_item">
            <h2>İlaç Hatırlatıcı</h2>
            <p>Ana Sayfa<span></span>>İlaç Hatırlatıcı</p>
          </div>
        </div>
      </div>
    </div>
  </div>
</section>
<br><br><br><br><br><br><br>
<section class="regervation_part">
  <div class="container">
    <div class="row align-items-center regervation_content">
      <div class="col-lg-7 col-md-6">
        <div class="regervation_part_iner">
          <form>
            <h2>ilaç Hatırlatıcı</h2>
            <div class="form-row">
              <div class="form-group col-md-6">
                <input class="form-control" id="inputGun" placeholder="Kaç günde bir kullanılacak">
              </div>
              <div class="form-group col-md-6">
                <input class="form-control" id="inputSaat" placeholder="Kaç saatte bir kullanılacak">
              </div>
              <div class="form-group col-md-6">
                <select class="form-control" id="Select">
                  <option value="1" selected>İlaç 1</option>
                  <option value="2">İlaç 2</option>
                  <option value="3">İlaç 3</option>
                  <option value="4">İlaç 4</option>
                  <option value="5">İlaç 5</option>
                </select>
              </div>
              <div class="form-group time_icon col-md-6">
                <select class="form-control" id="Select2">
                  <option value="" selected>Başlangıç Saati</option>
                  <option value="1">8-10</option>
                  <option value="1">10-12</option>
                  <option value="1">12-2</option>
                  <option value="1">2-4</option>
                  <option value="1">4-6</option>
                  <option value="1">6-8</option>
                  <option value="1">8-10</option>
                  <option value="1">1-12</option>
                </select>
              </div>
              <div class="form-group col-md-12">
                <textarea class="form-control" id="Textarea" rows="4"
                  placeholder="Notunuz"></textarea>
              </div>
            </form>
          </div>
        </div>
      </div>
    </div>
  </div>
</section>
```

```

        </div>
        <div class="regerv_btn">
          <a href="#" class="regerv_btn_iner">İlaç Ekle</a>
        </div>
      </form>
    </div>
  </div>
  <div class="col-lg-5 col-md-6">
    <div class=" rezervasyon_jpg ">
      
    </div>
  </div>
</div>
</div>
</section>

```

Kod Örneği – 6: Web Sitesi İlaç Hatırlatma Tasarımı(Arda Keyişoğlu)

2 g. Web Sitesi GoogleMaps Entegrasyonu(Arda Keyişoğlu)

Aşağıdaki kod bloğu, sitenin bize ulaşın bölümünde yer alan GoogleMap’e aittir. Bu bölümde google’ın hazır bir paketi entegre edilmiştir. Şu ana kadar herhangi bir sorun ile karşılaşılmamıştır. Bir sonraki raporda kodlar tamamlanıp tam fonksiyonel hale getirilecektir.

```

var map = new google.maps.Map(document.getElementById('map'), {
  center: {
    lat: 41.015137,
    lng: 28.979530
  },
  zoom: 9,
  styles: grayStyles,
  scrollwheel: false
});

```

Kod Örneği – 7: Web Sitesi GoogleMaps Entegrasyonu(Arda Keyişoğlu)

2 h. Mobil Uygulama İlaç Hatırlatma Tasarımı(Fatih Göktuğ Yıldırım)

Alternatif bir geçici giriş ekranı tasarladım. Ardından İlaç hatırlatma ekranı tasarımı yapmaya başladım zamanımın en çok kısmını bu ekranda harcadım. İçerisinde hoş bir tasarım olması için bileşenleri hepsini ayrı ayrı dosyalarda Classlar oluşturarak tasarladım. Tüm ekranlarda aynı görünmesini sağlamak için stylelendirmelerini ekran boyutu ve bileşen oranlarını hesaplayarak belirttim. Bugün panelinde gelen verilerde bugün kullanmanız gereken ilaçları belirterek ileride sizi bildirim göndermeyi hedeflemektedir. Geçmiş kısmında o günün tarihinden önceki kayıtlı ilaçlarını ve kullanıp kullanmadığını size geri bildirim olarak ekranda göstermektedir. En büyük sorun ilaç listesinde şimdilik arama yapamamaktadır.

```

searchFilter = text =>
{
  const newData = this.state.allData.filter( item => {
    const listItem = `${item.ilac_ad.toLowerCase()}`;
    return listItem.indexOf(text.toLowerCase()) > -1;
  });
  this.setState({
    Data: newData,
  });
};

btnAktif = (text) =>

```

```

{
  this.setState({ ara:text});
}
render() {
  return (
    <SafeAreaView style={styles.aComp}>
      <View style={styles.baslik}>
        <Text style={styles.basTxt}>İlaç Hatırlatıcı</Text>
      </View>
      <View style={styles.header}>
        <TextInput style={styles.txtInp} placeholder="İlaç ismi =>" value={this.state.text} onChangeText={ text => { this.setState({ text:text});} } placeholderTextColor= 'rgb(200,200,200)'/>
        <TouchableOpacity style={styles.btnAra} onPress={ () => this.btnAktif(this.state.text)}>
          <Text style={styles.btnTxt}>Ara</Text>
        </TouchableOpacity>
      </View>
      <View style={styles.icerik}>
        <IlacRoute ara={this.state.ara}/>
      </View>
    </SafeAreaView>
  )
}

```

Kod Örneği – 1: Mobil Uygulama İlaç Hatırlatma Tasarımı(Fatih Göktaş Yıldırım)

2 1. Mobil Uygulama Navigator Bloğu(Fatih Göktaş Yıldırım)

İlk 4 satırda üst panelin route yapısı oluşturulmuş daha sonra iki ekrandan bugün olan ekranın örnek tasarımı verilmiştir..Telefona göre kullanılan günün tarihini alıp TR standartına göre biçimlendirilmiş ardından gelen veri formatına göre çevrilip karşılaştırma yapıldığından listenin güncellenmesi sağlanmıştır.

```

<Tab.Navigator >
  <Tab.Screen name="Bugün" component={BgnIlac} />
  <Tab.Screen name="Geçmiş" component={GcmIlac} />
</Tab.Navigator>
state =
{
  Data : "",
}
zaman = () =>
{
  var today = new Date();
  var date = "";
  if(today.getDate()<10) { date += '0' + today.getDate();}else{ date += today.getDate();}
  if((today.getMonth()+1)<10) { date += '0' + (today.getMonth()+1);}else{ date += (today.getMonth()+1);}
  date += '.' + today.getFullYear();
  return(date);
}

```

```

    }

    aktar = () =>
    {
        var bugunEleman = this.bugunData();
        this.setState({Data:bugunEleman});
    }

    bugunData = () =>
    {
        var g , d , e;
        var da = [];
        DataIlac.forEach(item => {
            g = item.tarih;
            d = g.split('.')
            e = this.kontrol(d);
            if(e)
            {
                da.push(item);
            }
        });
        return da;
    }

    kontrol = (d) =>
    {
        var z = this.zaman();
        var sp2 = z.split('.');
        if( (d[0] == sp2[0]) && (d[1] == sp2[1]) && (d[2] == sp2[2]) )
        {
            return true;
        }
        else
        {
            return false;
        }
    }

    componentDidMount()
    {
        this.aktar();
    }

    render() {
        return (
            <View>
            <FlatList style={ styles.flt}
                data={ this.state.Data}
                keyExtractor={item => item.ilac_id}
                renderItem={({item}) => <IlacComp ad={item.ilac_ad} tarih={item.tarih} kullanım={item.Kullanım} saat={item.saat} />}
            />
        )
    }

```



```

        <Button      buttonStyle={styles.plsBtn}      icon={<Icon      name="plus"      size={45}/>}      onPress={this.zaman}
        iconContainerStyle={{ width:50,backgroundColor:'white'}}/>

        </View>

    )

```

Kod Örneği – 2: Mobil Uygulama Navigator Bloğu(Fatih Göktuğ Yıldırım)

2.j. Mobil Uygulama Component Listeleme(Fatih Göktuğ Yıldırım)

Bu ekranda yüzlerce listelenicek verinin tek bir component şeklinde listelenmesi sağlandı.CheckBox ın aktif olup olmaması ve tıklanıldığında değişmesi sağlandı.

```

state = {
    checked:false,
}
changeCheck = () =>
{
    if(this.state.checked)
    {
        this.setState({checked:false});
    }else
    {
        this.setState({checked:true});
    }
}
aktar = () =>
{
    if(this.props.kullanım == 'true')
    {
        this.setState({checked:true});
    }else
    {
        this.setState({checked:false});
    }
}
componentDidMount()
{
    this.aktar();
}
render() {
    return (
        <View style={styles.aVie}>
            <Text style={styles.ilac}> {this.props.ad} </Text>
            <View>
                <Text style={styles.tarih}> {this.props.tarih} </Text>
                <Text style={styles.tarih}> {this.props.saat} </Text>
            </View>
        </View>
    )
}

```

```

</View>

<CheckBox
  center
  title='Kullanım'
  uncheckedIcon='times-circle'
  iconRight
  checkedIcon='check-square'
  checkedColor='green'
  uncheckedColor='red'
  containerStyle={ styles.cntChk }
  checked={ this.state.checked }
  onPress={this.changeCheck}
/>
</View>
)

```

Kod Örneği – 3: Mobil Uygulama Component Listeleme(Fatih Göktuğ Yıldırım)

2 k. Mobil Uygulama Kitle Endeksi Tasarımı(Fatih Göktuğ Yıldırım)

Kullanıcıların Kitle Endeksi hesabı yapabileceği basit bir ara yüz tasarladım. Endeks oranı global kabul edilen aralıklarla değerlendirilmiştir.

```

<View style={styles.aComp}>
  <View style={styles.baslik}>
    <Text style={styles.basTxt}>Kitle Endeksi</Text>
  </View>
  <View style={styles.icerik}>
    <TextInput
      mode="flat"
      style={styles.Inp}
      label="Boy (cm)"
      value={this.state.boy}
      onChangeText={text=>this.setState({ boy: text})}
      selectionColor="white"
      underlineColor={'white'}
      keyboardType="number-pad"
      theme={{
        colors: {
          placeholder: 'white',
          primary: 'white',
        },
      }}
    />
    <TextInput
      mode="flat"
      style={styles.Inp}
      label="Kilo (kg)"
      value={this.state.kilo}
      onChangeText={text=>this.setState({ kilo: text})}
      selectionColor="white"
      underlineColor={'white'}
      keyboardType="number-pad"
      theme={{
        colors: {
          placeholder: 'white',
          primary: 'white',
        },
      }}
    />
    <Button

```

```

        uppercase={ false}
        mode="outlined"
        loading={ this.state.loginLoading}
        onPress={ this.btnAktif}
        color={ 'black'}
        style={styles.btnStyle}
        labelStyle={ { fontSize:17,alignItems:'center',alignContent:'center'} }>

        Hesapla
    </Button>
</View>

    <Modal isVisible={this.state.modalVisible} style={styles.modalSty}>
        <View style={styles.modalVie}>
            <Text style = {[styles.yazi , {fontSize:20,fontWeight:'bold'}]}>Kitle Endeksi Reçetesi</Text>
            <Text style = {styles.yazi}>{ 'Endeks : ' + this.state.endeks}</Text>
            <Text style = {styles.yazi}>{ 'Durum : ' + this.state.durum}</Text>
            <Text style = {styles.yazi}>{ 'İdeal Kilonuz : ' + this.state.ideal}</Text>
            <Button style={styles.btnStyle} title="Saklan" onPress={ this.toggleModal}
            labelStyle={ { fontSize:17,alignItems:'center',alignContent:'center'} }>
                Kapat</Button>
        </View>
    </Modal>
</View>

```

Kod Örneği – 4: Mobil Uygulama Kitle Endeksi Tasarımı(Fatih Göktuğ Yıldırım)

2.1. Mobil Uygulama Kitle Endeksi Hesaplanması(Fatih Göktuğ Yıldırım)

Sabit componentlerimizi responsive şekilde tasarladım.Sonuç kısmını tasarım yüzünde ‘Modal’ olarak adlandırılan bir pop up ile sundum.Hesaplamalar Class içerisinde tanımladığım methodlar ile sonuçlandırdım.Hesaplama butonu aktif edildiğinde bir method tetiklenir.Method hesaplama işlemlerini yaptıktan sonra Modal’ı aktif ederek kullanıcıya sonucu aktarır.

```

state =
{
    boy : "",
    kilo : "",
    modalVisible:false,
    endeks : "",
    durum: "",
    ideal : "",
}
btnAktif = () =>
{
    if(this.state.boy !== "" && this.state.kilo !== "")
    {
        this.hesapla();
        this.toggleModal();
    }else
    {
        alert('Alanlar Boş Geçilemez...');
    }
}
toggleModal = () =>
{
    this.setState({ modalVisible:!this.state.modalVisible});
}
hesapla = () =>
{
    let oran = this.state.kilo / (this.state.boy * this.state.boy);
    oran = oran * 10000;
    oran = oran.toFixed(1);
    let durum = "", ideal = 0;
    if( oran <= 18.5)
    {
        durum = 'Düşük Kilolu';
    }
    else if(oran <= 24.9)
    {
        durum = 'İdeal Kilolu';
    }
}

```

```

else if(oran <= 29.9)
{
    durum = 'İdeal Kilolunun Üzeri';
}
else if(oran < 40)
{
    durum = 'Obez';
}
else if(oran >= 40)
{
    durum = 'Aşırı Obez';
}
ideal = 22 / (this.state.boy * this.state.boy);
ideal = ideal * 100000;
ideal = ideal.toFixed(1);
this.setState({ indeks : oran , durum , ideal});
}

```

Kod Örneği – 5: Mobil Uygulama Kitle Endeksi Hesaplanması(Fatih Göktuğ Yıldırım)

2 m. Mobx State Yönetimi(Fatih Göktuğ Yıldırım)

Stateler, React-native platformunda hayati önem taşıyan yapılardır. Bu yapılar sayfalarınızda geçici veriler tutmanızı sağlayan bir yapıdır. Fakat sayfalar arası taşınmaları için farklı şeylere ihtiyaç duyarlar. Tüm sayfalar arasında veri kullanımı gerçekleşebilir. Örneğin hangi kullanıcı için işlem yapılacaksa bunu her arayüzde sorgulayamayız. Bir kere sorguladıktan sonra onu her sayfada kullanmak gerekli bunu Mobx ile yapabiliriz. Mobx i entegre etmiş bulunmaktayım. İstedğimiz sayfalara iletişimini kurabilmek için 'observer' yapısını eklemeliyiz. Javascriptte getter ve setter yazmadan verilerimize ulaşabiliyoruz. Fakat Yazılım mimarilerine göre kullanıcılarımızı getter ve setter kullanarak yapmak daha uygundur. Classımız da İlaç Hatırlatma ve Ekleme ekranında kullanılacak verilerin renderlanmasını parçalı fonksiyonlar ile bir bütün şeklinde geçmişte kullandığımız ve bugün kullanılacak verilerin filtrelenmesi ve bunların bir dizide hazırlanması yapılmaktadır. Observable kullanmaz iseniz sayfalar arasındaki veri kullanımını sağlayamazsınız. Diğer bir özellik @action yapısıdır. Bu yapı methodun içerisindeki işlemler bitmeden observable kısımları güncellemez bu yüzden önemlidir. Eğer kullanmazsanız her parametre aktarımı yaptığınızda bütün method tekrar baştan çalışır ve işlem yükü birikir. Methodlarımızdan biri olan 'searchFilter' attribute olan aranacak değeri ile bir filtreleme yaparak aranan geçmişte ve günümüzü kapsayan listeyi günceller.

```

import { observable , action ,configure ,autorun ,runInAction} from 'mobx';
import DataIlac from '../data/DataIlac';
configure({enforceActions:"observed"});
class IlacStore{
    @observable aranan = "";
    @observable data = DataIlac;
    @observable allData = DataIlac;
    @observable allTodayData = DataIlac;
    @observable allPastData = "";
    @observable time = '00:00';
    constructor()
    {
        autorun( () => {
            this.aktar();
        });
    }
    @action getAra = () =>
    {
        return this.aranan;
    }
    @action getTime = () =>
    {
        return this.time;
    }
    @action setAra = (ara) =>
    {
        this.aranan = ara;
    }
    @action setTime = (e) =>
    {
        IlacStore.time = e;
    }
}

```

```

}
@action searchFilter = () =>
{
  this.data = this.allData;
  const newData = this.allData.filter( item => {
    const listItem = item.ilac_ad.toLowerCase();
    return listItem.indexOf(this.aranan.toLowerCase()) > -1;
  });
  this.data = "";
  this.data = newData;
  this.allData = DataIlac;
}
@action sifirla = () =>
{
  this.data = this.allTodayData;
}
##region Ortak Zaman
@action aktar = () =>
{
  var bugunEleman = this.bugunData();
  this.data = "";
  this.data = bugunEleman;
  this.allTodayData = "";
  this.allTodayData = bugunEleman;
  var gecmisEleman = this.GecmisData();
  this.allPastData = gecmisEleman;
}
zaman = () =>
{
  var today = new Date();
  var date = "";
  if(today.getDate()<10) { date += '0' + today.getDate();}else{ date += today.getDate();}
  if((today.getMonth()+1)<10) { date += '0' + (today.getMonth()+1);}else{ date += (today.getMonth()+1);}
  date += '-' + today.getFullYear();
  return(date);
}
##endregion
##region Bugün
bugunData = () =>
{
  var g , d , e;
  var da = [];
  this.data.forEach(item => {
    g = item.tarih;
    d = g.split('.')
    e = this.kontrol(d);
    if(e)
    {
      da.push(item);
    }
  });
  return da;
}

kontrol = (d) =>
{
  var z = this.zaman();
  var sp2 = z.split('.');
  if( (d[0] == sp2[0]) && (d[1] == sp2[1]) && (d[2] == sp2[2]) )
  {
    return true;
  }
  else
  {
    return false;
  }
}
##endregion

##region Geçmiş
GecmisData = () =>
{
  var g , d , e;
  var da = [];
  this.allData.forEach(item => {
    g = item.tarih;

```

```

        d = g.split('.')
        e = this.gecmisKontrol(d);
        if(e)
        {
            da.push(item);
        }

    });
    return da;
}

gecmisKontrol = (d) =>
{
    var z = this.zaman();
    var sp2 = z.split('.');
    if (parseInt(sp2[2]) > parseInt(d[2])) )
    {
        return true;
    }else
    {
        if ( parseInt(sp2[1]) > parseInt(d[1]))
        {
            return true;
        }else
        {
            if( parseInt(sp2[0]) > parseInt(d[0])) )
            {
                return true;
            }else
            {
                return false;
            }
        }
    }
}

}

}

//endregion
}

export default new IlacStore()

```

Kod Örneği – 6: Mobil Uygulama Mobx State Yönetimi(Fatih Göktuğ Yıldırım)

2 n. Mobil Uygulama Navigasyon Tasarımı (Mehmet Uğur Atmaca,Fatih Göktaş Yıldırım)

Modern bir tasarım olan yandan kaydırmalı menü kullandık sebebi bir çok farklı ekran olacağı için hepsini bir liste halinde menüye getirmeyi hedefledik. Güncellenen navigasyon kütüphanesi yüzünden navigasyon iskeleti yapısını tekrar oluşturup son sürüm gerekliliklerini yerine getirdik. Her ekranda aynı boyutlandırmaya sahiptir.

```
function Feed() {  
  return (  
    <View style={{ flex: 1, justifyContent: 'center', alignItems: 'center' }}>  
      <Text>Feed Screen</Text>  
    </View>  
  );  
}  
  
const Drawer = createDrawerNavigator();  
  
function MyDrawer() {  
  return (  
    <Drawer.Navigator initialRouteName="Feed">  
      <Drawer.Screen  
        name="Feed"
```

```

    component={Feed}
    options={{ drawerLabel: 'Home' }}
  />
  <Drawer.Screen
    name="Notifications"
    component={Account}
    options={{ drawerLabel: 'Updates' }}
  />
  <Drawer.Screen
    name="Profile"
    component={MyCarousel}
    options={{ drawerLabel: 'Profile ',atmaca:'atmaca' }}
  />
  <Drawer.Screen
    name="IlacEkle"
    component={Hatırlat}
    options={{ drawerLabel: 'İlaç Ekranı',goktug:'goktug' }}
  />
</Drawer.Navigator>
);
}
export default function App() {
  return (
    <NavigationContainer>
      <MyDrawer />
    </NavigationContainer>
  );
}

```

Kod Örneği – 1: Mobil Uygulama Navigasyon Tasarımı(Mehmet Uğur Atmaca,Fatih Göktuğ Yıldırım)

2 o. Mobil Uygulama Randevu Ekranı Tasarımı(Mehmet Uğur Atmaca)

Hastalar randevu alma işlemlerini gerçekleştirdiği ekrandır, Ekran üzerinde hastahane, bölüm, tarih seçimi yapan hastalar ikinci aşamaya ilerler ve doktor seçimlerini yaparlar ardından son adımda ise randevuları ilgili detayları kontrol eder ve hastahane ile ilgili bilgiler alırlar.

```

import React from "react";
import { StyleSheet, Text, View, SafeAreaView, ScrollView, TouchableOpacity } from "react-native";
import { FontAwesome, MaterialCommunityIcons } from "@expo/vector-icons";
import { ProgressSteps, ProgressStep } from 'react-native-progress-steps';
import Card from "../components/Card";
import DatePicker from 'react-native-datepicker';
import { SwipeListView } from '@nvthai/react-native-swipe-list-view';
import RNPicker from "rn-modal-picker";

```

```

export class MyCarousel extends React.Component {
  constructor(props)
  {
    super(props);
    this.state = {
      dataSourceHospital: [
        {
          id: 1,
          name: "Lütfi Kıdar Eğitim Araştırma Hastanesi"
        },
        {
          id: 2,
          name: "Lütfi Kıdar Eğitim Araştırma Hastanesi"
        },
      ],
      dataSourceCategory: [
        {
          id: 1,
          name: "Dahiliye"
        },
        {
          id: 2,
          name: "Beyin Cerrahi"
        },
      ]],
      category:null,
      hospital:null,
      date:null
    }
  }
  _selectedValueHospital(index, item) {
    this.setState({ hospital: item.name });
  }
  _selectedValueCategory(index, item) {
    this.setState({ category: item.name });
  }
  render()
  {
    return (
      <View style={styles.container}>
        <ProgressSteps previousBtnStyle={{ display:'none'}} nextBtnTextStyle={{ color:'red'}}>
          <ProgressStep label="İlk Adım" previousBtnDisabled={'disable'}>
            <View style={{ alignItems: 'center'}}>
              <Text style={{ fontSize:20}}>Tarih Seçimi Yap !</Text>
              <DatePicker
                style={{
                  width: '100%',
                  color: 'white',

```



```

        fontSize: 50,
        marginBottom: 50
    }}
    date={ this.state.date}
    mode="date"
    placeholder="Gidiş Tarihi"
    format="YYYY-MM-DD"
    minDate={ this.state.localTime}
    maxDate="2020-06-01"
    confirmBtnText="Confirm"
    cancelBtnText="Cancel"
    customStyles={{
        dateIcon: {
            position: 'absolute',
            left: 0,
            top: 4,
            marginLeft: 3,
        },
        dateInput: {
            borderColor: 'black',
            borderWidth: 1,
            borderRadius: 10,
            marginLeft: 45,
            color: 'black',
            margin: 10
        },
        placeholderText: {
            color: 'black',
        },
        dateText: {
            color: 'black',
            justifyContent: 'flex-start',
        },
    }}
    onChange={ date => {
        this.setState({ date: date });
    }}
    />

    <Text style={{ fontSize: 20 }}>Hastane Seçimi Yap !</Text>

    <RNPicker
        dataSource={ this.state.dataSourceHospital}
        dummyDataSource={ this.state.dataSourceHospital}
        defaultValue={ false}
        pickerTitle={ "Hastane" }
        showSearchBar={ true}
        disablePicker={ false}
    />

```

```

changeAnimation={"fade"}
searchBarPlaceholder={"Arama"}
showPickerTitle={true}
searchBarContainerStyle={styles.searchBarContainerStyle}
pickerStyle={styles.pickerStyle}
itemSeparatorStyle={styles.itemSeparatorStyle}
pickerItemTextStyle={styles.listTextViewStyle}
selectedLabel={this.state.hospital}
placeholderLabel={"Hastane Seçiniz !"}
selectLabelTextStyle={styles.selectLabelTextStyle}
placeholderTextStyle={styles.placeholderTextStyle}
dropDownImageStyle={styles.dropDownImageStyle}
dropDownImage={ { uri:'https://img.icons8.com/dotty/80/000000/category.png'} }
selectedValue={(index, item) => this._selectedValueHospital(index, item)}
/>
<Text style={{fontSize:20,marginTop:'10%'}}>Bölüm Seçimi Yap !</Text>
<RNPicker
  dataSource={this.state.dataSourceCategory}
  dummyDataSource={this.state.dataSourceCategory}
  defaultValue={false}
  pickerTitle={"Bölüm"}
  showSearchBar={true}
  disablePicker={false}
  changeAnimation={"fade"}
  searchBarPlaceholder={"Arama"}
  showPickerTitle={true}
  searchBarContainerStyle={styles.searchBarContainerStyle}
  pickerStyle={styles.pickerStyle}
  itemSeparatorStyle={styles.itemSeparatorStyle}
  pickerItemTextStyle={styles.listTextViewStyle}
  selectedLabel={this.state.category}
  placeholderLabel={"Bölüm Seçiniz !"}
  selectLabelTextStyle={styles.selectLabelTextStyle}
  placeholderTextStyle={styles.placeholderTextStyle}
  dropDownImageStyle={styles.dropDownImageStyle}
  dropDownImage={ { uri:'https://img.icons8.com/dotty/80/000000/category.png'} }
  selectedValue={(index, item) => this._selectedValueCategory(index, item)}
/>
</View>
<SwipeListView
  style={{marginTop:30}}
  useFlatList
  data={this.state.dataSourceHospital}
  renderItem={ (data) => (
    <View style={styles.rowFront}>
      <Text>I am {data.id} in a SwipeListView</Text>

```

```

</View>

)}

renderHiddenItem={ (data, rowMap) => (
  <View style={styles.rowBack}>
    <Text>Left</Text>
    <Text>Right</Text>
  </View>
)}

leftOpenValue={ 50}
rightOpenValue={ -50}
/>
</ProgressStep>
<ProgressStep label="Second Step">
<View style={{ alignItems: 'center' }}>
  <Text>Doktor Seçimi Yap Ve İlerle !</Text>
  <View style={styles.content}>
<ScrollView horizontal style={styles.scroll} contentContainerStyle={styles.innerScroll}>
  <View style={styles.spacing}>
    <Card />
  </View>
  <View style={styles.spacing}>
    <Card />
  </View>
  <View style={styles.spacing}>
    <Card />
  </View>
</ScrollView>
</View>
<View style={styles.footer}>
  <View style={styles.footerBalance}>
    <Text style={styles.myBalance}>Randevu Ücreti</Text>
    <View style={styles.balanceContainer}>
      <Text style={styles.libraValue}>~163.20</Text>
      <Text>
        <Text style={styles.currency}>USD </Text>
        <Text style={styles.currencyValue}>166.00</Text>
      </Text>
    </View>
  </View>
</View>
<View style={styles.buttonRow}>
  <TouchableOpacity style={styles.footerButton}>
    <MaterialCommunityIcons name="clock" size={16} color="#FFF" />
    <Text style={styles.footerButtonText}>Randevu</Text>
  </TouchableOpacity>
  <TouchableOpacity style={styles.footerButton}>
    <FontAwesome name="heartbeat" size={16} color="#FFF" />

```

```

        <Text style={ styles.footerButtonText }>Muayene</Text>

        </TouchableOpacity>
      </View>
    </View>
  </View>
</View>
</ProgressStep>
<ProgressStep label="Third Step">
  <View style={{ alignItems: 'center' }}>
    <Text>Randevu Detay</Text>
  </View>
</ProgressStep>
</ProgressSteps>
</View>

```

Kod Örneği – 2: Mobil Uygulama Randevu Ekranı Tasarımı(Mehmet Uğur Atmaca)

2 p. Mobil Uygulama Giriş Ekranı ve Kullanıcı Kayıt Ekranı Tasarımı (Mehmet Uğur Atmaca)

Kullanıcılar kişisel bilgilerine, randevuları, tahlil sonuçlarına ve daha bir çok ekrana ve bilgiye ulaşmak için bu ekranı kullanmakta ve giriş gerçekleştirmektedir.

```

import {
  Platform,
  StyleSheet,
  Animated,
  Image } from 'react-native';
import React from 'react';
import { Button, TextInput } from 'react-native-paper';

class Account extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      kullanıcıAdi: "",
      sifre: "",
      loginLoading: false,
      kayıtDisplay: 'none',
      loginDisplay: 'flex',
      errorPlaceholder: 'white',
      newUsername:"",
      newPassword:"",
      newMailAdress:"",
      newTelephoneNumber:"",
      newPersonalName:"",
      newPersonalSurname:"
    };
  }
  fetchData3 = () => {
    this.setState({ loginLoading: true });
    fetch(
      'https://www.matmaca.com/api/Users/GetUser?username=' +
      this.state.kullanıcıAdi +
      '&password=' +
      this.state.sifre +
      "
    )
    .then(response => response.json())
    .then(responseJson => {
      this.setState({
        loginLoading: false,
      });
    });
  }
}

```

```

console.log(this.state.kullaniciAdi+this.state.sifre)
if (responseJson.length > 0) {
  this.props.navigation.navigate('Reservation', {
    username: this.state.kullaniciAdi,
    password: this.state.sifre,
  });
  AsyncStorage.setItem('username',this.state.kullaniciAdi);

  } else {
    this.setState({ errorPlaceholder: 'red' });
  }
})
.catch(error => console.log(error));
};
validateEmail = () => {
var re = /^(?=.*[A-Za-z])(?=.*\d)[A-Za-z\d$&+,:;=?@#|'<>.^*()%-]{8,}$/;
if(!re.test(this.state.newPassword))
{
  alert('En Az Bir Sayı,Karakter İçermeli Ve Uzunluğu En Az 8 Karakter Olmalıdır');
  this.setState({newPassword:''})
}
};
fetchData4 = (text) => {
this.setState({ loginLoading: true });
if(this.state.newPersonalName != "" && this.state.newPersonalSurname != "")
{
  fetch(
    'https://www.matmaca.com/api/Users/PostUser?userName=' +
    this.state.newUsername +
    '&userPassword=' +
    this.state.newPassword +
    '&userMailAdress=' +
    this.state.newMailAdress +
    '&userPhoneNumber=' +
    this.state.newTelephoneNumber +
    '&userMailAdress=' +
    this.state.newMailAdress +
    '&userPersonalName=' +
    this.state.newPersonalName +
    '&userPersonalSurname=' +
    this.state.newPersonalSurname +
    ''
  )
  .then(response => response.json())
  .then(responseJson => {
    this.setState({
      kullaniciAdi: this.state.newUsername,
      sifre:this.state.newPassword,
      newMailAdress:"",
      newPassword:"",
      newPersonalName:"",
      newPersonalSurname:"",
      newTelephoneNumber:"",
      newUsername:"",
      kayitDisplay:'none',
      loginDisplay:'flex',
      loginLoading:false
    });
    this.props.navigation.navigate('Reservation', {
      username: this.state.kullaniciAdi,
      password: this.state.sifre
    });
    AsyncStorage.setItem('username',this.state.kullaniciAdi);
  })
  .catch(error => console.log(error));
}
else
{
  alert('Alanlar Boş Geçilemez !');
  this.setState({ loginLoading: false });
}
}
render() {
  return (
    <View style={{ flex: 1, backgroundColor: 'transparent' }}>
    <View style={{ width: '100%', height: '100%', position: 'absolute' }}>
    <Image

```

```

    style={{
      width: '100%',
      height: '100%',
    }}
    resizeMode={'scale'}
    source={{
      uri: 'https://www.jakpost.travel/wimages/large/0-1432_download-samsung-one-ui-background-wallpapers-samsung-one.jpg',
    }}
  />
  <Image
    style={{
      width: '90%',
      height: '50%',
      marginLeft: '5%',
      marginTop: '5%',
      position: 'absolute',
      display: this.state.loginDisplay
    }}
    resizeMode={'scale'}
    source={{
      uri: 'https://upload.wikimedia.org/wikipedia/commons/3/36/Medicine_Barnstar_Hires.png',
    }}
  />
</View>
<View
  style={{
    width: '100%',
    height: '60%',
    marginTop: '95%',
    display: this.state.loginDisplay,
  }}>
  <TextInput
    label="Kullanıcı Adı"
    mode="flat"
    style={{ backgroundColor: 'transparent' }}
    selectionColor="white"
    underlineColor={this.state.errorPlaceholder}
    value={this.state.kullaniciAdi}
    onChangeText={text =>
      this.setState({ kullaniciAdi: text, errorPlaceholder: 'white' })
    }
    theme={{
      colors: {
        text: this.state.errorPlaceholder,
        placeholder: 'white',
        primary: 'white',
      },
    }}
  />
  <TextInput
    mode="flat"
    style={{
      backgroundColor: 'transparent',
      marginTop: 10,
      color: 'white',
    }}
    label="Sifre"
    value={this.state.sifre}
    onChangeText={text =>
      this.setState({ sifre: text, errorPlaceholder: 'white' })
    }
    selectionColor="white"
    underlineColor={this.state.errorPlaceholder}
    secureTextEntry={true}
    theme={{
      colors: {
        text: this.state.errorPlaceholder,
        placeholder: 'white',
        primary: 'white',
      },
    }}
  />
  <Button
    icon={{
      uri:
        'https://cdn2.iconfinder.com/data/icons/ios-7-icons/50/user_male2-512.png',
    }}
  />

```

```

    }}
    mode="outlined"
    loading={this.state.loginLoading}
    onPress={() => this.fetchData3()}
    color={'black'}
    style={{
      marginTop: 50,
      backgroundColor: 'white',
      marginLeft: '5%',
      width: '90%',
      color: 'black',
    }}>
    Giriş Yap
  </Button>
  <Button
    icon={{ uri: 'http://cdn.onlinewebfonts.com/svg/img_202009.png' }}
    mode="outlined"
    onPress={() =>
      this.setState({ kayitDisplay: 'flex', loginDisplay: 'none' })
    }
    color={'black'}
    style={{
      marginTop: 10,
      backgroundColor: 'white',
      color: 'black',
      marginLeft: '5%',
      width: '90%',
    }}>
    Kayıt Ol
  </Button>
</View>
<View
  style={{
    width: '100%',
    height: '100%',
    marginBottom: '60%',
    marginTop: '3%',
    display: this.state.kayitDisplay,
    paddingLeft: 20,
    opacity: 10
  }}>
  <TextInput
    label="Username"
    mode="flat"
    style={{ backgroundColor: 'transparent', borderRadius: 15, opacity: 5, marginTop: 10 }}
    selectionColor="white"
    underlineColor="white"
    value={this.state.newUsername}
    onChangeText={text => this.setState({ newUsername: text })}
    theme={{
      colors: { text: 'white', placeholder: 'white', primary: 'white' },
    }}
  />
  <TextInput
    mode="flat"
    style={{ backgroundColor: 'transparent', borderRadius: 15, marginTop: 15, opacity: 5 }}
    label="Password"
    secureTextEntry={true}
    value={this.state.newPassword}
    onChangeText={text => this.setState({ newPassword: text })}
    onBlur={() => {
      this.validateEmail();
    }}
    selectionColor="white"
    underlineColor="white"
    theme={{
      colors: { text: 'white', placeholder: 'white', primary: 'white' },
    }}
  />
  <TextInput
    label="E-Mail"
    mode="flat"
    style={{ backgroundColor: 'transparent', borderRadius: 15, marginTop: 15, opacity: 5 }}
    selectionColor="white"
    underlineColor="white"
    value={this.state.newMailAdress}
  />

```

```

onChangeText={text => this.setState({ newMailAdress: text })}
theme={{
  colors: { text: 'white', placeholder: 'white', primary: 'white' },
}}
/>
<TextInput
  mode="flat"
  style={{ backgroundColor: 'transparent',borderRadius:15,marginTop:15,opacity:5 }}
  label="Phone Number"
  value={this.state.newTelephoneNumber}
  onChangeText={text => this.setState({ newTelephoneNumber: text })}
  selectionColor="white"
  underlineColor="white"
  theme={{
    colors: { text: 'white', placeholder: 'white', primary: 'white' },
  }}
/>
<TextInput
  mode="flat"
  style={{ backgroundColor: 'transparent',borderRadius:15,marginTop:15,opacity:5 }}
  label="Personal Name"
  value={this.state.newPersonalName}
  onChangeText={text => this.setState({ newPersonalName: text })}
  selectionColor="white"
  underlineColor="white"
  theme={{
    colors: { text: 'white', placeholder: 'white', primary: 'white' },
  }}
/>
<TextInput
  mode="flat"
  style={{ backgroundColor: 'transparent',borderRadius:15,marginTop:15,opacity:5 }}
  label="Personal Surname"
  value={this.state.newPersonalSurname}
  onChangeText={text => this.setState({ newPersonalSurname: text })}
  selectionColor="white"
  underlineColor="white"
  theme={{
    colors: { text: 'white', placeholder: 'white', primary: 'white' },
  }}
/>
<Button
  icon={{
    uri:
      'https://cdn2.iconfinder.com/data/icons/ios-7-icons/50/user_male2-512.png',
  }}
  loading={this.state.loginLoading}
  mode="outlined"
  onPress={() => this.fetchData4("")}
  color={'black'}
  style={{
    marginTop: 50,
    backgroundColor: 'white',
    color: 'black',
    marginRight:'5%'
  }}>
  Kayit Ol
</Button>
<Button
  color={'black'}
  icon={{ uri: 'http://cdn.onlinewebfonts.com/svg/img_202009.png' }}
  mode="outlined"
  onPress={() =>
    this.setState({ kayitDisplay: 'flex', loginDisplay: 'flex',loading:true })
  }
  style={{
    marginTop: 10,
    backgroundColor: 'white',
    color: 'black',
    marginRight:'5%'
  }}>
  Giris Yap
</Button>
</View>
</View>
);

```



```
}  
}  
export default Account;
```

Kod Örneği – 3: Mobil Uygulama Giriş Ekranı ve Kullanıcı Kayıt Ekranı Tasarımı (Mehmet Uğur Atmaca)

2 q. Mobil Uygulama Kullanıcı Profil Ekranı (Mehmet Uğur Atmaca)

Kullanıcılar ekran üzerinde kişisel bilgilerini düzenleyebilir, kullanıcı hesapları ile ilgili değişiklikler ve eklemeler yapabilmektedir.

```
import React, { Component } from 'react';  
import {  
  StyleSheet,  
  Text,  
  View,  
  Image,  
  TouchableOpacity  
} from 'react-native';  
class Profile extends Component {  
  render() {  
    return (  
      <View style={styles.container}>  
        <View style={styles.header}></View>  
        <Image style={styles.avatar} source={{uri:  
'https://pbs.twimg.com/profile_images/952545910990495744/b59hSXUd_400x400.jpg'}}/>  
        <View style={styles.body}>  
          <Text style={styles.name}>Mehmet Uğur Atmaca</Text>  
          <Text style={styles.info}>Dahiliye Uzmanı</Text>  
          <Text style={styles.description}>Lorem ipsum dolor sit amet, saepe sapientem eu nam. Qui ne assum electram expetendis,  
omittam deseruisse consequuntur ius an,</Text>  
        </View>  
      </View>  
    );  
  }  
}  
const styles = StyleSheet.create({  
  header: {  
    backgroundColor: 'grey',  
    height: 200,  
  },  
  avatar: {  
    width: 130,  
    height: 130,  
    borderRadius: 63,  
    borderWidth: 4,
```

```

borderColor: "white",
marginBottom:10,
alignSelf:'center',
position: 'absolute',
marginTop:130
},
name:{
fontSize:22,
color:"#FFFFFF",
fontWeight:'600',
},
body:{
marginTop:60,
alignItems:'center'
},
bodyContent: {
flex: 1,
alignItems: 'center',
padding:30,
},
name:{
fontSize:28,
color: "#696969",
fontWeight: "600"
},
info:{
fontSize:16,
color: "#00BFFF",
marginTop:10
},
description:{
fontSize:16,
color: "#696969",
marginTop:10,
textAlign: 'center'
},
buttonContainer: {
marginTop:10,
height:45,
flexDirection: 'row',
justifyContent: 'center',
alignItems: 'center',
marginBottom:20,
width:250,
borderRadius:30,
backgroundColor: "#00BFFF",
},
});
export default Profile;

```

Kod Örneği – 4: Mobil Uygulama Kullanıcı Profil Ekranı(Mehmet Uğur Atmaca)

2 r. Mobil Uygulama Sayfa Yönlendime Ekranı-Drawer Navigation(Mehmet Uğur Atmaca)

Kullanıcılar ekranlarının sol tarafında bulunan navigasyon yani yönlendirme ekranından uygulama içerisinde ki sayfalar arasında geçiş yapabilmektedir, aynı zamanda kullanıcılar giriş yaptıklarında navigasyon ekranında kişileştirilmiş olarak kendi fotoğraflarını, adlarını ve ünvanlarını görebilmektedir.

```
function Feed() {
  return (
    <View style={{ flex: 1, justifyContent: 'center', alignItems: 'center' }}>
      <Text>Feed Screen</Text>
    </View>
  );
}

const Drawer = createDrawerNavigator();

function MyDrawer() {
  return (
    <Drawer.Navigator initialRouteName="Feed" drawerType={ 'front' }
      drawerContentOptions={{ activeTintColor:'black',userName:'Deneme Doc',title:'Doctor' }}>
      <Drawer.Screen
        name="Home"
        component={Feed}
        options={{ drawerLabel: 'Home' }}
      />
      <Drawer.Screen
        name="Notifications"
        component={Account}
        options={{ drawerLabel: 'Updates' }}
      />
      <Drawer.Screen
        name="Profile"
        component={MyCarousel}
        options={{ drawerLabel: 'Profile',atmaca:'atmaca' }}
      />
      <Drawer.Screen
        name="İlacEkle"
        component={Hatırlat}
        options={{ drawerLabel: 'İlaç Ekranı' }}
      />
      <Drawer.Screen
        name="Profile 2"
        component={Profile}
        options={{ drawerLabel: 'Profile' }}
      />
    </Drawer.Navigator>
  );
}

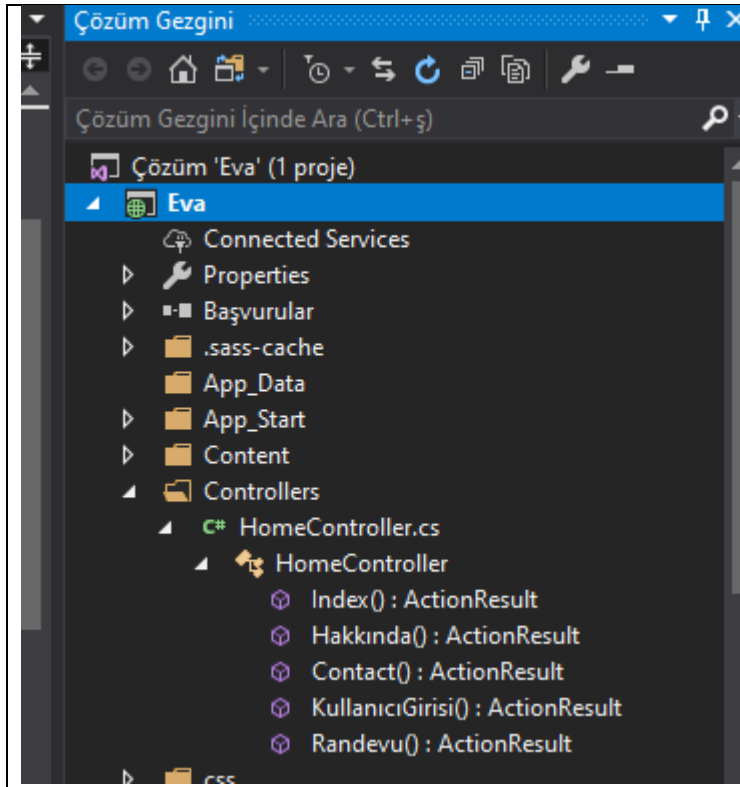
export default function App() {
  return (
    <NavigationContainer>
      <MyDrawer />
    </NavigationContainer>
  );
}
```

Kod Örneği – 5: Mobil Uygulama Sayfa Yönlendime Ekranı-Drawer Navigation(Mehmet Uğur Atmaca)

2 s. Web Sitesi MVC Structure Tasarımı[Controller] (Mert EFE)

Sistemin katmanlı bir MVC mimarisine sahip olabilmesi için Websitenin MVC [Controller] tabanlı tasarımı ve entegrasi gerçekleştirilmiştir. Sayfa Tab'lerine özel Route'lar ve View'ler oluşturulup kodların structure tabanlı değişimi gerçekleştirilmiştir. Günümüz güncel teknoloji versiyonları kullanılmıştır. Alt kısım'a anlaşılabilirlik açısından fotoğraf eklenmiştir.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
namespace EVA.Controllers
{
    public class HomeController : Controller
    {
        [Route:/Index/]
        public ActionResult Index()
        {
            return View();
        }
        [Route:/Hakkında/]
        public ActionResult Hakkında()
        {
            ViewBag.Message = "Hakkında-Page.";
            return View();
        }
        [Route:/Contact]
        public ActionResult Contact()
        {
            ViewBag.Message = "İletişim-Page.";
            return View();
        }
        [Route:/login]
        public ActionResult KullanıcıGirisi()
        {
            return View();
        }
        [Route:/Randevu]
        public ActionResult Randevu()
        {
            return View();
        }
    }
}
```



Kod Örneği – 1: Web Sitesi MVC Structure Tasarımı[Controller] (Mert EFE)

2 t. Web Sitesi MVC Structure Tasarımı [Views] (Mert EFE)

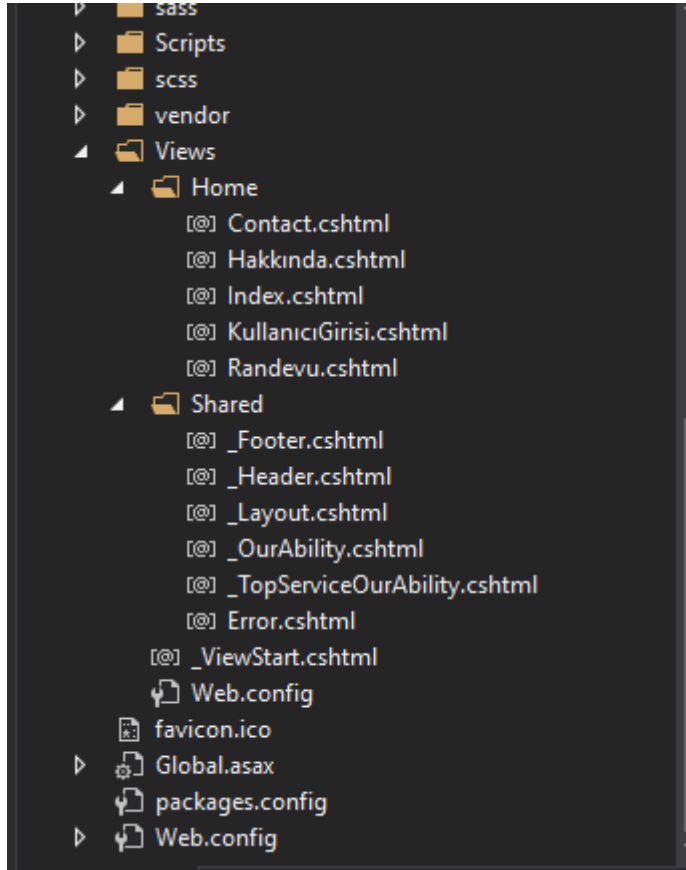
Sistemin katmanlı bir MVC mimarisine sahip olabilmesi için Websitenin MVC [Views] tabanlı tasarımı ve entegrasi gerçekleştirilmiştir. Sayfalar Partial View'lar olarak parçalanarak sistemin anlaşılabilirliği ve tekrar düzenlenebilirliği artırılmıştır. Günümüz güncel teknoloji versiyonları kullanılmıştır. Alt kısım'a anlaşılabilirlik açısından fotoğraf eklenmiştir.

```
<!--:header part start:-->
@Html.Partial("_Header")
<!-- Header part end-->
@RenderBody()
<!-- banner part start-->
@Html.Partial("_Banner")
<!-- banner part start-->
<!-- feature_part start-->
@Html.Partial("_Feature")
<!-- feature_part start-->
<!-- our_ability part start-->
@Html.Partial("_OurAbility")
<!-- our_ability part end-->
<!-- top_service part start-->
@Html.Partial("_Top-service")
<!-- top_service part end-->
<!--:review_part start:-->
```

```

@Html.Partial("_Review")
<!--:review_part end:-->
<!--:doctor_part start:-->
    @Html.Partial("_Doctor")
<!--:doctor_part end:-->
<!-- intro_video_bg start-->
    @Html.Partial("_Video")
<!-- intro_video_bg part start-->
<!-- footer part start-->
    @Html.Partial("_Footer")
<!-- footer part end-->
<!-- jquery plugins here-->

```



Kod Örneği – 2: Web Sitesi MVC Structure Tasarımı [Views] (Mert EFE)

2 u. Web Sitesi Model Katmanı(Mert Efe)

Aşağıdaki kod parçaları MVC'nin Model sayfalarına aittir. Kullanıcı Giriş ekranı için Model ekranı, Kayıt ol Ekranı için Model ekranı, Kayıt ol ekranında girdilerin onayları ve gereksinimleri için Model yazılması ve son olarak yazılan Modellerin kullanıcıya gösterimi.

Giriş Model

Kullanıcının giriş girdilerini tutan model ekranı. (Eposta, Parola ve Beni anımsa?)

```

public class LoginViewModel
{
    [Required]
    [Display(Name = "E-posta")]
    [EmailAddress]
    public string Email { get; set; }

    [Required]
    [DataType(DataType.Password)]
    [Display(Name = "Parola")]
    public string Password { get; set; }

    [Display(Name = "Beni anımsa?")]
    public bool RememberMe { get; set; }
}

```

Kod Örneği – 3: Web Sitesi Model Katmanı(Mert Efe)

Kayıt ol Model

Kullanıcı kayıt olurken girdilerin tutarlı ve istenilen şekilde olup olmadığını tespit edip ona göre hata mesajı gösteren model ekranı.

```

public class RegisterViewModel
{
    [Required]
    [EmailAddress]
    [Display(Name = "E-posta")]
    public string Email { get; set; }

    [Required]
    [StringLength(100, ErrorMessage = "{0} en az {2} karakter uzunluğunda olmalıdır.", MinimumLength = 6)]
    [DataType(DataType.Password)]
    [Display(Name = "Parola")]
    public string Password { get; set; }

    [DataType(DataType.Password)]
    [Display(Name = "Parolayı onaylayın")]
    [Compare("Password", ErrorMessage = "Parola ve onay parolası aynı değil.")]
    public string ConfirmPassword { get; set; }
}

```

Kod Örneği – 4: Web Sitesi Model Katmanı(Mert Efe)

Şifremi Unuttum Model

Şifresinin unutan bir kullanıcının e-posta adresi ile tekrar şifre alma isteği.

```

public class ForgotPasswordViewModel
{
    [Required]
    [EmailAddress]
    [Display(Name = "E-posta")]
    public string Email { get; set; }
}

```

Kod Örneği – 5: Web Sitesi Model Katmanı(Mert Efe)

Şifre değiştir Model

Şifrenin değiştirilmesi için var olan model ekranı.

```
public class ResetPasswordViewModel
{
    [Required]
    [EmailAddress]
    [Display(Name = "E-posta")]
    public string Email { get; set; }

    [Required]
    [StringLength(100, ErrorMessage = "{0} en az {2} karakter uzunluğunda olmalıdır.", MinimumLength = 6)]
    [DataType(DataType.Password)]
    [Display(Name = "Parola")]
    public string Password { get; set; }

    [DataType(DataType.Password)]
    [Display(Name = "Parolayı onayla")]
    [Compare("Password", ErrorMessage = "Parola ve onay parolası eşleşmiyor.")]
    public string ConfirmPassword { get; set; }

    public string Code { get; set; }
}
```

Kod Örneği – 6: Web Sitesi Model Katmanı(Mert Efe)

2 v. Web Sitesi Kullanıcı Ekranı Tasarımı(Mert Efe)

Aşağıdaki kod bloğu, sitenin Kullanıcı ekranını oluşturur. Kullanıcının kendi resmini ekleyebileceği ve bazı kişisel bilgilerini barındıran, ayrıca son zamanda yaptığı değişiklikleri ve işlemlerden haberdar olmasını sağlayan, Randevu zamanları gibi önemli tarihleri hatırlatan ve Kullanıcının kayıt olurken girmiş olduğu verileri değiştirip, düzenleme yapabileceği ekranlara sahip bir arayüzdür.

```
<div class="tab-pane" id="messages">
<div class="alert alert-info alert-dismissible">
    <a class="panel-close close" data-dismiss="alert">×</a> Önemli <strong> mesajlar </strong> burada kullanıcıya
    gösterilir.
</div>
<table class="table table-hover table-striped">
    <tbody>
        <tr>
            <td>
                <span class="float-right font-weight-bold">3 saat önce</span> Kullanıcının son raporu...
            </td>
        </tr>
        <tr>
            <td>
                <span class="float-right font-weight-bold">Düneyin</span> Son hesap işlemleri..
            </td>
        </tr>
    </tbody>
</table>
</div>
<div class="tab-pane" id="edit">
    <form role="form">
        <div class="form-group row">
            <label class="col-lg-3 col-form-label form-control-label">Ad</label>
            <div class="col-lg-9">
                <input class="form-control" type="text" value="Adınız" placeholder="Adınız">
            </div>
        </div>
        <div class="form-group row">
            <label class="col-lg-3 col-form-label form-control-label">SoyAd</label>
            <div class="col-lg-9">
                <input class="form-control" type="text" value="" placeholder="Soyadınız">
            </div>
        </div>
    </form>
</div>
```



```

<div class="form-group row">
  <label class="col-lg-3 col-form-label form-control-label">Email</label>
  <div class="col-lg-9">
    <input class="form-control" type="email" value="" placeholder="email@gmail.com">
  </div>
</div>
<div class="form-group row">
  <label class="col-lg-3 col-form-label form-control-label">Adres</label>
  <div class="col-lg-9">
    <input class="form-control" type="text" value="" placeholder="Şehir">
  </div>
</div>
<div class="form-group row">
  <label class="col-lg-3 col-form-label form-control-label"></label>
  <div class="col-lg-6">
    <input class="form-control" type="text" value="" placeholder="İlçe">
  </div>
  <div class="col-lg-3">
    <input class="form-control" type="text" value="" placeholder="Cadde">
  </div>
</div>
</div>

```

Kod Örneği – 7: Web Sitesi Kullanıcı Ekranı Tasarımı(Mert Efe)

2 w. Web Sitesi Randevu Ekranı Tasarımı(Mert Efe)

Aşağıdaki kod bloğu, sitenin Randevu Al kısmında yer alan ve kullanıcıların randevu alması için gerekli bilgileri girdiği ekrana aittir. Bir sonraki raporda veri tabanına bağlanıp olarak kayıt edilecektir

```

<section class="regervation_part" style="margin-bottom:30px">
  <div class="container">
    <div class="row align-items-center regervation_content">
      <div class="col-lg-7 col-md-6">
        <div class="regervation_part_iner">
          <form>
            <h2>Randevu Al</h2>
            <div class="form-row">
              <div class="form-group col-md-6">
                <input type="email" class="form-control" id="inputEmail4" placeholder="Ad">
              </div>
              <div class="form-group col-md-6">
                <input type="password" class="form-control" id="inputPassword4" placeholder="Email">
              </div>
              <div class="form-group col-md-6">
                <select class="form-control" id="Select">
                  <option value="1" selected>Servis 1</option>
                  <option value="2">Servis 2</option>
                  <option value="3">Servis 3</option>
                  <option value="4">Servis 4</option>
                  <option value="5">Servis 5</option>
                </select>
              </div>
              <div class="form-group time_icon col-md-6">
                <select class="form-control" id="Select2">
                  <option value="" selected>Saat</option>
                </select>
              </div>
              <div class="form-group col-md-12">
                <textarea class="form-control" id="Textarea" rows="4" placeholder="Notunuz"></textarea>
              </div>
            </div>
            <div class="regerv_btn">
              <a href="#" class="regerv_btn_iner">Randevu Al</a>
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>

```

```

<div class="col-lg-5 col-md-6">
  <div class="reservation_img">
    
  </div>
</div>
</div>
</div>
</section>

```

Kod Örneği – 8: Web Sitesi Randevu Ekranı Tasarımı(Mert Efe)

2 x. Web sitesi Login[Sign Up] Tasarımı(Semih EFE)

Kullanıcıların kayıt olabilmeleri için Bootstrap ve JQuery teknolojileri ile modern bir tasarıma sahip Login [Sign-up] ekranı tasarlanmıştır.Mvc Structure entegrasi yapılmıştır.Mobil kullanıcılar için yapılan tasarım kodları tekrar değiştirilmek zorunda kalmış, font değiştirilmiş ve her platform için uyumlu hale getirilmiştir. Kullanılan teknolojilerin son versiyonları projeye entegre edilmiştir.

```

<!-- Sign up form -->
<section class="signup">
  <div class="container">
    <div class="signup-content">
      <div class="signup-form">
        <h2 class="form-title">Kayıt Ol</h2>
        <form method="POST" class="register-form" id="register-form">
          <div class="form-group">
            <label for="name"><i class="zmdzmdi-accountmaterial-icons"></i></label>
            <input type="text" name="name" id="name" placeholder="Ad" />
          </div>
          <div class="form-group">
            <label for="email"><i class="zmdi zmdi-email"></i></label>
            <input type="email" name="email" id="email" placeholder="Email" />
          </div>
          <div class="form-group">
            <label for="pass"><i class="zmdi zmdi-lock"></i></label>
            <input type="password" name="pass" id="pass" placeholder="Şifre" />
          </div>
          <div class="form-group">
            <label for="re-pass"><i class="zmdi zmdi-lock-outline"></i></label>
            <input type="password" name="re_pass" id="re_pass" placeholder="Şifrenizi tekrar girin" />
          </div>
        </form>
      </div>
    </div>
  </div>
</section>

```

```

</div>

<div class="form-group">

  <input type="checkbox" name="agree-term" id="agree-term" class="agree-term" />
  <label for="agree-term" class="label-agree-term h6"><span><span></span></span>Kullanıcı koşullarını kabul ediyorum.</label>

</div>

<div class="form-group form-button">

  <input type="submit" name="signup" id="signup" class="form-submit h6" value="Kayıt ol" />

</div>

</form>

</div>

<div class="signup-image">

  <figure></figure>

  <a href="#Already" class="signup-image-link h5"> <i class="zmdi zmdi-account material-icons-name"></i> - Zaten
  hesabım var!</a>

</div>

</div>

</div>

</section>

```

Kod Örneği – 1: Web sitesi Login[Sign Up] Tasarımı(Semih EFE)

2 y. Web site Login [Sign in] Tasarımı(Semih EFE)

Kullanıcıların giriş yapabilmesi için Bootstrap ve JQuery teknolojileri ile modern bir tasarıma sahip Login [Sign-in] ekranı tasarlanmıştırMvc Strucute entegresi yapılmıştır.Mobil kullanıcılar için yapılan tasarım kodları tekrar değiştirilmek zorunda kalınmış, font değiştirilmiş her platform için uyumlu hale getirilmiştir. Kullanılan teknolojilerin son versiyonları projeye entegre edilmiştir.

```

<!-- Sing in Form -->

<section class="sign-in" id="Already">

  <div class="container">

    <div class="signin-content">

      <div class="signin-image">

        <figure></figure>

        <a href="#" class="signup-image-link h5"> <i class="zmdi zmdi-account material-icons-name"></i> - Hesap oluştur</a>

      </div>

      <div class="signin-form">

        <h2 class="form-title">Giriş Yap</h2>

        <form method="POST" class="register-form" id="login-form">

          <div class="form-group">

            <label for="your_name"><i class="zmdi zmdi-account material-icons-name"></i></label>

            <input type="text" name="your_name" id="your_name" placeholder="Ad" />

          </div>

          <div class="form-group">

            <label for="your_pass"><i class="zmdi zmdi-lock"></i></label>

            <input type="password" name="your_pass" id="your_pass" placeholder="Şifre" />

          </div>

        </form>

      </div>

    </div>

  </div>

</section>

```

```

</div>

<div class="form-group">

  <input type="checkbox" name="remember-me" id="remember-me" class="agree-term" />
<label for="remember-me" class="label-agree-term h6"><span><span></span></span>Beni hatırla</label>

</div>

<div class="form-group form-button h6">

<input type="submit" name="signin" id="signin" class="form-submit" value="Giriş yap" />

</div>

</form>

<div class="social-login">

  <span class="social-label h6">Var olan hesabınızla girin</span>

  <ul class="socials">

<li><a href="#"><i class="display-flex-center zmdi zmdi-facebook"></i></a></li>

<li><a href="#"><i class="display-flex-center zmdi zmdi-twitter"></i></a></li>

<li><a href="#"><i class="display-flex-center zmdi zmdi-google"></i></a></li>

  </ul>

</div>

</div>

</div>

</div>

</section>

</div>

```

Kod Örneği – 2: Web site Login [Sign in] Tasarımı(Semih EFE)

2 z. Web Sitesi Haberler Ekran Tasarımı(Semih Efe)

Aşağıdaki kod bloğu, sitenin “Haber” sekmesine aittir. Buradaki amaç kullanıcılarımızın keyifli zaman geçirebilmeleri ve yeni bilgiler öğrenebilmesidir. Güncel ve genel olarak sağlık haberleri yer alacaktır. Gelecek raporlarda geleceğine tam olarak karar verilecektir.

```

      <aside class="single_sidebar_widget search_widget">
<form action="#">
  <div class="form-group">
    <div class="input-group mb-3">
      <input type="text" class="form-control" value="Arama"
        onfocus="this.placeholder = ''"
        onblur="this.placeholder = 'Arama'">
      <div class="input-group-append">
<button class="btn" type="button"><i class="ti-search"></i></button>
      </div>
    </div>
  </div>
<button class="button rounded-0 primary-bg text-white 100 btn_1"
  type="submit">
  BUL
</button>
</form>
</aside>
<aside class="single_sidebar_widget post_category_widget">
<h4 class="widget_title">Kategoriler</h4>
<ul class="list cat-list">
  <li>
    <a href="#" class="d-flex">

```

```

        <p>Sağlık</p>
      </a>
    </li>
    <li>
      <a href="#" class="d-flex">
        <p>Spor</p>
      </a>
    </li>
    <li>
      <a href="#" class="d-flex">
        <p>Modern Teknoloji</p>
      </a>
    </li>
    <li>
      <a href="#" class="d-flex">
        <p>Kalp Sağlığı</p>
      </a>
    </li>
    <li>
      <a href="#" class="d-flex">
        <p>İlham</p>
      </a>
    </li>
  </ul>
</aside>

```

Kod Örneği – 3: Web Sitesi Haberler Ekran Tasarımı(Semih Efe)

2 aa. Web Sitesi Blog Ekran Tasarımı(Semih Efe)

Aşağıdaki kod bloğu, sitenin “Blog” sekmesine aittir. Buradaki amaç kullanıma sitemizde yapılan ve yapılacak olan güncellemeler hakkında bilgi verip onların geri dönüşünü ve yorumlarını almaktır. Bu sayede aldığımız geri dönüşler bizleri geliştirecek ve sistemimiz daha kullanıcı odaklı bir hale gelecektir.

```

<aside class="single_sidebar_widget search_widget">
  <div class="form-group">
    <div class="input-group mb-3">
      <input type="text" class="form-control" placeholder='Arama'
        onfocus="this.placeholder = ''"
        onblur="this.placeholder = 'Arama'">
      <div class="input-group-append">
        <button class="btn" type="button"><i class="ti-search"></i></button>
      </div>
    </div>
  </div>
  <button class="button rounded-0 primary-bg text-white w-100 btn_1"
    type="submit">
    BUL
  </button>
</aside>
<aside class="single_sidebar_widget instagram_feeds">
  <h4 class="widget_title">Instagram</h4>
  <ul class="instagram_row flex-wrap">
    <li>
      <a href="#">
        
      </a>
    </li>
    <li>
      <a href="#">
        
      </a>
    </li>
    <li>
      <a href="#">
        
      </a>
    </li>
  </ul>

```

```

        </li>
</aside>
<aside class="single_sidebar_widget newsletter_widget">
    <h4 class="widget_title">Son haberler</h4>

    <form action="#">
        <div class="form-group">
            <input type="email" class="form-control" onfocus="this.placeholder = ''"
            onblur="this.placeholder = 'Email'" placeholder='Email' required>
        </div>
        <button class="button rounded-0 primary-bg text-white w-100 btn_1"
            type="submit">
            Subscribe
        </button>
    </form>
</aside>

```

Kod Örneği – 4: Web Sitesi Blog Ekran Tasarımı(Semih Efe)

2 ab. Web Sitesi Kullanıcı Giriş Tasarımı (Semih Efe)

Aşağıdaki kod bloğu, sitenin “Giriş” sekmesine aittir. Bu kısımda giriş bölümümüzün güncellenmiş versiyonu yer almaktadır. Kullanıcı bu kısımdan giriş yapabilir, kayıtlı değilse kayıt ol bölümüne yönlendirilir.

```

<h2 class="form-title">Giriş Yap!</h2>
<hr />

@using (Html.BeginForm("Login", "Account", new { returnUrl = ViewBag.ReturnUrl }, FormMethod.Post, new { @class = "form-horizontal", role = "form" }))
{
    @Html.AntiForgeryToken()
    <h4>Yerel bir hesapla oturum açın.</h4>
    <hr />
    @Html.ValidationSummary(true, "", new { @class = "text-danger" })
    <div class="form-group">
        <div class="col-md-10">
            <label for="email"><i class="zmdi zmdi-email"></i></label>
            @Html.EditorFor(m => m.Email, new { htmlAttributes = new { @class = "zmdi zmdi-email", placeholder = "Email" } })
            @Html.ValidationMessageFor(m => m.Email, "", new { @class = "text-danger" })
        </div>
    </div>
    <div class="form-group">
        <div class="col-md-10">
            <label for="pass"><i class="zmdi zmdi-lock"></i></label>
            @Html.EditorFor(m => m.Password, new { htmlAttributes = new { @class = "zmdi zmdi-lock", placeholder = "Şifreniz", type = "password" } })
            @Html.ValidationMessageFor(m => m.Password, "", new { @class = "text-danger" })
        </div>
    </div>
    <div class="form-group">
        <div class="col-md-offset-2 col-md-10">
            <div class="checkbox">
                @Html.CheckBoxFor(m => m.RememberMe)
                @Html.LabelFor(m => m.RememberMe)
            </div>
        </div>
    </div>
    <div class="form-group">
        <div class="col-md-offset-2 col-md-10">
            <input type="submit" value="Oturum aç" class="btn btn-primary" />
        </div>
    </div>
    <p>
        @Html.ActionLink("Yeni kullanıcı olarak kaydol", "Register")
    </p>
    <p>
        @Html.ActionLink("Parolanızı mı unuttunuz?", "ForgotPassword")
    </p>

```

Kod Örneği – 5: Web Sitesi Kullanıcı Giriş Tasarımı (Semih Efe)

2 ac. Web Sitesi Kullanıcı Kayıt Ol Tasarımı (Semih Efe)

Aşağıdaki kod bloğu, sitenin “Kayıt ol” sekmesine aittir. Bu kısımda kayıt ol bölümümüzün güncellenmiş versiyonu yer almaktadır. Kullanıcı bu kısımdan kayıt olabilir yapabilir, kayıtlı ise giriş bölümüne yönlendirilir.

```
<h2 class="form-title">Kayıt Ol</h2>
    @Html.AntiForgeryToken()
    <h4>Yeni hesap oluşturun.</h4>
    <hr />
    @Html.ValidationSummary("", new { @class = "text-danger" })
    <div class="form-group">
    <label for="name"><i class="zmdi zmdi-account material-icons-name"></i></label>
    <input type="text" name="name" id="name" placeholder="Ad" />
    </div>
    <div class="form-group">
    <label for="name"><i class="zmdi zmdi-account material-icons-name"></i></label>
    <input type="text" name="name" id="name" placeholder="SoyAd" />
    </div>
    <div class="form-group">
    <label for="email"><i class="zmdi zmdi-email"></i></label>
    @Html.EditorFor(m => m.Email, new { htmlAttributes = new { @class = "zmdi zmdi-email", placeholder = "Email" } })
    </div>
    <div class="form-group">
    <label for="pass"><i class="zmdi zmdi-lock"></i></label>
    @Html.EditorFor(m => m.Password, new { htmlAttributes = new { @class = "zmdi zmdi-lock", placeholder =
    "Şifreniz", type = "password" } })
    </div>
    <div class="form-group">
    <label for="re-pass"><i class="zmdi zmdi-lock-outline"></i></label>
    @Html.EditorFor(m => m.ConfirmPassword, new { htmlAttributes = new { @class = "zmdi zmdi-lock", placeholder =
    "Şifrenizi tekrar girin", type = "password" } })
    </div>
    <div class="form-group">
    <input type="checkbox" name="agree-term" id="agree-term" class="agree-term" />
    <label for="agree-term" class="label-agree-term h6"><span><span></span></span></label>
    ediyorum.</div>
    <div class="form-group form-button">
    <input type="submit" name="signup" id="signup" class="form-submit h6" value="Kaydol" />
    </div>
    <div class="signup-image">
    <figure></figure>
    @Html.ActionLink("Zaten hesabım var!", "Login", "Account", htmlAttributes: new { @class = "form - submit h6",
    style = "margin-left:130px; font-size:20px;" })
    </div>
```

Kod Örneği – 6: Web Sitesi Kullanıcı Kayıt Ol Tasarımı (Semih Efe)

3 Sonuç

3 a. Başarı ile tamamlanan işler

- Web sitesi uygun başlıkları, sayfa içeriklerini, bileşenlerini araştırmış ve seçmiştir.
- Web sitesine uygun görselleri tasarlayıp sayfalara eklemiştir.
- Ana sayfa-Hakkımızda-Bize ulaşın ekran tasarımı ve gereken bileşen kodlamasını yerine getirmiştir.
- Website için CSS , JQuery ve Bootstrap kodlarına özel eklemeler yapılmış eklenen sayfaların mobil uyumlu olmaları sağlanmıştır.
- Website için Kullanıcı Kayıt [Sign up] ve Kullanıcı Giriş [Sign in] bölümleri modern bir tasarımla eklenmiş ve entegre edilmiştir.

- React-Native;
Giriş ,Alternatif Giriş ,İlaç Hatırlatma,İlaç Ekleme,Kayıt,Randevu Alma tasarımı ekranları başarılı bir şekilde tasarlanmıştır.Ekran Navigasyonu iskeleti oluşturulmuş ve ekranlar eklendikçe tasarımı geliştirilecektir.Şuan belirtilen ekranlar arasında geçiş sağlanmakta olup geçici veriler ile işlemler gerçekleştirilmektedir.
- MVC;Structure sistemi website tasarımına entegre edilmiştir.Her bir parça ayrı ayrı Partial View'lara ayrılmış Controller ile ilişkilendirilmiştir.Sayfalar arası actionlink bağlantıları yapılmış sorunsuz çalışması sağlanmıştır. Login page bölümleri MVC structure'a entegre edilmiştir.
- Web sitesi uygun başlıklar, sayfa içerikleri, bileşenler araştırılmış ve seçilmiştir.
- Web sitesine uygun görseller tasarlanıp sayfalara eklemiştir.
- Ana sayfa-Hakkımızda-Bize ulaşın-Gündem-Doktorlar-Ek Hizmetler-Profil ekran tasarımı ve gereken bileşen kodlamasını yerine getirilmiştir.
- Website için CSS , JQuery ve Bootstrap kodlarına özel eklemeler yapılmış eklenen sayfaların mobil uyumlu olmaları sağlanmıştır.
- Website için Kullanıcı Kayıt [Sign up] ve Kullanıcı Giriş [Sign in] bölümleri modern bir tasarımla eklenmiş ve entegre edilmiştir.
- Model için requirement gereksinimleri tamamlanmıştır.
- Veritabanına alınan şifre gibi gizli veriler encode şekilde tutularak kullanıcının güvenliği sağlanmıştır.
- Websitesi için haber ve blog sayfaları kullanıcılarımızın keyifli zaman geçirebilmeleri ve yeni bilgiler öğrenebilmesi için tasarlanmıştır.
- React-native Mobx State yönetimi kurulmuş ve çalışmıştır.
- React-native İlaç ekleme ve Kitle endeksi Sayfaları eklenmiştir.
- Önceki raporda ilaç arama hatası düzeltilmiştir.
- Navigasyon yapısı güncellendi.
- Ortak bağlantılı bir veritabanı yaratıldı.Tablolar üstünde yaptığımız çalışmalar create edilecektir

3 b. Başarı ile tamamlanamayan işler

- Randevu sistemi, İlaç hatırlatıcı ve Tahlil sorgulama ekran tasarımları değişeceğinden sadece alternatif tasarımlar oluşturulmuştur, iş daha bitmemiştir
- Logo tam olarak tasarlanmamıştır, değişecektir
- Login sayfasına admin[doktor] özelleştirmesi entegre edilmeli.
- MVC tabanlı veritabanı sistemi entegre edilmeli.
- React-Native;İlaç Hatırlatma Ekranı = İlaç Arama girdi değişkeni hala aktif değildir.Dosyalar arası veri taşıma için güncel teknolojiler entegre edilmeli.
- Websitesi sayfalarının tamamı veritabanına bağlı değildir.
- Yapılan çalışmalardaki tablolar veritabanımıza aktarılmadı.
- Api hazırlanmadı.

3 c. Kişilerin görevlerini yerine getirme durumu

	Numara	Ad Soyad	Projedeki Görevini Yerine Getirme Durumu
1	160313008	Arda Keyişoğlu	<ul style="list-style-type: none"> • Web sitesi uygun başlıkları, sayfa içeriklerini, bileşenlerini araştırılmış ve seçilmiştir. • Web sitesine uygun görselleri tasarlanıp sayfalara eklemiştir. • Ana sayfa, Hakkımızda, Bize ulaşın, Doktorlar, Randevu sistemi, İlaç hatırlatıcı, Tahlil

			<p>sorgulama, Vücut Kitle Endeksi ekranları gerçekleştirildi.</p> <ul style="list-style-type: none"> • Ana sayfa, Hakkımızda, Bize ulaşın, Doktorlar, Randevu sistemi, İlaç hatırlatıcı ve Tahlil sorgulama, Vücut Kitle Endeksi ekranları için gereken MVC bağlantıları gerçekleştirildi. • Logo tasarımı yerine getirmiştir. • Google Maps entegrasyonu yerine getirmiştir.
2	160313049	Fatih Göktuğ Yıldırım	<ul style="list-style-type: none"> • Alternatif giriş, ilaç ekleme ekranı görevini %100 yerine getirmiştir. • İlaç hatırlatma ekranını %95 yerine getirmiştir. • Navigasyon iskeletininin oluşturulmasına yardım etmiş ve görevini %100 yerine getirmiştir. • React-Native ilaç ekleme ekran tasarımı ve kodlamasını yapmıştır..(%100) • React-Native Ekran navigasyon mimarisi güncellemesini yapmıştır(%100). • React-Native Mobx uygunluğu araştırma ve entegre edilmesini yapmıştır(%100).
3	160313043	Mehmet Uğur Atmaca	<ul style="list-style-type: none"> • Hastalar randevu alma işlemleri için mobil uygulama üzerinde kullanılacak ekran güncellendi. • Giriş ekranı tasarımı yapıldı. • Ekran yönlendirmesi ekranında kullanıcıya göre kişiselleştirilmiş alanlar eklenmiş ve düzenlenmiştir. • Anasayfa tasarımı oluşturuluyor. • Doktorlar listelendi, randevu detayları eklendi.
4	160313005	Mert Efe	<ul style="list-style-type: none"> • Mvc structure kapsamında Controller oluşturma görevini yerine getirmiştir. • Mvc structure kapsamında Partial-View ve ActionLink oluşturma görevini %100 oranında yerine getirmiştir. • Mvc structure kapsamında Layout Page oluşturma görevini %100 oranında yerine getirmiştir • Kayıt ol ekranında girdilerin onayları ve gereksinimleri için Model entegre edilmiş ve yazılan Modellerin kullanıcıya gösterimi sağlanmıştır. • Yanlış girilen bilgilerin uyarı mesajı döndermesi için gerekli düzenlemeler yapılmıştır. • Kullanıcı ekranı, randevu al, bize ulaş ekranlarının tasarımı ve kodları eklenmiştir. • Veritabanının model ile ilişkisi sağlanmıştır. • Girilen şifrenin güvenlik için gizli olarak kayıt edilmesi siteye entegre edilmiştir. • Bütün ekranların route ve yönlendirme kısımlarının yazılması, kontrol edilmesi ve testleri tamamlanmıştır.
5	160313009	Semih Efe	<ul style="list-style-type: none"> • Website tasarımı CSS, JS geliştirme görevini

			<p>%100 oranında yerine getirmiştir.</p> <ul style="list-style-type: none"> • Website Route structure yeniden tasarlama görevini %100 oranında yerine getirmiştir. • Website'ye Login Page ekleme görevini %100 oranında yerine getirmiştir. • Website'ye Sign Up Page ekleme görevini %100 oranında yerine getirmiştir. • Website'ye "Haber Sayfası" ekleme görevini %100 yerine getirmiştir. • Website'ye "Blog Sayfası" ekleme görevini %100 yerine getirmiştir. • Website jquery scriptleri güncelleme görevini %100 yerine getirmiştir.
--	--	--	--

3 d. İlerlemeler

- Randevu ekranında kullanıcıların daha hızlı aksiyon alabilmeleri için hızlı randevu sliderı eklenmiştir.
- Ekran yönlendirmesi ekranında kullanıcıya göre kişiselleştirilmiş alanlar eklenmiş ve düzenlenmiştir.
- Kullanıcıların Kayıt ve Giriş sayfalarında belirtilen standartlara göre giriş yapabilmeleri entegre edilmiştir.
- Sayfa geçişleri esnasında oluşan bug'lar tespit edilip çözüm sağlanmıştır. (Route'lar düzeltilmiştir.)
- Kullanıcı Ekranı için fotoğraf ekleme ve kişisel bilgilerini değiştirebileceği yeni alanlar eklenmiştir.
- Şifremi unuttum ve beni hatırla butonları eklenip aktif hale getirilmiştir.