

Beykent Üniversitesi
Yazılım Mühendisliği
Yazılım Mühendisliği Tasarım Projesi
2020 – Bahar

Rapor 2

EVA Kişisel Sağlık Asistanı (EKSA)

Grup Elemanları

ARDA KEYİŞOĞLU 160313008
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.....	3-5
2 c. Projenin Aşamaları.....	5-6
2 Dönem İçinde Yapılan İşler.....	7
2 a. Web Sitesi Tasarımı(Arda Keyişoğlu).....	7-10
2 b. Web Sitesi Model(Mert EFE).....	10-11
2 c. Web Sitesi Tasarımı (Mert EFE)	11-13
2 d. Web Sitesi Tasarımı(Semih EFE).....	13-17
2 e. Mobil Uygulama Kitle Endeksi Tasarımı (Fatih Göktuğ Yıldırım).....	18-20
2 f . Mobx State Yönetimi(Fatih Göktuğ Yıldırım).....	20-23
2 g. Kullanıcı Profil Ekranı (Mehmet Uğur Atmaca)	23-25
2 h. Sayfa Yönlendirme Ekranı (Mehmet Uğur Atmaca)	25-26
3 Sonuç.....	26
3 a. Başarı ile tamamlanan işler.....	26-27
3 b. Başarı ile tamamlanamayan işler.....	27
3 c. Kişilerin görevlerini yerine getirme durumu.....	27-29
3 d. İlerlemeler	29

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 ekran tasarımı ve kodlamasıHTML CSS JavaScript, Alternatif Randevu sistemi, İlaç hatırlatıcı, Tahlil sorgulama ve Vücut kitle endeksi ekran tasarımı ve kodlamasıGoogle Maps navigasyon entegrasyonu ve ekran tasarımıAdobe Photoshop, logo ve web sitesi içerisinde kullanılan projeye uygun görsellerin hazırlanmasıMVC Kullanıcı giriş, Kullanıcı kayıt, Kullanıcı profili.MVC Randevu sistemi, Vücut kitle endeksi, İlaç hatırlatıcı ve Tahlil sorgulama kodlaması.Veri Tabanı oluşturulmasında bölünen kısmı Create etme.MVC Ses sistemi teknolojisine uygun tasarımlar ve

			kontrol testleri. • MVC OCR teknolojisine uygun tasarımlar ve kontrol testleri.
2	160313049	Fatih Göktuğ Yıldırım	• React-Native ilaç hatırlatma ekran tasarımı ve kodlaması. • React-Native Alternatif Giriş ekran tasarımı ve kodlaması. • React-Native ilaç ekleme ekran tasarımı ve kodlaması. • React-Native Ekran navigasyon mimarisi tasarımı ve kodlaması. • React-Native mobx uygunluğu araştırama ve uygunsa entegre edilmesi. • React-Native Profil ve Ayarlar ekran tasarımı ve kodlaması. • React-Native ses sistemi entegre çalışması. • React-Native OSR teknoloji entegre çalışması. • Veri Tabanı oluşturulmasında bölünen kısmı Create etme. • Api' kodlarının doğruluğunu test etme ve hayata geçirme. • MVC Ses sistemi entegresi ve çalışma doğruluğu
3	160313043	Mehmet Uğur Atmaca	• 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. • React-Native ses sistemi entegre çalışması. • React-Native OSR teknoloji entegre çalışması. • Veri Tabanı oluşturulmasında bölünen kısmı Create etme. • Api' kodlarının doğruluğunu test etme ve hayata geçirme. • MVC Ses sistemi entegresi ve kontrollü çalışması. • MVC OSR teknolojisi entegre çalışması.
4	160313005	Mert Efe	• HTML CSS JsScript, 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 Ses sistemi teknolojisine uygun tasarımlar ve kontrol testleri. • MVC OCR teknolojisine uygun tasarımlar ve kontrol testleri.

			<ul style="list-style-type: none"> 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. MVC Ses sistemi teknolojisine uygun tasarımlar ve kontrol testleri. MVC OCR teknolojisine uygun tasarımlar ve kontrol testleri. 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> <p>Güvenilirlik şartlarının sağlanması.</p> <p>Gereksinimlere göre görevlere ayrılıp proje bazlı geliştirmeler yapılması.</p> <p>Rakiplerin arayüzleri ve kullanılabilirliği analiz edilmesi.</p>
2	Tasarım	01.15.2020 – 11.03.2020	<p>Anasayfa ve diğer ekranların geliştirilmesi.</p> <p>Hatırlatma ve Tahlil sorgulama ekranlarının tasarlanması.</p>

			<p>Tasarım şablonları oluşturularak en uygun arayüz belirlenmesi.</p> <p>Web site tasarımı için bootstrap framework, jquery, css, html, javascript kullanılması.</p> <p>Web sayfası için Mvc'ye entegre edilip Mvc mimarisinin kullanılması.</p> <p>Mobil-App için React Native kullanılması. Arayüzlerin entegre edilmesi.</p> <p>Web ve Mobil için responsive olması.</p>
3	Ön Uç Geliştirme	10.02.2020 – 30.03.2020	<p>Mobil programlama hybrid bir platform olduğu için arka uç geliştirmeyi de kapsamaktadır bu durum yapılacak tasarımların React ile bütünleştirilmesi yapılır.</p> <p>Web programlama kısmında güncel bootstrap.js kütüphaneleri ile belirlenen ara yüzler geliştirilir.</p>
4	Arka Uç Geliştirme	10.02.2020 – 30.03.2020	<p>Ön yüzü geliştirilen arayüzler her iki platformda da fonksiyonellikleri çalışır hale getirilmelidir.</p>
5	Tasarım Testleri	31.03.2020 – 03.04.2020	<p>Fonksiyonlar çalıştığı zaman tasarımı problemlerin çıkıp çıkmadığı, farklı ekran boyutlarında şekil bozuklukları var mı bunlar kontrol edilir.</p>
6	Veri Tabanı Tasarımı	20.03.2020 – 23.03.2020	<p>Raporda tasarlanan veritabanı tablo modelleri ve özellikleri sunucu da Create edilir ve yayına alınır.</p>
7	Api Geliştirilmesi	31.03.2020 – 05.04.2020	<p>Veritabanı modelinin verilerini erişme ve veri kaydetme seçeneklerini hayata geçirebilmek ve güvenliği sağlamak için API geliştirilir.</p>
8	Test	20.04.2020	<p>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.</p>

2 Dönem İçinde Yapılan İşler

2 a. Web Sitesi 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>
          </form>
        </div>
      </div>
    </div>
  </div>
</section>
```

```

<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 – 1: Web Sitesi 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">4-10</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>
</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 – 2: Web Sitesi Tasarımı(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 – 3: Web Sitesi Tasarımı(Arda Keyişoğlu)

2 b. Web Sitesi Model(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 – 4: Web Sitesi Model 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 – 5: Web Sitesi Model(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 – 6: Web Sitesi Model(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 – 7: Web Sitesi Model (Mert Efe)

2 c. Web Sitesi Tasarım(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">x</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>
    <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>
  </form>
</div>
```

```

        <div class="col-lg-3">
            <input class="form-control" type="text" value=""
placeholder="Cadde">
        </div>
    </div>

```

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

Randevu Al

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

```

Kod Örneği – 9: Web Sitesi Model(Mert Efe)

2 d. Web Sitesi 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" = '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 – 10: Web Sitesi 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>
</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 – 11: Web Sitesi 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 – 12: Web Sitesi 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>Kullanıcı koşullarını kabul ediyorum.</label>
    </div>
    <div class="form-group form-button">
    <input type="submit" name="signup" id="signup" class="form-submit h6"
value="Kaydol" />
    </div>
    </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>
```

2 e. 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 – 5: Kitle Endeksi Tasarımı

Sabit componentlerimizi responsive şekilde tasarladım.Sonuç kısmını tasarım yüzünde ‘Modal’ olarak adlandırılan bir pop up ile sundum.Hesaplamalar Class içerisinde tanımladığım methodlar ile sonuçlandırdım.

```

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(orand <= 24.9)
{
    durum = 'İdeal Kilolu';
}
else if(orand <= 29.9)
{
    durum = 'İdeal Kilolunun Üzeri';
}
else if(orand < 40)
{
    durum = 'Obez';
} else if(orand >= 40)
{
    durum = 'Aşırı Obez';
}
ideal = 22 / (this.state.boy * this.state.boy);
ideal = ideal * 100000;
ideal = ideal.toFixed(1);
this.setState({ endeks : oran , durum , ideal});
}

```

Kod Örneği – 6: Kitle Endeksi Hesaplanması

Hesaplama butonu aktif edildiğinde bir method tetiklenir. Method hesaplama işlemlerini yaptıktan sonra Modal'ı aktif ederek kullanıcıya sonucu aktarır.

2 f. 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 taşınmları 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. İstedığımız sayfalara iletişimini kurabilmek için 'observer' yapısını eklemeliyiz.

```

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 – 7: Mobx State Yönetimi

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.Methodlarımdan biri olan ‘searchFilter’ attribute olan aranacak değeri ile bir filtreleme yaparak aranan geçmişte ve günümüzü kapsayan listeyi günceller.

2 g. 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 – 8: Kullanıcı Profil Ekranı

2 h. 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',tittle:'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 – 9: Sayfa Yönlendime Ekranı

3 Sonuç

3 a. Başarı ile tamamlanan işler

- 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

- 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 araştırmış ve seçmiştir.(%100) • Web sitesine uygun görselleri tasarlayıp sayfalara eklemiştir.(%100) • Ana sayfa-Hakkımızda-Bize ulaşın ekran tasarımı ve gereken bileşen kodlamasını yerine getirmiştir.(%100) • İlaç hatırlatıcı ve Tahlil sorgulama ve Vücut kitle endeksi ekran tasarımlarını yerine getirmiştir.(%100) • Logo tasarımını yerine getirmiştir.(%100)

			<ul style="list-style-type: none"> • Google Maps entegrasyonu(%50)
2	160313049	Fatih Göktuğ Yıldırım	<ul style="list-style-type: none"> • 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"> • Profil ekranı tasarlanmış gerekli style kodları yazılmıştır.. • Navigasyon ekranında değişiklikler yapılmış kaynak kodlarında ihtiyaca yönelik düzenlemeler getirilmiştir. • Randevu ekranında iyileştirilmeler yapılmış hızlı randevu alma sliderı eklenmiştir.
4	160313005	Mert Efe	<ul style="list-style-type: none"> • Website tasarımını Mvc Structure tabanlı yeniden oluşturma görevini %100 oranında getirmiştir. • 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 • Account işlemleri için Model sayfaların ve Kullanıcı Giriş ekranı için Model ve Kaydol Ekranı için Model ekranları eklenmiştir. • 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. • Katmanlı Mimari ile veritabanı kurulmasına başlanılmıştır. • 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 %100 oranında yerine getirmiştir. • 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 Login ve Sign Up sayfalarını 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.