

T.C.
DÜZCE ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ



LARAVEL İLE DÜZCE ÜNİVERSİTESİ WEB SİTESİ GELİŞTİRME

Esra TEKİR

BİTİRME TEZİ

Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü

DANIŞMAN

DR. Öğretim Üyesi Ahmet Albayrak

DÜZCE, 2021

İÇİNDEKİLER

	<u>Sayfa No</u>
ŞEKİL LİSTESİ	3
ÇİZELGE LİSTESİ.....	4
ÖZET	5
ABSTRACT.....	6
1. GİRİŞ.....	7
2. BÖLÜM2	9
2.1. LARAVEL	9
2.2. Laravel Dosya Yapısı	9
2.2.1. Genel Dizin	9
3. BÖLÜM3	11
2.1.BACKEND	11
2.2.FRONTEND	18
4. SONUÇLAR VE ÖNERİLER	23
5. KAYNAKLAR	24
ÖZGEÇMİŞ.....	26

ŞEKİL LİSTESİ

<u>No</u>	<u>Sayfa</u>
Şekil 2.1 Admin Login Sayfası	11
Şekil 2.2.	12
Şekil 2.3.	12
Şekil 2.4. Yönetici Login Route'si	12
Şekil 2.5.	13
Şekil 2.6. Admin Paneli	14
Şekil 2.7. Duyurular Veritabanı Tablosu	14
Şekil 2.8.	15
Şekil 2.9. Slider Veritabanı Tablosu	16
Şekil 2.10.	16
Şekil 2.11.	16
Şekil 2.12.	17
Şekil 2.13.	18
Şekil 2.14.	19
Şekil 2.15.	19
Şekil 2.16.	20
Şekil 2.17.	20
Şekil 2.18.	21
Şekil 2.19.	22
Şekil 2.20.	22
Şekil 2.21.	23
Şekil 2.22.	24

ÇİZELGE LİSTESİ

No		Sayfa
Çizelge 2.1.	Laravel kök dizin yapısı	10
Çizelge 2.2.	Laravel'deki Uygulama Dizini Yapısı	10

ÖZET

LARAVEL İLE DÜZCE ÜNİVERSİTESİ WEB SİTESİ GELİŞTİRME

Esra TEKİR

Düzce Üniversitesi

Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü

Bitirme Tezi

Danışman: Dr. Öğretim Üyesi Ahmet ALBAYRAK

Ocak 2022

Bu tez, Laravel Framework kullanılarak yeni bir Düzce Üniversitesi web sitesi geliştirilmesini amaçlamaktadır. Laravel, yerleşik özellikler aracılığıyla web uygulamaları geliştirmeyi daha kolay ve daha hızlı hale getirmek için tasarlanmış açık kaynaklı bir PHP framework'tür. MVC tabanlı bir framework olduğundan yapısı gereği çok sade ve temiz kod yazma olanağı sağlamıştır. Dahili birçok yapı bulundurmakla kalmıyor bunları karmaşadan ve hantallıktan uzak şekilde sunan bir framework'tür. Auth, Redis bunlardan sadece bazılarıdır. Projeyi çok kısa zamanda ayağa kaldırır vakit kaybetmeyi ortadan kaldırır. Eski usul kod geliştirmekten modern kabul görmüş programcılığa adım atmaya yardımcı olmuştur. ORM, OOP, Template Engine, Redis, Composer, Unit Test vb yapılara aşina oldurmuştur.

Anahtar sözcükler: MVC, Auth, OOP, Composer.

ABSTRACT

BURAYA TEZ BAŐLIĐI İNGİLİZCE YAZILMALIDIR

Esra TEKİR

Düzce University

Faculty of Engineering, Department of Computer Engineering

Supervisor: Dr. Ahmet ALBAYRAK

January 2022, 30 pages

This thesis aims to develop a new Duzce University website using Laravel Framework. Laravel is an open source PHP framework designed to make developing web applications easier and faster through built-in features. Since it is an MVC-based framework, it provides the opportunity to write very simple and clean code due to its structure. Not only does it contain many internal structures, it presents them away from clutter and cumbersomeness. Auth, Redis are just some of them. It makes the project stand up in a very short time and eliminates wasting time. It helped to move from developing old-fashioned code to modern accepted programming. It become familiar with ORM, OOP, Template Engine, Redis, Composer, Unit Test etc.

Keywords: MVC, Auth, OOP, Composer.

1.GİRİŞ

Bu bölümde projeye geçmeden önce Laravel tarihçesi ve nelerden oluştuğuna dair araştırmalara yer verilmiştir. İlk olarak;

Laravel nedir?

Laravel ihtiyaç duyulan, gelişmiş birçok özellik ve yapıyı üzerinde barındıran, PHP ve OOP'nin tüm özelliklerinden yararlanan, web uygulamaları geliştirmeyi sağlayan açık kaynak PHP framework' tür. Birçok gelişmiş özelliği içerisinde barındıran bu sistem sayesinde pek çok faydalı kod sistemlerine yardımcı olur. Sistemsel kullanımda öğrenilmesi oldukça kolay olan bu sistem birçok kişi tarafından tercih edilir.

Her alanda kişi ve kişiler bu sistemi kullanılarak sade kodlar yazarak büyük küçük uygulamaların geliştirilmesine ve uygulamaların daha kaliteli olmasını sağlayan kodlar yazar. Bu sistem sayesinde sade kodlar yazarak büyük küçük uygulamalar geliştirilir. İster bireysel olarak isterse proje amaçlı yazılan uygulamalarda kurumsal bir uygulama adına sahip olan bu sistem sayesinde uygun olan sistemler düzenlenebilir.

Açık kod kaynaklı bu sisteme sahip olan sistemlerin geliştiricileri tarafından kullanılabilir. Bu sistem ve gereksinimlere sahip olmak için bazı belirli işlemlerin bilinmesi gerekir. Açık kaynaklı bir koda sahip olan MVC yapısına sahip olan bu sistem profesyoneller ve geliştiriciler tarafından kullanılır. Laravel'in özellikleri:

1)ORM

ORM (Object Relational Mapping), Database ile uygulamamızda (Object-Oriented) nesnelerimiz sayesinde bağlantı kurup yönetmemizi sağlayan bir yapıdır. Klasik SQL cümleleri yazmadan nesnelerimiz üzerinden veri tabanına erişim sağlayıp kontrol edebiliyor sorgular çalıştırabilir. ORM database'den bağımsız çalışır. Yani Mysql, SQLite, postgresql, MSSql, Oracle gibi birçok database için aynı kodları kullanırsınız.

2)Blade Template

Laravel, Blade adı verilen template engine'ye sahiptir. Projeyi yine çok kolay bir şekilde arayüzle bütünleştirebilir. Projeye temiz ve dinamik arayüzler hazırlama kolaylığı sağlar.

3)Route

Laravel'in iyi bir route (yönlendirme) mekanizması vardır. Yorulmadan temiz URL elde eder Api'ler için uygun erişim yönlendirmeleri yapılabilir. Route yapısıyla ister yönlendirmeler yapılabilir ister filtrelerden kontrollerden geçirebilir ister Controller class'larına yönlendirme yapılabilir.

4)Migrations (Sürüm Kontrolü, Göçler)

Veri tabanı sürüm kontrol sistemidir. Migration ile PHP Artisan Komut Satırı ile uygulama veritabanına şemalar ekleyebilme ve düzenleyebilme imkanı sağlar. Veri tabanı yönetim sistemine gitmeden sisteminizdeki veritabanınızı oluşturmaya yada güncellemeye yarayan yapı. Örneğin uygulamanızı başka bir ortama taşıdınız, veri tabanını oluşturmaya çalışmaktansa uygulamanızda ki hali hazırdaki yapıyı çalıştırarak sistemi hazır hale getirmiş oluyoruz.

5)Automatic Pagination (Otomatik Sayfalandırma)

Laravel sayfalama sistemini düşünmüş ve geliştiriciyi zahmete sokmadan el atmıştır. Kolay şekilde kullanabilir ve istenirse değiştirilebilir.

6)Performans

Laravel Cache mekanizmaları sunar. Redis ile bütünleşmiş bir yapısı vardır. Projenize Redis dahil etmek için ekstra uğraşmanıza gerek yoktur.Çok hızlı ve basit şekilde yapılandırılabilir.

Dahili olarak auth, filter gibi birçok yapıyı içinde barındırır. Bunu yine kendi yapısına uygun ve basit şekilde sunar. Hemen hemen her uygulamada Auth mekanizması olur ve bunu yapmak için ekstra çabaya girilir. İşte Laravel bunu da pas geçmemiş bunun gibi yapılarda sunmuş. Birçok Symfony bileşeni üzerine kurulmuştur. Güvenli, hızlı, sitemizi ayağa kaldırmak çok kolaydır.

2.BÖLÜM 2

Bu bölümde Laravel ile geliştirdiğim web sitenin ayrıntılarına, nasıl yapıldığına, nelerden oluştuğuna dair bilgilere yer verilmiştir.

2.1. Laravel

Laravel, MVC (Model-View-Controller) yapısında web uygulamaları geliştirme için tasarlanmış ücretsiz, açık kaynak bir PHP web uygulaması iskeletidir. MVC, yazılım mühendisliğinde kullanılan bir mimari desendir. Kullanıcıya yüklü miktarda verinin sunulduğu karmaşık uygulamalarda veri ile gösterimin soyutlanması esasına dayanır. Laravel, kişiselleştirilmiş web yazılımı oluşturmak için hızlı ve verimli bir şekilde kullanıldığı bilinmektedir.

Laravel, bağımlılıklarını yönetmek için Composer'ı kullanır. Bu nedenle, Laravel'i kullanmadan önce, Composer yüklenir. Kurulduktan sonra laravel new komutu belirlenen dizinde yeni bir Laravel kurulumu oluşturacaktır. Örneğin, laravel new blog, Laravel'in önceden kurulmuş tüm bağımlılıklarını içeren yeni bir Laravel kurulumu içeren blog adında bir dizin oluşturacaktır. PHP'nin yerleşik geliştirme sunucu içerisinde kullanıldığından, php artisan serve komutunu kullanılır. Bu komut <http://localhost:8000> adresinde bir geliştirme sunucusu başlatır.

Öncelikle üstünde kodlarımızı yazacağımız program olan **PhpStorm**'u indirerek başlıyoruz. Daha sonra Laravel'i içine kuracağımız AppServ'i bilgisayarımıza indiriyoruz. **Appserv** Php, Laravel, Codeigniter veya başka programlama dillerini tarayıcımızda çalıştırmamızı sağlayan aracı programdır. Php, Apache ve PhpMyAdmin Paketlerini bilgisayarımıza kurulumunu bizim için tek tıkla yapar. Daha sonra AppServ klasörü içerisinde www klasörü altında adı 'ecms' olan projemiz için yeni bir klasör oluşturularak Composer yardımıyla Laravel' i kurarız.

2.2.Laravel Dosya Yapısı

2.2.1.Genel Dizin

Laravel'i kurduktan sonra web sunucunuzun doküman / web kökünü public dizin olacak şekilde konfigüre etmelisiniz. Bu dizindeki index.php, uygulamaya giren tüm HTTP istekleri için ön denetleyici görevi görür.

Bu dizin (klasör, directory), içinde başka bazı alt dizinleri bulabileceğiniz en önemli dizinlerden biridir (çizelge 2.1). Çizelge 2.2’de Laravel'deki uygulama dizini yapısı verilmiştir.

Çizelge 2.1. Laravel kök dizin yapısı

Dizin	Açıklama
app	Uygulama dizini Laravel uygulamanızın temel kodunu tutar.
bootstrap	Bootstrap dizini, uygulamanız için kullanılan tüm bootstrapping komut dosyalarını tutar.
config	Config dizini, tüm proje konfigürasyon dosyalarınızı (.config) tutar.
database	Veritabanı dizini, veritabanı dosyalarınızı tutar.
public	Genel dizin, Laravel projenizi başlatmanıza yardımcı olur ve ayrıca projeniz için gereken resimlerle birlikte diğer komut dosyalarını (JavaScript ve CSS) tutar.
resources	Resources dizini tüm Sass dosyalarını, dil (yerelleştirme) dosyalarını, (varsa) şablonları tutar.
routes	Routes dizininde, konsol.php, api.php, channel.php vb. Gibi tüm yönlendirme dosyalarınızı yönlendirin.
storage	Depolama dizini, oturum dosyalarınızı, önbelleği, derlenmiş şablonları ve çerçeve tarafından oluşturulan çeşitli dosyaları tutar.
test	Test dizini, tüm test durumlarınızı tutar.
vendor	Satıcı dizini, tüm besteci (composer) bağımlılık dosyalarını (dependencies) tutar.

Çizelge 2.2. Laravel'deki Uygulama Dizini Yapısı

Dizin	Açıklama
Console	Konsol dizini, tüm proje uzman komutlarınızı içerir.
Events	Events dizini, laravel uygulamanızın açabileceği olay dosyalarını tutar. Etkinlikler, projedeki herhangi bir işlemin gerçekleştiği laravel projesinin diğer bölümlerine mesaj veya sinyal göndermek için kullanılır.
Exceptions	İstisnalar dizini, laravel projenizin istisna istisnalarını, Laravel projenizin attığı tüm istisnaları ele alan dosyaları tutar.
Http	Http dizini farklı filtreler, istekler ve denetleyicileri tutar.
Jobs	İşler dizini, bu dizindeki tüm sanatçılığı işleri tutar. Ancak başlangıçta yaratılmaz, bunun yerine, bu artisan komutunu yazmanız ve çalıştırmanız gerekir: make:job
Listeners	Dinleyiciler dizini, olayları almak ve işlemek için kullanılan tüm projenizin işleyici sınıfını tutar.
Mail	Ana dizin, Laravel projeniz aracılığıyla gönderilen tüm e-postaları tutar ve bu dizinin şu komutu kullanarak oluşturulması gerekir: make:mail
Notifications	Bildirimler dizini Laravel projeniz aracılığıyla gönderilen tüm işlem bildirimlerinizi içerir ve bu dizinin bu komut kullanılarak oluşturulması gerekir: make:notification
Policies	Politikalar dizini laravel projeniz için farklı politikalar tutar.
Providers	Sağlayıcılar dizini, farklı servis sağlayıcıları içermek için kullanılır.
Rules	Kurallar dizini, özel doğrulama kurallarıyla ilgili tüm farklı nesneleri tutar ve bu dizinin bu komut kullanılarak oluşturulması gerekir: make:rule

Proje için yapılan ön hazırlıklar bittikten sonra yönetici kontrollerinin yapıldığı admin sayfasını yaparak başlıyoruz.

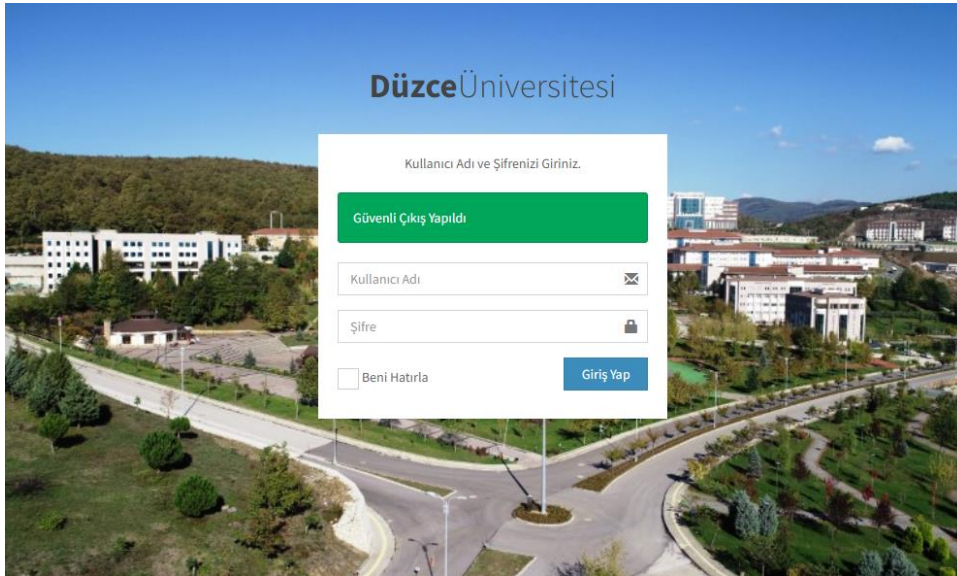
3.BÖLÜM

3.1. BACKEND

Backend, bir yazılımda kullanıcıların görmediği kısımların tümü ve ana sistemdir. Bu bölümde örneğin kayıtlı olduğunuz bir web sitesine giriş yapmak istiyorsunuz, ilk olarak bu Giriş Yap isteği sunucuya iletilir daha sonra ise sunucunun veri tabanında isminiz ile şifrenizin uyuşup uyuşmadığına bakılır ve ona göre bir cevap gelir. Siz giriş yapma isteği gönderdiğinizde bunu veri tabanında arayan, bulan ve nasıl bir cevap verileceğini belirleyen şey backend'tir. Laravel'de Veritabanı işlemlerini 'Migration'larla kolayca kurabiliriz. Önce phpMyAdmin'den veritabanını oluştururuz. Laravel'den veritabanına bağlarız. Şifremiz ve kullanıcı adımızı kaydederek. Daha sonra migration ile otomatik tablolarımızı oluştururuz. Terminal üzerinden 'php artisan migrate' komutunu çalıştırarak tablolarımızı migrate ederiz.

Login Logout işlemleri için Laravel Auth ile Node.js kolayca kurulabilir. Bize otomatik olarak login/logout işlemlerini sunar. Kurulumu, terminale 'composer require laravel/ui --dev' ve 'php artisan ui vue --auth' yazarak gerçekleştiririz.

Admin sayfamıza erişmek için 'ecms/nedmin' yazarak admin sayfamıza giriş yapmak için bir login sayfası karşımıza çıkıyor.



Şekil 2.1. Admin Login Sayfası

Login yapabilmek için veritabanında kayıtlı bir admin bilgisi kaydediyoruz. Kayıtlı admin kullanıcı adı kısmına geçerli belirlenen mail adresi ve belirlenen şifreyi girerek admin

```

<!DOCTYPE html>
<html>
<head>

  <meta charset="utf-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <title>Admin | Log In</title>

  <!-- Tell the browser to be responsive to screen width -->
  <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport">

  <!-- Bootstrap 3.3.7 -->
  <Link rel="stylesheet" href="/backend/bower_components/bootstrap/dist/css/bootstrap.min.css">

  <!-- Font Awesome -->
  <Link rel="stylesheet" href="/backend/bower_components/font-awesome/css/font-awesome.min.css">

  <!-- Ionicons -->
  <Link rel="stylesheet" href="/backend/bower_components/ionicons/css/ionicons.min.css">

  <!-- Theme style -->
  <Link rel="stylesheet" href="/backend/dist/css/adminLTE.min.css">

  <!-- iCheck -->
  <Link rel="stylesheet" href="/backend/plugins/iCheck/square/blue.css">

```

```
<link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro" />
<style type="text/css">
    .login-page{
        background: url("/images/settings/DuzceUni1.jpg") no-repeat center center fixed;
        background-size: cover;
        -webkit-background-size: cover;
        -moz-background-size: cover;
        -o-background-size: cover;
    }

    body {
        overflow: hidden;
    }
</style>
</head>
<body class="hold-transition login-page">
<div class="login-box">
    <div class="login-logo">
        <a href="{{route('admin.Login')}}"><b>Düzce</b>Üniversitesi</a>
    </div>
```

```
Route::namespace('Backend')->group(function () {

    Route::prefix('nedmin')->group(function () {

        Route::get('uri: '/', action: 'DefaultController@index')->name('nedmin.Index')->middleware('middleware');
        Route::get('uri: /login', action: 'DefaultController@login')->name('nedmin.Login');
        Route::get('uri: /logout', action: 'DefaultController@logout')->name('nedmin.Logout');

        Route::post('uri: /login', action: 'DefaultController@authenticate')->name('nedmin.Authenticate');

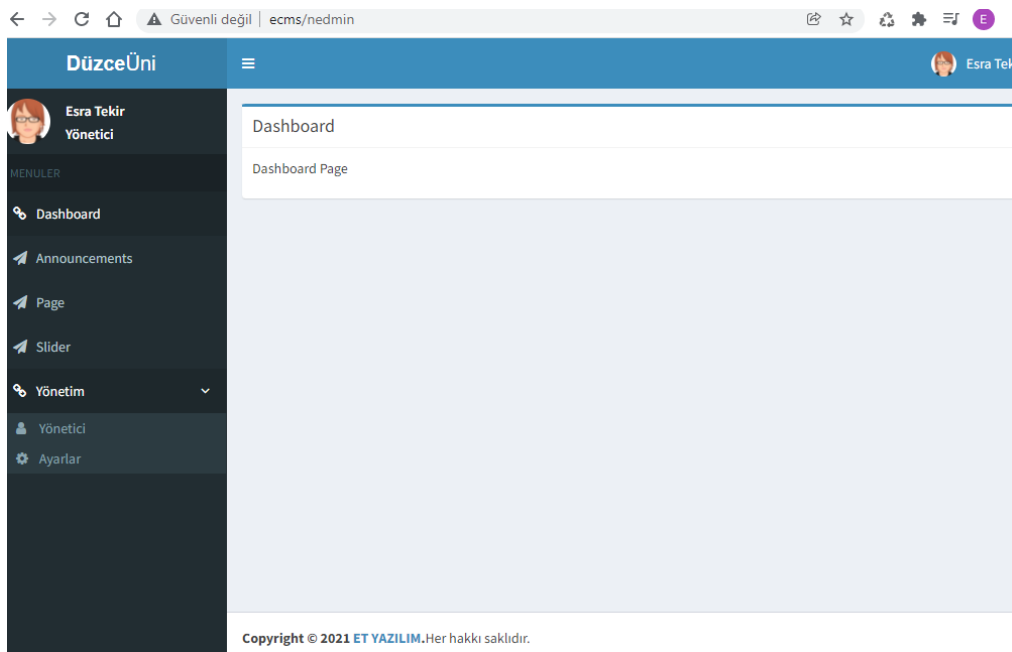
    });

});
```

Şekil 2.4. Yönetici Login Route'si

Oluşturulan ilk route'muz, kontrollerinin yapıldığı DefaultController içerisinde index metodunu çalıştıracaktır. İkinci route'miz DefaultController içerisinde oluşturduğumuz login metodunu, üçüncü route'miz ise logout metodunu çalıştırarak çıkış yönlendirmesini çalıştıracaktır.

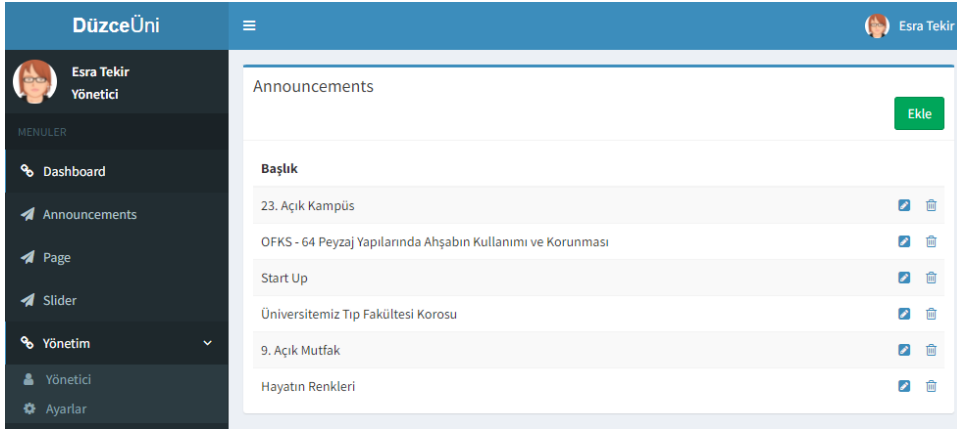
En son route'mizde ise kimlik doğrulama işlemlerini içeren authenticate metodunu çalıştıracaktır. Authenticate metodunda eğer kullanıcı bilgileri doğru ise yönetici işlemlerinin yapıldığı anasayfaya giriş yapılacak yanlıssa hatalı kullanıcı uyarısını verecektir. Başarılı olarak giriş yapıldıysa eğer yönetici paneli karşımıza çıkacaktır.



Şekil 2.5. Admin Paneli

Admin panelimizin görünümünü içinde barındıran view'imiz resources-views-backend' in içerisinde bulunan layout.blade.php dosyasıdır. Admin panelinde Düzce Üniversitesi web sitesine ekleyeceğimiz duyurular, sliderların, yöneticilerin ve ayarların kontrolünü ve düzenlemesini yaptığımız bölümlerden oluşmaktadır.

Duyurular bölümünde yeni duyurular ekleme, duyurularımızın içeriğini düzenleme ve silme işlemleri yapılabilecektir.



Şekil 2.6.

Duyurularımızı ekleme view'ımız resources-views-backend-blogs' un içinde olan create.blade.php dosyamızdır. Duyurularımızı düzenleme view'ımız resources-views-backend-blogs' un içinde olan edit.blade.php dosyamızdır. Duyurularımızı görüntülediğimiz ve silme işlemlerinin yapıldığı view'ımız resources-views-backend-blogs' un içinde olan index.blade.php dosyamızdır.

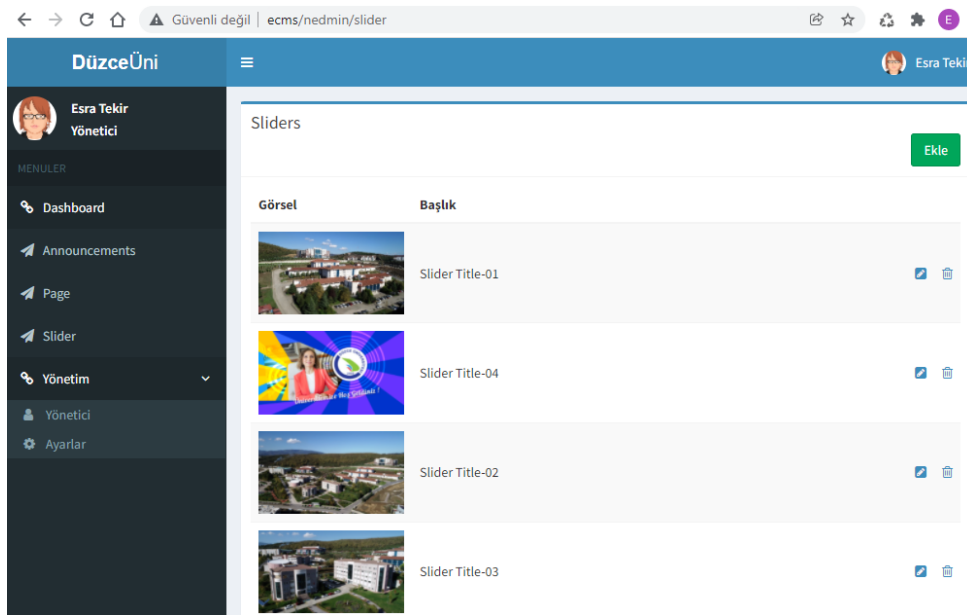
Duyurularımızın kontrollerinin yapıldığı Controller'imiz BlogController'dir. BlogController'in içerisinde index, create, store, edit, update, sortable metodları vardır. Bu metodlar view'larımızın kontrolünü yapmaktadır. Duyuru veritabanı tablosu oluşturmak için migrate ediyoruz. Daha sonra tohumlama işlemi olan Seed işlemleri için kodumuz olan 'php artisan make:seeder BlogsTableSeeder' yazarak tohumlama işlemlerini yapıyoruz. **Seeder**'lar migrationlarımız ile de birlikte çalıştığı için tek komut yardımı ile hem veritabanı inşasını hem de varsayılan kayıtların oluşturulmasını sağlayabilmektedir. Duyurularımızın veritabanı da şu şekildedir:

	id	created_at	updated_at	blog_title	blog_slug	blog_file	blog_must	blog_content	blog_status
<input type="checkbox"/>	1	2021-12-24 16:00:00	2022-01-02 13:00:59	23. Açık Kampüs	23-acik-kampus	61cb1a42a9071.jpg	0	<p>23. Açik Kampüs</p>	1
<input type="checkbox"/>	2	2021-12-23 16:00:00	2022-01-02 13:00:59	OFKS - 64 Peyzaj Yapılarında Ahşabın Kullanımı ve ...	ofks-64-peyzaj-yapilarinda-ahsabın-kullanımı-ve-ko...	61cb1a8939965.jpg	1	<p>OFKS - 64 Peyzaj Yapılarında Ahşabın Kullanımı ...	1
<input type="checkbox"/>	3	2021-12-22 16:00:00	2022-01-02 13:00:59	Start Up	start-up	61cb1ad8d218d.jpg	2	<p>Start Up</p>	1
<input type="checkbox"/>	4	2021-12-22 16:00:00	2022-01-02 13:01:00	Üniversitemiz Tıp Fakültesi Korosu	universitemiz-tip-fakultesi-korusu	61cb1b1401d87.jpg	3	<p>Üniversitemiz Tıp Fakütesi Korosu</p>	1
<input type="checkbox"/>	5	2021-12-21 16:00:00	2022-01-02 13:01:00	9. Açık Mutfak	9-acik-mutfak	61cb1b462fe8e.jpg	4	<p>9. Açik Mutfak</p>	1
<input type="checkbox"/>	10	2021-12-21 16:00:00	2022-01-02 13:01:00	Hayatın Renkleri	hayatin-renkleri	61cb1b61a5a1b.jpg	5	<p>Hayatın Renkleri</p>	1

Şekil 2.7. Duyurular Veritabanı Tablosu

Veritabanının da oluşturulmasıyla birlikte verileri çekmemiz gerekiyor. Bunun için Model oluşturarak verilerimizi çekeriz. Model oluşturmak için ‘php artisan make:model Blogs’ komutunu çalıştırırız. Model oluşturduğumuzda verileri çekmek için Controller yardımıyla artık verileri çekebiliriz. Ayarlarımızı sürükle bırak mantığıyla JQuery UI Sortable işlemlerini projeye ekleyerek gerçekleştirilmiştir. Duyurular kısmında veri silme işlemleri için AJAX ile yapılmıştır.

Slider bölümünde web site anasayfamıza ekleyebileceğimiz sliderlarımızı ekleme, sliderlarımızı güncelleme, düzenleme ve silme işlemlerini gerçekleştirebiliyoruz.

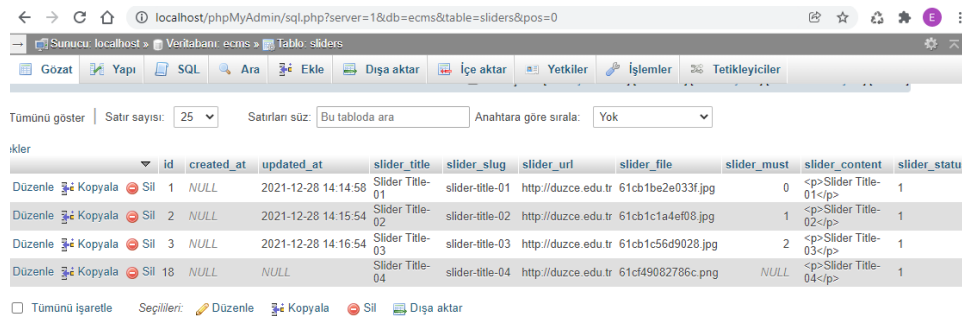


Şekil 2.8.

Sliderlerimizi ekleme view’ımız resources-views-backend-sliders’ in içinde olan ‘create.blade.php’ dosyamızdır. Sliderlerimizi düzenleme view’ımız resources-views-backend-sliders’in içinde olan ‘edit.blade.php’ dosyamızdır. Sliderlerimizi görüntülediğimiz ve silme işlemlerinin yapıldığı view’ımız resources-views-backend-sliders’ in içinde olan index.blade.php dosyamızdır.

Sliderlerimizin kontrollerinin yapıldığı Controller’imiz SliderController’dır. SliderController’in içerisinde index, create, store, edit, update, destroy ve sortable metodları vardır. Bu metodlar view’larımızın kontrolünü yaparak yeni slider ekleme, oluşturma, editleme ve güncelleme yapmamıza olanak sağlar. Slider veritabanı tablosu oluşturmak için migrate ediyoruz. Daha sonra tohumlama işlemi olan Seed işlemleri için kodumuz olan ‘php artisan

make:seeder SlidersTableSeeder' yazarak tohumlama işlemlerini yapıyoruz. **Seeder**'lar migrationlarımız ile de birlikte çalıştığı için tek komut yardımı ile hem veritabanı inşasını hem de varsayılan kayıtların oluşturulmasını sağlayabilmektedir. Sliderlarımızın veritabanı da şu şekildedir:

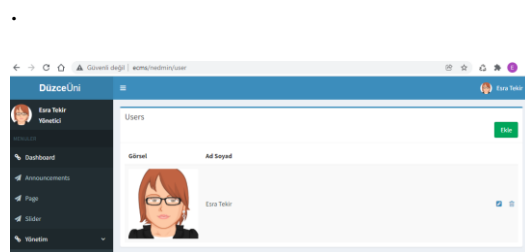


	id	created_at	updated_at	slider_title	slider_slug	slider_url	slider_file	slider_must	slider_content	slider_status
Düzenle Kopyala Sil	1	NULL	2021-12-28 14:14:58	Slider Title-01	slider-title-01	http://duzce.edu.tr	61cb1be2e033f.jpg	0	<p>Slider Title-01</p>	1
Düzenle Kopyala Sil	2	NULL	2021-12-28 14:15:54	Slider Title-02	slider-title-02	http://duzce.edu.tr	61cb1c1a4ef08.jpg	1	<p>Slider Title-02</p>	1
Düzenle Kopyala Sil	3	NULL	2021-12-28 14:16:54	Slider Title-03	slider-title-03	http://duzce.edu.tr	61cb1c56d9028.jpg	2	<p>Slider Title-03</p>	1
Düzenle Kopyala Sil	18	NULL	NULL	Slider Title-04	slider-title-04	http://duzce.edu.tr	61cf49082786c.png	NULL	<p>Slider Title-04</p>	1

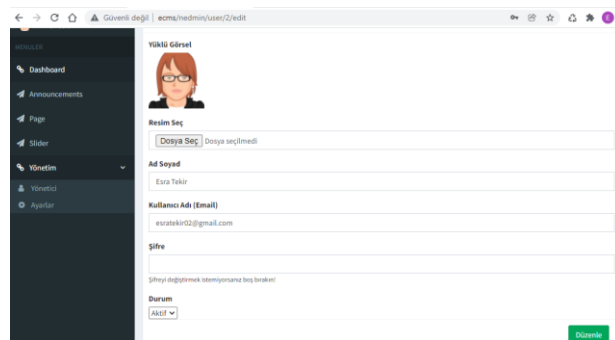
Şekil 2.9. Slider Veritabanı Tablosu

Veritabanının da oluşturulmasıyla birlikte verileri çekmemiz gerekiyor. Bunun için Model oluşturarak verilerimizi çekeriz. Model oluşturmak için 'php artisan make:model Sliders' komutunu çalıştırırız. Model oluşturduğumuzda verileri çekmek için Controller yardımıyla artık verileri çekebiliriz. Ayarlarımızı sürükleyip bırak mantığıyla JQuery UI Sortable işlemlerini projeye ekleyerek gerçekleştirilmiştir.

Yönetici kısmında ise admin ekleme, admin bilgilerini güncelleme, giriş şifresini değiştirme, admin silme işlemleri yapılmaktadır.



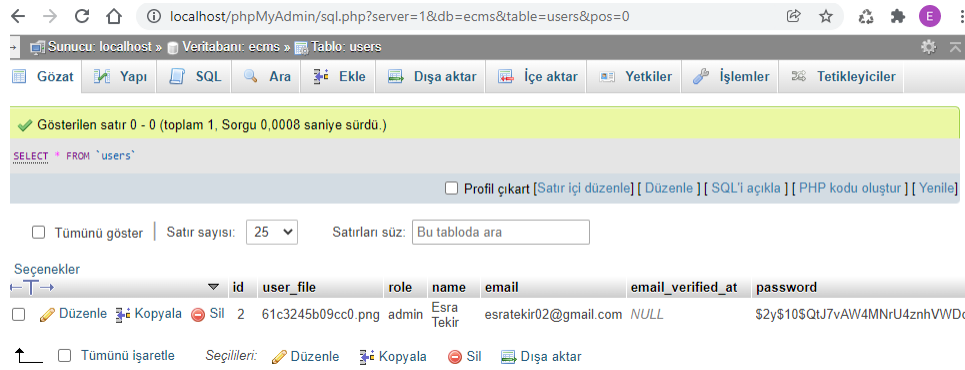
Şekil 2.10.



Şekil 2.11.

Yönetici ekleme view'ımız resources-views-backend-users' in içinde olan 'create.blade.php' dosyamızdır. Yönetici düzenleme view'ımız resources-views-backend-users'in içinde olan 'edit.blade.php' dosyamızdır. Yöneticilerimizi görüntülediğimiz ve silme işlemlerinin yapıldığı view'ımız ise resources-views-backend-users' in içinde olan index.blade.php dosyamızdır.

Yönetici işlemlerinin kontrollerinin yapıldığı Controller'imiz UserController'dir. UserController'in içerisinde index, create, store, edit, update, destroy ve sortable metodları vardır. Bu metodlar view'larımızın kontrolünü yaparak yeni yönetici ekleme, oluşturma, editleme ve güncelleme yapmamıza olanak sağlar. Yönetici veritabanı da şu şekildedir:





	id	user_file	role	name	email	email_verified_at	password
<input type="checkbox"/>	2	61c3245b09cc0.png	admin	Esra Tekir	esratekir02@gmail.com	NULL	\$2y\$10\$QJ7vAW4MnRU4znhVWDe

Şekil 2.12. Yönetici Veritabanı Tablosu

Veritabanının da oluşturulmasıyla birlikte verileri çekmemiz gerekiyor. Bunun için Model oluşturarak verilerimizi çekeriz. Model oluşturmak için 'php artisan make:model User' komutunu çalıştırırız. Model oluşturduğumuzda verileri çekmek için Controller yardımıyla artık verileri çekebiliriz. Ayarlarımızı sürükleyip bırak mantığıyla JQuery UI Sortable işlemlerini projeye ekleyerek gerçekleştirilmiştir.

Admin panelimizdeki son bölüm olan ayarlar web sitemizin genel ayar işlemlerinin yapıldığı kısımdır. Sitemizin başlığı, logosu, iletişim bilgilerinin kayıtlı olduğu kısımdır.

Id	Açıklama	İçerik	Anahtar Değer	Type	
1	Başlık	Düzce Üniversitesi	title	text	<input checked="" type="checkbox"/>
2	Açıklama	Düzce Üniversitesi Web Sitesi Ayarları	description	text	<input checked="" type="checkbox"/>
3	Logo		logo	file	<input checked="" type="checkbox"/>
4	Icon		icon	file	<input checked="" type="checkbox"/>
5	Anahtar Kelimeler	laravel,Düzce Üniversitesi,esra tekir	keywords	text	<input checked="" type="checkbox"/>
6	Telefon Sabit	0850 800 81 81	phone_sabit	text	<input checked="" type="checkbox"/>
7	Telefon GSM	0543 XXX XX XX	phone_gsm	text	<input checked="" type="checkbox"/>
8	Mail	esratekir02@gmail.com	mail	text	<input checked="" type="checkbox"/>
9	İlçe	Merkez	ilçe	text	<input checked="" type="checkbox"/>
10	İl	Düzce	il	textarea	<input checked="" type="checkbox"/>
11	Açık Adres	ANA KAMPÜS Düzce Üniversitesi Konuralp Yerleşkesi Yörük Mah. Merkez/DÜZCE, 81620	adres	textarea	<input checked="" type="checkbox"/>

Şekil 2.13.

Ayarlarımızı güncelleme view’ımız resources-views-backend-settings’in içinde olan ‘edit.blade.php’ dosyamızdır. Ayarları görüntülediğimiz view’ımız ise resources-views-backend-settings' in içinde olan index.blade.php dosyamızdır.

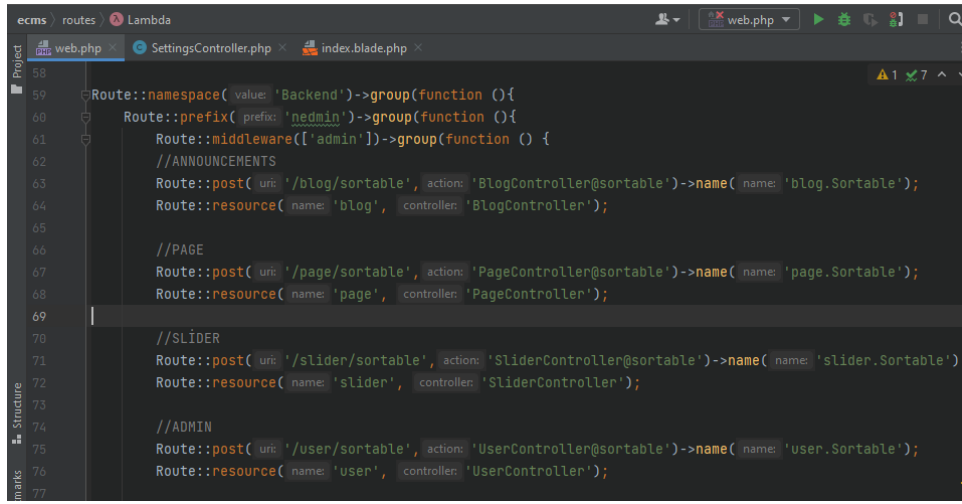
Ayarların kontrollerinin yapıldığı Controller’imiz SettingsController’dir. SettingsController’in içerisinde index, sortable, destroy, edit, update metodları vardır. Bu metodlar view’larımızın kontrolünü yaparak ayarlarımızı güncelleme işlemlerinin yapılmasına olanak sağlar. Ayarlar tablosu oluşturmak için migrate ediyoruz. Daha sonra tohumlama işlemi olan Seed işlemleri için kodumuz olan ‘php artisan make:seeder SettingsTableSeeder’ yazarak tohumlama işlemlerini yapıyoruz. **Seeder**'lar migrationlarımız ile de birlikte çalıştığı için tek komut yardımı ile hem veritabanı inşasını hem de varsayılan kayıtların oluşturulmasını sağlayabilmektedir. Ayarların veritabanı da şu şekildedir:

	id	created_at	updated_at	settings_description	settings_key	settings_value	settings_type	settings_must	settings_delete	settings_stat
Düzenle Kopyala Sil	1	NULL	2021-12-28 13:29:22	Başlık	title	Düzce Üniversitesi	text	0	0	1
Düzenle Kopyala Sil	2	NULL	2021-12-28 14:36:43	Açıklama	description	Düzce Üniversitesi Web Sitesi Ayarları	text	1	0	1
Düzenle Kopyala Sil	3	NULL	2021-12-28 14:35:49	Logo	logo	61cb20c56df6f.PNG	file	2	0	1
Düzenle Kopyala Sil	4	NULL	2021-12-20 12:17:46	Icon	icon	61c0746a3e074.png	file	3	0	1
Düzenle Kopyala Sil	5	NULL	2022-01-03 12:13:31	Anahtar Kelimeler	keywords	laravel,Düzce Üniversitesi,esra tekir	text	4	0	1
Düzenle Kopyala Sil	6	NULL	2021-12-27 13:53:11	Telefon Sabit	phone_sabit	0850 800 81 81	text	5	0	1
Düzenle Kopyala Sil	7	NULL	2021-12-19 16:58:26	Telefon GSM	phone_gsm	0543 XXX XX XX	text	6	0	1
Düzenle Kopyala Sil	8	NULL	2021-12-19 16:58:24	Mail	mail	esratekir02@gmail.com	text	7	0	1
Düzenle Kopyala Sil	9	NULL	2021-12-27 13:52:41	İlçe	ilçe	Merkez	text	8	0	1
Düzenle Kopyala Sil	10	NULL	2021-12-27 13:52:08	İl	il	Düzce	textarea	9	0	1
Düzenle Kopyala Sil	11	NULL	2021-12-27 13:57:06	Açık Adres	adres	ANA KAMPÜS Düzce Üniversitesi	textarea	10	0	1

Şekil 2.14. Ayarlar Veritabanı Tablosu

Veritabanının da oluşturulmasıyla birlikte verileri çekmemiz gerekiyor. Bunun için Model oluşturarak verilerimizi çekeriz. Model oluşturmak için ‘php artisan make:model Settings’ komutunu çalıştırırız. Model oluşturduğumuzda verileri çekmek için Controller yardımıyla artık verileri çekebiliriz. Ayarlarımızı sürükleyip bırak mantığıyla JQuery UI Sortable işlemlerini projeye ekleyerek gerçekleştirilmiştir.

Backend kısmında son olarak yönlendirme işlemlerinin yapıldığı Route oluşturma kısmımız da şu şekildedir:



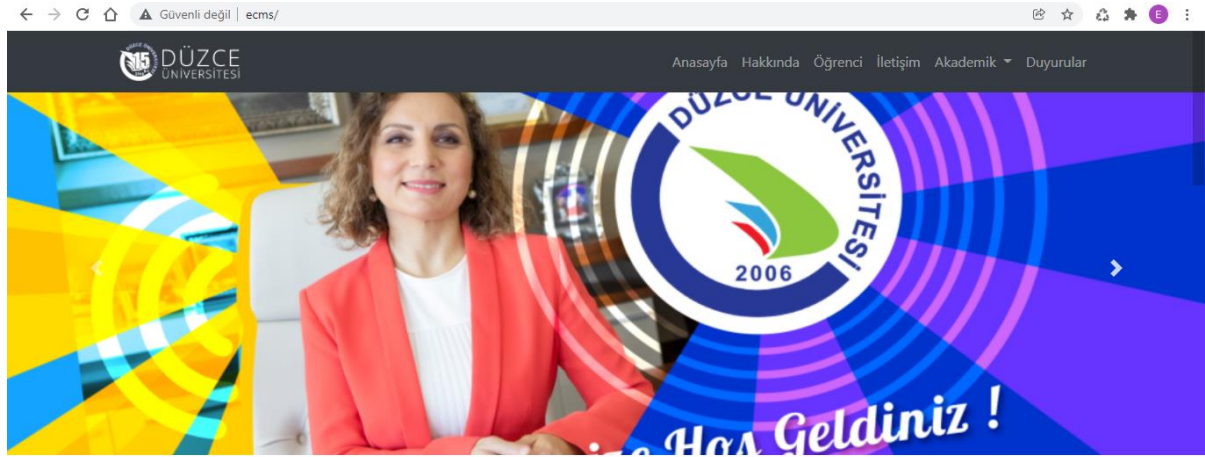
```
58
59 Route::namespace('Backend')->group(function () {
60     Route::prefix('nedmin')->group(function () {
61         Route::middleware(['admin'])->group(function () {
62             //ANNOUNCEMENTS
63             Route::post('/blog/sortable', action: 'BlogController@sortable')->name('blog.Sortable');
64             Route::resource('blog', controller: 'BlogController');
65
66             //PAGE
67             Route::post('/page/sortable', action: 'PageController@sortable')->name('page.Sortable');
68             Route::resource('page', controller: 'PageController');
69
70             //SLIDER
71             Route::post('/slider/sortable', action: 'SliderController@sortable')->name('slider.Sortable');
72             Route::resource('slider', controller: 'SliderController');
73
74             //ADMIN
75             Route::post('/user/sortable', action: 'UserController@sortable')->name('user.Sortable');
76             Route::resource('user', controller: 'UserController');
77
```

Şekil 2.15. Backend

3.2.FRONTEND

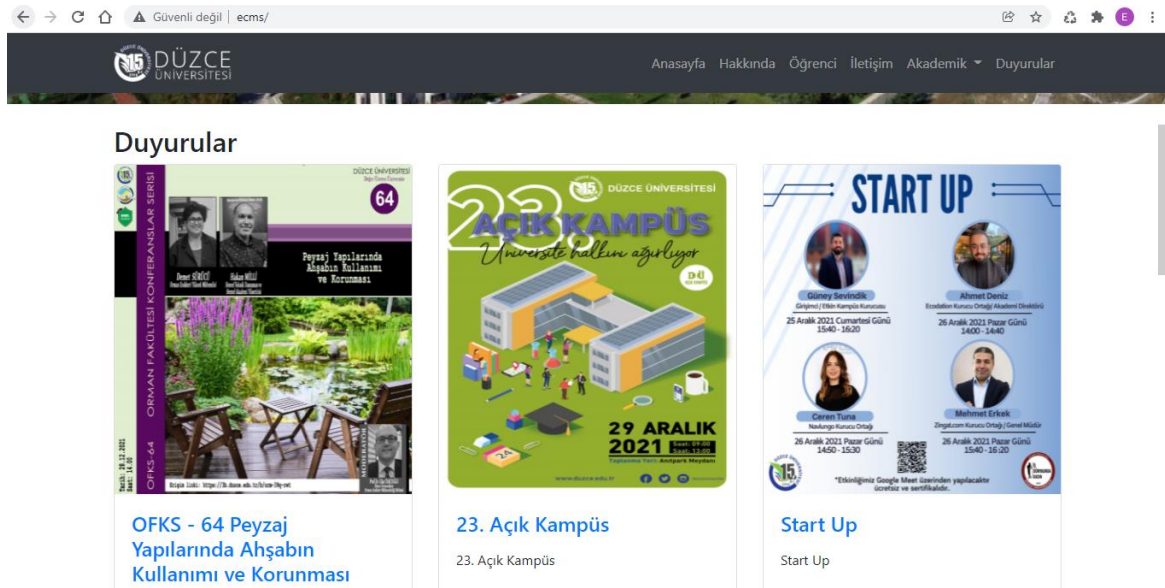
Frontend’in Türkçe karşılığı “Önyüz”dür. Yapılma aşamasındaki bir web sitesinin ön yüzünü (client-side) HTML, CSS ve JavaScript gibi teknolojileri kullanarak web sitesinin görsel tarafını oluşturan kişilere ise front-end developer (Ön yüz geliştirici) denir. Web sitesinde kullanılacak renkler, içeriklerin yerleşimi, yazı tipinin seçilmesi ve uygulanması gibi birçok görevi bünyesinde barındırır. Projemizin frontend kısmı ise şöyledir;

İlk olarak tarayıcımıza ‘http://ecms/’ yazarak web sitemizin ana sayfasına giriş yapıyoruz. Ana sayfada karşımıza ilk çıkan backend kısmında kontrolünü de yaptığımız Düzce Üniversitesi’mizin resimlerini içinde bulunduran sliderlerimizdir.



Şekil 2.16. Anasayfa

Aşağı inildikçe Duyurular kısmı karşımıza çıkacaktır. Duyurulara ister ana sayfadan ister sağ üst köşede bulunan Duyurular kısmından daha ayrıntılı bir şekilde incelenme, içeriğini okuyabilme şansı sunulmuştur.



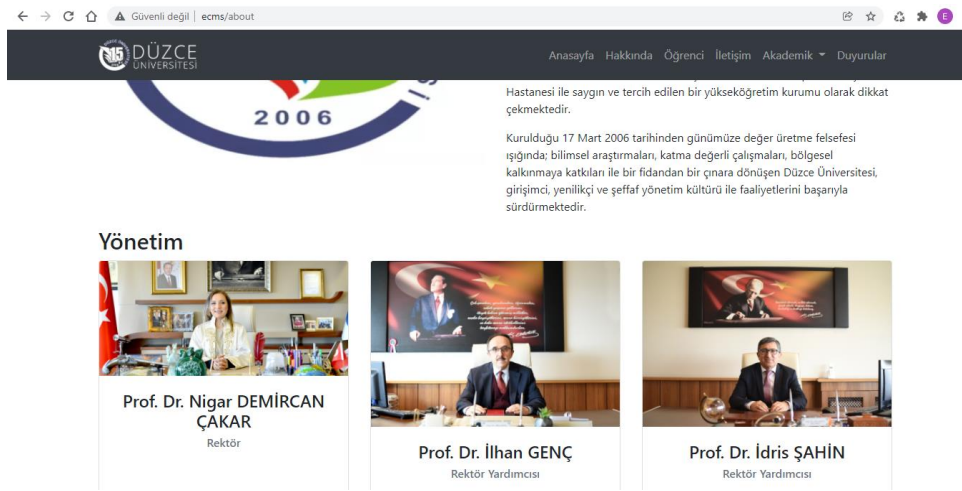
Şekil 2.17. Anasayfa

Ana sayfamızın en aşağı kısmında ise Üniversite'mizin bilgi güvenliği politikasının açıklandığı kısım vardır. Ana sayfamızın görünümünü sağlayan view'imiz 'resources/views/frontend/default' un içinde bulunan 'index.blade.php' dosyası ve sağ üstteki ana sayfa, hakkımızda, iletişim, duyurular kısımlarını görüntülememizi sağlayan dosya ise 'resources/views/frontend' un içinde bulunan 'layout.blade.php' dosyamızdır.

Ana sayfanın kontrolünü yapmamızı sağlayan Controller'ımız 'Controllers/frontend' in içinde bulunan DefaultController' da bulunan index metodudur. Index metodu ana sayfamızdaki duyurular ve sliderların listelenmesi ve viewsımızın çağırıldığı metoddur.

```
public function index()
{
    $data['blog']=Blogs::all()->orderby('blog_must');
    $data['slider']=Sliders::all()->orderby('slider_must');
    return view('frontend.default.index',compact('data'));
}
```

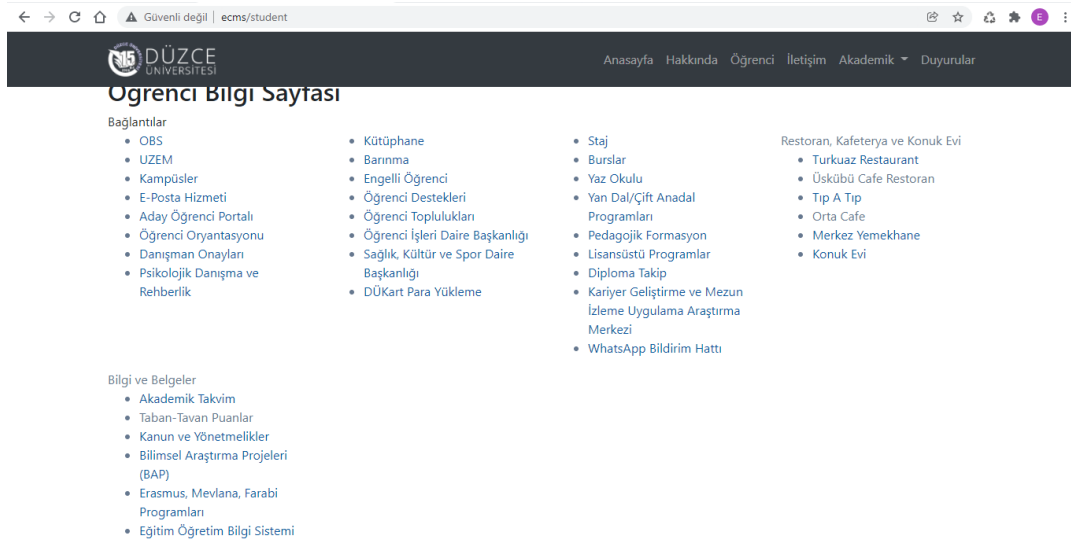
Hakkımızda kısmına geldiğimizde Üniversitemizin kuruluşu ile ilgili bilgiler ve yönetimde bulunan rektör ve rektör yardımcılarının tanıtıldığı ve resminin olduğu kısım karşımıza çıkacaktır.



Şekil 2.18. Hakkımızda Sayfası

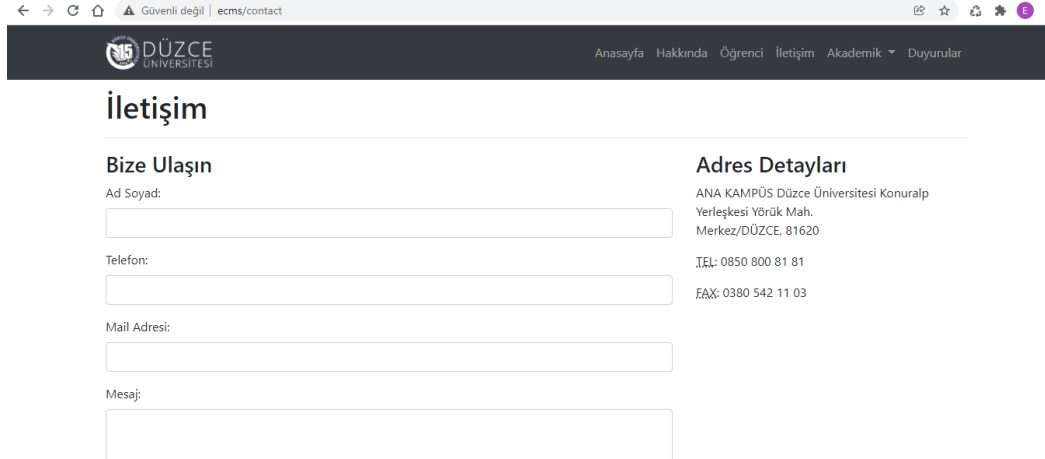
Hakkımızda sayfasının görünümünü sağlayan view'imiz 'resources/views/frontend/default' un içinde bulunan 'about.blade.php' dosyasıdır. Hakkımızda sayfamızın kontrolünü yapmamızı sağlayan Controller'imiz 'Controllers/frontend' in içinde bulunan DefaultController' da bulunan about metodudur. About metodu viewsımızın çağırıldığı metoddur.

Öğrenci kısmına geldiğimizde Üniversitemizin öğrencilerinin ihtiyacı olan her şeyin olduğu kısım karşımıza çıkacaktır. Sayfamızın viewi 'resources/views/frontend/default' da bulunan student.blade.php'dir. Controlleri DefaultController'in içindeki student metodudur.



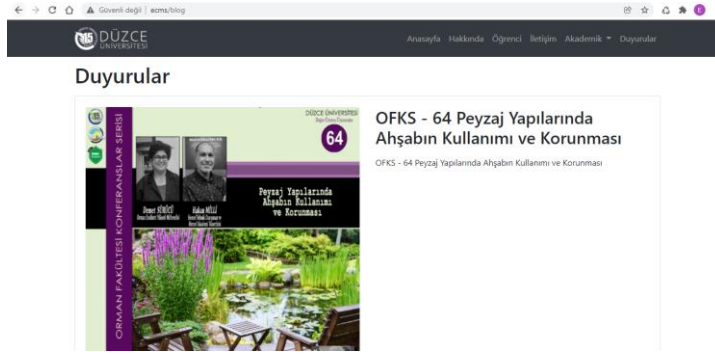
Şekil 2.19. Öğrenci Sayfası

İletişim kısmına geldiğimizde Üniversitemizin iletişim bilgilerinin ve adresinin olduğu kısım karşımıza çıkacaktır. Sayfamızın viewi ‘resources/views/frontend/default’ da bulunan contact.blade.phpdir. Controlleri DefaultController’in içindeki contact metodudur.



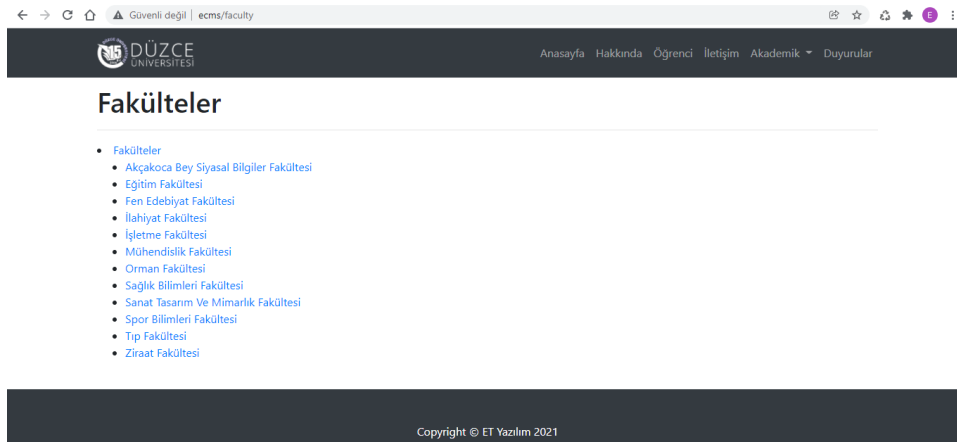
Şekil 2.20. İletişim Sayfası

Duyurular kısmına geldiğimizde Üniversitemizin yaptığı etkinlik vb. İçinde bulunduğu duyuruların olduğu kısım karşımıza çıkacaktır. Sayfamızın viewi ‘resources/views/frontend/blog’ da bulunan index.blade.phpdir. Controlleri BlogController’dir.



Şekil 2.21. Duyurular Sayfası

Son olarak Akademik kısmına geldiğimizde Üniversitemizde bulunan Fakülteler, Enstitüler, Yüksek Okullar, Meslek Yüksekokulları ve Kurullar karşımıza çıkacaktır. Fakülteler sayfamızın viewi 'resources/views/frontend/default' da bulunan 'faculty.blade.php' dir. Controlleri DefaultController'dir.



Şekil 2.21

Son olarak anasayfa, hakkında, iletişim, öğrenci, duyurular sayfalarına yönlendirme, link vermemizi sağlayan Route'larımız vardır.

```
Route::namespace( value: 'Frontend')->group(function () {
    Route::get( uri: '/', action: 'DefaultController@index')->name( name: 'home.Index');
    Route::get( uri: '/blog', action: 'BlogController@index')->name( name: 'blog.Index');

    //CONTACT
    Route::get( uri: '/contact', action: 'DefaultController@contact')->name( name: 'contact.Detail');

    //STUDENT
    Route::get( uri: '/student', action: 'DefaultController@student')->name( name: 'student.Detail');

    //ABOUT
    Route::get( uri: '/about', action: 'DefaultController@about')->name( name: 'about.Detail');
```

Şekil 2.22.Frontend Routes

4.SONUÇLAR VE ÖNERİLER

Bu tez çalışmasının sonucunda Laravel Framework kullanılarak Düzce Üniversitesi web sitesi tasarlanmış ve geliştirilmiştir. Web sitemiz buna uygun bir veritabanına, backend'e ve frontend'e sahip olacak şekilde hazırlanmıştır. Laravel frameworkünün sunduğu kolaylıklar özelleştirilmiş olan web uygulamalarında kolay ve hızlı bir biçimde tasarlanmış olan, oldukça kullanışlı, açık kaynaklı olan modern bir web uygulama çerçevelerinden bir tanesi olmasıdır. Geliştiricileri, sunmuş oldukları özellikler, performans sebebi ile Laravel'i diğer çerçeve sistemlerine tercih ediyorlar. Model İzleme denetleyici olarak bilinen MVC yapısını içerisinde bulundurur. Bu sayede PHP'den çok daha fazla kullanışlı hale gelmesini sağlar. Yönlendirme (Routing), kimlik doğrulama(Auth), oturumlar ve ön belleğe alınma işlemleri gibi web projelerinin birçoğunda kullanılan ortak amaçta yürütülen görevleri kolaylaştırır. Bu sayede ağrıyı geliştirme dışarısına çıkarmayı hedefler. Benzersiz bir mimari alt yapısına sahiptir. Bu sistemler büyük projeler için değil aynı anda küçük projeler için de kullanılabilir.

5.KAYNAKLAR

- [1] Bean, M. (2015) Laravel 5 essentials, Packt Publishing Ltd, Unknown, 0s.
- [2] L Welling, L Thomson, PHP and MySQL development, 2003, books.google.com.
- [3] S McCool,Laravel Starter,2012, data.dammio.com.
- [4]HR Yu, Designed and implementation of web based on Laravel Framework, ACSR, Advances in Computer Science,2015,0s.
- [5]Laravel, , “Laravel Documentation”, Erişim: 3 Aralık 2021,<https://laravel.com/docs>.
- [6] P DuBois, MySQL,2008, books.google.com.
- [7] MKI BAGWAN, SD GHULE, A Modern Review on Laravel- PHP Framework, 2019.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Esra TEKİR
Doğum Tarihi ve Yeri : 12.11.1997 Eminönü
Yabancı Dili : B1 Seviyesi
E-posta : esratekir02@gmail.com

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Lisans	Bilgisayar Müh.	Düzce Üniversitesi	2022
Lise		Haraççı İMKB Anadolu İmam Hatip Lisesi	2015