**T.C.**

**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

BSM 498 BİTİRME ÇALIŞMASI

SÜRÜCÜ KURSU  
PLATFORMU

G151210097 – EMRE KARA

G151210055– MAKBULE AMİNE BOSNALI

|  |  |  |
| --- | --- | --- |
| Fakülte Anabilim Dalı  Tez Danışmanı | :  : | BİLGİSAYAR MÜHENDİSLİĞİ  Doç.Dr. DEVRİM AKGÜN |

2018-2019 Bahar Dönemi

T.C.

SAKARYA ÜNİVERSİTESİ

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

SÜRÜCÜ KURSU

PLATFORMU

BSM 498 - BİTİRME ÇALIŞMASI

Emre KARA

Makbule Amine BOSNALI

|  |  |  |
| --- | --- | --- |
| Fakülte Anabilim Dalı | : | BİLGİSAYAR MÜHENDİSLİĞİ |

Bu tez .. / .. / … tarihinde aşağıdaki jüri tarafından oybirliği / oyçokluğu ile kabul edilmiştir.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| ………………. | ………………. | ………………. |
| Jüri Başkanı | Üye | Üye |

**ÖNSÖZ**

Günümüzde birçok işletme rekabet üstünlüğünü elde etmede Bulut Bilişim teknolojisinin önemini anlamış ve gerek tedarikçileri gerekse müşterileriyle olan ilişkilerini karşılıklı işbirliği ve menfaat esasına bağlı olarak yeniden yapılandırmaya başlamışlardır. Özellikle tedarikçilerle geliştirilen teknolojik işbirliğinin veri hızının arttırılması, iletişim maliyetinin düşürülmesi, veri güvenliğinin sağlanması ve müşteri memnuniyetinin arttırılması gibi konularda son derece olumlu katkılar sağladığı görülmektedir. Tez çalışmamızın planlanmasında, araştırılmasında, yürütülmesinde ve oluşumunda ilgi ve desteğini esirgemeyen sayın hocamız Doç. Dr. Devrim AKGÜN’e sonsuz teşekkürlerimizi sunarız.

**İÇİNDEKİLER**

|  |  |
| --- | --- |
| ÖNSÖZ……...................................................................................................... | iii |
| İÇİNDEKİLER.................................................................................................. | iv |
| SİMGELER VE KISALTMALAR LİSTESİ.................................................... | vi |
| ŞEKİLLER LİSTESİ......................................................................................... | vii |
| TABLOLAR LİSTESİ....................................................................................... | viii |
| ÖZET................................................................................................................. | ix |
|  |  |
|  |  |
| BÖLÜM 1. |  |
| GİRİŞ................................................................................................................ | 1 |
|  |  |
| BÖLÜM 2. |  |
| SİSTEMATİK YAKLAŞIM…………………………...................................... | 3 |
| 2.1. Otomasyon........................................................................................ | 7 |
| 2.2. Programlama Dilleri.......................................................................... | 8 |
| 2.2.1 C# Programlama Dili................................................................ | 8 |
| 2.3. Asp.net Teknolojisi………………………………………………... |  |
| 2.3.1. Mvc Framework……………………………………………... |  |
| 2.3.2. Asp.net Core Teknolojisi…………………………………….. |  |
| 2.4. ReactJs Kütüphanesi………………………………………………. |  |
| 2.4.1. Redux Kütüphanesi…………………………………………. |  |
| 2.5. Sql Veritabanı……………………………………………………… |  |
| 2.5.1. Entity Framework…………………………………………… |  |
| 2.6. Visual Studio………………………………………………………. |  |
| 2.7. Versiyon Kontrol Sistemleri……………………………………….. |  |
| 2.7.1. Git-Github…………………………………………………... |  |
| 2.7.2. TFS………………………………………………………….. |  |
| 2.8. Test………………………………………………………………… |  |
| 2.8.1. Unit Test……………………………………………………. |  |
| BÖLÜM 3. |  |
| EHLİYET TÜRKİYE PLATFORMUNUN TASARIMI................................... | 10 |
| 3.1. Giriş…………………………………………………....................... | 10 |
| 3.2. Kursiyer Uygulaması………............................................................. | 11 |
| 3.2.1. Seo……................................................................................... | 10 |
| 3.3. Admin Paneli………………………................................................. | 11 |
| 3.4. Sürücü Kursu Uygulaması…………………………………………. |  |
|  |  |
| BÖLÜM 4. |  |
| SONUÇLAR………………...………….…………………………………...... | 12 |
|  |  |
|  |  |
| KAYNAKLAR……………………………………………………………….. | 13 |
| EK A………………………………………………………………………….. | 14 |
| ÖZGEÇMİŞ……………………………………………….………………….. | 29 |
|  |  |
|  |  |
| BSM 498 BİTİRME ÇALIŞMASI DEĞERLENDİRME VE SÖZLÜ SINAV TUTANAĞI………………………………………………………………… | 17 |
|  |  |

SİMGELER VE KISALTMALAR LİSTESİ

|  |  |
| --- | --- |
| SQL | : Structured Query Language |
| JS | : Javascript |
| TFS | : Team Foundation Server |
| URL | : Uniform Resource Locator |
| CSRF | : Cross Site Request Forgery |
| XSS | : Cross-Site Scripting |
| SEO | : Search Engine Optimization |
| SPA | : Single Page Applications |
| UI | : User Interface |
| MSSQL | : Microsoft SQL Server |
| VS | : Visual Studio |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**ŞEKİLLER LİSTESİ**

|  |  |  |
| --- | --- | --- |
| Şekil 1.1. | NET Framework İşleyiş…….……………………………... | 18 |
| Şekil 1.2. | Reactjs Özellikleri…...…………………………………….. | 11 |
| Şekil 1.3. | Redux Veri Akışı…...……………………………………… | 27 |
| Şekil 1.4. | Sıkıştırılmış Html Görüntüsü...……………………………. | 28 |
| Şekil 1.5. | Kursiyer Uygulaması..……….……………………………. | 28 |
| Şekil 1.6. | Kurs Detay Sayfası………………………………………… |  |
| Şekil 1.7. | Swagger Api Sayfası………………………………………. |  |
| Şekil 1.8. | Öğrenciler Sayfası…………………………………………. |  |
| Şekil 1.9. | Use Case Diyagramı……………………………………….. |  |
| Şekil 2.1. | Sürücü Kursu Veritabanı Şeması………………………….. |  |
| Şekil 2.2. | Kursiyer Uygulaması Veritabanı Şeması………………….. |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**TABLOLAR LİSTESİ**

|  |  |  |
| --- | --- | --- |
|  |  |  |

**ÖZET**

Anahtar kelimeler Sürücü Kursu Otomasyonu. Kolay Ödeme Sistemi, Web Uygulaması

Günümüzde online uygulamarın yeri tartışılmaz bir öneme sahiptir. Özellikle kolay ve çabasız olarak işleri birkaç tıkla çözebilecek uygulamalar zaman tasarrufuyla birlikte büyük kolaylık sağlamıştır.

Projemizde ehliyet almak için sürücü kursu arayan insanların; evlerine en yakın sürücü kursu hakkında bilgi alarak, kolay bir biçimde kayıt olmaları ve ödeme yapmaları sağlanmaktadır. Müşteriye sınav tarihleri ve ders günleri hakkında hatırlatmalar ve bilgilendirmeler yapılmaktadır. Sisteme kayıtlı sürücü kursları müşteri potansiyelini arttırarak, uzaktan kolay bir şekilde kayıt hizmetini müşterilerine sunmaktadırlar.

Uygulama “ehliyetturkiye.com” üzerinden hizmet vermektedir. Uygulama geliştirilirken; ASP.NET CORE ve React.js teknolojileri kullanılmıştır. Projemiz, yönetici paneli, sürücü kursu paneli ve başvuru sayfası olmak üzere üç ana kısımdan oluşmaktadır. Sistem şu an Türkiye’nin tüm ve il ve ilçelerinde hizmet verebilme kapasitesine sahiptir. Amacımız ülkemizde ehliyet sahibi olmak isteyen herkesin kullanabileceği bir merkez oluşturmaktır.

Sonuç olarak; sürücü kurslarını ve ehliyet almak isteyenleri buluşturan bir uygulama geliştirilmiştir. Ehliyet alma işleminde ve sürücü kurslarının müşteri kayıt sürecinde kolaylık sağlayan bir inovasyon sağlanmıştır.

# GİRİŞ

Teknoloji insanların hayatını kolaylaştırmak için durmaksızın gelişmeye

devam etmektedir. Günümüzde internet üzerinden bir çok yorucu işi kolay bir şekilde yapabilmekteyiz. Her geçen gün geliştirilen uygulamalar sayesinde vaktimizi asıl işlerimize ayırmak daha kolay bir hal almaktadır.

Bu durumun en güzel örneklerinden biri; Yemeksepeti.com’un ülkemizde

oluşturduğu inovasyondur. Yemeksepeti.com’un kurucu ortağı ve CEO’su olan Nezat Aydın [1] : “Online alışveriş Türkiye’deyken de kullandığım şeylerdi ancak ABD’ye gittiğimde internet üzerinden paket servisle tanıştım ve insanların hayatını ne kadar kolaylaştırdığını yerinde gördüm. Türkiye gibi genç nüfuslu olan bir ülkede, paket servis ve internet gibi çok hızlı gelişmekte olan iki sektörü içinde barındıran bir modelin başarılı olacağını düşündüm. Türkiye’ye döndüğümde güvendiğim yakın arkadaşlarımla bu fikri paylaştım ve yemeksepeti.com’u kurduk, 11 senedir hizmet veriyoruz. Şu anda Türkiye’de 41 ilde, onun dışında Rusya’da ve Birleşik Arap Emirlikleri’nde hizmet vermekteyiz.” [2] sözleriyle insanların hayatını kolaylaştıran küçük bir yeniliğin ne kadar etkin olabileceğini açıklamaktadır.

Biz projemizde herhangi bir sürücü kursundan ehliyet alma sürecinin nasıl

daha kolay gerçekleştirebileceğini düşündük. Online bir şekilde ehliyet alma fikri üzerine sürücü kurslarıyla anlaşmalar yaptık. Sürücü adaylarının kendilerine en uygun kursu seçerek, kayıt yapabileceği ve PayTR ile ödeme işlemini gerçekleştirebileği bir platform olan ehliyetturkiye.com’u gerçekleştirdik.

Bu uygulama ile sisteme başvuru yaparak kayıt olan sürücü kursları ile ehliyet

almak isteyen müşterilerimizin iletişim kurmasını sağladık. İlgili sürücü kursu sınav taihlerini sistem üzerinden paylaşabilir, böylece müşterilerine sistem tarafından çeşitli platformlar aracılığıyla bildirimler gönderilir.

Gerçekleştirilen projeler;

\*Sürücü kursu sorgusunun ve ödeme işleminin gerçekleştirilebildiği anasayfa

\*Sürücü kursunun yönetim paneli

\*Ehliyettürkiye.com platformunun yönetici paneli

Bu çalışmada ülkemizdeki sürücü kursları ve ehliyet almak isteyen

insanlara kolaylık sağlayamak amacıyla ehliyetturkiye.com platformu geliştirilmiştir.

Proje esnasında izlediğimiz çalışma takvimi EK-3'te verilmiştir. Bu çalışma

takvimine uymak programlı çalışmamızı sağlamıştır.

# TEKNOLOJİLER VE ARAÇLAR

Bölüm içerisinde geliştirilen yazılımın mantalitesinden ve yazılımda kullanılan yardımcı kütüphanelerden , geliştirilen ortamdan bahsedilmiştir.

* 1. Otomasyon

Otomasyon, bir işin insan ile makine arasında paylaşılmasıdır. Toplam işin

paylaşım yüzdesi otomasyonun düzeyini belirler. İnsan gücünün yoğun olduğu otomasyon sitemleri yarı otomasyon, makinenin yoğun olduğu sitemler de tam otomasyon olarak adlandırılırlar. Sanayi devriminin başlangıcının aksine yakın geçmişimizde üretmek tek başına yeterli olmaktan uzaklaştı. Tüm dünyanın açık bir pazar haline geldiği rekabetçi koşullarda üretimi; hızlı, standart, güvenli, nihayet verimli kılmak, bir zorunluluk haline geldi. Endüstride bu zorunluluğun karşılığı şüphesiz ki otomasyondur.[3]

Otomasyonun yazılım projelerindeki hellerine crm, erp ,mrp gibi projeler örnek olarak verilebilir.Burada geliştirilen uygulamada sürücü kurslarının manuel yaptığı şeyleri otomatikleştirmek ve işlerinin daha iyi yönetilmesini sağlamak hedeflenmiştir. Yazılım modül modül geliştirilmeye çalışılmıştır bu sayede ilerleyen süreçlerde uygulamanın kapsamının ileri seviyelere çıkması planlanmaktadır.

* 1. Programlama Dilleri

Programcının belirli bir algoritmayı ifade etmek için kullandığı standartlaşmış notasyona programlama dili denilmektedir. Programcı komutları yazmak için farklı programlar kullanabilmektedir. Bu diller aracılığıyla programcının bilgisayara hangi veri üzerinde işlem yapacağını, verinin nasıl depolanarak iletileceğini, hangi koşullarda işlem yapılacağını anlatmasını sağlamaktadır. Günümüze kadar 2500’ den fazla programlama dili yapılmıştır.

Programlama dillerini makine yakınlık derecesine göre; alt seviye programlama dilleri, orta seviye programlama dilleri, üst seviye programlama dilleri olmak üzere üçe ayırmaktayız.

Bizim projemizde orta seviye programlama dillerinden biri olan C# dili kullanılmaktadır.

* + 1. C# Programlama Dili

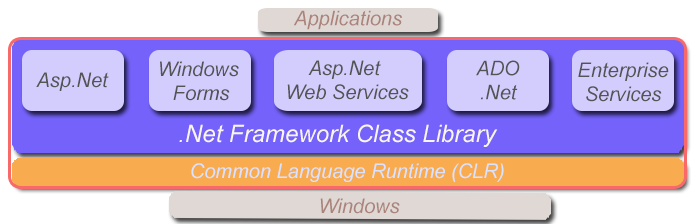
C# programlama dili, C++ gibi bir C türevidir. C++ dili adını, C diliyle

Nesneye yönelimli programlama yapabilmek için eklentiler (C With Classes) almıştır. Benzer şekilde C++ diline yeni eklentiler yapılarak ((C++)++) bir adım daha ileriye götürülmüş ve tamamen nesneye yönelik tasarlanmış C# dilinin isimlendirilmesinde, + karakterinin birbirlerine yakınlaştırılmış hali ve bir melodi anahtarı olan C# Major kullanılmaktadır.

Özellikle nesne yönelimli programlama kavramının gelişmesine katkıda bulunan en aktif programlama dillerinden biri olan C#, basit, modern, genel-amaçlı, nesneye yönelik bir programlama dili olarak tasarlanmıştır. [4]

* 1. ASP.NET Teknolojisi

ASP’ nin açılımı Activate Server Page olarak isimlendirilir. Server taraflı bir web geliştirme teknolojisidir. Microsoft tarafından geliştirilmektedir. **ASP .Net** ile sadece Html tabanlı statik sayfalar değil, dinamik web sayfaları da oluşturabilmektedir.  **ASP.NET** kodları Common Language Runtime(CLR) yani ortak dil çalışma zamanı olarak kullanılmaktadırlar.  CLR, Microsoft .Net framework çatısı altındaki tüm dilleri ASP.Net uygulamalarında ortak bir dile çeviren katmana verilen isimdir. Yani kodlar, diğer diller ile güvenli bir şekilde çalışabilmek için ortak bir dile çevrildikten sonra makine kodlarına çevrilmektedirler. Böylelikle platform bağımsız bir şekilde çalışabilmekteyiz.[5]

****

* + 1. MVC Framework

Model-view-controller (MVC), yazılım mühendisliğinde kullanılan bir  “mimari desen” dir. Bu desen Trygue Reenskaug tarafından ilk olarak tanımlanan bir desendir. Daha sonra Smalltalk üzerine yapılan araştırmalar Xerox araştırma laboratuarlarında devam etmiştir.

“**Model**: Uygulamaya logic olarak eklenecek veri katmanını oluşturduğumuz bölümdür. Veri tabanı işlemlerimiz için tablolarımızı ve saklı yordamlarımızı (store procedure) Ado.Net Entity model veya LinQ model yardımı ile kullanabileceğimiz katman olarak tanımlayabiliriz.

**View**: Kullanıcı kontrollerinin, bileşenlerin gösterildiği ve son kullanıcının gördüğü katmandır. Genel olarak model katman üzerinde yer alan tablo kolonlarına göre oluşturulur. Genel yapısı dışında Asp.Net uygulamalarında da yapabildiğimiz gibi tasarımsal öğeleri ekleyerek (css, resim, animasyonlar (Flash, Silverlight, v.b. ) gibi) daha zengin web uygulamaları da oluşturabilmemiz mümkündür.  
  
**Controller**: View katman üzerinde yapılan işlemler sonucunda sayfanın hangi sayfaya yönleneceğine, ya da veri tabanı üzerinde işlem mi yapacağına ilişkin işlemlerin tutulduğu katmandır. ”

MVC Framework kullanılarak gerçekleştirilen uygulamalar; kullanıcı katmanı, iş katmanı ve veri katmanı olarak oluşmaktadırlar. Bu yapıda geliştirilen uygulamalar, katmanlı yapı dahilinde geliştirilmiş olmaktadırlar.[6]

* + 1. ASP.NET CORE Teknolojisi

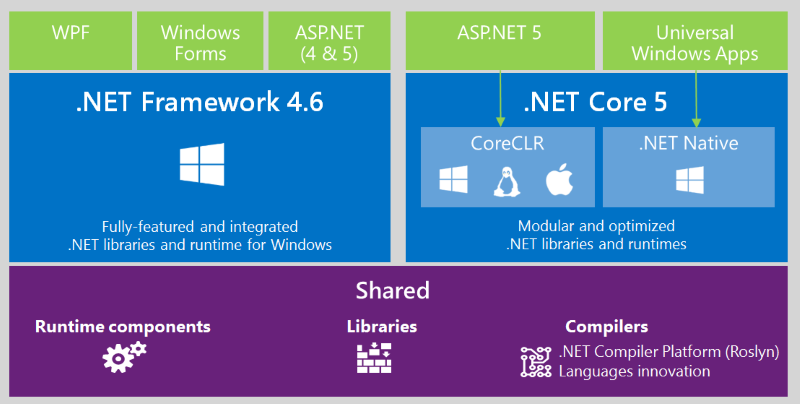
ASP.NET 5 adı yerine ASP.NET CORE adının tercih edileceği belirtilerek ASP.NET teknolojisinin en son sürümü kullanılmaya başlanmıştır.

Bu platformun .NET dünyasından geçiş yapan yazılımcılar için getirdiği en büyük yeniliklerden biri, web uygulamasının sunulabilmesi için IIS’e ihtiyaç duyulmamasıdır. Bu şekilde uygulamanın yayını [Kestrel web sunucusu](https://docs.microsoft.com/en-us/aspnet/core/fundamentals/servers/kestrel" \t "_blank) üzerinden açılabilmekte veya kendi kendini yayımlayacak bir uygulama geliştirilebilmektedir. Yine bu platform içerisinde varsayılan olarak gelen [dependency injection](https://msdn.microsoft.com/en-us/library/dn178469(v=pandp.30).aspx" \t "_blank) desteği ile uygulama içerisindeki bağımlılıkların birbirinden izole edilmesi, test ve bakım maliyetlerinin azalması sağlanmış olur.

ASP.NET Core ‘ün göze çarpan bir diğer özelliği performansıdır. Tüm web frameworkleri ile yapılan benchmark testlerinde, saniyede 7 milyon isteğe cevap verebilmeyi başarmıştır. Bu sıralamada şuan diğer frameworkler arasında 3. sıradadır.

ASP.NET Core MVC’de göze çarpan özellikleri routing, model binding, model validation, dependency injection, filters, areas, Web APIs, strongly typed views, tag helpers, ve view components şeklinde sıralamak mümkündür.

Bizim projemizde ASP.NET CORE teknolojisi ve C# dili kullanılmaktadır. Detaylı bilgi için üçüncü bölüme bakılmalıdır.[7]



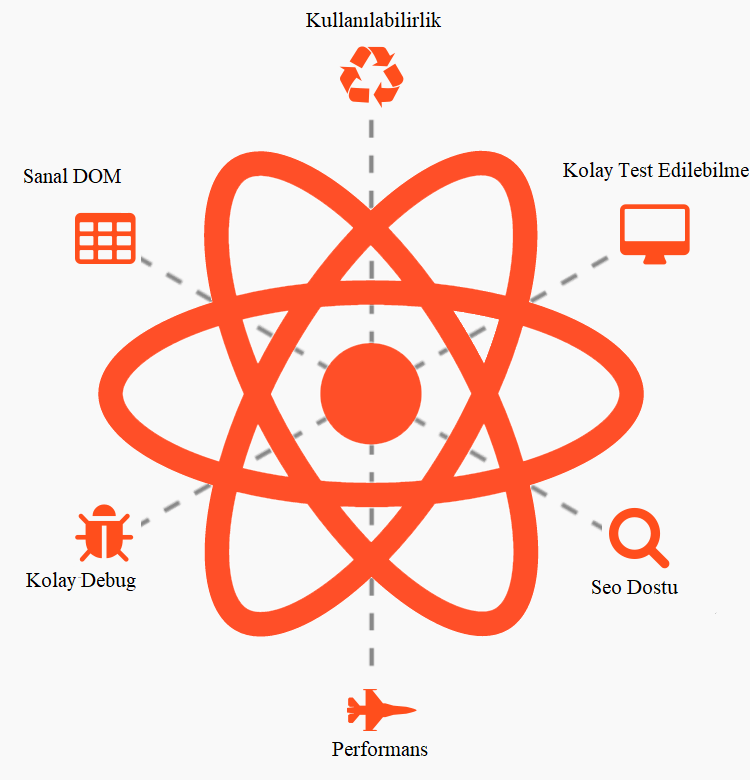
Şekil .Net Framework-.Net Core Karşılaştırılması

Şekil 1.2.’de klasik .net framework’ü ile .net core framework ünün işletim sistemi üzerinde nasıl çalıştığı anlatılmaktadır

.Net Core’u bağlantıdan indirebilirsiniz.[1]

* 1. React.js Kütüphanesi

React; Facebook tarafından üretilmiş, kullanıcı ara yüzleri oluşturmak için kullanılan bir javascript kütüphanesidir. React bir framework değil, bir kütüphanedir. Bunun nedeni, React sadece view (görünüm/sayfa) katmanına odaklanır. Bunun dışında herhangi bir mimari katman ile bir ilgisi yoktur. React’ın tek görevi; ara yüzdeki bütün işlemleri en mantıklı, en kolay, en maliyetsiz, en performanslı yolla yapıyor olmaktır. React’ı biraz daha teknik açıdan tanımlarsak; React, component (element) bazlı, Virtual DOM (Sanal DOM) mimarisini kullanan bir frontend kütüphanesidir.[8]



Şekil .2. Reactjs Özellikleri.

Sürücü kursu yönetim paneli reactjs ve redux kullanılarak geliştirilmiştir.

* + 1. Redux

Redux, Dan Abramov ve Andrew Clark tarafından geliştirilen bir yönetim

kütüphanesidir. Redux’la birlikte uygulamanız dahilinde hem veri yönetimini yapabilir hem de ui tarafını yönetebilirsiniz. Böylelikle Redux kullanımıyla birlikte, zamanla uygulama üzerindeki hakimiyetinizi rahatlıkla koruyabilirken, kapsamı gittikçe büyüyen uygulamaları da çok daha rahat yönetmeniz mümkün olabilir.

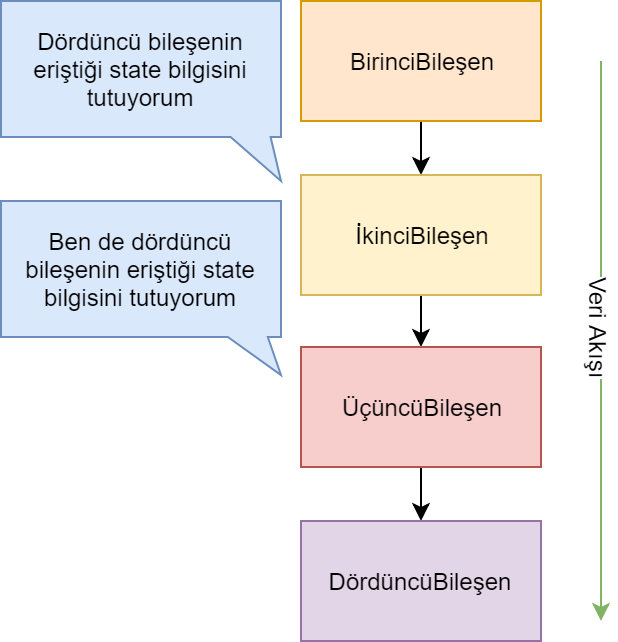
Redux’la kullanacağımız bazı nesneler ya da kavramlar var. Öncelikle

onlardan kısaca bahsetmek gerekebilir. Mantığını anlayabilmemiz için ihtiyacımız olan üç ana kavram; “store”, “reducer” ve “action” diyebiliriz.

**store**: Redux kütüphanesiyle birlikte yaratacağımız verilerin tutulacağı alan diyebiliriz. Uygulamamızda tek bir adet “store”umuz olacak ve state’lerimizi bu store’un içinde depolayacağız.

**reducer**: Action’dan gelen verileri süzgeçleyip store’da belirtilen veriyi güncellememizi sağlayan bir araç.

**action**: Reducer’lara ulaşarak onları tetikleyen, store’daki güncellemesi gereken veriyi yollan bir araç. Neyi güncellemesi gerektiğini adlandırırken “type” ile belirtmemiz gerekirken, değiştirmesi gereken veriyi payload’larla taşır.[9]



Şekil .3. Redux Veri Akışı.

* 1. SQL Veri Tabanı

Veri, bilgisayar ortamına aktarılan işlenmemiş bilgilerdir. Bilgisayara girilen ve bilgisayar tarafından saklanabilen her şeye veri denilebilir. Veri saklamaktaki amaç, daha sonra saklanan verilere yeniden ulaşabilmek ve kullanabilmektir.

RAM bellekte saklanan veriler anlık işlemler için tercih edilir ve bilgisayarın gücü kesildiğinde veriler silinecektir. Bu nedenle programlarım verileri Harddisk adı verilen donanımda saklanmalıdır. Harddiskler herhangi elektrik enerjine ihtiyaç duymadıklarından güç kesilmesi durumunda veriler korunmaktadır.

İlişkisel veritabanı yönetim sistemlerine örnek olarak SQL Server, Oracle, MYSQL gösterilebilir.

Bizim projemizde Microsoft SQL Server veritabanı yönetim sistemi tercih edilmektedir. Tercih etmede ana nedenimiz yazılım geliştirdiğimiz asp.net platformunun mssql ‘e tam olarak destek vermesidir. Detaylı bilgi için üçüncü bölüme bakabilirsiniz.[10]

* + 1. Entity Framework

Entity Framework ORM(Object Relational Mapping) araçlarındandır. ORM; ilişkisel veri tabanı ile nesneye yönelik(OOP) arasında köprü görevi gören araçtır. Bu köprüyle ilişkisel veritabanındaki bilgilerimizi yönetmek için nesne modellerimizi kullandığımız yapıdır. Özetle, veritabanına bizim nesnelerimizi bağlayarak bizim için veri alışverişi yapan Microsoft tarafından geliştirilmiş bir faramework’tür.

Projemizde entity framework kullanılmıştır.Projede Entity Framework Code-First kullanılmıştır.Bu sayede veritabanı ile boğuşmadan sadece kod üzerinden veritabanı sınıfları ,tablolar arasındaki ilişkeleri oluşturup bunları yönetmemiz sağlanmıştır.[11]

* 1. Visual Studio

Visual Studio, Microsoft tarafından üretilen ve konsollar, grafik kullanıcı

arayüzleri, Windows formları, Web servisleri ya da Web uygulamaları oluşturmak için kullanılan bir IDE’dir. Visual Studio programı içerisinde yalnızca Microsoft Windows tarafından desteklenen yerel kodlar kullanılmaktadır.

Visual Studio’nun bir nevi yazılım üretmek için yazılım olduğunu

söyleyebiliriz. Visual Studio yazılımları elbette kendisi kodlamıyor ancak geliştiricilere ileri seviye özellikleri sayesinde yardımcı olarak daha kısa süre içerisinde programlarını hazırlama imkanı sunuyor. Visual Studio programını kullanarak bilgisayar yazılımları, web uygulamaları ve web servislerini çok daha hızlı ve kolay bir şekilde hazırlayabilirsiniz.

Visual Studio son derece fonksiyonel ve şık ara yüz tasarımına sahiptir. Bu

fonksiyonel arayüzle birlikte Visual Studio yazılımcıların yükünü hafifletecek bir kod editörü, debugger, GUI tasarlama aracı, veri tabanı şema tasarım aracı ve öncül revizyon kontrol sistemlerini sunuyor. Visual Studio yazılımın ticari amaçla kullanılan ticari program şeklinin yanı sıra topluluk bazlı olarak geliştirilen ücretsiz sürümü de bulunmaktadır.[12]

* 1. Versiyon Kontrol Sistemleri

Versiyon kontrol sistemleri yazılım geliştirirken aynı anda birden fazla

yazılımcının birbirlerinin yaptığı değişikliklerden etkilenmeden kod yazmasını sağlayan sistemlerdir.Aynı zamanda uygulamayı eski bir kod sürümüne çok rahat bir şekilde geçirmeyide sağlamaktadırlar. Bu ve benzeri sebeplerden dolayı günümüz yazılım geliştirme süreçlerinin en önemli parçalarından biridir. Bu projede github ve tfs versiyon kontrol sistemi kullanılmıştır.

* + 1. Git – Github

Git , yazdığımız projeleri ve uygulamaları, bilgisayarımızda ya da harici

disklerde değilde internet üzerinde tutmamızı ve yönetmemizi sağlayan bir versiyon kontrol sistemidir.

Git, GNU Genel Kamu Lisansı’nın 2. sürümüyle lisanslanmış bir özgür

yazılımdır .

Git ,Linus Torvalds tarafından tasarlanıp geliştirilmiştir.

GitHub, sürüm kontrol sistemi olarak Git kullanan yazılım projeleri için bir

depolama servisidir. GitHub özel depolar için ücretli üyelik seçenekleri sunarken, açık kaynaklı projeler için ücretsizdir. Öğrencilere özel education hesabı ile üye olunursa ücretsiz bir şekilde özel depolara sahip olabilirler.

Şubat 2008'de kurulan GitHub, Nisan 2008'de hizmet vermeye

başlamıştır.Mayıs 2011 tarihinden itibaren GitHub açık kaynaklı(open source) projeler tarafından tercih edilen en popüler depolama servisidir.

Github hesaplarının istatistiklerini bulabileceğiniz , Türkiye’de ve dünya

genelinde farklı programlama dillerindeki sıralamalarınızı aşağıdaki link üzerinden öğrenebilirsiniz.

* + 1. TFS

TFS; takım halinde çalışılan projelerde, takım içerisindeki iletişimi ve verimi

artıracak, projelerin yönetimini kolaylaştıracak bir Microsoft ürünüdür.

Genel amacı iş birliği sağlamak ve ürün gelişimini kolaylaştırmaktır. TFS ile

birlikte ortak server’da tutulan projeye tüm ortak proje üzerinde çalışanlar ulaşabilir ve işlerini buradan görerek diğer takım arkadaşlarının bilgisine sunabilmektedir. Yani farklı geliştirme ortamları üzerinden ekip halinde proje üzerinde çalışılması ve değişiklerin herkese yansıtılması sağlanmaktadır.

TFS, projenizi düzen içerisinde, sağlıklı bir ortam sunarak geliştirmenizi

sağlar. Program her zaman proje geliştiricilerine en güncel hali sunar ve istenildiği taktirde önceki sürümlere ulaşabilme imkanı sağlar.

* 1. Test

Yazılım testi(Software testing) test altında hizmetlerin veya ürünlerin kalitesi

hakkında paydaşlara bilgi sağlamak için yürütülen bir araştırmadır, yazılım testi aynı zamanda, yazılım uygulamalarının risklerini anlamak için yazılımı bağımsız ve nesnel olarak incelemektir. Test teknikleri yazılım böceklerini(hatalar ve diğer kusurlar) bulma niyetiyle uygulama veya bir programı çalıştırma süreçlerini kapsar.[15]

* + 1. Unit Test

Unit test geliştirmiş olduğunuz kodu test etmek için yazdığınız koddur ve esas

olarak kodun veya fonksiyonun belirli bir biriminin davranışını kontrol eder ve geliştirmeyi yapan developer tarafından yazılır.

Bir fonksiyonun birimi bir uygulamanın test edilebilir en küçük parçası

olarakta tanımyalabiliriz aslında. Prosedürel programlamada, bir ünite bütün bir modül olabilir, ancak daha genel olarak tek bir fonksiyon veya prosedürdür. Nesne yönelimli programlamada, bir birim genellikle bir sınıf gibi bütün bir arabirimdir, ancak tek bir yöntem olabilir. Birim testleri, programcılar tarafından veya bazen geliştirme süreci boyunca beyaz kutu test edenler tarafından yaratılan kısa kod parçalarıdır . Bileşen testi için temel oluşturur.[16]

Projede unit test yazılmıştır.Unit Test yazmak için .net core paketlerinden olan xunit paketi kullanılarak yazılmıştır.

[Fact]

**public** **async Task Get\_Course\_Cars()**

{

var list = await \_courseCarAppService.GetListAsync(CourseId);

list.ShouldBeOfType<List<CourseCar>>();

list.ShouldNotBeNull();

}

Yukarıda projede yazılan örnek bir test kodu bulunmaktadır. C# dilinde test methodlarının yazım standartlarında “\_” kullanılması önerilmektedir.

**Get\_Course\_Cars** methodunda , kursa ait arabaların getirilmesi testi yazılmıştır.

**Should** kısmı testi çalıştıran kısımdır.

ShouldBeOfType **‘**list**’** değişkenin List<CourseCar> olup olmadığının kontrolünü yapar.

ShouldNotBeNull kısmında ‘list’ değişkeninin boş olup olmadığını kontrol eder.VS editöründeki testi çalıştır komutu ile bize başarılı yada başarısız olarak sonuç döner buna göre bir başarısız durumunda henüz uygulama kullanıcıya geçmeden bu hatayı düzeltebilme fırsatını yakalamış oluruz. Yazılım geliştirme süreçlerinde TDD(Test Driven Development) yaklaşımı yani önce testin yazılması projelerdeki hata oranını büyük oranda azaltmayı hedeflemektedir.

# EHLİYET TÜRKİYE PLATFORMUNUN TASARIMI

* 1. Giriş

Kursiyerlerin kullanacağı uygulama, admin paneli ve web service .net core 2

framework’ü ile geliştirilmiştir. Sürücü kurslarının önyüzü react.js kütüphanesi ile geliştirilmiştir.

Veritabanı olarak MSSQL kullanılmıştır ve sürücü kurslarının veritabanı ile kursiyer uygulamasının veritabanı farklı olarak 2 adet tasarlanmıştır. Veritabanları Entity Framework Code-First(Önce Kod) yaklaşımı ile hazırlanmıştır. Bu sayede veritabanı bağımlılığı (Mysql,Mssql,Postgresql) kısmen ortadan kaldırılmıştır.

Önyüz tasarımları admin paneli ve kursiyer uygulamasında Bootstrap 4 ile Sürücü kursu platformunda ise Ant Design kütüphaneleri kullanılmıştır.Kursiyer uygulamasının tasarımı önce Photoshop olarak çizilmiştir daha sonra Html,Css(Bootstrap 4) ‘e dökme işlemi gerçekleştirilmiştir.

Versiyon Kontrol Sistemi olarak projede TFS ve GitHub kullanılmıştır.Kursiyer uygulaması TFS ‘de kontrol edilirken diğer 3 uygulama GitHub üzerinden kontrol edilmiştir.

* 1. Kursiyer Uygulaması

Ana Domain’de barındırılarn uygulamadır. Asp.net Core 2 ile geliştirilip,TFS

de kontrol edilmiştir.

Bu uygulama ehliyet almak isteyen kişilerin il,ilçe parametrelerine göre sürücü

kursu aradıkları, buldukları sürücü kurslarının detaylarını görebildiği iletişime geçebildiği , ön kayıt yaptırabildikleri kısımdır.

Kullanıcıların öncelikle sisteme ad-soyad,telefon,email bilgilerini

girerek üye olmaları gerekmektedir.Üye olmadan kullancılar ödeme işlemi veya bir kursa ön kayıt başvurusu yapamaktadırlar.Giriş yapan kullanıcılar anasayfadaki arama butonlarından seçtikleri il , ilçe ye göre sürücü kurslarını arayabilirler.Üyelerin yaptıkları filtreye göre sürücü kursu veritabanında uygun bir sürücü kursu var ise liste ekranına eğer yok ise kullanıcıya bir uyarı gösterilmektedir.Eğer belirlenen mevkiide bir sürücü kursu bulanmadı ise ,üye eğer ister ise karşısına çıkan formdan talep oluşturabilir bu talep karşısında yönetim ekibinden görevlendirelecek kişi veya kişiler bu mevkiideki kurslarla iletişime geçerek onlarıda sisteme katmak üzere çalışma gerçekleştirmektedir.

Seçilen mevkiide eğer bir adet kurs var ise sistem otomatik olarak o kursun

detay sayfasına , 1’den fazla kurs var ise listeleme ekranı açılmaktadır. Açılan listeleme ekranında kurslar hakkında özet olarak nitelendirilebilecek bilgiler yer almaktadır. Kursiyer seçtiği kursun detay sayfasına geldiğinde. Kurs hakkında detaylı bir bilgi ekranı bulunmaktadır.

Kursiyer kurs detay sayfasındaki iletişim bilgilerinden kurs ile iletişime

geçebilir veya ön kayıt yaptırabilir. Ön kayıt işlemi yaptıran kursiyerin bilgileri sürücü kursuna bildirim olarak gönderilir sürücü kursu kendi panelinden gördüğü bilgiler eşliğinde kursiyerle iletişime geçer. Eğer kayıt gerçekleşecek ise sürücü kursu kursiyerin ön kayıt başvurusunu panelinden onay vererek işlemi sonlandırmaktadır.

Kursiyer profil sayfasında kendi bilgilerini görüntüleyebilmektedir. Bunun

haricinde eğer bir kursa kayıt yaptırmış ise o kurs ondan belge(Kimlik fotokopisi vs) talep edebilme ktedir. Talep edilen bu belgeleri kursiyer profilinden sürücü kursuna gönderebilmektedir. Böylece iki taraf içinde bir kolaylık sağlanması planlanmıştır.Profil sayfasında kendi fotoğraflarını yükleyebilmektedirler.

Ders tarihleri sürücü kursu panelinden belirlenmektetir. Belirlenen tarihler

doğrultusunda kursiyer bu bilgileri içeren bildirim almaktadır. Gelen bildirime göre kursiyerler derslere katılım göstermek durumundadır.

Kursiyerler sistem hakkında veya sürücü kursları hakkında herhangi bir

şikayetleri olması durumunda sistem yetkilileri ile iletişime geçerek şikayetini dile getirebilmektedir.

Kursiyerler eğer isterler ise anlaştıkları sürücü kurslarının ödemelerini de bu

sistem üzerinden yapabilmektedirler. Bunun için kredi kartı veya havale eft seçeneği kullanabilmektedir.Ödeme sistemi için PayTR ödeme çözüm yöntemi kullanılmıştır. Ödeme yapılan miktar öncelikle sistem sahibinin PayTR hesabına aktarılmaktadır.Aktarılan bu miktar daha sonra sistem yetkilisi tarafından manuel olarak sürücü kurslarına gerekli komisyon işlemi kesilerek gönderilmektedir. Ödeme işleminde PayTR platformuda yüzde 3 komisyon almaktadır.Ödeme işleminde oluşabilecek bir çok hata test edilip transactionlar ile güvence altına alınmıştır.

Sistemin güvenlik testi bir bilgisayar mühendisine yaptırılmıştır.Verilen rapor

şu şekildedir.

1. XSS

Profil adresinde kullanıcı paneli mevcut ,mevcut olan bu sayfada yapılan “Stored-XSS” açık testleri sonunda açığın mevcut olmadığı tespit edilmiştir.

Sitede kullanılan parametrelerde “Reflected-XSS” saldırısı uygulanamamakta, güvenlik mevcuttur.

Test edilen süre zarfında web sitesine uygulanabilecek bir “Self-XSS” yahut “Blind-XSS” gibi saldırı methodlarına karşı korunaklı olduğu tespit edilmiştir.

1. CSRF

Sitede bilgi girişine izin verilen sayfalarda “\_\_RequestVerificationToken” isimli bir token mevcuttur.Bu Token,CSRF saldırılarına karşı önlem niteliği taşımaktadır.Sonuç olarak web sitesi olabilecek ktirik CSRF saldırılarına karşı korunaklı olduğu tespit edilmiştir.

1. Parameter Tempering

Web sitesi genelinde kullanılmakta olan parametrelerin değiştirilmesine izin vermemektedir.

1. Url Redirections

Web sitesi içerisinde yer alan, URL Redirections parametrelerin manipüle edilmesine karşın güvenlik mevcuttur. Herhangi bir şekilde parametreler değiştirilmemektedir.

1. Local File Inclusion

Web sitesi içerisinde bulunan sadece yetkililerin erişim sağladığı dosyalara herhangi bir erişim sağlanmamaktadır. Sunucu içerisindeki dosyalar herhangi bir saldırıya karşı güvenli görünmektedir.

1. File Uploading

Web sitesi içerisindeki dosya yükleme kısımlarında saldırganların herhangi bir “shell” atmasına karşın korunaklı olduğu tespit edilmiştir.

1. Host Header Attacks

Web sitesi genelinde Host-Header saldırıları mevcut değildir. Herhangi bir deneme yapıldığında sayfa tarafından“ Bad-Request” verilip bunun önüne geçilmiştir.

1. SQL Injection

Kontrol edilen sayfalarda herhangi bir SQL Injection uygulanabilecek bir parametreye rastlanmamıştır.

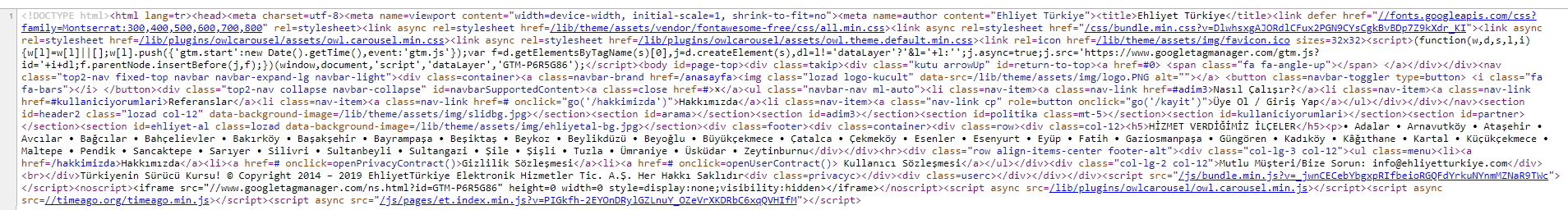
Web sitesi bilgim dahilinde gerçekleştirilen SQL Injection saldırılarına karşı korunaklı olduğu tespit edilmiştir.

* + 1. Seo

Arama Motoru Optimizasyonu (SEO), Web sitelerinin arama motorlarındadaha iyi performans göstermesi için yapılan çalışmaların tümüne verilen isimdir.[17]

Web sitesinde seo konusunda çalışmalar yapılmıştır. Sitenin kaynak

kodlarında sıkıştırma işlemi gerçekleştirilmiştir böylece KB olarak tasarrufa gidilip sitenin daha hızlı açılması sağlanmıştır.



Şekil .4. Sıkıştırılmış Html Görüntüsü.

Web sitesindeki resimlerin lazy-load(gecikmeli yükleme) ile sayfanın kodları

yüklendikten sonra yüklenmesi sağlanmıştır.Bu sayede boyutu büyük olan resim dosyalarının siteyi yavaşlatmasının önüne geçilmiştir. Düşük internet hızına sahip kullanıcıların mağdur olmaması hedeflenmiştir.

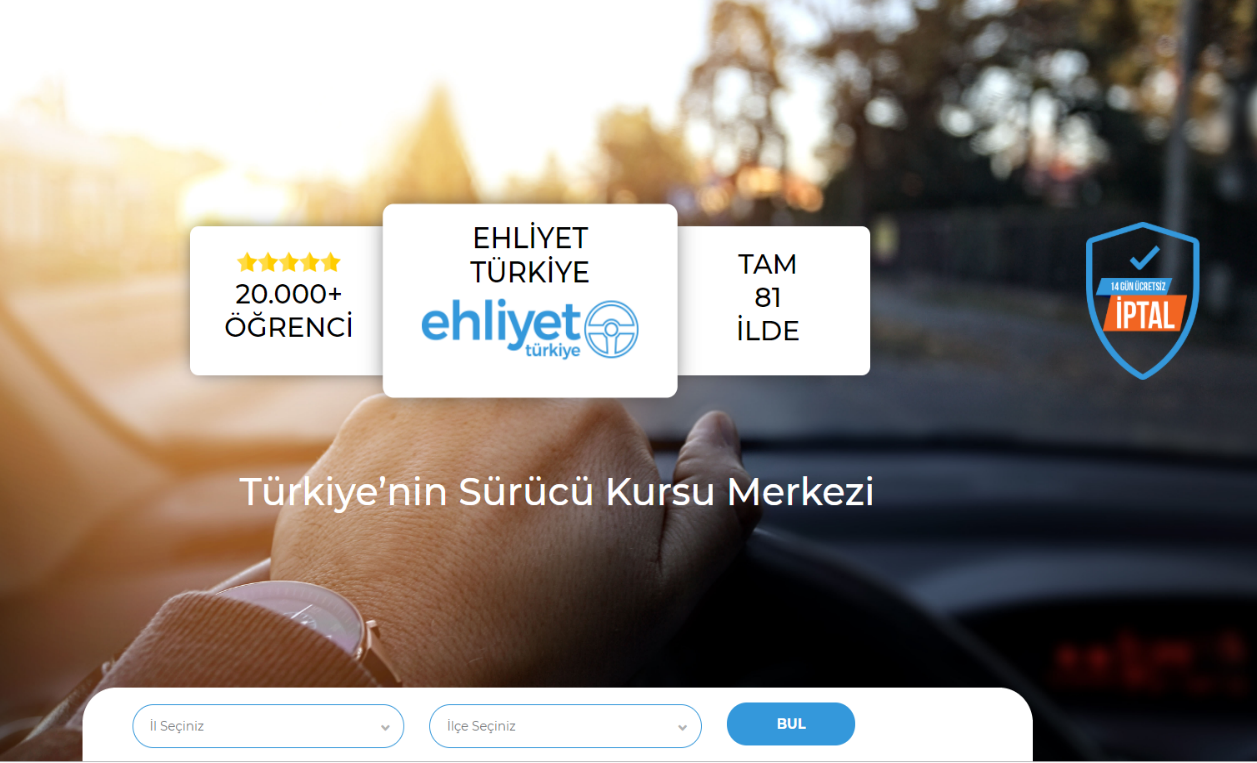
Web sitesinde mobil uyumluluk konusuna dikkat edilmiştir. Bootstrap4

sayesinde site tamamen tüm mobil cihazlara(ios,android) uyumlu ve kullanışlı olarak geliştirilmiştir. 2018 yılında yapılan araştırmalara göre Türkiye’de internet kullanıcıların yüzde 36’sı Bilgisayar yüzde 62’si ise mobil cihazlardan girmektedir.[18]

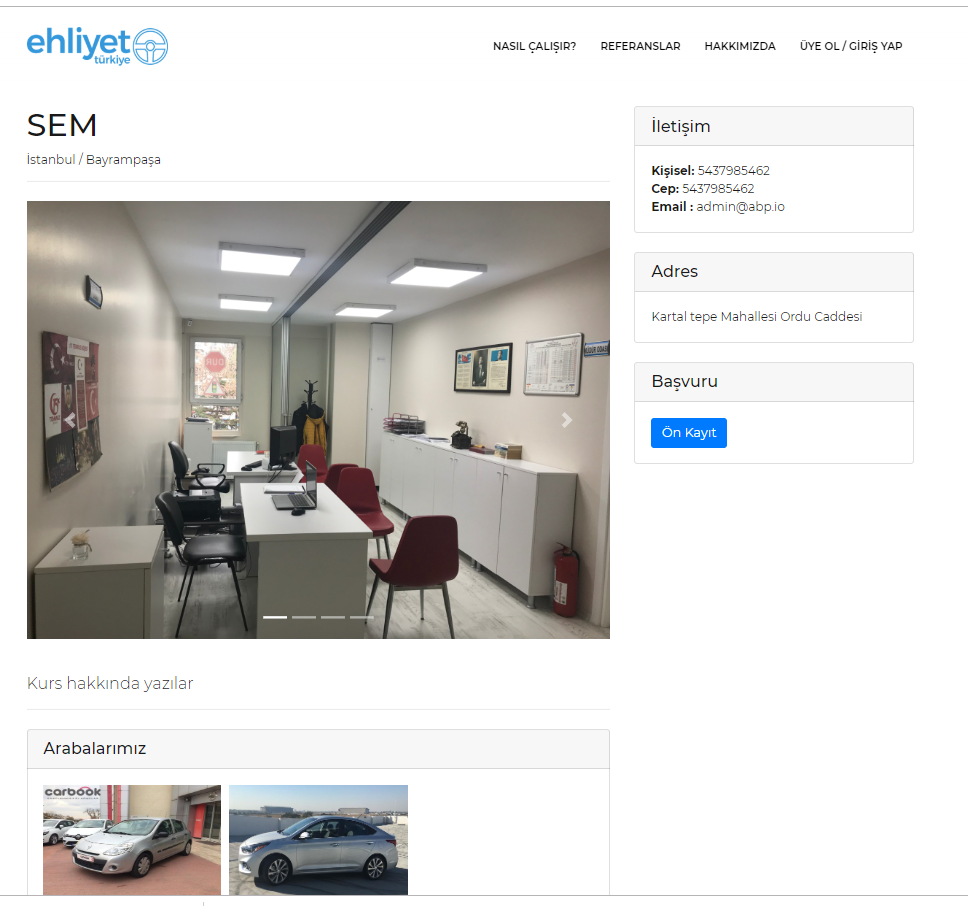
Web sitesinde genel seo taglarına dikkat edilmiştir.Yapılan analizlerde Seo

puanı ortalamanın üstündedir. Bu sayede arama motorlarında (Google,Yandex) üst sıralarda çıkılması hedeflenmektedir.

İlerleyen zamanlarda projenin google arama motorunda daha üst sıralarda gözükmesi için google adwords platformuna reklam verilmesi planlanmaktadır.



Şekil .5. Kursiyer Uygulaması.



Şekil .6. Kurs Detay Sayfası.

* 1. Admin Paneli

Admin paneli bir subdomain olarak barındırılmaktadır. Kursiyer

uygulaması ile aynı veritabanını kullanmaktadır.Admin panelinin kodlanmasında .net core ile hazırlanmış bir framework olan abp.io framework’ü kullanılmıştır.Abp.io framework’ü .net dünyasında oldukça popular bir hale gelmiştir, ve şirketin sahibi Sakarya Üniversitesi’nden mezun olan Halil İbrahim Kalkan’dır.Admin bir nevi sistemin sahibi olmaktadır.Kursiyer uygulamasındaki üye olan kişilere admin rolü verilerek bu sistem kullanımlarına açılmaktadır.Admin olan kişilerin yetkileri şu şekildedir;

Sistemdeki üye sayısını üyelerin bilgilerini görebilir ve aynı zamanda bu

bilgileri değiştirme yetkisine de sahiptirler.Bir üyeyi sistemden yasaklayabilir.Üyeler ile iletişime(mail) de geçebilmektedir.

Sistemde varolan kayıtlı sürücü kurslarını görüntüleyebilir , bu sürücü

kurslarının bilgilerini değiştirebilir. Yeni kayıt için başvuran sürücü kurslarını görebilir ve bu kurslar ile iletişimlerin ardından onay işlemi veya red işlemini gerçekleştirebilir.Admin tarafından onaylanan sürücü kursuna veritabanında bir üyelik açılıp bunun bilgileri başvuruda belirtilen yerlere bildirim olarak gönderilmektedir.

Sitenin teknik ayarlarını, mail ayarlarını , seo ayarlarını değiştirebilmektedir.

Gerekdiği durumlarda site üzerinden sürücü kurslarına bildirimler gönderebilmektedir.

Admin panelinde ödeme ekranlarıda mevcuttur.Kullanıcıların yaptığı ödemeler admin kontrolünden geçmektedir.Kontrolünden geçen ödemeler onaylandıktan sonra sürücü kurslarına gerekli pay alındıktan sonra aktarılmaktadır.

* 1. Sürücü Kursu Uygulaması

Sürücü kursu uygulaması, projenin en kapsamlı olan uygulamasıdır. Bu

uygulamanın önyüzü son dönemlerde popüler olan spa kütüphanelerinden reactjs kütüphanesi ile geliştirilmiştir.Projenin backend denilen veritabanı vs. işlerin yapıldığı arka tarafı ise .net core 2 ile geliştirilmiştir.Backend projesi katmanlı mimari yapısına uygun şekilde geliştirilmiştir.Bu şekilde geliştirilmenin amacı projenin daha rahat kontrol edilmesi yeni özelliklerin sisteme daha kolay şekilde entegre edilmesini sağlamak ve daha profesyönel bir kod yazım şekli oluşturmaktır. Backend bir webservice projesidir. Projenin backend kısmına bundan sonra api denilecektir. Reactjs uygulaması backend ile api üzerinden http istekleri ile iletişime geçmektedir.Uygulamada herhangi bir ekranın işleyişi şu şekilde olmaktadır.Öncelikle ekranın tasarımı reactjs uygulaması ile geliştirilir.Burada teknik olarak reactjs durumlarını yönetmek için neredeyse tüm reactjs uygulamalarında kullanılan redux kütüphanesi kullanılmıştır.Redux kütüphanesi sayesinde global state yönetimi yapılabilmektedir.Tasarımı yapılan ekranın ardınan redux kodları yazılmaktadır. Api ile iletişime geçerek sayfanın işlevselliği sağlanır.Bu sayede api projesi ve önyüz projesi tamamen birbirinden bağımsız bir hale getirilmiştir.Birinde oluşan hata diğer projenin nispeten çalışmasına engel olmamaktadır.Günümüz web yazılım dünyasında projeler artık herhangi bir spa framework(react,angular,vue) ve bir backend tarafı olacak şekilde geliştirilmektedir.Sürücü kursu uygulaması admin paneli gibi bir subdomain de tutulmaktadır.Ön yüz uygulamasının tasarımı isomorphic tema olarak adı geçen temelinde ant.design kurallarını baz alan bir tasarımdır. Bu uygulamada sürücü kurslarının her türlü isteğine cevap verebilmek hedeflenmiştir. Uygulamayı kullanabilmek için öncelikle admin panelinden sürücü kurs başvurunuzun onaylanıp bir şifre edinmeniz gerekmektedir.

Sürücü kursları kendisine kayıtlı olan öğrencileri görebilmekte,bu öğrencilerle iletişime geçebilmekte ,öğrencilerin belgelerini görüntüleyebilmekte,belge talep edebilmekte ve artık geçmişte kalan öğrenciler ile ilişkiyi kesebilmektedir.Sürücü kursu eğer isterse kursiyer uygulamasına kayıt olmayan kişileride öğrencisi olarak sisteme ekleyebilmektedir, bu şekilde eklene kişiler kursiyer uygulamasına giriş yapamazlar ama panelde gözükmektedir.

Sürücü kursu kendisine ön kayıt yaptıran kişileri görebilir bunlarla iletişime geçtikten sonra bunları onaylayabilir.Onay işleminin ardından öğrenciler kısmında bu kişiyi görebilmektedir.Ön kayıt yaptıran kişiler onay/red işleminin ardından bildirim ile haberdar edilmektedir.

Sürücü kursları panelde bulunan notlar bölümünden işleri ile alakalı herhangi bir konuda notlar alıp onları görebilmektedir.

Sürücü kursları paneldeki takvim bölümünde öğrencilere dersler atamaktadır.

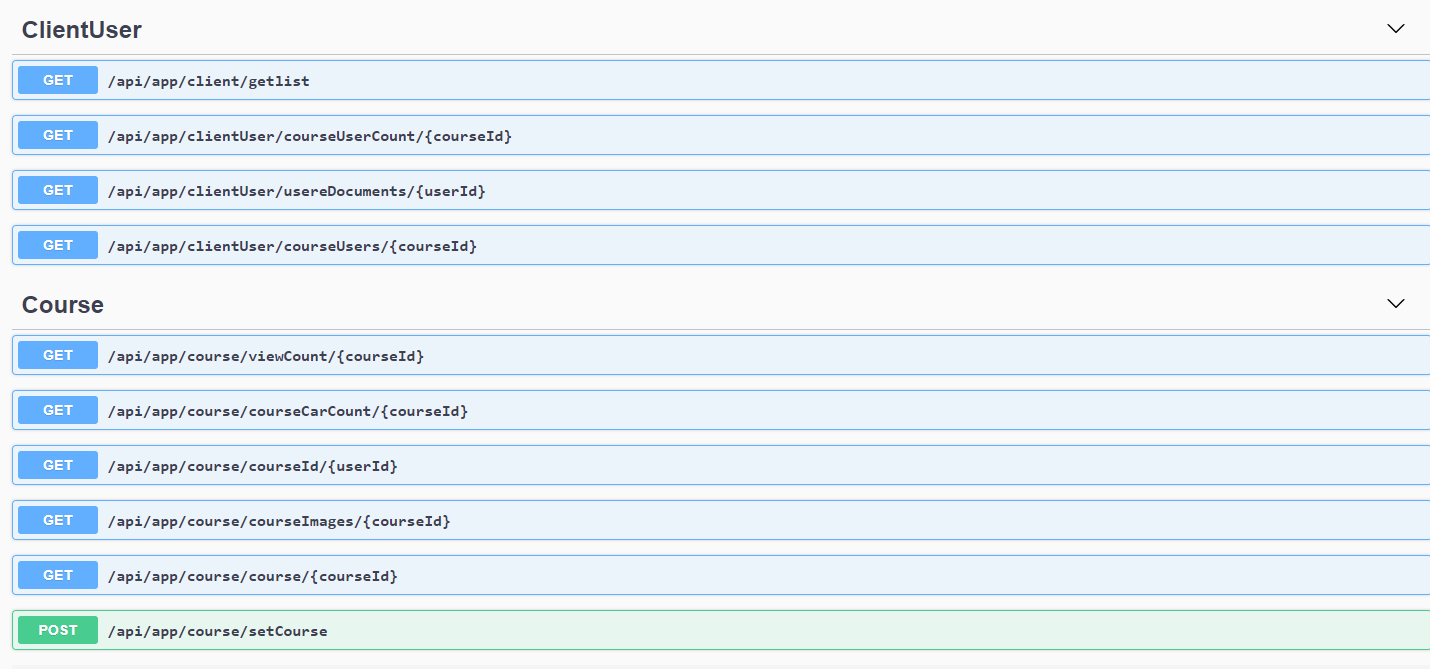
Öncelikle saat aralığı seçilmektedir. Seçilen saat aralığında örneğin saat 15:00-18:00 aralığında hangi öğrencilerin derse çağırılacağı açılan menüden en az 1 öğrenci olacak şekilde seçilmelidir , ilgili başlık ve açıklama kısmı doldurulduktan sonra işlem tamamlanmalıdır.Tamamlanan bu işlemin saati açıklaması seçilen öğrencilere bildirim olarak gitmektedir.

Sürücü kursu panelden, kursiyer uygulamasındaki kurs detay sayfasını

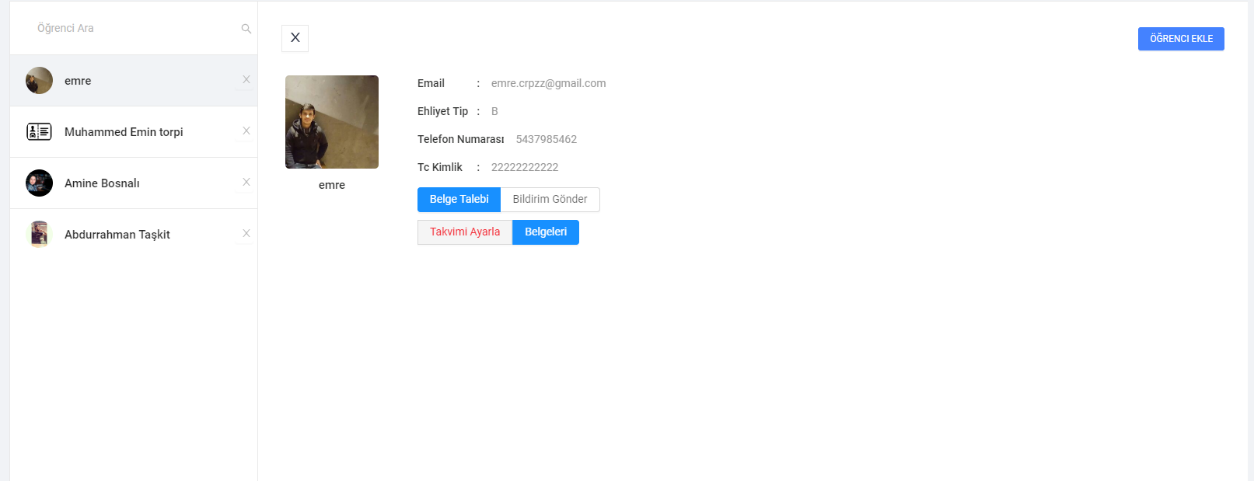
tasarlayabilmektedir. Hakkında yazısını ,kurs resimlerini, kursa ait olan arabaları ekleyip düzenleyebilmektedir.Kurs iletişim ayarlarını yine sistem üzerinden düzenleyebilmektedir.

Sürücü kursu panelden admin tarafından gelen bildirimleri okuyabilmektedir. Sürücü kursu panelinden yüklenen dosyaların hepsi bir subdomain altına yüklenmektedir. Kodların olduğu hosting ile , dosyaların olduğu hosting dosyalar cdn. Öneki ile başlayan bir domaine yüklenmiştir, buradaki amaç daha yönetilebilir, daha düzenli bir işleyiş olmasıdır.

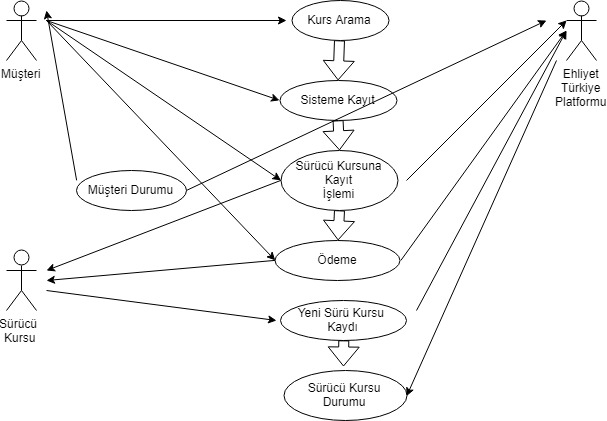
Ön yüz projesinin api ile daha rahat anlaşılır bir iletişim kurması için yazılan .net methodlarının gereken parametreleri , methodların dönüş değerlerini gösteren örnek istekler atılabilmesini sağlayan swagger kütüphanesi kullanılmıştır. /swagger url ine giderek yazılan bütün methodlar bir tasarıma dökülmüş halde bulunmaktadır.



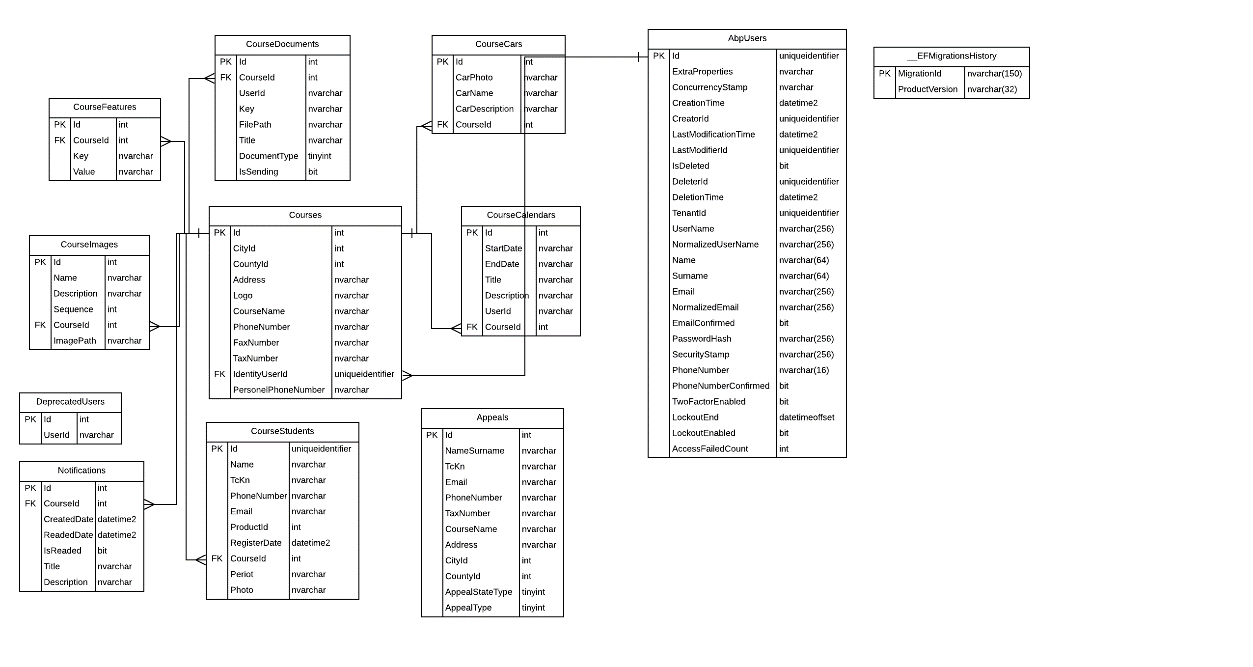
Şekil 1.7. Swagger Api Sayfası.



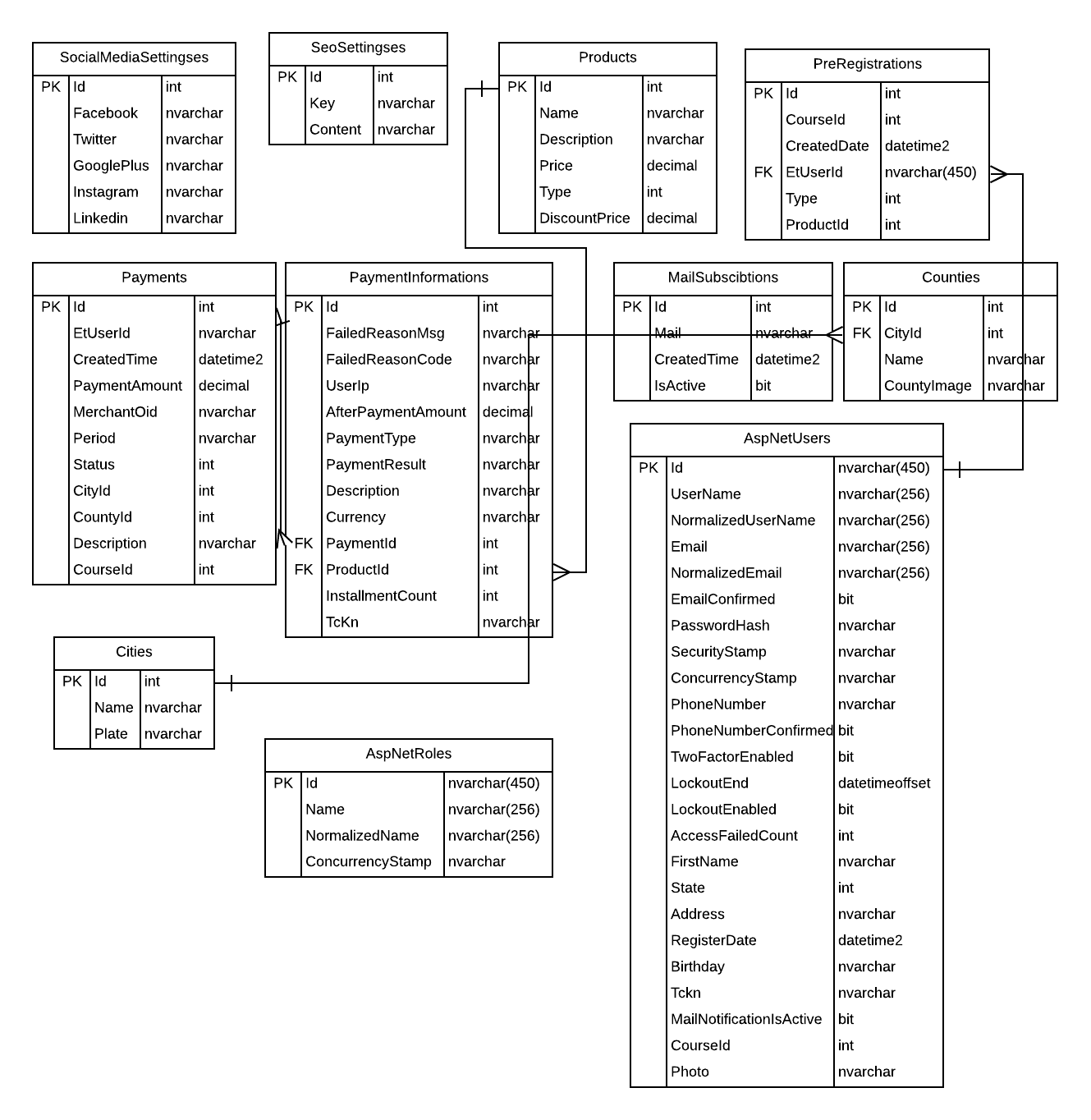
Şekil 1.8. Öğrenciler Sayfası.

****

Şekil .9. Use Case Diyagramı.

****

Şekil 2.1. Sürücü Kursu Veritabanı Şeması.

****

Şekil 2.2. Kursiyer Uygulaması Veritabanı Şeması.

# SONUÇLAR

Projede, ehliyet alma sürecine yönelik zaman tasarrufu ve kolaylık sağlayabilecek bir inovasyon geliştirildi. Sürücü kurslarının ve ehliyet sahibi olmak isteyen insanları buluşturan Ehliyet Türkiye platformuyla aşağıdaki sonuçlara ulaşılmıştır:

Ehliyet alma sürecinde ki işlemler tespit edilerek, bu sürecin online biçimde nasıl geliştirilebileceği araştırılmıştır.

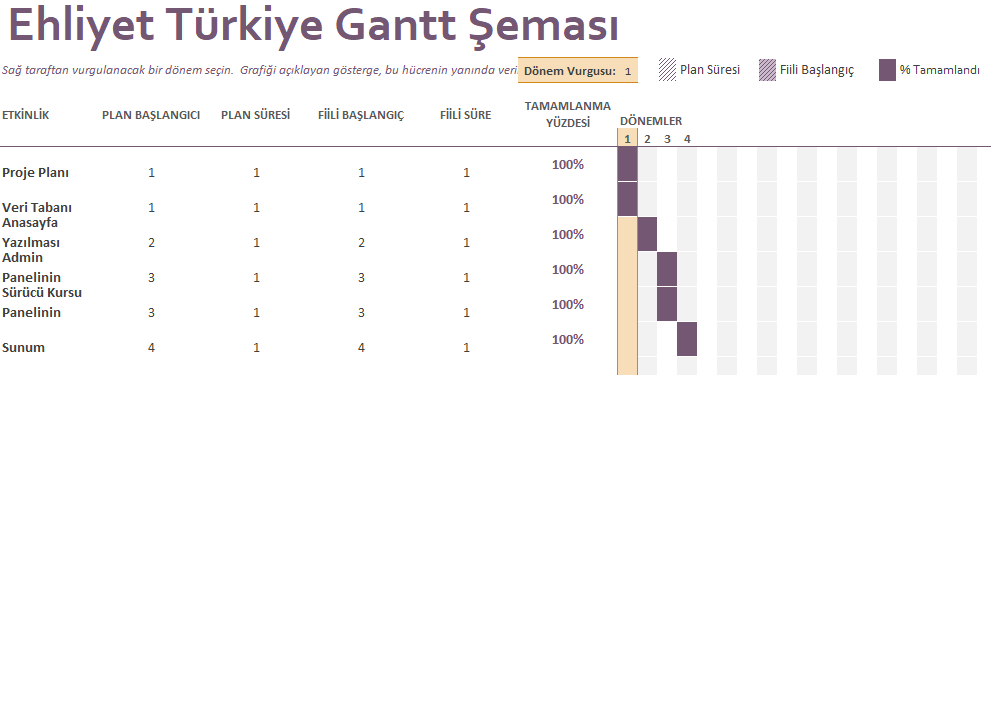
* Ülkemizde fazla sayıda sürücü kursunun müşterilerine online ehliyet sınavları için web sitesi kullandıkları tespit edilmiştir.[16]
* Ehliyet almak isteyen bireyin kendisine en yakın kursu bulmak için uzun araştırma süresine ihtiyacı olduğu gözlemlenmiştir.
* Ehliyet Türkiye platformunun hızlı ve minimum güvenlik zafiyetiyle geliştirilmesi için ASP.NET Core, React.js teknolojileri ve SQL SERVER veritabanı yönetim sistemi tercih edilmiştir.
* Ehliyet Türkiye platformu üç ayrı projeye ayrılarak geliştirilmiştir.
* Katmanlı mimarinin daha efektif kullanılabilmesi ve projenin daha kolay geliştirilebilmesi için abp.io framework’ü kullanılmıştır.
* Geliştirme sürecinde Github,TFS tercih edilmiştir.
* Proje, ehliyetturkiye.com üzerinden hizmete sunulması planlanmaktadır.

**KAYNAKLAR**

|  |  |  |
| --- | --- | --- |
| [1] |  | <https://www.konseptprojeler.com/turkiyenin-en-eglenceli-ofisinin-oykusu-yemeksepeti-com> |
| [2] |  | <https://www.biyografi.info/kisi/nevzat-aydin> |
| [3] |  | <https://www.wikiwand.com/tr/Otomasyon> |
| [4] |  | <https://gelecegiyazanlar.turkcell.com.tr/konu/c-sharp/egitim/c-101/c-dilinin-tarihcesi>  <http://www.csharpnedir.com/articles/read/?id=922> |
| [5] |  | <https://webmaster.kitchen/nedir-bu-asp-net/> |
|  |  |  |
| [6] |  | <https://www.iztim.com/blog/yazilimteknolojisi/mvc-nedir?page=2s> |
| [7] |  | <http://devnot.com/2017/asp-net-core-mvc-ye-giris/> |
|  |  |  |
| [8] |  | <https://www.mediaclick.com.tr/blog/react-js-nedir> |
|  |  |  |
| [9] |  | <https://medium.com/@erdemuslu/redux-nedir-nas%C4%B1l-kullan%C4%B1l%C4%B1r-b%C3%B6l%C3%BCm-1-4dad7a0f3b57>s |
|  |  |  |
| [10] |  | <https://smartpro.com.tr/sql-veritabani-nedir/> |
| [11] |  | <https://medium.com/t%C3%BCrkiye/entity-framework-nedir-bff3943bec72> |
|  |  |  |
| [12] |  | <https://wmaraci.com/nedir/visual-studio> |
|  |  |  |
| [13] |  | <https://medium.com/@halilozel1903/git-github-ve-gitlab-nedir-e51a87e8e4e7> |
|  |  |  |
| [14] |  | <https://emrullahmert.wordpress.com/tag/tfs-nedir/> |
|  |  |  |
| [15] |  | <https://www.wikiwand.com/tr/Yaz%C4%B1l%C4%B1m_testi> |
|  |  |  |
| [16] |  | <https://medium.com/@resulzel/unit-test-nedir-ff353aac240a> |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**EKLER**

**EK A:** Gantt Şeması

****

**ÖZGEÇMİŞ**

Emre KARA, 26.11.1995 de İstanbul’da doğdu. İlk, orta ve lise eğitimini Bayrampaşa’da tamamladı. 2015 yılında Tuna Lisesi’nden mezun oldu. 2015 yılında Sakarya Üniversitesi Bilgisayar Mühendisliği Bölümü’nü kazandı. 2017 yılında yazılım stajını Teknoliva Ltd. Şti. de ve 2018 yılında da Win Young Ltd. Şti. ‘de donanım stajını yapmıştır. Şu anda SAÜ Bilgisayar Mühendisliği son sınıf öğrencisidir.

Makbule Amine BOSNALI, 02.06.1997 de Afyon’da doğdu. İlk, orta ve lise eğitimini Bolvadin’de tamamladı. 2015 yılında Açık Öğretim Lisesi’nden mezun oldu. 2015 yılında Sakarya Üniversitesi Bilgisayar Mühendisliği Bölümü’nü kazandı. 2017 yılında yazılım stajını Altı Harf Ltd. Şti. de ve 2018 yılında da Win Young Ltd. Şti. ‘de donanım stajını yapmıştır. Şu anda SAÜ Bilgisayar Mühendisliği son sınıf öğrencisidir.

**BSM 498 BİTİRME ÇALIŞMASI**

**Değerlendİrme ve Sözlü Sınav Tutanağı**

KONU : SÜRÜCÜ KURSU OTOMASYONU

ÖĞRENCİLER (Öğrenci No/AD/SOYAD): G151210097/EMRE/KARA

G151210055/MAKBULE AMİNE/BOSNALI

|  |  |  |  |
| --- | --- | --- | --- |
| Değerlendirme Konusu | İstenenler | Not Aralığı | Not |
| **Yazılı Çalışma** |  |  |  |
| **Çalışma klavuza uygun olarak hazırlanmış mı?** | x | 0-5 |  |
| **Teknik Yönden** |  |  |  |
| **Problemin tanımı yapılmış mı?** | x | 0-5 |  |
| Geliştirilecek yazılımın/donanımın mimarisini içeren blok şeması (yazılımlar için veri akış şeması (dfd) da olabilir) çizilerek açıklanmış mı? |  |  |  |
| Blok şemadaki birimler arasındaki bilgi akışına ait model/gösterim var mı? |  |  |  |
| Yazılımın gereksinim listesi oluşturulmuş mu? |  |  |  |
| Kullanılan/kullanılması düşünülen araçlar/teknolojiler anlatılmış mı? |  |  |  |
| Donanımların programlanması/konfigürasyonu için yazılım gereksinimleri belirtilmiş mi? |  |  |  |
| UML ile modelleme yapılmış mı? |  |  |  |
| Veritabanları kullanılmış ise kavramsal model çıkarılmış mı? (Varlık ilişki modeli, noSQL kavramsal modelleri v.b.) |  |  |  |
| Projeye yönelik iş-zaman çizelgesi çıkarılarak maliyet analizi yapılmış mı? |  |  |  |
| Donanım bileşenlerinin maliyet analizi (prototip-adetli seri üretim vb.) çıkarılmış mı? |  |  |  |
| Donanım için gerekli enerji analizi (minimum-uyku-aktif-maksimum) yapılmış mı? |  |  |  |
| Grup çalışmalarında grup üyelerinin görev tanımları verilmiş mi (iş-zaman çizelgesinde belirtilebilir)? |  |  |  |
| Sürüm denetim sistemi (Version Control System; Git, Subversion v.s.) kullanılmış mı? |  |  |  |
| Sistemin genel testi için uygulanan metotlar ve iyileştirme süreçlerinin dökümü verilmiş mi? |  |  |  |
| Yazılımın sızma testi yapılmış mı? |  |  |  |
| Performans testi yapılmış mı? |  |  |  |
| Tasarımın uygulamasında ortaya çıkan uyumsuzluklar ve aksaklıklar belirtilerek çözüm yöntemleri tartışılmış mı? |  |  |  |
| **Yapılan işlerin zorluk derecesi?** | x | 0-25 |  |
| **Sözlü Sınav** |  |  |  |
| **Yapılan sunum başarılı mı?** | x | 0-5 |  |
| **Soruları yanıtlama yetkinliği?** | x | 0-20 |  |
| **Devam Durumu** |  |  |  |
| **Öğrenci dönem içerisindeki raporlarını düzenli olarak hazırladı mı?** | x | 0-5 |  |
| **Diğer Maddeler** |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| **Toplam** |  |  |  |

Danışman (Jüri adına):

danışman imzası: