

**PAMUKKALE ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ**

OYUN ARKADAŞIM

LİSANS TEZİ

**Berkay ŞENOL
(16253065)**

Bilgisayar Mühendisliği

Tez Danışmanı: Öğr.Gör. Şevket Umut Çakır

OCAK 2021

Pamukkale Üniversitesi Bilgisayar Mühendisliği Bölümü, 16253065 numaralı Lisans Öğrencisi Berkay Ş ENOL, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “OYUN ARKADAŞ IM” başlıklı tezini aşağıdaki imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Öğr. Gör. Şevket Umut Çakır**
Pamukkale Üniversitesi

Jüri Üyeleri : **Prof. Dr. Sezai TOKAT**
Pamukkale Üniversitesi

Dr. Öğrt. Ü. Gökçen YILMAZ
Pamukkale Üniversitesi

Teslim Tarihi : **24 Ocak 2021**
Savunma Tarihi : **19 Ocak 2021**

Önsöz

Bu çalışmayı yaparken bana desteğini esirgemeyen tez danışmanım Öğr.Gör. Şevket Umut ÇAKIR'a teşekkür ederim.

19 Ocak 2021

Berkay ŞENOL

İçindekiler

Sayfa

ÖNSÖZ.....	v
İÇİNDEKİLER.....	vii
ÇİZELGE LİSTESİ	xi
ŞEKİL LİSTESİ	xiii
ÖZET.....	xv
SUMMARY	xvii
1. GİRİŞ	1
1.1 Tezin Amacı.....	1
1.1.1 Tezin ikincil amaçları.....	1
1.2 Literatür Araştırması.....	1
1.2.1 Oyuncuilan.com.....	1
1.2.2 Duobul.com.....	2
1.2.3 Discord Kanalları.....	3
2. UYGULAMANIN VERİTABANI VE MODELİ.....	5
2.1 IR Diagramı	5
2.2 Veritabanı Tabloları	5
2.3 Prosedürler.....	5
2.4 Model.....	6
3. UYGULAMA EKРАНLARI ve LAYOUTLAR.....	7
3.1 Üye Layout	7
3.2 Profilim Layout.....	7
3.3 Admin Layout	7
3.4 Kayıt Ekranı.....	8
3.5 Üye Giriş Ekranı	8
3.6 Üye Anasayfa Ekranı	8
3.7 Profilim Ekranı	9
3.8 Profilim Ekranı	9
3.9 Profilim - Oyun Profilleri Ekranı.....	10
3.10 Profilim - Oyun Profilleri Oluşturma Ekranı	10
3.11 Profilim - Oyun Profilleri Düzenleme Ekranı.....	11
3.12 Gönderilen Mesajlar Ekranı.....	12
3.13 Gelen Mesajlar Ekranı	12
3.14 Üye Arama Ekranı	12
3.15 Oyun Profili Bulma - LoL Profili Bulma Ekranı	13
3.16 Oyun Profili Bulma - CS:GO Profili Bulma Ekranı	13
3.17 Oyun Profili Bulma - Valorant Profili Bulma Ekranı.....	14
3.18 Üye'ye Mesaj Gönderme Ekranı	15

3.19 Üye Profil Ekranı.....	15
3.20 Üye Oyun Profili Ekranları.....	15
3.21 Sıralamalar - Lol Profilleri Sıralama Ekranı.....	16
3.22 Sıralamalar - CS:GO Profilleri Sıralama Ekranı	17
3.23 Sıralamalar - Valorant Profilleri Sıralama Ekranı	17
3.24 Yönetici Giriş Ekranı	17
3.25 Admin Paneli - Kullanıcılar Ekranı	17
3.26 Admin Paneli - Üye Güncelleme Ekranı	18
3.27 Admin Paneli - Oyun Profilleri Ekranı	18
3.28 Admin Paneli - Oyun Profili Düzenleme Ekranı	20
3.29 Admin Paneli - Gelen Mesajlar Ekranı.....	20
3.30 Admin Paneli - Gönderilen Mesajlar Ekranı	21
3.31 Admin Paneli - İstatistikler Ekranı.....	21
4. CONTROLLERS.....	23
4.1 Login Kontroller	23
4.1.1 Giriş Yap	23
4.1.2 Admin Giriş Yap.....	23
4.1.3 Çıkış Yap.....	24
4.2 Kayıt Ol Kontroller.....	24
4.3 Anasayfa Kontroller.....	24
4.4 Üye Kontroller	24
4.4.1 Admin Paneli Üye Görüntüleme ve Üye Ekleme	25
4.4.2 Admin Paneli Üye Bilgileri Güncelleme	25
4.4.3 Üye ve Oyun Profilleri Görüntüleme	25
4.4.4 Üyelerin Oyun Profillerini Puanlama	25
4.4.5 Üyelerin Oyun Profillerini Puanlama	25
4.4.6 Sıralamalar Tablosu	27
4.4.7 Belirli Üyeye Mesaj Gönderme	27
4.5 Oyun Profilleri Kontroller	27
4.5.1 Admin Paneli - Oyun Profili Ekleme ve Silme.....	27
4.5.2 Admin Paneli - Oyun Profili Düzenleme.....	27
4.5.3 Oyun Profilleri Bulma	28
4.5.4 Oyun Profilleri Bulma JsonResult	28
4.6 İstatistik Kontroller	29
4.7 Profilim Kontroller	30
4.7.1 Profilim - Gösterme ve Düzenleme	30
4.7.2 Profilim - Oyun Profillerini Gösterme	31
4.7.3 Profilim - Oyun Profili Oluşturma.....	32
4.7.4 Profilim - Oyun Profili Düzenleme	32
4.7.5 Profilim - Oyun Profili Silme	32
4.8 Mesajlar Kontroller.....	32
4.8.1 Üye Mesaj İşlemleri.....	32
4.8.2 Admin Paneli Mesaj İşlemleri.....	32
5. KULLANILAN TEKNOLOJILER.....	35
5.1 Visual Studio.....	35
5.2 Asp .Net MVC.....	36

5.2.1 Model.....	36
5.2.2 View	37
5.2.3 Controller.....	37
5.2.4 Routing.....	38
5.3 MVCGrid.....	38
5.3.1 Özellikleri	38
5.4 Bootstrap.....	38
5.5 JQuery.....	39
5.6 Microsoft SQL Server.....	39
6. UYGULAMADA YAPILABİLECEK GELİŞTİRMELER	41
KAYNAKLAR.....	43
EKLER.....	45
ÖZGEÇMİŞ	47

Şekil Listesi

Sayfa

Şekil 1.1	: oyuncuilan.com.....	2
Şekil 1.2	: duobul.com	2
Şekil 1.3	: duobul.com	3
Şekil 2.1	: IR Diagramı	5
Şekil 2.2	: Veritabanı Tabloları	5
Şekil 2.3	: Prosedürler.....	6
Şekil 2.4	: OyunArkadasım Model	6
Şekil 3.1	: Üye Layout	7
Şekil 3.2	: Profilim Layout Sol Menü	8
Şekil 3.3	: Admin Layout.....	8
Şekil 3.4	: Kayıt Olma Ekranı.....	9
Şekil 3.5	: Giriş Yapma Ekranı	9
Şekil 3.6	: Anasayfa Ekranı	10
Şekil 3.7	: Profilim Ekranı	10
Şekil 3.8	: Profilim Ekranı	11
Şekil 3.9	: Profilim - Oyun Profilleri Ekranları.....	11
Şekil 3.10	: Profilim - Oyun Profilleri Oluşturma Ekranları.....	11
Şekil 3.11	: Profilim - Oyun Profilleri Düzenleme Ekranları.....	12
Şekil 3.12	: Gönderilen Mesajlar Ekranı.....	12
Şekil 3.13	: Gelen Mesajlar Ekranı.....	13
Şekil 3.14	: Üye Arama Ekranı.....	13
Şekil 3.15	: LoL Profili Bulma Ekranı	14
Şekil 3.16	: CS:GO Profili Bulma Ekranı Kodları	14
Şekil 3.17	: Valorant Profili Bulma Ekranı	15
Şekil 3.18	: Üye Mesaj Gönderme Ekranı	15
Şekil 3.19	: Üye Profili Ekranı.....	16
Şekil 3.20	: Üye Oyun Profili Ekranları.....	16
Şekil 3.21	: LoL Profilleri Sıralama Ekranı	16
Şekil 3.22	: CS:GO Profilleri Sıralama Ekran Kodları	17
Şekil 3.23	: Valorant Profilleri Sıralama Ekran Kodları.....	18
Şekil 3.24	: Yönetici Girişi Ekran Kodları.....	18
Şekil 3.25	: Admin Paneli - Kullanıcılar Ekranı	19
Şekil 3.26	: Üye Güncelleme Ekranı.....	19
Şekil 3.27	: Admin Paneli - Oyun Profilleri Ekranı	20
Şekil 3.28	: Admin Paneli - Oyun Profili Düzenleme Ekran Kodları	20
Şekil 3.29	: Admin Paneli - Gelen Mesajlar Ekranı.....	21

Şekil 3.30	: Admin Paneli - Gönderilen Mesajlar Ekranı	21
Şekil 3.31	: Admin Paneli - İstatistikler Ekranı	22
Şekil 4.1	: Üye Girişi Action Result	23
Şekil 4.2	: Admin Girişi Action Result	23
Şekil 4.3	: Çıkış Yap Action Result	24
Şekil 4.4	: Kayıt Ol Action Result	24
Şekil 4.5	: Anasayfa Action Result'ları	25
Şekil 4.6	: Admin Paneli Üye Görüntüleme ve Ekleme ActionResult'ları	25
Şekil 4.7	: Admin Paneli - Üye Bilgileri Güncelleme ActionResult	26
Şekil 4.8	: Üye ve Oyun Profilleri Görüntüleme ActionResult	26
Şekil 4.9	: Üye Oyun Profilleri Puanlama ActionResult	27
Şekil 4.10	: Üye Oyun Profilleri Puanlama ActionResult	28
Şekil 4.11	: Oyun Profilleri Sıralamalar ActionResult	28
Şekil 4.12	: Belirli Üyeye Mesaj Gönderme ActionResult	29
Şekil 4.13	: Admin Paneli - Oyun Profili Ekleme/Silme ActionResult	29
Şekil 4.14	: Admin Paneli - Oyun Profili Düzenleme ActionResult	29
Şekil 4.15	: Oyun Profilleri Bulma ActionResult	30
Şekil 4.16	: Oyun Profilleri Bulma JsonResult	30
Şekil 4.17	: Admin Paneli - İstatistik ActionResult	30
Şekil 4.18	: Profilim - Profilim Gösterme/Düzenleme ActionResult	31
Şekil 4.19	: Profilim - Profilim Gösterme/Düzenleme ActionResult	31
Şekil 4.20	: Profilim - Oyun Profili Oluşturma	32
Şekil 4.21	: Profilim - Oyun Profili Düzenleme	33
Şekil 4.22	: Profilim - Oyun Profili Silme	34
Şekil 4.23	: Profilim - Üye Mesajlar ActionResult'ları	34
Şekil 4.24	: Profilim - Admin Paneli Mesajlar ActionResult'ları	34
Şekil 5.1	: Visual Studio	35
Şekil 5.2	: Model View Controller	37
Şekil 5.3	: MVC Life Cycle	37
Şekil 5.4	: Bootstrap	39
Şekil 5.5	: MS-SQL	40

OYUN ARKADAŞIM

Özet

Günümüzde online oyunlar inanılmaz sayıda kullanıcıya sahiptir. Birçok oyunda derecelendirme sistemi vardır. İnsanlar oynadıkları oyunda yükselmek için hayatından oldukça vaktini bu oyunlara harcıyor ve de bu durum onların hayatını olumsuz etkileyebiliyor. OyunArkadaşım; üye olduktan sonra, üyenin sahip olduğu oyuncu profilleriyle, oyuncuların kendine uygun filtrelerle takım arkadaşı bulmasını ve onların oynadıkları oyunda daha kısa sürede hedeflerine ulaşmasını sağlar. Kendi bünyesinde puan ve sıralama sistemi vardır. Bu sistemde sıralamaya girerek ödüller kazanmasını sağlar.

Anahtar Kelimeler: oyun, duo, lol, cs, valorant, derece, takım, oyun arkadaşı.

OYUN ARKADAŞIM

SUMMARY

Online games today have an incredible number of users. Many games have a rating system. People spend quite a bit of time in their lives in order to rise in the game they play, and this can negatively affect their lives. OyunArkadasım, After becoming a member, it enables the players to find teammates with suitable filters and reach their goals in a shorter time in the game they play with the player profiles that the member has. It has its own score and ranking system. In this system, it allows him to win prizes by ranking.

Keywords: game, duo, lol, cs, valorant, ranking, team, teammate.

1. GİRİŞ

Günlük hayatımızda online oyunlar, birçok kişi için oldukça önemli bir yere sahip. Online oyun oynayan insanların sayısı oldukça fazla. Bu oyunlar da genellikle derecelendirme sistemine sahip. Bu derecelendirme sistemi insanların günlük hayatını olumsuz etkileyebiliyor. Bu kişiler oynadığı oyuna çok fazla zaman ayırdığı için gece geç yatabiliyor, sabah işine geç kalabiliyor, oyuna sinirlenip daha agresif bir kişi olabiliyor. OyunArkadasım bunlar gibi olumsuz etkilere çözüm arayan bir projedir.

1.1 Tezin Amacı

Bu proje online oyunlara oldukça vakit ayıran insanların, ayırdıkları vakti maksimum verimle değerlendirmesini sağlamayı hedefler.

1.1.1 Tezin ikincil amaçları

Sitemize üye olan kullanıcılar kendi oyun profillerini oluştururlar. Bu oyun profilleri sayesinde kendilerine uygun kişileri bulur ve iletişime geçer. Böylece kendine uygun kişilerle oyunlarını oynarak oyun içinde daha hızlı yükselirler. Sitemizde oyuncu profilleri puanlanabilmektedir. Puanlamayı teşvik etmek adına, oyun profilleri sıralamalarında ilk 5'te yer alan kullanıcılara oyun içinde ödüller verilmektedir. Puanlama sistemi, kullanıcıların daha güvenilir bir oyun arkadaşı bulması için önemlidir.

1.2 Literatür Araştırması

1.2.1 Oyuncuilan.com

Oyuncuilan.com bir "Oyun Buluşma Noktası" platformudur. Temel olarak oyuncuları buluşturmayı hedeflerken, oyun dünyasıyla ilgili son gelişmeleri paylaşan bir oyun portalıdır.







oyuncu ilan ver

Anasayfa Blog İlanlar Oyun Kodları Serverler Hesap Satış Forum


Ücretsiz İlan Ver Giriş Yap Kayıt Ol

oyun arkadaşı ilanları : arama sonuçları

7672 sonuç bulundu.

İlan	Yaş	Şehir	Cinsiyet	Seviye	İlan Tarihi
 Açıklamayı ful okuyun ekleyin Light in Shadows	23	Kayseri	Erkek	Unranked	11.01.2021
 Gümüş Altın Elo Bot lane oyuncusuyum. BuLLeDeRateD	24	Adana	Erkek	Altın IV	10.01.2021
 Beraber oynayacağım takım arkadaşı arıyorum strongerthanall	21	İzmir	Erkek	Unranked	10.01.2021
 Solo Q Ranked Krito	30	Adana	Erkek	Altın IV	10.01.2021
 (ADC MAIN) hafta içi akşamları lol de duo ahmetmohurcu	26	İstanbul	Erkek	Gümüş III	10.01.2021
 Soloq ve esnek oyunlar için lukeng	25	İstanbul	Erkek	Unranked	09.01.2021

ARAMAYI ÖZELLEŞTİR



İlan Ara

Seviye Aralığı

En Düşük

En Yüksek

Şehir

Tüm Şehirler
Yurtdışı
Adana
Adıyaman
Afyonkarahisar

Şekil 1.1 : oyuncuilan.com

1.2.2 Duobul.com

DUO BUL

Oyunlar GİRİŞ

DC Mordecia

Dereceli Tek/Gift (5v5)

Açıklama

ADC mainim agresif oynarım iyi ve agresif oynayan sup mainler varsa ekleyebilir sesli sohbet olursa daha iyi olur

Oynadığı Roller

ALT

Aradığı Roller

DESTEK

İletişim Programları

discord | skype | teamspeak3 | curse

PROFİLE GİT DETAYLI BİLGİ (0 YORUM)

BloodSorry

Dereceli Tek/Gift (5v5)

Açıklama

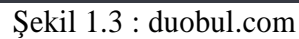
adc mainim agresif oynarım iyi ve agresif oynayan sup mainler varsa ekleyebilir sesli sohbet olursa daha iyi olur

Oynadığı Roller

Şekil 1.2 : duobul.com

Takımında tanıdığın birilerinin olması, tanımadığın kişilerle oynamaktan çok daha iyidir. Oyunlarında hem daha çok kazanabilir, hem de daha çok eğlenebilirsin. Bunun nedeni çok basit; Takım Oyunu. Eğer oyunlarına bir duo ile girersen, rakiplerinden bir

1.2.3 Discord Kanalları



3


```

SQLQuery1.sql - D...DAQBA\bseno (56)
USE [DBTez]
GO
/***** Object: StoredProcedure [dbo].[Top5cs]    Script Date: 14.1.2021 01:47:37 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER procedure [dbo].[Top5cs]
as
select top 5 NICKNAME,PUAN,UYEID from TBLUYEbyUNBILGI where OYUNID=2 order by PUAN desc

SQLQuery2.sql - D...DAQBA\bseno (56)
USE [DBTez]
GO
/***** Object: StoredProcedure [dbo].[Top5lo1]    Script Date: 14.1.2021 01:48:49 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER procedure [dbo].[Top5lo1]
as
select top 5 NICKNAME,PUAN,UYEID from TBLUYEOYUNBILGI where OYUNID=1 order by PUAN desc

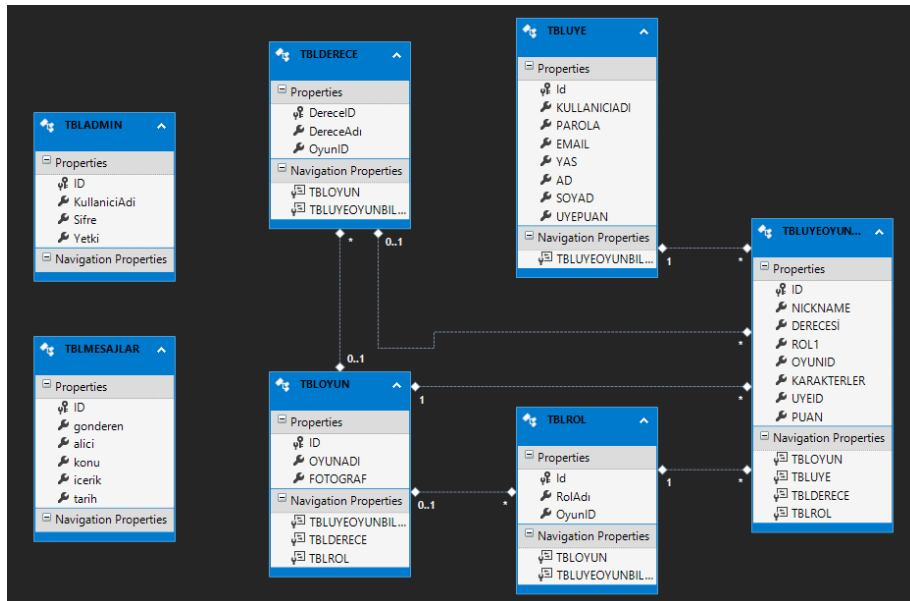
SQLQuery3.sql - D...DAQBA\bseno (57)
USE [DBTez]
GO
/***** Object: StoredProcedure [dbo].[Top5valo]    Script Date: 14.1.2021 01:49:25 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER procedure [dbo].[Top5valo]
as
select top 5 NICKNAME,PUAN,UYEID from TBLUYEOYUNBILGI where OYUNID=3 order by PUAN desc

```

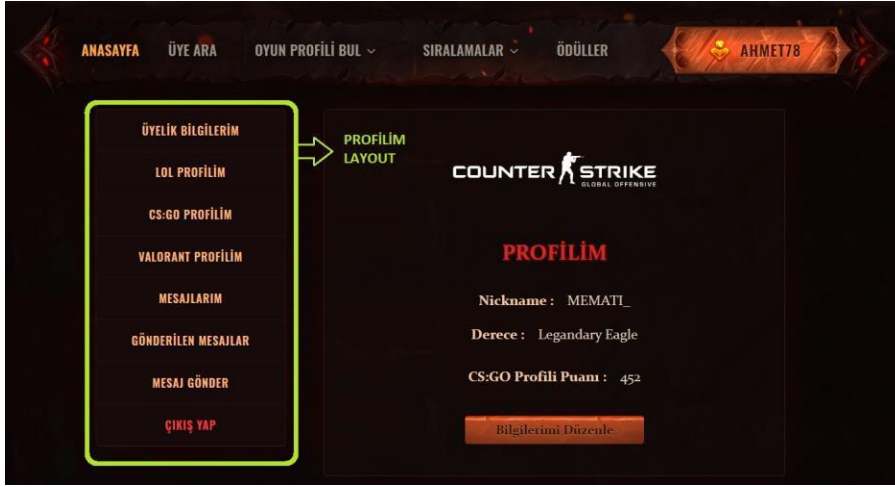
Şekil 2.3 : Prosedürler

2.4 Model

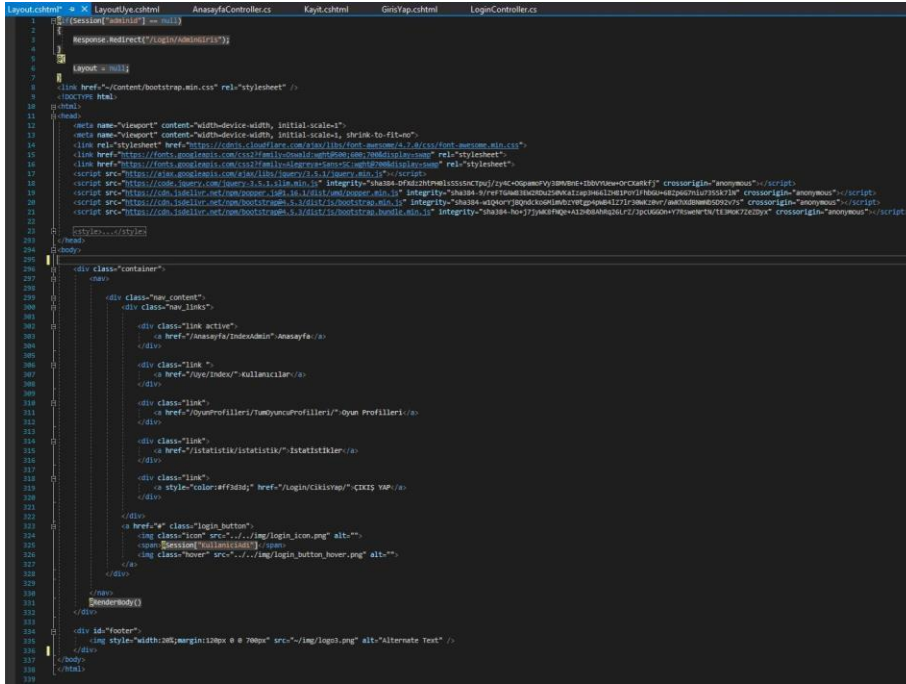
OyunArkadasım projesinde kullanılan model.



Şekil 2.4 : OyunArkadasım Model



Şekil 3.2 : Profilim Layout Sol Menü



Şekil 3.3 : Admin Layout

3.4 Kayıt Ekranı

Siteye kayıt olma sayfası.

3.5 Üye Giriş Ekranı

Üye girişinin yapıldığı ekran.

3.6 Üye Anasayfa Ekranı

KAYIT OL

Kullanıcı Adı

Şifre

E-Mail

Adınız

Soyadınız

Yaşınız

KAYIT OL

Şekil 3.4 : Kayıt Olma Ekranı

GİRİŞ YAP

Kullanıcı Adı

Şifre

[Şifreni mi Unuttun?](#) [Yönetici Girişi](#)

GİRİŞ YAP

[Üyelğin Yok Mu? Hemen Üye Ol](#)

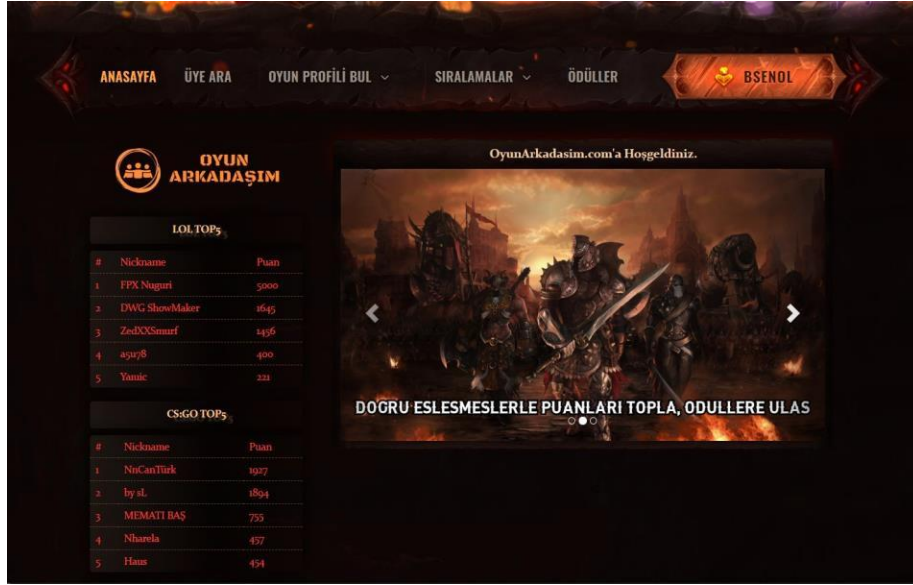
Şekil 3.5 : Giriş Yapma Ekranı

Misafir kullanıcıların ve üyelerin anasayfası. Sol tarafta oyunlardaki en yüksek puana sahip 5 kullanıcı yer almakta. Site tanıtım amaçlı slider da bulunmakta.

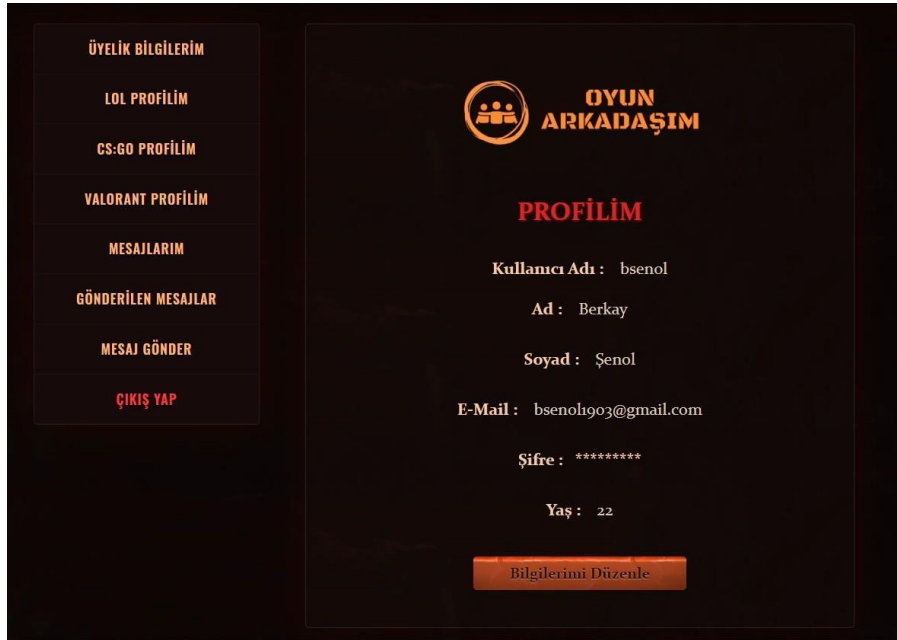
3.7 Profilim Ekranı

Üyelerin bilgilerini görüntüleyebildiği ve oyun profillerini görüntüleme, düzenleme ve oluşturma sayfalarına ulaştığı ekran.

3.8 Profilim Ekranı



Şekil 3.6 : Anasayfa Ekranı



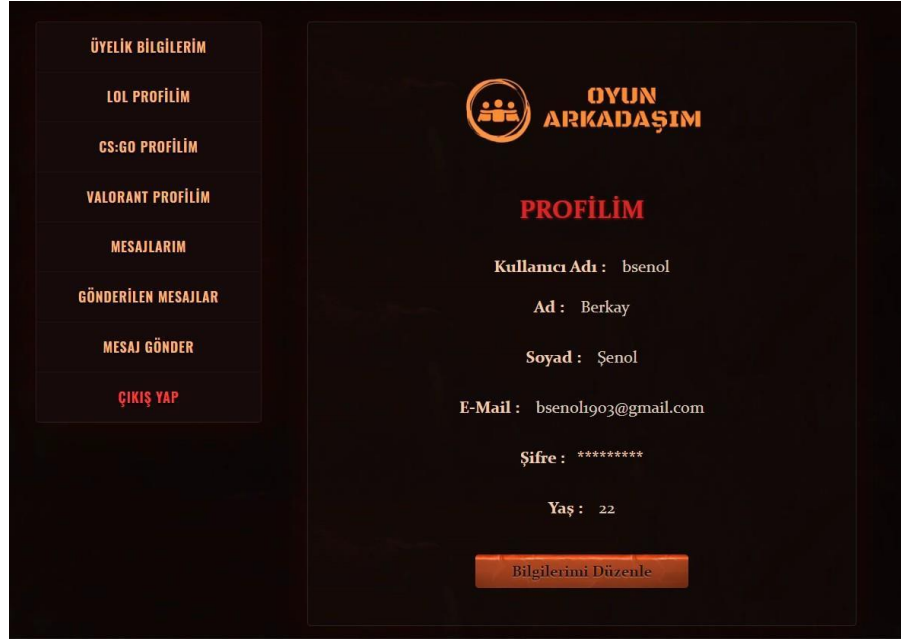
Şekil 3.7 : Profilim Ekranı

Üyelerin bilgilerini görüntüleyebildiği ve oyun profillerini görüntüleme, düzenleme ve oluşturma sayfalarına ulaştığı ekran.

3.9 Profilim - Oyun Profilleri Ekranı

Üyelerin kendi oyun profillerini görüntüleyebildiği ve kendi oyun profillerini düzenleme ve oluşturma sayfalarına ulaştığı ekran.

3.10 Profilim - Oyun Profilleri Oluşturma Ekranı

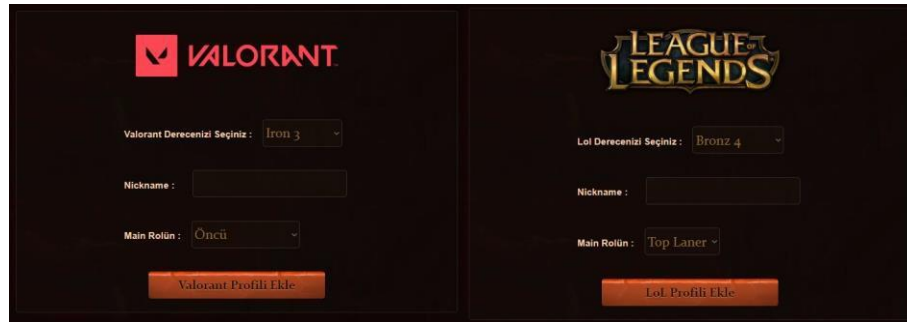


Şekil 3.8 : Profilim Ekranı



Şekil 3.9 : Profilim - Oyun Profilleri Ekranları

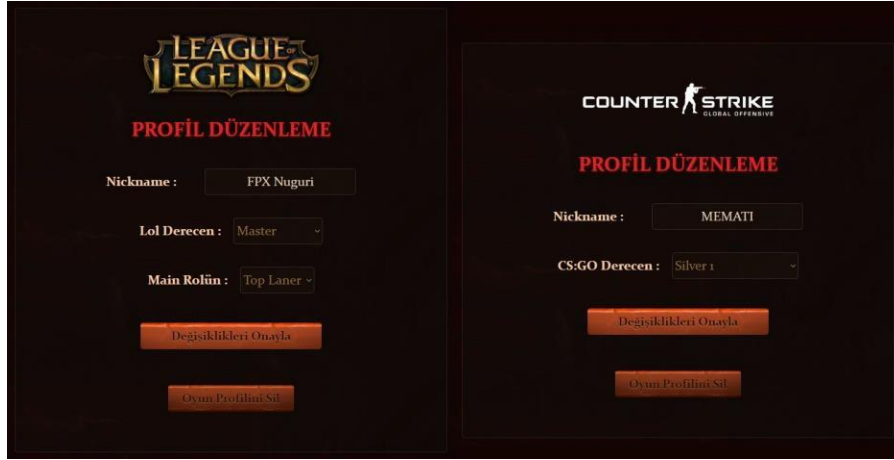
Üyelerin kendi oyun profillerini oluşturduğu ekran.



Şekil 3.10 : Profilim - Oyun Profilleri Oluşturma Ekranları

3.11 Profilim - Oyun Profilleri Düzenleme Ekranı

Üyelerin kendi oyun profillerini düzeleyebildiği ve mevcut profilini silebildiği ekran.



Şekil 3.11 : Profilim - Oyun Profilleri Düzenleme Ekranları

3.12 Gönderilen Mesajlar Ekranı

Üyelerin diğer üyelere gönderdiği mesajları görüntülediği ekran.

ÜYELİK BİLGİLERİM

LOL PROFİLİM

CS:GO PROFİLİM

VALORANT PROFİLİM

MESAJLARIM

GÖNDERİLEN MESAJLAR

MESAJ GÖNDER

ÇIKIŞ YAP

GÖNDERİLEN MESAJLAR

ALICI	KONU	MESAJ	ZAMAN	SİL
ahmet78	selam	Selam iyi oyunlar	21.1.2021 00:00:00	
ahmet78	Duo Ariyorum	Support Gold3 elo duo arıyorum.	21.1.2021 00:00:00	
ahmet78	Puan	Profiline olumlu puan kullandım. Sende kullanır mısın?	21.1.2021 00:00:00	

Şekil 3.12 : Gönderilen Mesajlar Ekranı

3.13 Gelen Mesajlar Ekranı

Üyelerin diğer üyelere aldığı mesajları görüntülediği ekran.

3.14 Üye Arama Ekranı

Üyelerin diğer üyeleri görüntülediği ekran. Üyeler bu ekranda kullanıcı adı araması yapabilir, yaşa göre filtreleme ve sıralama yapabilir. Üyelere Mesaj Gönderebilir.

3.15 Oyun Profili Bulma - LoL Profili Bulma Ekranı



Şekil 3.13 : Gelen Mesajlar Ekranı



Şekil 3.14 : Üye Arama Ekranı

Üyelerin diğer üyelerin LoL Profillerini görüntülediği ekran. Üye, LoL Profili bulmak için oyun profillerinin sahip olduğu nickname, derece, oyun rolü, puan özelliklerini filtreleyebilir ve sıralayabilir. Oyun Profili Bulma sayfalarında Grid.Mvc kullanılmıştır. Grid.Mvc sayesinde oyun profilleri özellikleri hakkında filtreleme ve sıralama yapılabilmektedir.

3.16 Oyun Profili Bulma - CS:GO Profili Bulma Ekranı

Üyelerin diğer üyelerin CS:GO profillerini görüntülediği ekran. Üye, CS:GO Profili bulmak için oyun profillerinin sahip olduğu nickname, derece, oyun rolü, puan özelliklerini filtreleyebilir ve sıralayabilir. Oyun Profili Bulma sayfalarında Grid.Mvc kullanılmıştır. Grid.Mvc sayesinde oyun profilleri özellikleri hakkında filtreleme ve sıralama yapılabilmektedir.

NICKNAME	DERECE	MAIN ROLÜ	PUANI	MESAJ	PROFİL
ttt	Silver 4	Mid Laner	2		
Kolon_lol	Silver 3	AD Carry	26		
ZedXXSmurf	GrandMaster	Mid Laner	1456		
asur8	Diamond 4	Mid Laner	400		
testname	Bronz 1	AD Carry	74		
yeniolola	Platinum 4	Jungler	2		
Yjdp42	Gold 2	Support	45		
Manu_Rh	Gold 1	AD Carry	215		
IAJAS	Platinum 4	AD Carry	9		
DWG ShowMaker	Challenger	Mid Laner	1645		

Şekil 3.15 : LoL Profili Bulma Ekranı

```

1  using GridMvc.Html
2  @model IEnumerable<OyunArkadasim.Models.Entity.TBLUYEOYUNBILGI>
3
4  ViewBag.Title = "CsProfilBul";
5  Layout = "~/Views/Shared/LayoutUye.cshtml";
6
7
8  <style>...</style>
114 <html xmlns="http://www.w3.org/1999/xhtml">
115 <head>...</head>
123 <body>
124     <h2>CS:GO PROFİLLERİ</h2>
125     <div id="liste">
126         @Html.Grid(Model).Named("csprofilleri").Columns(col =>
127         {
128             col.Add(x => x.NICKNAME).Titled("NICKNAME").Sortable(true).Filterable(true);
129             col.Add(x => x.TBLDERECE.DereceAdi).Titled("DERECE").Sortable(true).Filterable(true).SetFilterWidgetType("csdereceler");
130             col.Add(x => x.PUANI).Titled("PUANI").Sortable(true);
131             col.Add().Titled("MESAJ").Encoded(false).Sanitized(false).RenderValueAs(html =>
132             {
133                 string htmlstr = "<a class='mesaj-icon' href='/Uye/MesajGonder/' + html.UYEID.ToString() + '></a>";
134                 return MvcHtmlString.Create(htmlstr);
135             })
136             .SetWidth(200);
137             col.Add().Titled("PROFİL").Encoded(false).Sanitized(false).RenderValueAs(html =>
138             {
139                 string htmlstr = "<a href='/Uye/UyeProfilGoster/' + html.UYEID.ToString() + '></a>";
140                 return MvcHtmlString.Create(htmlstr);
141             })
142             .SetWidth(200);
143         })
144         .WithPaging(10).SetLanguage("tr")
145     </div>
146
147     <script>
148     $(function () {
149         pageGrids.csprofilleri.addFilterWidget(new CustomersFilterWidgetCsDereceler());
150     })
151     </script>
152
153 </body>
154 </html>
155
156
157

```

Şekil 3.16 : CS:GO Profili Bulma Ekranı Kodları

3.17 Oyun Profili Bulma - Valorant Profili Bulma Ekranı

Üyelerin diğer üyelerin Valorant profillerini görüntülediği ekran. Üye, Valorant Profili bulmak için oyun profillerinin sahip olduğu nickname, derece, oyun rolü, puan özelliklerini filtreleyebilir ve sıralayabilir. Oyun Profili Bulma sayfalarında Grid.Mvc kullanılmıştır. Grid.Mvc sayesinde oyun profilleri özellikleri hakkında filtreleme ve sıralama yapılabilir.

VALORANT PROFİLLERİ					
NICKNAME	DERECE	MAIN ROLÜ	PUANI	MESAJ	PROFİL
qa588da	Iron 3	Öncü	755		
yenivlao	Gold 1	Düello	78		
Valosu42	Diamond 2	Düello	101		
ManuIIIli	Legandary Eagle M.	Support	154		
IIIIII	Immortal 1	Düello	457		
Yoaousj	Gold 1	Gözcü	241		
SÜLEYMAN ÇAKIR	Bronze 3	Düello	1450		
SelanikPOWER	Gold 1	Kontrol Uzmanı	442		
kkkloon_VAL	Iron 2	Düello	757		
SaaLy	Immortal 2	Kontrol Uzmanı	0		

Şekil 3.17 : Valorant Profili Bulma Ekranı

3.18 Üye'ye Mesaj Gönderme Ekranı

Üyelerin, diğer üyelere mesaj gönderdiği ekran.

TEST123

ÜYE PROFİLİ

LOL PROFİLİ

CS:GO PROFİLİ

VALORANT PROFİLİ

MESAJ GÖNDER

MESAJ GÖNDER

Alıcı :

Konu :

Mesaj İçeriği :

Mesaj Gönder

Şekil 3.18 : Üye Mesaj Gönderme Ekranı

3.19 Üye Profil Ekranı

Üyenin profilini görüntülediğimiz ekran.

3.20 Üye Oyun Profili Ekranları

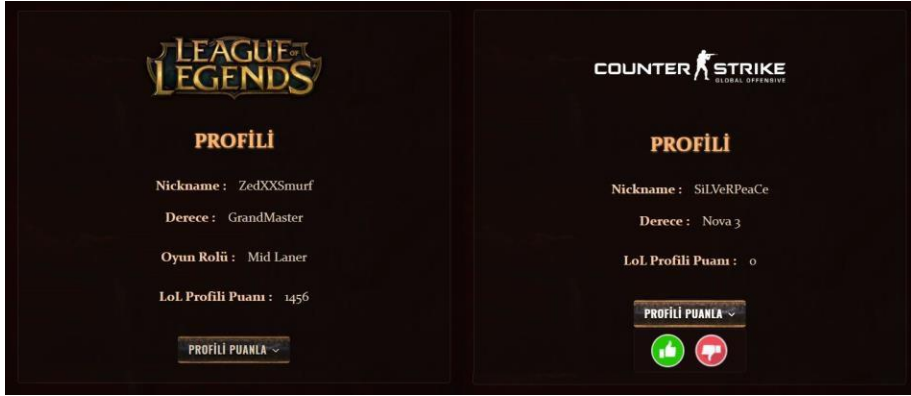
Üyenin oyun profillerini görüntülediğimiz ekran.

3.21 Sıralamalar - Lol Profilleri Sıralama Ekranı

Sistemdeki LoL Profillerinin puana göre sıralandığı ekran.



Şekil 3.19 : Üye Profili Ekranı



Şekil 3.20 : Üye Oyun Profili Ekranları

LOL PUAN SIRALAMASI						
#	NICKNAME	DERECESİ	MAIN ROL	ÜYE ADI	PROFİL PUANI	PROFİL
1	DWG ShowMaker	Challenger	Mid Laner	caner	1645	✦
2	ZedXXSmurf	GrandMaster	Mid Laner	caglar45	1456	✦
3	asuy78	Diamond 4	Mid Laner	raststdty	400	✦
4	Yauic	Gold 2	Support	ahmet78	221	✦
5	Manu_Rh	Gold 1	AD Carry	qwert	215	✦
6	testname	Bronz 1	AD Carry	test123	74	✦
7	Yjdp42	Gold 2	Support	sas53	45	✦
8	iyioldu23	Diamond 2	Mid Laner	iyioldu2	42	✦
9	Kolon_lol	Silver 3	AD Carry	kkolon_23	26	✦
10	IAJAS	Platinum 4	AD Carry	qwe	9	✦
11	yeniolol12	Platinum 4	Jungler	yeni	2	✦
12	ttt	Silver 4	Mid Laner	Hasan13	2	✦
13	FPX Nuguri	Master	Top Laner	bsenol	0	✦
14	DeaDLy	Diamond 2	Support	tuly23	0	✦

Şekil 3.21 : LoL Profilleri Sıralama Ekranı

3.22 Sıralamalar - CS:GO Profilleri Sıralama Ekranı

Sistemdeki CS:GO Profillerinin puana göre sıralandığı ekran.

```
1  using OyunArkadasim.Models.Entity
2  @model List<TBLUYEOYUNBILGI>
3  @{
4      ViewBag.Title = "CsSıralama";
5      Layout = "~/Views/Shared/LayoutUye.cshtml";
6  }
7  @if (int satirNo = 0; )
8  <style>...</style>
71 <h2>CS:GO PUAN SIRALAMASI</h2>
72
73
74 <table class="table" style="border: 1px solid #ffb676;">
75 <tr>
76 <th>#</th>
77 <th>NICKNAME</th>
78 <th>DERECESİ</th>
79 <th>ÜYE ADI</th>
80 <th>PROFİL PUANI</th>
81 <th>PROFİL</th>
82 </tr>
83 @foreach (var p in Model)
84 {
85 <tr>
86 <td>@satirNo += 1</td>
87 <td>@p.NICKNAME</td>
88 <td>@p.TBLDERECE.DereceAdı</td>
89 <td>@p.TBLUYE.KULLANICIADI</td>
90 <td>@p.PUAN</td>
91 <td><a href="/Uye/UyeProfilGoster/@p.UYEID"></a></td>
92 </tr>
93 }
94 </table>
95
96
```

Şekil 3.22 : CS:GO Profilleri Sıralama Ekran Kodları

3.23 Sıralamalar - Valorant Profilleri Sıralama Ekranı

Sistemdeki Valorant Profillerinin puana göre sıralandığı ekran.

3.24 Yönetici Giriş Ekranı

Sistemdeki kayıtlı yöneticilerin, admin paneline giriş yaptığı ekran.

3.25 Admin Paneli - Kullanıcılar Ekranı

Yöneticinin sistemdeki üyeleri görüntülediği ekran. Ekranda üye bilgilerini düzenleme, mesaj gönderme, üyenin sahip olduğu oyun profillerini görüntüleme sayfasına gidebilir, aynı zamanda sistemdeki bir üyeyi silebilir.

```

1  @using OyunArkadasim.Models.Entity
2  @model List<TBLUYEOYUNBILGI>
3  @{
4      ViewBag.Title = "ValorantSiralama";
5      Layout = "~/Views/Shared/LayoutUye.cshtml";
6  }
7  @{ int satirNo = 0; }
8  <style>...</style>
71 <h2>VALORANT PUAN SIRALAMASI</h2>
72
73
74 <table class="table" style="border: 1px solid #ffb676;">
75 <tr>
76 <th>#</th>
77 <th>NICKNAME</th>
78 <th>DERECESİ</th>
79 <th>MAIN ROL</th>
80 <th>ÜYE ADI</th>
81 <th>PROFİL PUANI</th>
82 <th>PROFİL</th>
83 </tr>
84 @foreach (var p in Model)
85 {
86 <tr>
87 <td>@{satirNo += 1}</td>
88 <td>p.NICKNAME</td>
89 <td>p.TBLDERECE.DereceAdı</td>
90 <td>p.TBLROL.RolAdı</td>
91 <td>p.TBLUYE.KULLANICIADI</td>
92 <td>p.PUAN</td>
93 <td><a href="/Uye/UyeProfilGoster/@p.UYEID"></a></td>
94 </tr>
95 }
96 </table>
97

```

Şekil 3.23 : Valorant Profilleri Sıralama Ekran Kodları

```

1
2
3  @using OyunArkadasim.Models.Entity
4  @model AdminGiris
5  @{
6      ViewBag.Title = "AdminGiris";
7      Layout = "~/Views/Shared/LayoutUye.cshtml";
8  }
9  <style>...</style>
89 @using (Html.BeginForm("AdminGiris", "Login", FormMethod.Post, new { @class = "inner_form" }))
90 {
91 <div class="title">YÖNETİCİ GİRİŞİ</div>
92
93 <div class="form-group">
94 <input type="text" name="KullaniciAdi" class="form-control" placeholder="Kullanıcı Adı" required="required" />
95 </div>
96
97 <div class="form-group">
98 <input type="password" name="Sifre" class="form-control" placeholder="Şifre" required="required" />
99 <a href="#" class="form_link">Şifreni mi Unuttun?</a>
100 </div>
101
102 <button type="submit" name="login-button" class="btn">
103 <span>GİRİŞ YAP</span>
104 </button>
105 <p class="alert text-danger">@ViewBag.uyari</p>
106
107

```

Şekil 3.24 : Yönetici Girişi Ekran Kodları

3.26 Admin Paneli - Üye Güncelleme Ekranı

Yöneticinin sistemdeki üye bilgilerini değiştirebildiği ekran.

3.27 Admin Paneli - Oyun Profilleri Ekranı

[ANASAYFA](#) [KULLANICILAR](#) [OYUN PROFİLLERİ](#) [İSTATİSTİKLER](#) [MESAJLAR](#) [ÇIKIŞ YAP](#) [ADMIN](#)

ÜYE PROFİLLERİ

Aramak İstediğiniz Kullanıcı Adını Giriniz : Ara

UyeId	Kullanıcı Adı	Parola	Email	Yaş	Ad	Soyad	Üye Sil	Üye Güncelle	Oyun Profillerini Güncelle	Mesaj Gönder
2	bsenol	123123	bsenol1903@gmail.com	22	Berkay	Şenol	Sil	Güncelle	Oyun Profilleri	Mesaj
6	ahmet78	123455	ahmet3r@hotmail.com	40	Ahmet	Türkü	Sil	Güncelle	Oyun Profilleri	Mesaj
15	Hasan13	hwert1234333	ahmet30@gmail.com	30	Hasan	AYO	Sil	Güncelle	Oyun Profilleri	Mesaj
22	rastdty	454545	dfgstrutsr5@gmail.com	43	Utku	ıysal	Sil	Güncelle	Oyun Profilleri	Mesaj
23	test123	test1234	test13@gmail.com	25	testad	testsoyad	Sil	Güncelle	Oyun Profilleri	Mesaj
25	kkolon_23	ayarlaz7236	ayarajsf@hotmail.com	21	Cevdet	Kolon	Sil	Güncelle	Oyun Profilleri	Mesaj
26	caglar45	csvaglar1495983	caglar45@gmail.com	21	Caglar	akgün	Sil	Güncelle	Oyun Profilleri	Mesaj

[Yeni Üye Ekle](#)

1 2 3 4 »

Şekil 3.25 : Admin Paneli - Kullanıcılar Ekranı

ÜYE BİLGİLERİ GÜNCELLEME

Üye Adı :

Üye Şifresi :

Üye Maili :

Üye Yaşı :

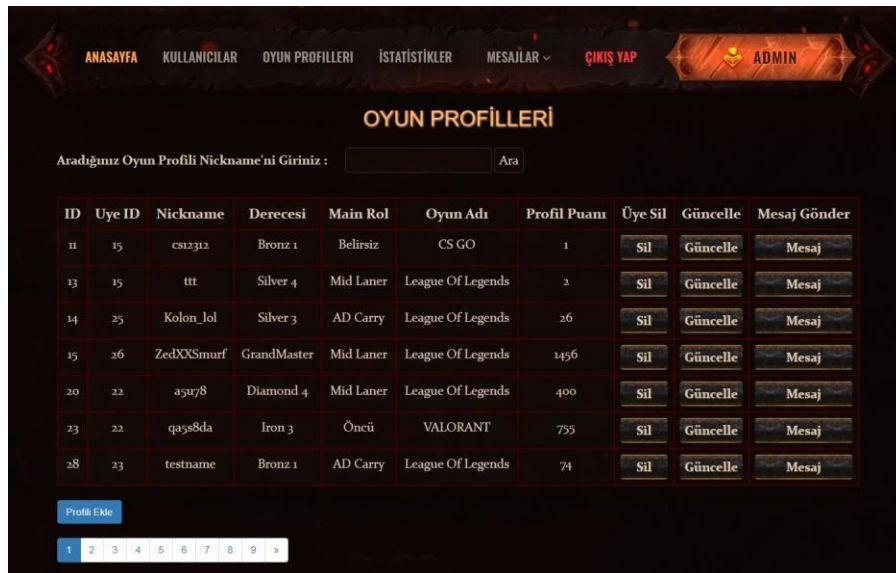
Üyenin Adı :

Üyenin Soyadı :

[Güncelle](#)

Şekil 3.26 : Üye Güncelleme Ekranı

Yöneticinin sistemdeki oyun profillerini görüntülediği, silebildiği, üyeye mesaj gönderebildiği ve oyun profilini düzenleme sayfasına gidebildiği ekran.



Şekil 3.27 : Admin Paneli - Oyun Profilleri Ekranı

3.28 Admin Paneli - Oyun Profili Düzenleme Ekranı

Yöneticinin sistemdeki oyun profillerinin bilgilerini düzenleyebildiği ekranın kodları.

```

1  @model OyunArkadasim.Models.Entity.TBLUYEUYUNUSITGI
2
3  ViewBag.Title = "OyunProfiliGetir";
4  Layout = "~/Views/Shared/Layout.cshtml";
5
6  @{
7      @{style="margin-top:15px;"}
8  }
9  <div class="form">
10     <h2>Oyun Profili Güncelle</h2>
11     @using (Html.BeginForm("OyunProfiliGuncelle", "OyunProfilleri", FormMethod.Post))
12     {
13         <div style="margin-top:15px;"></div>
14         <form class="form-group">
15             @Html.HiddenFor(p => p.ID, new { @class = "form-control" })
16             <br />
17             @Html.HiddenFor(p => p.TBLUYE.Id, new { @class = "form-control" })
18             @Html.HiddenFor(p => p.TBLUYUN.ID, new { @class = "form-control" })
19             <label>Nickname :&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</label>
20             @Html.TextBoxFor(p => p.NICKNAME, new { @class = "text-box" })
21             <br />
22             <label>Derecesi :&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</label>
23             @Html.DropDownListFor(p => p.TBLDERECE.DereceID, (list<SelectListItem>)ViewBag.dgr1, new { @class = "text-box text-warning" })
24             <br />
25             @Html.Main RolId :&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~</label>
26             @Html.DropDownListFor(p => p.TBLROL.Id, (list<SelectListItem>)ViewBag.dgr2, new { @class = "text-box text-warning" })
27             <br />
28             <label>Profil Puanı :&nbsp;&nbsp;&nbsp;</label>
29             @Html.TextBoxFor(p => p.PUAN, new { @class = "text-box" })
30             <br />
31             <div style="margin-top:15px">
32                 <button class="btn btn-warning"><span>Oyuncu Profilini Güncelle</span></button>
33             </div>
34         </form>
35     }
36 </div>

```

Şekil 3.28 : Admin Paneli - Oyun Profili Düzenleme Ekran Kodları

3.29 Admin Paneli - Gelen Mesajlar Ekranı

Yöneticinin, üyelere gelen mesajları görüntülediği ekran.

3.30 Admin Paneli - Gönderilen Mesajlar Ekranı



Şekil 3.29 : Admin Paneli - Gelen Mesajlar Ekranı

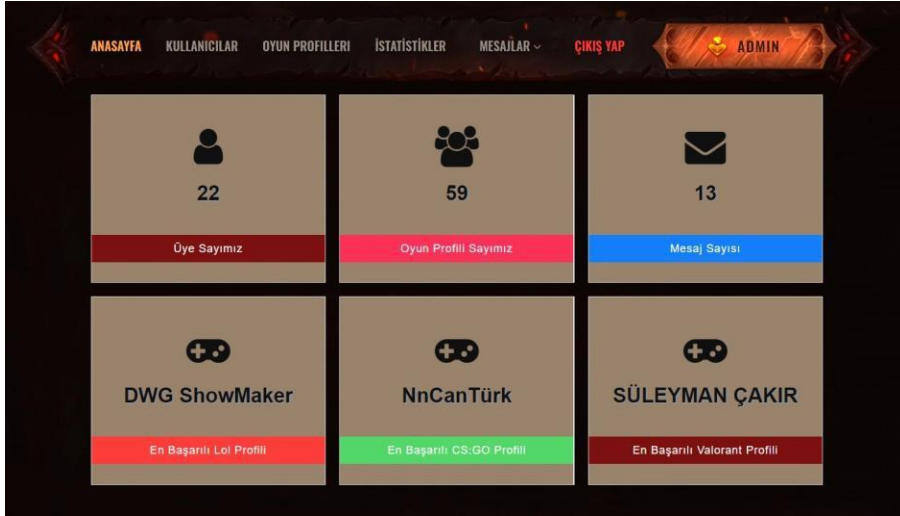
Yöneticinin, üyelere gönderdiği mesajları görüntülediği ekran.



Şekil 3.30 : Admin Paneli - Gönderilen Mesajlar Ekranı

3.31 Admin Paneli - İstatistikler Ekranı

Yöneticinin, siteyle ilgili istatistikleri görüntüleyebildiği ekran.



Şekil 3.31 : Admin Paneli - İstatistikler Ekranı

4. CONTROLLERS

4.1 Login Kontroller

Siteye üye, admin girişi ve çıkış yapma ActionResultların'ın bulunduğu controller.

4.1.1 Giriş Yap

Üye girişi yapılmasını sağlayan ActionResult.

```
public ActionResult GirişYap()
{
    return View();
}

[HttpPost]
public ActionResult GirişYap(TBLUYE u)
{
    var bilgiler = db.TBLUYE.FirstOrDefault(x => x.KULLANICIADI == u.KULLANICIADI && x.PAROLA == u.PAROLA);

    if (bilgiler != null)
    {
        FormsAuthentication.SetAuthCookie(bilgiler.KULLANICIADI, false);
        Session["KullaniciAdi"] = bilgiler.KULLANICIADI.ToString();
        Session["UyeId"] = bilgiler.Id.ToString();
        return RedirectToAction("Index", "Anasayfa");
    }
    else
    {
        ViewBag.uyari = "Kullanıcı Adı yada şifre yanlış.";
        return View();
    }
}
```

Şekil 4.1 : Üye Girişi Action Result

4.1.2 Admin Giriş Yap

Yönetici girişi yapılmasını sağlayan ActionResult.

```
public ActionResult AdminGiris()
{
    return View();
}

[HttpPost]
public ActionResult AdminGiris(TBLADMIN a)
{
    var login = db.TBLADMIN.Where(x => x.KullaniciAdi == a.KullaniciAdi && x.Sifre == a.Sifre).SingleOrDefault();
    if (login != null)
    {
        Session["KullaniciAdi"] = login.KullaniciAdi.ToString();
        Session["adminid"] = login.ID.ToString();
        FormsAuthentication.SetAuthCookie(login.KullaniciAdi, false);
        return RedirectToAction("IndexAdmin", "Anasayfa");
    }
    else
    {
        ViewBag.uyari = "Kullanıcı Adı yada şifre yanlış.";
        return View(a);
    }
}
```

Şekil 4.2 : Admin Girişi Action Result

4.1.3 Çıkış Yap

Siteden üye veya yönetici çıkışını sağlayan ActionResult.

```
public ActionResult CikisYap()
{
    Session["KullaniciAdi"] = null;
    Session["adminid"] = null;
    Session["UyeId"] = null;
    Session.Abandon();
    FormsAuthentication.SignOut();
    return RedirectToAction("GirisYap");
}
```

Şekil 4.3 : Çıkış Yap Action Result

4.2 Kayıt Ol Kontroller

Siteye üye olunmasını sağlayan ActionResult.

```
public ActionResult Kayit()
{
    return View();
}

[HttpPost]
public ActionResult Kayit(TBLUYE u)
{
    if (!ModelState.IsValid)
    {
        return View(u);
    }

    db.TBLUYE.Add(u);
    db.SaveChanges();
    return View("../Login/GirisYap");
}
```

Şekil 4.4 : Kayıt Ol Action Result

4.3 Anasayfa Kontroller

Üye ve admin anasayfalarını sağlayan ActionResult. Anasayfada, oyunlardaki en yüksek puana sahip ilk 5 kullanıcı yer almaktadır. Bu sıralama veritabanında prosedür kullanarak sağlanmıştır.

4.4 Üye Kontroller

```

public ActionResult Index()
{
    var prf1 = db.Top5lol();
    ViewBag.dgr1 = prf1;
    var prf2 = db.Top5cs();
    ViewBag.dgr2 = prf2;
    var prf3 = db.Top5valo();
    ViewBag.dgr3 = prf3;
    return View();
}

public ActionResult IndexAdmin()
{
    var prf1 = db.Top5lol();
    ViewBag.dgr1 = prf1;
    var prf2 = db.Top5cs();
    ViewBag.dgr2 = prf2;
    var prf3 = db.Top5valo();
    ViewBag.dgr3 = prf3;
    return View();
}

```

Şekil 4.5 : Anasayfa Action Result'ları

4.4.1 Admin Paneli Üye Görüntüleme ve Üye Ekleme

Admin üyeleri görüntüleme ve yeni üye ekleme ActionResult.

```

public ActionResult Index(string u, int sayfa = 1)
{
    var uye = from k in db.TBLUYE select k;
    if (!string.IsNullOrEmpty(u))
    {
        uye = uye.Where(m => m.KULLANICIADI.Contains(u));
    }
    return View(uye.ToList().ToPagedList(sayfa, 7));
}

public ActionResult UyeSil(int id)
{
    var uye = db.TBLUYE.Find(id);
    db.TBLUYE.Remove(uye);
    db.SaveChanges();
    return RedirectToAction("Index");
}

[HttpGet]
public ActionResult UyeEkle()
{
    return View();
}

[HttpPost]
public ActionResult UyeEkle(TBLUYE u)
{
    if (!ModelState.IsValid)
    {
        return View("UyeEkle");
    }
    db.TBLUYE.Add(u);
    db.SaveChanges();
    return RedirectToAction("Index");
}

```

Şekil 4.6 : Admin Paneli Üye Görüntüleme ve Ekleme ActionResult'ları

4.4.2 Admin Paneli Üye Bilgileri Güncelleme

Admin üye bilgilerini güncelleme ActionResult.

4.4.3 Üye ve Oyun Profilleri Görüntüleme

Üyelerin profilinin ve sahip olduğu oyun profillerinin görüntülenmesini sağlayan ActionResult.

4.4.4 Üyelerin Oyun Profillerini Puanlama

Üyelerin, oyun profillerine puan vermesini sağlayan ActionResult.

4.4.5 Üyelerin Oyun Profillerini Puanlama

Üyelerin, oyun profillerine puan vermesini sağlayan ActionResult.

4.4.6 Sıralamalar Tablosu

```

public ActionResult UyeGuncelle(int id)//UyeGetirme işlemi
{
    var uye = db.TBLUYE.Find(id);
    ViewBag.dgr = id;
    return View("UyeGuncelle", uye);
}

public ActionResult UyeGuncelle1(TBLUYE u)
{
    if (!ModelState.IsValid)
    {
        return View("UyeGuncelle");
    }
    var uye = db.TBLUYE.Find(u.Id);
    uye.Id = u.Id;
    uye.KULLANICIADI = u.KULLANICIADI;
    uye.PAROLA = u.PAROLA;
    uye.EMAIL = u.EMAIL;
    uye.YAS = u.YAS;
    uye.AD = u.AD;
    uye.SOYAD = u.SOYAD;
    db.SaveChanges();
    return RedirectToAction("Index");
}

```

Şekil 4.7 : Admin Paneli - Üye Bilgileri Güncelleme ActionResult

```

[Authorize]
public ActionResult UyeProfilGoster(int id)
{
    var p = db.TBLUYE.Find(id);
    ViewData["uyeid"] = id;
    ViewData["kullaniciadi"] = p.KULLANICIADI;
    return View("UyeProfilGoster", p);
}

[Authorize]
public ActionResult UyeLolProfilGoster(int id)
{
    var uye = db.TBLUYE.Find(id);
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(1)) && (x.UYEID == uye.Id)).FirstOrDefault();
    ViewData["uyeid"] = id;
    return View("UyeLolProfilGoster", op);
}

[Authorize]
public ActionResult UyeCsProfilGoster(int id)
{
    var uye = db.TBLUYE.Find(id);
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(2)) && (x.UYEID == uye.Id)).FirstOrDefault();
    return View("UyeCsProfilGoster", op);
}

[Authorize]
public ActionResult UyeValorantProfilGoster(int id)
{
    var uye = db.TBLUYE.Find(id);
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(3)) && (x.UYEID == uye.Id)).FirstOrDefault();
    ViewData["uyeid"] = id;
    return View("UyeValorantProfilGoster", op);
}

```

Şekil 4.8 : Üye ve Oyun Profilleri Görüntüleme ActionResult

```

[Authorize]
public ActionResult LolPuanlama(int id)
{
    var uye = db.TBLUYE.Find(id);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(1)) && (x.UYEID == uye.Id)).FirstOrDefault();
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    op.PUAN = op.PUAN + 1;
    db.SaveChanges();
    return View("UyeProfilGoster", uye);
}

[Authorize]
public ActionResult LolPuanlama2(int id)
{
    var uye = db.TBLUYE.Find(id);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(1)) && (x.UYEID == uye.Id)).FirstOrDefault();
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    op.PUAN = op.PUAN - 1;
    if (op.PUAN <= 0)
    {
        op.PUAN = 0;
    }
    db.SaveChanges();
    return View("UyeProfilGoster", uye);
}

[Authorize]
public ActionResult CsPuanlama(int id)
{
    var uye = db.TBLUYE.Find(id);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(2)) && (x.UYEID == uye.Id)).FirstOrDefault();
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    op.PUAN = op.PUAN + 1;
    db.SaveChanges();
    return View("UyeProfilGoster", uye);
}

[Authorize]
public ActionResult CsPuanlama2(int id)
{
    var uye = db.TBLUYE.Find(id);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(2)) && (x.UYEID == uye.Id)).FirstOrDefault();
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    op.PUAN = op.PUAN - 1;
    if (op.PUAN <= 0)
    {
        op.PUAN = 0;
    }
    db.SaveChanges();
    return View("UyeProfilGoster", uye);
}

```

Şekil 4.9 : Üye Oyun Profilleri Puanlama ActionResult

Oyun profillerinin puanlarına göre sıralanmasını sağlayan ActionResult.

4.4.7 Belirli Üyeye Mesaj Gönderme

Belirli bir üyeye yada belirli bir oyun profiline sahip olan üyeye mesaj gönderilmesini sağlayan ActionResult.

4.5 Oyun Profilleri Kontroller

4.5.1 Admin Paneli - Oyun Profili Ekleme ve Silme

Yöneticinin belirli bir üyeye oyun profili eklemesini ve belirli bir oyun profilini silmesini sağlayan ActionResult.

4.5.2 Admin Paneli - Oyun Profili Düzenleme


```

[Authorize]
public ActionResult LolPuanlama(int id)
{
    var uye = db.TBLUYE.Find(id);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(1)) && (x.UYEID == uye.Id)).FirstOrDefault();
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    op.PUAN = op.PUAN + 1;
    db.SaveChanges();
    return View("UyeProfilGoster", uye);
}

[Authorize]
public ActionResult LolPuanlama2(int id)
{
    var uye = db.TBLUYE.Find(id);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(1)) && (x.UYEID == uye.Id)).FirstOrDefault();
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    op.PUAN = op.PUAN - 1;
    if (op.PUAN <= 0)
    {
        op.PUAN = 0;
    }
    db.SaveChanges();
    return View("UyeProfilGoster", uye);
}

[Authorize]
public ActionResult CsPuanlama(int id)
{
    var uye = db.TBLUYE.Find(id);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(2)) && (x.UYEID == uye.Id)).FirstOrDefault();
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    op.PUAN = op.PUAN + 1;
    db.SaveChanges();
    return View("UyeProfilGoster", uye);
}

[Authorize]
public ActionResult CsPuanlama2(int id)
{
    var uye = db.TBLUYE.Find(id);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(2)) && (x.UYEID == uye.Id)).FirstOrDefault();
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    op.PUAN = op.PUAN - 1;
    if (op.PUAN <= 0)
    {
        op.PUAN = 0;
    }
    db.SaveChanges();
    return View("UyeProfilGoster", uye);
}

```

Şekil 4.10 : Üye Oyun Profilleri Puanlama ActionResult

```

public ActionResult LolSiralama()
{
    var lolprofilleri = db.TBLUYEOYUNBILGI.Where(x => x.OYUNID.Equals(1)).OrderByDescending(i => new { i.PUAN, i.DERECEİ }).ToList();
    return View(lolprofilleri);
}

public ActionResult CsSiralama()
{
    var csprofilleri = db.TBLUYEOYUNBILGI.Where(x => x.OYUNID.Equals(2)).OrderByDescending(i => new { i.PUAN, i.DERECEİ }).ToList();
    return View(csprofilleri);
}

public ActionResult ValorantSiralama()
{
    var valoprofilleri = db.TBLUYEOYUNBILGI.Where(x => x.OYUNID.Equals(3)).OrderByDescending(i => new { i.PUAN, i.DERECEİ }).ToList();
    return View(valoprofilleri);
}

```

Şekil 4.11 : Oyun Profilleri Sıralamalar ActionResult

Yöneticinin belirli bir oyun profilini düzenlemesini sağlayan ActionResult.

4.5.3 Oyun Profilleri Bulma

Üye panelinde kullanılan oyun profilleri bulma sayfalarını sağlayan ActionResult.

4.5.4 Oyun Profilleri Bulma JsonResult


```

[Authorize]
[HttpGet]
public ActionResult MesajGonder(int id)
{
    var uye = db.TBLUYE.Find(id);
    ViewData["kullaniciadi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    return View();
}

[Authorize]
[HttpPost]
public ActionResult MesajGonder(TBLMESAJLAR m)
{
    var uyeadi = (string)Session["KullaniciAdi"].ToString();
    m.gonderen = uyeadi.ToString();
    m.tarih = DateTime.Parse(DateTime.Now.ToShortDateString());
    db.TBLMESAJLAR.Add(m);
    db.SaveChanges();
    var uye = db.TBLUYE.Find(ViewData["uyeid"]);
    return RedirectToAction("GidenMesajlar", "Mesajlar");
}

```

Şekil 4.12 : Belirli Üyeye Mesaj Gönderme ActionResult

```

[HttpGet]
public ActionResult OyunProfiliEkle()
{
    List<SelectListItem> deger1 = (from i in db.TBLOYUN.ToList()
    select new SelectListItem
    {
        Text = i.OYUNADI,
        Value = i.ID.ToString()
    }).ToList();

    ViewBag.dgr1 = deger1;

    List<SelectListItem> deger2 = (from i in db.TBLUYE.ToList()
    select new SelectListItem
    {
        Text = i.ID.ToString(),
        Value = i.ID.ToString()
    }).ToList();

    ViewBag.dgr2 = deger2;
    List<SelectListItem> deger3 = (from i in db.TBLDERECE.ToList()
    select new SelectListItem
    {
        Text = i.DereceAdi,
        Value = i.DereceID.ToString()
    }).ToList();

    ViewBag.dgr3 = deger3;
    List<SelectListItem> deger4 = (from i in db.TBLROL.ToList()
    select new SelectListItem
    {
        Text = i.RolAdi,
        Value = i.ID.ToString()
    }).ToList();

    ViewBag.dgr4 = deger4;
    return View();
}

[HttpPost]
public ActionResult OyunProfiliEkle(TBLUYEOYUNBILGI n)
{
    var prf = db.TBLOYUN.Where(p => p.ID == n.TBLOYUN.ID).FirstOrDefault();
    var uye = db.TBLUYE.Where(u => u.ID == n.TBLUYE.ID).FirstOrDefault();
    var drc = db.TBLDERECE.Where(u => u.DereceID == n.TBLDERECE.DereceID).First();
    var rol = db.TBLROL.Where(u => u.ID == n.TBLROL.ID).FirstOrDefault();
    n.TBLOYUN = prf;
    n.TBLUYE = uye;
    n.TBLDERECE = drc;
    n.TBLROL = rol;
    db.TBLUYEOYUNBILGI.Add(n);
    db.SaveChanges();
    return RedirectToAction("Index", "uye");
}

public ActionResult OyunProfiliSil(int id)
{
    var p = db.TBLUYEOYUNBILGI.Find(id);
    var uye = db.TBLUYE.Where(u => u.ID == p.TBLUYE.ID).FirstOrDefault();
    db.TBLUYEOYUNBILGI.Remove(p);
    db.SaveChanges();
    return RedirectToAction("Index", "uye");
}

```

Şekil 4.13 : Admin Paneli - Oyun Profili Ekleme/Silme ActionResult

```

public ActionResult OyunProfiliGetir(int id)
{
    var p = db.TBLUYEOYUNBILGI.Find(id);
    List<SelectListItem> deger1 = (from i in db.TBLDERECE.ToList()
    select new SelectListItem
    {
        Text = i.DereceAdi,
        Value = i.DereceID.ToString()
    }).ToList();

    ViewBag.dgr1 = deger1;
    List<SelectListItem> deger2 = (from i in db.TBLROL.ToList()
    select new SelectListItem
    {
        Text = i.RolAdi,
        Value = i.ID.ToString()
    }).ToList();

    ViewBag.dgr2 = deger2;
    return View("OyunProfiliGetir", p);
}

public ActionResult OyunProfiliGuncelle(TBLUYEOYUNBILGI p)
{
    var prf = db.TBLUYEOYUNBILGI.Find(p.ID);
    var uye = db.TBLUYE.Where(y => y.ID == p.TBLUYE.ID).FirstOrDefault();
    var drc = db.TBLDERECE.Where(x => x.DereceID == p.TBLDERECE.DereceID).FirstOrDefault();
    var rol = db.TBLROL.Where(x => x.ID == p.TBLROL.ID).FirstOrDefault();
    prf.DERECEID = uye.ID;
    prf.TBLDERECE = drc;
    prf.TBLROL = rol;
    prf.NICKNAME = p.NICKNAME;
    prf.PUAN = p.PUAN;
    db.SaveChanges();
    return RedirectToAction("Index", "uye");
}

```

Şekil 4.14 : Admin Paneli - Oyun Profili Düzenleme ActionResult

Üye panelinde kullanılan oyun profilleri bulma sayfalarında oyun rolü, oyun derecesi özelliklerinde filtreleme sağlayan JsonResult'lar.

4.6 İstatistik Kontroller

```

[Authorize]
public ActionResult LolProfiliBul(int sayfa = 1)
{
    var lolprofilleri = db.TBLUYEOYUNBILGI.Where(x => x.OYUNID.Equals(1)).ToList();
    return View(lolprofilleri);
}

[Authorize]
public ActionResult CsProfiliBul(int sayfa = 1)
{
    var csprofilleri = db.TBLUYEOYUNBILGI.Where(x => x.OYUNID.Equals(2)).ToList();
    return View(csprofilleri);
}

[Authorize]
public ActionResult ValorantProfiliBul(int sayfa = 1)
{
    var valprofilleri = db.TBLUYEOYUNBILGI.Where(x => x.OYUNID.Equals(3)).ToList();
    return View(valprofilleri);
}

```

Şekil 4.15 : Oyun Profilleri Bulma ActionResult

```

public JsonResult LolDereceGetirAjax()
{
    var drc = db.TBLDERECE.Where(x => x.OyunID == 1).Select(x => x.DereceAdi);
    return Json(drc);
}

public JsonResult LolRolGetirAjax()
{
    var roller = db.TBLROL.Where(x => x.OyunID == 1).Select(x => x.RolAdi);
    return Json(roller);
}

public JsonResult CsDereceGetirAjax()
{
    var drc = db.TBLDERECE.Where(x => x.OyunID == 2).Select(x => x.DereceAdi);
    return Json(drc);
}

[Authorize]
public JsonResult ValorantDereceGetirAjax()
{
    var drc = db.TBLDERECE.Where(x => x.OyunID == 3).Select(x => x.DereceAdi);
    return Json(drc);
}

[Authorize]
public JsonResult ValorantRolGetirAjax()
{
    var roller = db.TBLROL.Where(x => x.OyunID == 3).Select(x => x.RolAdi);
    return Json(roller);
}

```

Şekil 4.16 : Oyun Profilleri Bulma JsonResult

Admin Panelinde kullanılan istatistik sayfasını sağlayan ActionResult.

```

public ActionResult istatistik()
{
    var uyesayisi = db.TBLUYE.Count();
    ViewBag.dgr1 = uyesayisi;
    var opsayisi = db.TBLUYEOYUNBILGI.Count();
    ViewBag.dgr2 = opsayisi;
    var mesajsayisi = db.TBLMESAJLAR.Count();
    ViewBag.dgr3 = mesajsayisi;
    var loltop1 = db.TBLUYEOYUNBILGI.Where(x => x.OYUNID.Equals(1)).OrderByDescending(z => z.PUAN).Select(y => y.NICKNAME).FirstOrDefault();
    ViewBag.dgr4 = loltop1;
    var cstop1 = db.TBLUYEOYUNBILGI.Where(x => x.OYUNID.Equals(2)).OrderByDescending(z => z.PUAN).Select(y => y.NICKNAME).FirstOrDefault();
    ViewBag.dgr5 = cstop1;
    var valotop1 = db.TBLUYEOYUNBILGI.Where(x => x.OYUNID.Equals(3)).OrderByDescending(z => z.PUAN).Select(y => y.NICKNAME).FirstOrDefault();
    ViewBag.dgr6 = valotop1;
    return View();
}

```

Şekil 4.17 : Admin Paneli - İstatistik ActionResult

4.7 Profilim Kontroller

Profilim Kontroller'da yer alan Action Result'ların viewlarında LayoutProfilim kullanılır. LayoutProfilim sayesinde üyelerin profilim sayfasında kendi sahip olduğu oyun profilleri ve mesajlarım menüsünü sağlar.

4.7.1 Profilim - Gösterme ve Düzenleme

Üyenin Profilim sayfası ve kendi üyelik bilgilerini düzenlemesini sağlayan ActionResultlar.

```

[Authorize]
public ActionResult ProfilGoster()// Üyenin Kendi Profilini Gösterme
{
    var kullanıcıadi = (string)Session["KullaniciAdi"];
    var degerler = db.TBLUYE.FirstOrDefault(z => z.KULLANICIADI == kullanıcıadi);
    return View(degerler);
}

public ActionResult Index()//Profilim Düzenleme
{
    var kullanıcıadi = (string)Session["KullaniciAdi"];
    var degerler = db.TBLUYE.FirstOrDefault(z => z.KULLANICIADI == kullanıcıadi);
    return View(degerler);
}

[Authorize]
[HttpPost]
public ActionResult Index2(TBLUYE u)
{
    var kullanıcı = (string)Session["KullaniciAdi"];
    var uye = db.TBLUYE.FirstOrDefault(x => x.KULLANICIADI == kullanıcı);
    uye.PAROLA = u.PAROLA;
    uye.AD = u.AD;
    uye.SOYAD = u.SOYAD;
    uye.YAS = u.YAS;
    uye.EMAIL = u.EMAIL;
    //uye.KULLANICIADI = u.KULLANICIADI;
    db.SaveChanges();
    return RedirectToAction("ProfilGoster");
}

```

Şekil 4.18 : Profilim - Profilim Gösterme/Düzenleme ActionResult

4.7.2 Profilim - Oyun Profillerini Gösterme

Üyenin sahip olduğu oyun profillerini göstermesini sağlayan ActionResultlar.

```

[Authorize]
public ActionResult LolProfiliGoster()
{
    var prf = (string)Session["UyeId"];
    var prf2 = db.TBLUYEOYUNBILGI.FirstOrDefault(z => z.UYEID.ToString() == prf);

    try
    {
        var dd = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(1)) && (x.UYEID == prf2.UYEID)).FirstOrDefault();
        if (dd == null)
        {
            return RedirectToAction("ProfilimLolProfiliEkle");
        }
        else
        {
            return View(dd);
        }
    }
    catch (Exception)
    {
        return RedirectToAction("ProfilimLolProfiliEkle");
    }
}

[Authorize]
public ActionResult ValorantProfiliGoster()
{
    var prf = (string)Session["UyeId"];
    var prf2 = db.TBLUYEOYUNBILGI.FirstOrDefault(z => z.UYEID.ToString() == prf);

    try
    {
        var dd = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(3)) && (x.UYEID == prf2.UYEID)).FirstOrDefault();
        if (dd == null)
        {
            return RedirectToAction("ProfilimValorantProfiliEkle");
        }
        else
        {
            return View(dd);
        }
    }
    catch (Exception)
    {
        return RedirectToAction("ProfilimValorantProfiliEkle");
    }
}

```

Şekil 4.19 : Profilim - Profilim Gösterme/Düzenleme ActionResult

4.7.3 Profilim - Oyun Profili Oluřturma

Üye profilim ekranından, oyun profili sekmesine geçtiğinde, o oyuna ait bir profili yoksa o oyunun profil oluřturma sayfasına yönlendirilir. Üyenin oyun profili oluřturmasını saėlayan ActionResultlar.

```
[HttpGet]
public ActionResult ProfilimValorantProfililekle()
{
    List<SelectListItem> deger1 = (from i in db.TBLROL.Where(r => r.OyunID == 3).ToList()
    select new SelectListItem
    {
        Text = i.RolAdi,
        Value = i.Id.ToString()
    }).ToList();

    ViewBag.dgr1 = deger1;

    List<SelectListItem> deger2 = (from i in db.TBLDERECE.Where(d => d.OyunID == 3).ToList()
    select new SelectListItem
    {
        Text = i.DereceAdi,
        Value = i.DereceID.ToString()
    }).ToList();

    ViewBag.dgr2 = deger2;
    return View();
}

[Authorize]
[HttpPost]
public ActionResult ProfilimValorantProfililekle(TBLUYEVUNBILGI n)
{
    var prf = db.TBLUYUN.Where(p => p.ID == 3).FirstOrDefault();
    var drc = db.TBLDERECE.Where(p => p.DereceID == n.TBLDERECE.DereceID).FirstOrDefault();
    var rol = db.TBLROL.Where(r => r.Id == n.TBLROL.Id).FirstOrDefault();
    var k = (string)Session["uyeId"];
    var uye = db.TBLUYE.FirstOrDefault(z => z.Id.ToString() == k);
    n.TBLUYUN = prf;
    n.TBLDERECE = drc;
    n.TBLROL = rol;
    n.UYEDID = uye.Id;
    db.TBLUYEVUNBILGI.Add(n);
    db.SaveChanges();
    return RedirectToAction("ValorantProfilimGoster");
}

[Authorize]
[HttpGet]
public ActionResult ProfilimLoLProfililekle()
{
    List<SelectListItem> deger1 = (from i in db.TBLROL.Where(r => r.OyunID == 1).ToList()
    select new SelectListItem
    {
        Text = i.RolAdi,
        Value = i.Id.ToString()
    }).ToList();

    ViewBag.dgr1 = deger1;

    List<SelectListItem> deger2 = (from i in db.TBLDERECE.Where(d => d.OyunID == 1).ToList()
    select new SelectListItem
    {
        Text = i.DereceAdi,
        Value = i.DereceID.ToString()
    }).ToList();

    ViewBag.dgr2 = deger2;
    return View();
}

[Authorize]
[HttpPost]
public ActionResult ProfilimLoLProfililekle(TBLUYEVUNBILGI n)
{
    var prf = db.TBLUYUN.Where(p => p.ID == 1).FirstOrDefault();
    var drc = db.TBLDERECE.Where(p => p.DereceID == n.TBLDERECE.DereceID).FirstOrDefault();
    var rol = db.TBLROL.Where(r => r.Id == n.TBLROL.Id).FirstOrDefault();
    var k = (string)Session["uyeId"];
    var uye = db.TBLUYE.FirstOrDefault(z => z.Id.ToString() == k);
    n.TBLUYUN = prf;
    n.TBLDERECE = drc;
    n.TBLROL = rol;
    n.UYEDID = uye.Id;
    db.TBLUYEVUNBILGI.Add(n);
    db.SaveChanges();
    return RedirectToAction("LoLProfilimGoster");
}
```

řekil 4.20 : Profilim - Oyun Profili Oluřturma

4.7.4 Profilim - Oyun Profili Düzenleme

Üyenin sahip olduėu oyun profillerinin düzenlenmesini saėlayan ActionResultlar.

4.7.5 Profilim - Oyun Profili Silme

Üyelere oyun profili düzenleme sayfasından oyun profillerini silebilir. Oyun Profillerinin silinmesini saėlayan ActionResultlar.

4.8 Mesajlar Kontroller

4.8.1 Üye Mesaj İşlemleri

Üyenin gönderdiėi ve aldıėı mesajları gösteren ve silen, yeni mesaj göndermesini saėlayan ActionResultlar.

4.8.2 Admin Paneli Mesaj İşlemleri

Yöneticilerin gönderdiėi ve aldıėı mesajları gösteren ve silen, yeni mesaj göndermesini saėlayan ActionResultlar.

```

[Authorize]
[HttpGet]
public ActionResult ValorantProfilimiDuzenle()
{
    List<SelectListItem> deger1 = (from i in db.TBLROL.Where(r => r.OyunID == 3).ToList()
    select new SelectListItem
    {
        Text = i.RolAdi,
        Value = i.Id.ToString()
    }).ToList();

    ViewBag.dgr1 = deger1;

    List<SelectListItem> deger2 = (from i in db.TBLDERECE.Where(d => d.OyunID == 3).ToList()
    select new SelectListItem
    {
        Text = i.DereceAdi,
        Value = i.DereceID.ToString()
    }).ToList();

    ViewBag.dgr2 = deger2;
    var kullaniciadi = (string)Session["KullaniciAdi"];
    var degerler = db.TBLUYE.FirstOrDefault(z => z.KULLANICIADI == kullaniciadi);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(3)) && (x.UYEID == degerler.Id)).FirstOrDefault();
    return View(op);
}

[Authorize]
[HttpPost]
public ActionResult ValorantProfilimiDuzenle2(TBLUYEOYUNBILGI p)
{
    var k = (string)Session["UyeId"];
    var u = db.TBLUYE.FirstOrDefault(z => z.Id.ToString() == k);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(3)) && (x.UYEID == u.Id)).FirstOrDefault();
    var drc = db.TBLDERECE.Where(x => x.DereceID == p.TBLDERECE.DereceID).FirstOrDefault();
    var rol = db.TBLROL.Where(x => x.Id == p.TBLROL.Id).FirstOrDefault();
    op.NICKNAME = p.NICKNAME;
    op.TBLDERECE = drc;
    op.TBLROL = rol;
    db.SaveChanges();
    return RedirectToAction("ValorantProfilimiGoster");
}

[Authorize]
[HttpGet]
public ActionResult CsProfilimiDuzenle()
{
    List<SelectListItem> deger2 = (from i in db.TBLDERECE.Where(d => d.OyunID == 2).ToList()
    select new SelectListItem
    {
        Text = i.DereceAdi,
        Value = i.DereceID.ToString()
    }).ToList();

    ViewBag.dgr2 = deger2;
    var kullaniciadi = (string)Session["KullaniciAdi"];
    var degerler = db.TBLUYE.FirstOrDefault(z => z.KULLANICIADI == kullaniciadi);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(2)) && (x.UYEID == degerler.Id)).FirstOrDefault();
    return View(op);
}

[Authorize]
[HttpPost]
public ActionResult CsProfilimiDuzenle2(TBLUYEOYUNBILGI p)
{
    var k = (string)Session["UyeId"];
    var u = db.TBLUYE.FirstOrDefault(z => z.Id.ToString() == k);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(2)) && (x.UYEID == u.Id)).FirstOrDefault();
    var drc = db.TBLDERECE.Where(x => x.DereceID == p.TBLDERECE.DereceID).FirstOrDefault();
    op.NICKNAME = p.NICKNAME;
    op.TBLDERECE = drc;
    db.SaveChanges();
    return RedirectToAction("CsProfilimiGoster");
}

```

Şekil 4.21 : Profilim - Oyun Profili Düzenleme


```

[Authorize]
public ActionResult LolProfiliniSil()
{
    var k = (string)Session["UyeId"];
    var u = db.TBLUYE.FirstOrDefault(z => z.Id.ToString() == k);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(1)) && (x.UYEID == u.Id)).FirstOrDefault();
    db.TBLUYEOYUNBILGI.Remove(op);
    db.SaveChanges();
    return RedirectToAction("ProfilGoster");
}

[Authorize]
public ActionResult ValorantProfiliniSil()
{
    var k = (string)Session["UyeId"];
    var u = db.TBLUYE.FirstOrDefault(z => z.Id.ToString() == k);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(3)) && (x.UYEID == u.Id)).FirstOrDefault();
    db.TBLUYEOYUNBILGI.Remove(op);
    db.SaveChanges();
    return RedirectToAction("ProfilGoster");
}

[Authorize]
public ActionResult CsProfiliniSil()
{
    var k = (string)Session["UyeId"];
    var u = db.TBLUYE.FirstOrDefault(z => z.Id.ToString() == k);
    var op = db.TBLUYEOYUNBILGI.Where(x => (x.OYUNID.Equals(2)) && (x.UYEID == u.Id)).FirstOrDefault();
    db.TBLUYEOYUNBILGI.Remove(op);
    db.SaveChanges();
    return RedirectToAction("ProfilGoster");
}

```

Şekil 4.22 : Profilim - Oyun Profili Silme

```

[Authorize]
public ActionResult Index() // kullanıcıya gelen mesajlar.
{
    var uyeadi = (string)Session["KullaniciAdi"].ToString();
    var mesajlar = db.TBLMESAJLAR.Where(x => x.alici == uyeadi.ToString()).ToList();
    return View(mesajlar);
}

[Authorize]
public ActionResult GidenMesajlar()
{
    var uyeadi = (string)Session["KullaniciAdi"].ToString();
    var mesajlar = db.TBLMESAJLAR.Where(x => x.gonderen == uyeadi.ToString()).ToList();
    return View(mesajlar);
}

[Authorize]
public ActionResult MesajSil(int id)
{
    var msj = db.TBLMESAJLAR.Find(id);
    db.TBLMESAJLAR.Remove(msj);
    db.SaveChanges();
    return RedirectToAction("Index");
}

[HttpGet]
public ActionResult YeniMesaj()
{
    return View();
}

[Authorize]
[HttpPost]
public ActionResult YeniMesaj(TBLMESAJLAR m)
{
    var uyeadi = (string)Session["KullaniciAdi"].ToString();
    m.gonderen = uyeadi.ToString();
    m.tarih = DateTime.Parse(DateTime.Now.ToShortDateString());
    db.TBLMESAJLAR.Add(m);
    db.SaveChanges();
    return RedirectToAction("GidenMesajlar");
}

```

Şekil 4.23 : Profilim - Üye Mesajlar ActionResult'ları

```

[Authorize]
public ActionResult AdminGelenMesajlar() // kullanıcıya gelen mesajlar.
{
    var uyeadi = (string)Session["KullaniciAdi"].ToString();
    var mesajlar = db.TBLMESAJLAR.Where(x => x.alici == uyeadi.ToString()).ToList();
    return View(mesajlar);
}

[Authorize]
public ActionResult AdminGidenMesajlar()
{
    var uyeadi = (string)Session["KullaniciAdi"].ToString();
    var mesajlar = db.TBLMESAJLAR.Where(x => x.gonderen == uyeadi.ToString()).ToList();
    return View(mesajlar);
}

[Authorize]
public ActionResult AdminMesajSil(int id)
{
    var msj = db.TBLMESAJLAR.Find(id);
    db.TBLMESAJLAR.Remove(msj);
    db.SaveChanges();
    return RedirectToAction("AdminGelenMesajlar");
}

[Authorize]
[HttpGet]
public ActionResult AdminMesajGonder(int id)
{
    var uye = db.TBLUYE.Find(id);
    ViewData["KullaniciAdi"] = uye.KULLANICIADI;
    ViewData["uyeid"] = id;
    return View();
}

[Authorize]
[HttpPost]
public ActionResult AdminMesajGonder(TBLMESAJLAR m)
{
    var uyeadi = (string)Session["KullaniciAdi"].ToString();
    m.gonderen = uyeadi.ToString();
    m.tarih = DateTime.Parse(DateTime.Now.ToShortDateString());
    db.TBLMESAJLAR.Add(m);
    db.SaveChanges();
    var uye = db.TBLUYE.Find(ViewData["uyeid"]);
    return RedirectToAction("AdminGidenMesajlar", "Mesajlar");
}

```

Şekil 4.24 : Profilim - Admin Paneli Mesajlar ActionResult'ları

5. KULLANILAN TEKNOLOJILER

5.1 Visual Studio

Visual Studio, birçok programlama dilini kullanarak program, uygulama ya da web sitesi yapabileceğiniz bir IDE yani entegre geliştirme ortamıdır. Microsoft Windows için bilgisayar programları, web siteleri, web uygulamaları, web hizmetleri ve mobil uygulamalar geliştirmek için kullanılır.

Visual Studio, Windows API, Windows Forms, Windows Presentation Foundation, Windows Store ve Microsoft Silverlight gibi Microsoft yazılım geliştirme platformlarını kullanır. Hem yerel kod hem de yönetilen kod üretebilir.

Visual Studio, IntelliSense'i (kod tamamlama bileşeni) ve kod yeniden düzenleme işlemini destekleyen bir kod düzenleyici içerir. Entegre hata ayıklayıcı, hem kaynak düzeyinde hata ayıklayıcı hem de makine düzeyinde hata ayıklayıcı olarak çalışır. Diğer yerleşik araçlar arasında bir kod profili oluşturucu, GUI uygulamaları oluşturmak için form tasarımcısı, web tasarımcısı, sınıf tasarımcısı ve veritabanı şeması tasarımcısı bulunur. Neredeyse her düzeyde işlevselliği artıran eklentileri kabul eder.



Şekil 5.1 : Visual Studio

Visual Studio, farklı programlama dillerini destekler ve dile özgü bir hizmet olması koşuluyla, kod düzenleyicisinin ve hata ayıklayıcının hemen hemen tüm programlama dillerini desteklemesine olanak tanır. Yerleşik diller arasında C, C++ ve C++ / CLI

(Visual C ++ ile), VB.NET (Visual Basic .NET ile), C (Visual C ile), F (Visual Studio 2010'dan itibaren) ve TypeScript (Visual Studio 2013 Update 2'den itibaren) bulunur.

Diğer dillerin yanı sıra Python, Ruby, Node.js ve M gibi diğer dilleri destekleme, ayrı olarak yüklenen dil servisleri aracılığıyla sağlanır. Aynı zamanda XML / XSLT, HTML / XHTML, JavaScript ve CSS'yi de destekler. Java (ve J) geçmişte destekleniyordu.

5.2 ASP.Net MVC

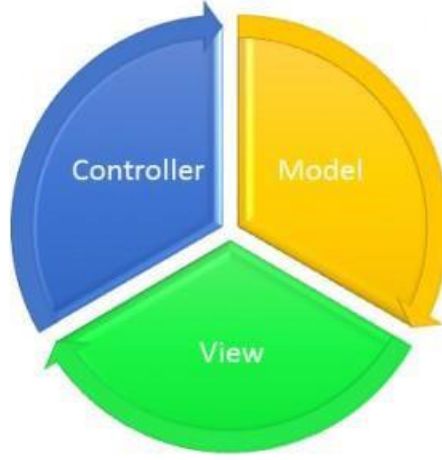
ASP.NET MVC, MVC pattern'ini ASP.NET'e eklemek için Microsoft'un geliştirdiği framework'tür. ASP.NET MVC'nin ne olduğunu anlamak için öncelikle MVC'nin ne olduğunu incelemekte fayda var.

MVC, uygulama geliştirmede (özellikle web uygulaması geliştirmede) önemli yere sahip mimari desenlerden biridir. Günümüzde MVC denince akla Microsoft'un geliştirdiği ASP.NET MVC Framework gelmektedir, oysa 1979 yılından beri (Microsoft 1975 yılında kurulmuştur) yazılım dünyasında yer almaktadır.

MVC, Model , View , Controller kelimelerinin baş harflerinden oluşur ve her kelime MVC'nin farklı bir katmanını ifade eder.

5.2.1 Model

MVC dünyasında model uygulama verisinin veya durumunun saklandığı yerdir, genellikle veritabanı veya xml/json dosyası formatındadır. Model, veri katmanını (database, xml, json dosyası, vb.) uygulamadan izole eder, böylece diğer katmanlarda veri katmanının neresi olduğunun bilinmesine gerek kalmaz. Model katmanı sıklıkla Entity Framework, Nhibernate, LLBLGen, vb. gibi araçlar kullanılarak oluşturulur.



Şekil 5.2 : Model View Controller

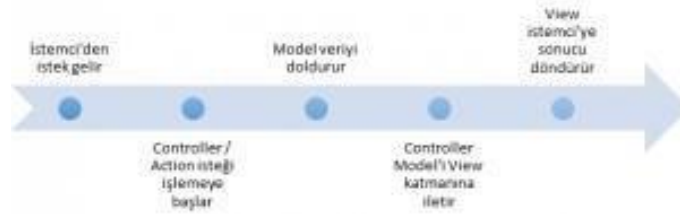
5.2.2 View

View, istemcinin gördüğü arayüzü içeren katmandır, genellikle Model katmanındaki verinin kullanılması ile oluşturulur. View katmanının Model ve Controller katmanlarından ayrılması ile arayüz değişikliklerinin uygulamanın diğer katmanlarını değiştirmeye gerek kalmadan yapılabilmesi sağlanmıştır.

View katmanında HTML5 ve CSS3 gibi son versiyon teknolojiler kullanmak mümkündür. HTML5 ve CSS3 ile masaüstü ve mobil tarayıcılarda çalışabilen uygulamalar geliştirmek çok kolaylaşmıştır. Hatta Windows Store uygulamaları geliştirmek için HTML5 ve CSS3 teknolojilerinden yararlanılabilir.

5.2.3 Controller

Controller, istemciden gelen isteği işlemek, Model ve View katmanları arasında köprü olmak gibi görevleri yerine getirir. Controller içerisinde bir veya daha fazla Action olabilir, genellikle her Action bir web sayfası üretmek için kullanılır.



Şekil 5.3 : MVC Life Cycle

MVC'nin diğer bir önemli yapıtaşı Routing mekanizmasıdır.

5.2.4 Routing

Routing, istemcinin uygulamaya yaptığı isteği uygun Controller ve Action'a yönlendiren yapıdır. İstemci, isteği uygulamanın belli bir adresine gönderir, routing mekanizması sayesinde ilgili adres için en uygun Controller ve içerisindeki Action tespit edilir ve çalıştırılır.

5.3 MVCGrid

Ajax sayfalama ve sıralama ile ASP.NET MVC ve Bootstrap için bir grid sistemi.

Ayrıca filtreleme desteği, csv'ye dışa aktarma, geri düğmesi desteği vardır.

MVCGrid'in birincil amacı, ihtiyaç duyduğunuz ek ön uç etkileşimini kolayca eklemenize olanak tanıyan çok basit ve kullanımı kolay bir istemci tarafı API'si ile birlikte temel grid işlevselliğini sağlamaktır. Projenize çok kolay bir şekilde ekleyebilir ve minimum çabayla çalıştırabilirsiniz, ancak gerektiğinde özelleştirmek için genişletilebilirliğe de sahiptir.

5.3.1 Özellikleri

- Mevcut model nesnelerinizi kullanır.
- AJAX kullanarak sunucu tarafında sıralama ve sayfalama yapar.
- Yerleşik csv'ye dışa aktarma.
- Minimum istemci koduyla filtreleme ve sütun desteği sağlar.

5.4 Bootstrap

Açık kaynak kodlu, ücretsiz bir CSS framework'tür, tasarım aracıdır. Kolayca telefonlar, tabletler ve masaüstü bilgisayarlar için farklı ve cihaz büyüklüğüne uygun şekilde sitenizin gözükmelerini sağlayan temalar/tasarımlar yapabilirsiniz. Bir site için gerekli olan tüm elemanları (form öğeleri, etiketler, tablolar, uyarı ve bilgi metinleri, navigasyon bar, sayfalandırma modülü, açılan menüler, farklı özellikte butonlar vb. bir çok tasarım öğesini) içinde barındıran Bootstrap tasarım yaparken bu hazır elementleri kullanarak bütün cihazlara uygun tasarımlar geliştirmenize yarar. Yani her şeyi hazır olan kodlarla yeni bir tasarım ortaya koymak çok basit ve pratiktir. Stiller, imajlar ,

jşler daha önceden sistemin içine monte edilmiştir. Sizin yapacağınız tek şey bunları çağırmaktır.



Şekil 5.4 : Bootstrap

Bir web sitesi için gerekli olan tüm öğeleri içerdiği için bir siteyi komple bir bütün olarak tasarlayabilirsiniz. Tipografik öğeler, tablolar, imajlar, slider, carousel, modal pencereler, butonlar, dropdown menüler, navigasyon bar, sayfalandırma, etiketler, thumbnail uyarı ve bilgilendirme balonları, yükleme barları vb. gibi bir çok tasarım ögesi hazır olarak sunulur.

5.5 JQuery

jQuery popüler bir JavaScript kütüphanesidir. 2006 yılında, John Resig tarafından geliştiricilerin web sitelerde JavaScript kullanımını kolaylaştırma amacıyla yaratılmıştır. Ayrı bir program dili değildir ve JavaScript ile birlikte çalışır. jQuery ile çok daha fazlasını çok daha azıyla yapabileceksiniz. jQuery nedir ve nasıl çalışır, bu makalede öğreneceksiniz.

Kod yazmak yorucu hale gelebilir, özellikle de birçok string varsa. jQuery çoklu kod satırlarını tek bir koda sıkıştırır, böylece tek bir göreve ulaşmak için bloklarca kodu yeniden yazmak zorunda kalmazsınız.

5.6 Microsoft SQL Server

Windows firmasına ait bilgi yönetim ve depolama ürünü olan MSSQL, işletmeler için önemli bir sistem. Microsoft SQL Server ile veri işleme, indeksleme, sorgulama, raporlama gibi çeşitli işlemler yapılabiliyor. Bu veritabanı gelişmiş, özellikleri belli meblağlar karşısında sizlere sağlıyor. Masaüstü uygulama ve programları kullanacak olanların ve internet sistemleri tasarlayıp hazırlayanların veritabanı gereksinimini iyi derecede karşılıyor.



Şekil 5.5 : MS-SQL

Sağladığı veritabanı özelliklerinin diğerlerine oranla daha iyi sonuçlar veriyor olması Microsoft temelli veritabanı yönetim sistemi ürününün başarısını olumlu etkiliyor. SQL Server, İngilizce tabiri ile Relational Database Management System olarak adlandırılan “İlişkisel Veritabanı Sistemi” ve kısaca RDBMS olarak da biliniyor. Bu sistem içerisinde veriler tablo şeklinde tutuluyor. Tablolar birbirleri ile ilişkisel açıdan bağlantılı olabiliyor.

Microsoft SQL Server 2017 sürümlerin ölçeklendirme özellikleri farklılık gösteriyor. Örneğin maksimum çekirdek sayıları, bellek kapasiteleri, arabellek havuzu boyutları, maksimum veritabanı boyutları versiyonlara göre değişiyor. Bunlar veri kaybı yaşanmaması için özel bir çalışma mantığından güç alıyor. Profesyonel yazılım uzmanları, büyük ölçekli e-ticaret ağları SQL Server sürümlerine yöneliyor, süreçlerin işleyişine uygun olarak paket versiyonları arasında değişikliğe gidebiliyor.

6. UYGULAMADA YAPILABİLECEK GELİŞTİRMELER

- Oyun profili puanlama sistemi geliştirilebilir. Mevcut sistemde üyeler birbirine sınırsız bir şekilde puan kullanabiliyor. Bu sistemde bir zaman kısıtlaması kullanılarak, kullanıcıların aynı kullanıcıya kötü niyetli puan kullanımını engelleyebilir.
- Kullanıcıların birbirlerine puan kullanmasını teşvik etme amacıyla ödüller sistemi eklenebilir. Oyun profilleri puan sıralamalarında ilk 5 sırada yer alan üyelere oyun içi ödüller verilebilir.
- Oyun profillerinin, oynadığı oyunlar bu profillerde gösterilebilir. Böylece kullanıcıların eşleşmelerden daha çok verim alması sağlanabilir.
- Oyun profillerine yorumlama sistemi gelebilir. Böylece kullanıcıların eşleşmelerden daha çok verim alması sağlanabilir.

Kaynakça

- 1- <https://dotnet.microsoft.com/apps/aspnet/mvc>
- 2- <http://www.borakasmer.com/asp-net-mvc-nedir-ne-ise-yarar/>
- 3- <https://visualstudio.microsoft.com/tr/>
- 4- <https://www.gencprogramci.org/microsoft-visual-studio-nedir-ne-ise-yarar/>
- 5- <https://archive.codeplex.com/?p=gridmvc>
- 6- <https://www.microsoft.com/tr-tr/sql-server/sql-server-downloads>
- 7- <https://www.penta.com.tr/markalar/microsoft/urunler/microsoft-sql-server/microsoft-sql-server-nedir/>
- 8- <https://jquery.com>
- 9- <https://www.argenova.com.tr/jquery-nedir-neye-yarar>
- 10- <https://getbootstrap.com>
- 11- <https://weebtasarim.com/Bootstrap-nedir-ve-nasil-kullanilir>
- 12- <https://www.udemy.com/course/mvc5-ile-3-panelli-kutuphane-yonetim-sistemi/>
- 13- <https://www.udemy.com/course/asp-net-mvc-5-ile-grid-mvc-kullanimini-ogrenin/>

ÖZGEÇMİŞ

Ad Soyad: Berkay ŞENOL

Doğum Tarihi ve Yeri: 13/04/1998

E-Posta: bsenol1903@gmail.com

ÖĞRENİM DURUMU:

- **Lise:** 2016, Bursa Fen Koleji
- **Lisans:** 2021, Pamukkale Üniversitesi, Bilgisayar Mühendisliği

MESLEKİ DENEYİMLER VE ÖDÜLLER:

- Staj I: Yeni Nesil Bilişim Teknolojileri, 2019