

**T.C.**

**DÜZCE ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**GERÇEK ZAMANLI BİLGİ YARIŞMASI**

**EMRE KAYMAK 141001022  
FATİH KARAÜZÜM 151001052**

**MERT SAİT VAROL 151001052**

**LİSANS BİTİRME TEZİ**

**DANIŞMAN**

**PROF. DR. RESUL KARA**

**DÜZCE, 2019**

T.C.

DÜZCE ÜNİVERSİTESİ

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**GERÇEK ZAMANLI BİLGİ YARIŞMASI**

Emre KAYMAK, Fatih KARAÜZÜM, Mert Sait VAROL tarafından hazırlanan bitirme tezi çalışması aşağıdaki danışman tarafından Düzce Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü’nde **LİSANS BİTİRME TEZİ** olarak kabul edilmiştir.

24 Mayıs 2019

(İmza)

Prof. Dr. Resul KARA

**BEYAN**

Bu tez çalışmasının kendi çalışmamız olduğunu, tezin planlanmasından yazımına kadar bütün aşamalarda etik dışı davranışımınız olmadığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimizi, bu tez çalışmasıyla elde edilen bütün bilgi ve yorumlara kaynak gösterdiğimizi ve bu kaynakları da kaynaklar listesine aldığımızı, yine bu tezin çalışılması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımızın olmadığını beyan ederiz.

24 Mayıs 2019

(İmza)

Emre KAYMAK

(İmza)

Fatih KARAÜZÜM

(İmza)

Mert Sait VAROL

**TEŞEKKÜR**

Lisan eğitimi kapsamında geliştirdiğimiz bu çalışmada bizlere değerli zamanını ayıran ve her konuda bizleri doğru şekilde yönlendiren değerli danışman hocamız Prof. Dr. Resul KARA ’ya en içten dileklerimizle teşekkürlerimizi sunarız

Aldığımız mühendislik eğitiminde, iyi bir mühendis olabilmemiz için gerekli olan değerli bilgilerini bizlerden esirgemeyen tüm bölüm hocalarımıza ve asistanlarına teşekkürlerimizi sunar ve saygılarımızı iletiriz.

Her şeyden öte, eğitimimiz süresince bize her konuda destek veren ailelerimize saygı ve sevgilerimizi sunarız.

**24 Mayıs 2019 Emre KAYMAK**

**Fatih KARAÜZÜM**

**Mert Sait VAROL**

İÇİNDEKİLER

ŞEKİL LİSTESİ

**Sayfa No**

KISALTMALAR

Git GitHub

IDE Integrated Development Environment – Tümleşik Geliştirme Ortamı

GUİ Graphical User Interface- Grafiksel Kullanıcı Arayüzü

TFVC [Team Foundation Sürüm Denetimi](https://docs.microsoft.com/tr-tr/azure/devops/repos/tfvc/index?view=azure-devops)

VB Visual Basic

HTML Hypertext Markup Language

CSS Cascading Style Sheets -Basamaklı Stil Şablonları

JS JavaScript

ORM Object Relational Mapper

MVC Model-View-Controller

EF Entity Framework

OOP Object Orient Programing

HTTP Hyper Text Transfer Protocol

TCP Transmission Control Protocol

PHP Hypertext Preprocessor

IOS iPhone/*iPad* Operating System

MSSQL Microsoft SQL Server

SQL Structured Query Language

ID Identification Number

DB Database

URL Uniform Resource Locator

ÖZET

Emre KAYMAK, Fatih KARAÜZÜM, Mert Sait VAROL

Düzce Üniversitesi

Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü

Lisans Tezi

Danışman: Prof. Dr. Resul KARA

Mayıs 2019, 100 sayfa

Eğlenerek öğrenmek eskiden masallar ve bilmeceler ile yapılırken günümüzde daha çok dijital dünyada gerçekleşmektedir. Bunu göz önüne alarak geliştirdiğimiz bu proje bir bilgi yarışması projesidir. Web tabanlı olan projemizde yarışmayı başlatması için bir kişi olması gerekiyor. Bu kişi daha önceden oluşturulan oyunları kullanabilirken kendisi de bir oyun oluşturabilir. Seçtiği oyunu başlattıktan sonra diğer kullanıcılar bu sınıfa giriş yaparak katılmış olurlar. Kullanıcılar giriş yaptıktan sonra yarışma başlamış olur. Kullanıcılara sorular tek tek gösterilerek cevaplaması beklenir. Kullanıcılar sorulara doğru cevap vermesi ile kazandığı puanlara göre sıralama oluşturulur ve en sonunda birinci belirlenir. Aynı zamanda projemiz üzerinden anket de yapılmaktadır. Anket oluşturma işlemleri aynı oyun sınıfındaki gibi gerçekleşmektedir. Ancak tek fark ankette puan ve sıralama işlemleri yapılmamaktadır.

Projemizi geliştirirken iki farklı programlama dili kullandık. Bunlardan birinci ASP.Net ikincisi ise JavaScript. ASP.Net ile projede sabit olan yerleri tasarlarken gerçek zamanlı işlemlerin yapıldığı kısımları tasarlamak için JavaScript kullandık. Gerçek zamanlı işlemlerini kodlarken JavaScript ’in Socket.IO kütüphanesini kullandık. Bu kütüphane bize istemci ile sunucu arasında gerçek zamanlı, çift yönlü iletişim sağlamaktadır. Bu işlemler sonunda oluşan verileri kaydetmek için veritabanı olarak MS SQL kullanıldı.

**Anahtar Kelimeler:** Bilgi Yarışması, ASP.Net, JavaScript, Socket.IO, MS SQL

ABSTRACT

Emre KAYMAK, Fatih KARAÜZÜM, Mert Sait VAROL

Düzce University

Engineering Faculty, Department of Computer Engineering

License Thesis

Supervisor: Prof. Dr. Resul KARA

May 2019, 100 pages

Eğlenerek öğrenmek eskiden masallar ve bilmeceler ile yapılırken günümüzde daha çok dijital dünyada gerçekleşmektedir. Bunu göz önüne alarak geliştirdiğimiz bu proje bir bilgi yarışması projesidir. Web tabanlı olan projemizde yarışmayı başlatması için bir kişi olması gerekiyor. Bu kişi daha önceden oluşturulan oyunları kullanabilirken kendisi de bir oyun oluşturabilir. Seçtiği oyunu başlattıktan sonra diğer kullanıcılar bu sınıfa giriş yaparak katılmış olurlar. Kullanıcılar giriş yaptıktan sonra yarışma başlamış olur. Kullanıcılara sorular tek tek gösterilerek cevaplaması beklenir. Kullanıcılar sorulara doğru cevap vermesi ile kazandığı puanlara göre sıralama oluşturulur ve en sonunda birinci belirlenir. Aynı zamanda projemiz üzerinden anket de yapılmaktadır. Anket oluşturma işlemleri aynı oyun sınıfındaki gibi gerçekleşmektedir. Ancak tek fark ankette puan ve sıralama işlemleri yapılmamaktadır.

Projemizi geliştirirken iki farklı programlama dili kullandık. Bunlardan birinci ASP.Net ikincisi ise JavaScript. ASP.Net ile projede sabit olan yerleri tasarlarken gerçek zamanlı işlemlerin yapıldığı kısımları tasarlamak için JavaScript kullandık. Gerçek zamanlı işlemlerini kodlarken JavaScript ’in Socket.IO kütüphanesini kullandık. Bu kütüphane bize istemci ile sunucu arasında gerçek zamanlı, çift yönlü iletişim sağlamaktadır. Bu işlemler sonunda oluşan verileri kaydetmek için veritabanı olarak MS SQL kullanıldı.

**Keywords:** Knowledge Contest, ASP.Net, JavaScript, Socket.IO, MS SQL

GİRİŞ

Günümüzde teknolojinin ilerlemesi ve eğitime verilen önemin artmasıyla, eğitim sorunlarının çözümünde teknolojik olanaklardan yararlanmak kaçınılmaz hale gelmiştir. Bu teknolojik olanaklardan birisi olan bilgisayar, içinde yaşadığımız yüzyılın temel kültür öğelerinden biri olup, kullanımı hızla yaygınlaşan bir araç haline gelmiştir [3].

Teknolojik alanda ve özellikle Internet’teki gelişmeler; bilginin, sınırsız ve kolay erişilebilir olmasının yanı sıra, ucuz, hızlı ve yaygınlaşan bir bilişim teknolojisi olmasına dikkat çekmektedir [4]. Bu şekilde kolay elde edilen bilginin kalıcı hale getirilebilmesi için geliştirdiğimiz bu sistemde öğrencilere bilgi aktarılırken hem de eğlendirmek amaçlanmaktadır. Öğrencilerin eğlenerek öğrendiği bilginin kalıcı olduğu görülmektedir. Bu bilgiler ışığında geliştirdiğimiz online bilgi yarışması sistemi öğrencilere hem öğretirken hem de eğlendirmektedir. Sistem sadece öğrenciler için değil bir araya gelen arkadaş grupları için de ortam sağlamaktadır. Yani kısacası eğlenerek öğrenmek isteyen herkese açık bir sistemdir. Sistemimiz aynı zamanda anket oluşturabilmek içinde tasarlanmıştır. Anket işlemlerinde kullanıcılardan cevapları aldıktan sonra sınıf otomatik olarak kapanır. Burada puanlama sistemi bulunmamaktadır.

Ekleme yapılıcak altarnetif

Sisteme giriş yapabilmek için ilk önce üye olmamız gerekir. Ancak oyunu katılacak tüm kullanıcıların üye olmasına gerek yoktur. Sadece oyunu veya anketi başlatan kişinin üye olması yeterlidir. Üye olma işlemi tamamlandıktan sonra kullanıcı, mail adresi ve şifresi ile sisteme giriş yapabilir. Kullanıcı giriş yaptıktan sonra daha önceden oluşturulan oyunları da başlatabilirken kendisi de oyun başlatabilir. Daha önceden oluşturulan oyunlar kategorilere göre sınıflandırıldığı için kullanıcı dostu olması sağlanmıştır. Kullanıcı bu sayede aradığı oyunu daha kolay şekilde bulabilmektedir. Bunun yanı sıra kullanıcının arama yapabilmesi için bir arama butonu bulunmaktadır. Kullanıcı buraya aramak istediği oyun veya anketi yazarak daha kolay şekilde bulabilmektedir. Eğer kullanıcı daha önceden oluşturulan sınıfların içerisinden istediğini bulamaz ise veya kendisi oyun oluşturmak isterse bunun içinde oyun oluşturma sekmesi bulunmaktadır. Burada kullanıcı burada kendisine bir sınıf oluşturup oyun ekledikten sonra yarışmayı başlatabilir. Oyun başlatıldıktan sonra diğer kullanıcıların giriş yapabilmesi beklenir. Bütün kullanıcılar giriş yaptıktan sonra yetkili kullanıcı oyunu başlatır ve ilk soru ekranda yansıtılır. Kullanıcılar ekrana yansıtılan sorudan doğru cevabı işaretleyerek puanları hesaplanır. Her sorunun sonunda sıralama gösterilerek kullanıcılara sıralama bilgisi verilir. Son sorudan sonra birinci belirlenerek oyun sonlandırılır.

1.SİSTEMİ ANALİZİ

1.1.GEREKSİNİMLERİN BELİRLENMESİ

Proje tasarlanmadan önce gereksinimler belirlendi. Bu gereksinimlerin en başında gelenler arasında yönetim ve güvenlik gelmektedir. Kullanıcıların bilgi güvenliğini sağlamak için kullanıcıların kayıt olmaları sağlanmıştır. Bu şekilde hem kendi kişisel bilgileri korunurken hem de oluşturduğu oyunların bilgileri korunmuştur. Diğer bir gereksinim olan yöneticiler içindir. Bu gereksinimde yöneticiler siteyi daha iyi yönetebilmesi için bir admin tarafı belirlenmiştir. Admin tarafında yöneticiler oyunlara müdahale edebilirken kullanıcı profillerini de görüntüleyebilmektedir. Bütün gereksinim listesi aşağıda ayrıntılı bir şekilde verilmiştir.

* Kullanıcı üye olması ve kullanıcı giriş
* Admin tarafı için ayrı bir panel
* Oyun ekleme, silme, güncelleme

Proje gereksinimleri belirlendikten sonra bu gereksinimlere uygun programlama dillerine karar verildi. Bunlar ASP.Net ve JavaScript tir. ASP.Net ile projede sabit olan yerleri tasarlarken gerçek zamanlı işlemlerin yapıldığı kısımları tasarlamak için JavaScript kullandık. Gerçek zamanlı işlemlerini kodlarken JavaScript ’in Socket.IO kütüphanesini kullandık. Bu kütüphane bize istemci ile sunucu arasında gerçek zamanlı, çift yönlü iletişim sağlamaktadır. Bu işlemler sonunda oluşan verileri kaydetmek için veritabanı olarak MS SQL kullandık. Bu programla dillerine uygun geliştirme ortamları olarak Microsoft Visual Studio, Visual Studio Code ve Microsoft SQL Server kullanıldı.



Şekil 1.1.1 Kullanılan Programlama Dilleri

1.2.VERİTABANI TABLOLARININ BELİRLENMESİ

Proje gereksinimleri belirlendikten sonra bu gereksinimlere uygun veri tabanı toplaları belirlendi. Toplam on sekiz tabloların isimleri alfabetik olarak şu şekildedir;

* Anket
* Anket Oturum
* Anket Soru
* Dil
* Favori
* İletişim
* Kullanıcı
* Kullanıcı Resim
* Menü
* Menü Rol
* Rapor
* Resim
* Resim Kategori
* Rol
* Sınıf
* Sınıf Kategori
* Sorular
* Takip

Veri tabanı tablolarının isimlerine karar verdikten sonra bu tabloların içine yazılacak sütunlara karar vermek için gerekli analizler yazıldı. Analizler sonunda sütunların isimlerine ve değişkenlerin tipleri belirlendi. İlk önce bütün tablolarda verilere erişmek için bir ID tanımlandı. Daha sonradan bütün tablolara kendi özelliklerine göre sütunlar yazıldı. Kâğıt üzerinde belirlenen tabloların birbirleriyle olan ilişkilerinin analiz edilmesi kaldı. Analiz sonunda birbiriyle ilişkili olan tablolar belirtilerek tabloları geliştirme ortamında oluşturmak kaldı. Bu işlemler daha ayrıntılı bir şekilde üçüncü bölümde anlatılacaktır.

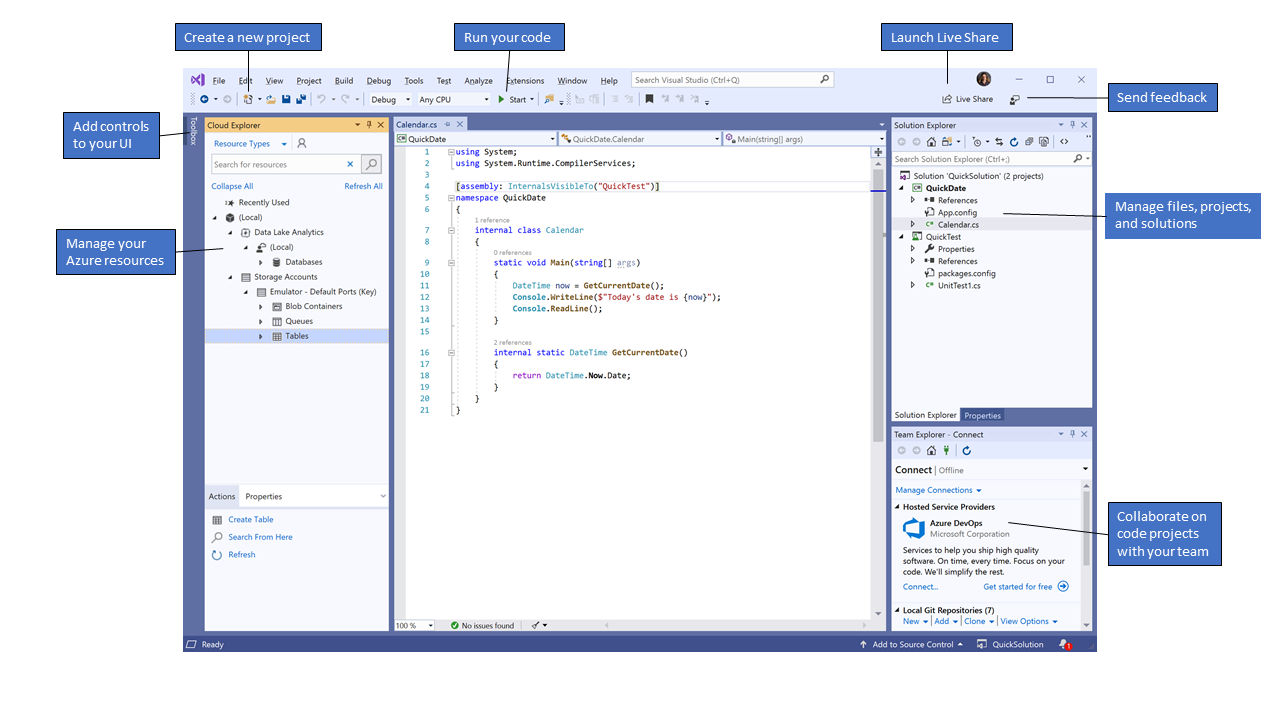
2.KULLANILAN GELİŞTİRME ORTAMLARI

2.1.MİCROSOFT VİSUAL STUDİO

2.1.1 Microsoft Visual Studio Nedir?

Visual Studio Microsoft tarafından geliştirilen ve Microsoft ürünlerinin üzerinde çalıştığı, web uygulamalı, konsol uygulamaları, web servisler ya da Windows formları gibi çeşitli uygulamalarının geliştirile bilineceği bir bütünleşik yazılım ortamıdır. Visual Studio aslında yazılım geliştirmek için bize kolaylıklar ve bazı imkanlar soran bir yazılım (IDE) dır.

Visual Studio aynı zamanda fonksiyonellikler sağlayan güzel bir arayüze sahiptir. Bu ara yüzde kod editörü, GUİ tasarlama aracı, veritabanı şema ekranı, Azure bağlantı seçenekleri, debugger gibi birçok seçenek bulunmaktadır.



Şekil 2.1.1.Visual Studio Ekranı

Visual Studio kodu düzenlemek, hata ayıklamak ve yayınlamak için yazılmış bir IDE dir.

2.1.2 Microsoft Visual Studio’nun Bazı Özellikleri

* [Solution Explorer](https://docs.microsoft.com/en-us/visualstudio/ide/solutions-and-projects-in-visual-studio?view=vs-2019): Sağ üstte bulunan projelerinizi görüntüleyebileceğiniz, kodlarınızı açabileceğiniz ve yönetmenizi sağlayan ekrandır. Projenizi ve dosyalarınızı gruplayarak uygulama geliştirme aşamasında size kolaylık sağlar.
* Editör Ekranı: Genellikle zaman geçirdiğimiz ekrandır(ortada). Burası kodlarımızı görüntüleyebildiğimiz ve düzenleyebildiğimiz ekrandır.
* Team Explorer: Sağ altta bulunan Git ve TFVC gibi teknolojileri kullanarak ekip halinde çalışmayı kolaylaştıran ve ortak proje geliştirilmesine olanak sağlayan bir seçenektir.

2.1.3 Neden Visual Studio Seçildi?

Visual Studio seçmemizin nedeni; projemizin bir kısmında kullandığımız C# dili ve MVC mimarisini geliştirebileceğimiz en uygun ortam olmasıydı. Hata ayıklamada ki başarısı ve canlı olarak hata göstermesi çok büyük kolaylık oluşturma. Aynı zamanda Git gibi bir platform içinde sunması da ortak proje geliştirmemizde ekip arasında kolaylık sağladı.

2.2 ASP .NET

2.2.1 Asp .Net Nedir?

Asp .Net Microsoft tarafından geliştirilen .NET Framework temelli bir çatıdır. Çatı denmesinin sebebi Asp .Net in bir dil olmayışı, farklı dillerde bu teknolojiyi kullanarak web uygulamaları geliştirilebilmesidir. ASP.NET ile sadece HTML tabanlı statik sayfalar dan ziyade, kullanıcı etkileşimli dinamik sayfalar geliştirebilmektedir. Örneğin günümüzde e-ticaret siteleri, haber siteleri, bazı kurumsal uygulamalar, forum siteleri ASP.NET ile geliştirilebilir ve yönetilebilir. Asp .Net server tabanlı çalışmaktadır. Yani Html, JavaScript, Css gibi istemci tarafından değil direk serverda çalışır ve istemci isteklerini istemci tarafında gönderir. Daha sonra istemci web uygulaması bunu derleyerek gösterir.

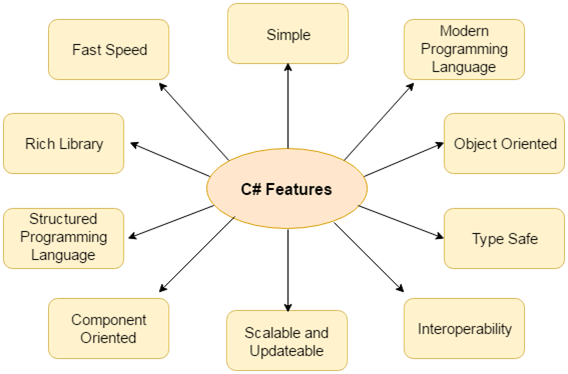
Bu sayede istemci kodlara ulaşamaz, güvenlik te sağlanmış olunur. Asp .Net kullanmak için illa bir dil bilmek gerekmez. Örneğin C# ile de bu yapıyı kullanabilirken, VB ile de Asp .Net kullanılabilmektedir. Çatı denmesinin nedeni de bundan kaynaklıdır.

2.3. C#

2.3.1 C# nedir?

C# Microsoft tarafından geliştirilen .Net Framework için hazırlanmış nesne yönelimli bir programlama dilidir. Yani nesneler önceden sınıflar halinde yazılıdır. Programcıya sadece o nesneyi kullanmak ve sonrasında nesneyi amaca uygun çalıştıracak kodları yazmak kalır.

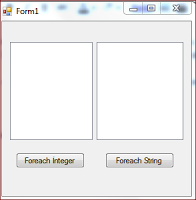
C# VB nin yüksek verimliliğini ve C++ ın sade gücünü birleştirmeyi amaçlar. C # kullanarak konsol uygulaması, Windows form uygulaması ve web uygulamaları yada servis uygulamaları geliştirebiliriz. Güvenlik amacıyla C++ da olduğu gibi doğrudan bellek yönetimine izin vermez. .Net temelli olduğu için otomatik bellek yönetimi yapılır. Sınıflar ve veri türleri, tüm .NET dillerinde olduğu gibi C# için de ortaktır.



Şekil 2.3.1.C# özellik tablosu

2.3.2 C# ile Neler Yapılabilir?

* Windows Form Uygulamaları (Ado.Net): Grafiksel kullanıcı ara yüzü olarak tanımlanır. Konsol uygulamasından ayıran özelliği burada bir GUİ ekranı bulunmasıdır. Bu ekranı toolbox seçeneğinden özelleştirebiliriz.



Şekil 2.3.2.1 C# GUI Ekranı

Windows form uygulamalarında veritabanı bağlantısı kurabilmek için kullanılan .Net kütüphanesinin adı Ado.Net kütüphanesidir. Bu kütüphaneyi kullanarak veritabanına entegre uygulamalar geliştirebiliriz.

* Web Uygulaması (Asp.Net): C# kullanarak geliştirebileceğimiz bir alan da web uygulamalarıdır bu uygulamalar Asp .Net çatısı altında yapılır. Bizim de kullandığımız kısım bu kısımdır.
* Mobil Programlama: Windows Phone işletim sistemine sahip telefonlar için uygulama geliştirme kısmıdır. Bu seçenek sayesinde mobil uygulama geliştirilebilir.
* Web Servisleri
* DLL yazma

2.3.3 Neden C# Dilini Seçtik?

* C# seçmemizin ana sebeplerinden birisi MVC mimarisini kullanacağımızdan C# kullanmanın çok kolaylık sağlayacak olmasıydı.
* Visual Studio da yazdığımız C# kodlarımızı, Visual Studio nun araçlarıyla hata ayıklamasının ve kod geliştirme süreci çok daha efektif olmakta.
* C# ın nesne yönelimli bir dil olmasıyla projemize tamamen uygun olması ve nesne class yapısını çok kolay uygulayabilmemiz.
* .Net Framework üzerine kurulduğu için Entity Framework gibi güçlü bir ORM aracını kullanabilecek olmamız.
* .Net Framework ün çok fazla kütüphanesi bulunması ve bunları C# ta kullanabilecek olmamız.

2.4 ENTİTY FRAMEWORK

2.4.1 Entity Framework Nedir?

Entity Framework Microsoft tarafından geliştirilen bir ORM(Object Relational Mapping) araçlarından biridir. ORM nedir dersek: İlişkisel veritabanı ile nesneye yönelik programlama(OOP) arasında bir köprü görevi gören araçtır. Bu köprü, ilişkisel veritabanında ki verilemizi yönetmek için nesne modellerimizi programımız da kullanabildiğimiz bir yapıdır. Yani veritabanına bizim ürettiğimiz nesnelerimizi bağlayan ve bizim için veri alışverişini yapan bir framework’tür. Aynı zamanda veritabanında ki tablolarımızı da class lar şeklinde projemize eklememize olanak sağlayan yapıdır. Bu yapı sayesinde veritabanında ki tablolarımız ile aynı özellikte class larımız oluşur ve biz bu class lardan nesne türeterek işlemlerimizi yapabiliriz.

Entity Framework ile 4 farklı yöntem kullanarak proje geliştirebiliriz. Bu yöntemler;

* Model First (Yeni Veritabanı Oluşturma Visual Studio İle): Bu yöntem ile Visual Studio üzerine boş bir model dosyası (.edmx) ekleyerek veritabanını bu model üzerinde tasarlayabilmeye imkan sağlıyor. Derleme anında verilen script dosyası veritabanını oluşturuyor. Aynı zamanda veritananında ki tablolarımızı projemize eklediğimiz de .edmx uzantılı bir model dosyası oluşmaktadır.
* Database First (Var Olan Veritabanını Kullanma): Bu yöndem önceden oluşturduğumuz veritabanını projeye model olarak bağlayarak gerekli model classlarımız Entity Framework tarafından oluşturulmaktadır.
* Code First (Yeni Veritabanı Kod Yazarak): Model classlarımızı Visual Studio tarafında oluşturmaya başladığımız bir yöntemdir. Veritabanımız bu classlardan oluşmaktadır. Yani ilk olarak veritabanı oluşturmadan tablolarımızı Visual Studio da yazarak projeyi derlediğimizde veritabanında oluşmasını sağlamaktayız.
* Code First(Var Olan Veritabanını Kullanma): Bu yöntemde ise var olan bir veritabanını projemize ekleyerek veritabanımızda ki her bir tablonun projemizde EF tarafından bir model class ları ile temsil edilmesi yani oluşturulması sağlanmaktadır.

2.4.2 Entity Framework Avantaj ve Dezavantajları

Avantajları;

* OOP olarak proje geliştirmemize imkan verir.
* Hiç SQL bilgisi olmayan birisi bile veritabanı işlemlerini EF ile yapabilir.
* Herhangi bir veritabanına bağımlılık yoktur. Oracle, MS SQL ile kullanılabilir.
* Code First sayesinde projenizde veritabanınızı yanınızda taşıma gereği duymadan istediğiniz yerde oluşturabilirsiniz. Buda size büyük bir esneklik kazandırmaktadır.
* Yazılım geliştirme zamanını kısaltır.
* Yazılım geliştirme maaliyetini azaltır.

Dezavantajları;

* En büyük sorunu performanstır. Ado.Net gibi hızlı bir performansı maalesef yoktur. Ancak bu yavaş olduğu manasına gelmez.
* Veritabanından veri alış-verişi yapacağımız zaman kontrol bizde değil Entity Frameworktedir. Yani arka planda veritabanı işlemleri için kendisi sorgu oluşturmaktadır. Basit bir işlem için karmaşık bir sorgu oluşturabilmektedir. Bu sorguları görüntüleyebilmekteyiz.
* Syntax’ı yeni kullanacak kişiler için karmaşık gelebilir ancak zamanla alışılacaktır.

2.4.3 Neden Entity Framework kullandık?

Entity Framework kullanmamızın ana sebebi veritabanı işlemlerini SQL sorguları ile uğraşmadan çok hızlı bir şekilde gerçekleştirebiliyor olmamız. Bu bize çok ciddi zaman kazancı sağlamaktadır. Ve proje geliştirme aşamasını çok kolaylaştırmaktadır. Ayrıca OOP ye gayet uygun yapısı ve güçlü bir ORM aracı olmasıyla çok büyük kazanç sağlamaktadır.

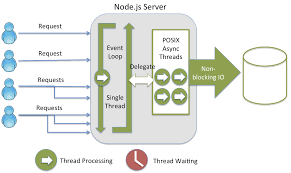
2.5.NODE.JS

2.5.1 Node.js Nedir?

Node.js Javascript kullanarak server side uygulamalar yapabileceğimiz, Joyent tarafından 2009 yılında geliştirilmeye başlanmış bir Javascript Runtime platformudur. V8 javascript motoru üzerinde çalışır. V8 motoru Google tarafından geliştirilmiş C, C++ ve javascript dilleri ile yazılan bir enginedir. Amacı javascript kodlarını makine koduna çevirmektir. Node.js in en büyük avantajlarından birisi de çok performanslı olmasıdır. Asenkron olarak çalışabilen çoklu eventler oluşturulabilir. Response hızları çok yüksektir.

Node.js ayrıca zaman ve kaynak konusunda oldukça iyidir. Mesela çok yüksek trafikli sitelerde anlık istekler, aynı signalR’da olduğu gibi web socket kullanılarak client’ın ekranına çok kısa zamanda ve az kaynak tüketilerek gönderilebilir. Mesela bir servisten çekilen borsa bilgilerinin tüm clientlara gönderilmesi gibi. Bu işlemi kullanıcı sorgulamadan borsa bilgileri değiştikçe yapılacaktır. Gerçek zamanlı (real time) uygulamalar web soket teknolojisi kullanılarak node.js ile performanslı bir şekilde yapılabilir.

Node.js, asenkron çalışan, [açık kaynaklı](https://tr.wikipedia.org/wiki/A%C3%A7%C4%B1k_kaynak), işlem bazlı (event based), ölçeklenebilir bir uygulama geliştirme çatısıdır. Uygulamaların sunucu (server) tarafında kullanılır.[1]



Şekil 2.5. 1.Node.js Yapısı

Node.js’e geçen popüler firmalardan biri Linkedin’dir. Mobil sunucu tarafında Node.js ’e geçmeden önce Rails kullanıyordu. Node.js ’e geçtikten sonra server maliyeti 1/10 düşmüştür. Bazı işlemlerde 20 kata varan hız artışı olmuştır. Paypall çalışma platformunu riske etmemek için, direkt Node.js’e geçişi yerine Java ve Node.js olarak 2 platformda paralel olarak geliştirme yapmıştır. Sonuçta Node.js java uygulamasına göre 2 kat daha fazla saniyelik istemi karşılamıştır.

Üstelik Node.js tek çekirdekli bir işlemci üzerinden çalışırken java uygulaması beş çekirdekli işlemci üzerinde çalışmaktadır. Yani Node.js maliyeti 1/5’e indirmiştir. Ayrıca response time’ı %35 hızlanmıştır.

Bir yazılımın hem arkayüz (backend) hem önyüz (frontend) kodlarını aynı dilde yazma gibi çok büyük bir avantajı vardır.

Günümüzde çoğu uygulama, zamanının çoğunu veritabanlarına veya Internet’deki çeşitli servislere istek yapıp gelen sonuçları beklemekle geçirir. Node.js, asenkrondur. Diğer platformlar gibi beklemez. İstekleri teker teker yapmak yerine paralel olarak yapar. İstekler bittikleri zaman Node’u “geri ararlar” (callback). Bu sayede anlık olarak çok sayıda istek hızlı bir şekilde işlenebilir.

Örnek vermek gerekirse diğer senkron servisler bir markette ki kasa gibidir. Yani birisinin işini bitirmeden diğerine geçemez. Ancak node.js ise starbucks gibi önce siparişleri alır işlem bitince müşteriyi çağırıp kahvesinin vermektedir. Bu sayede asenkron çalışarak çok büyük bir zaman tasarrufu sağlar.

2.5.2 Neden Node.js Kullandık?

Node.js kullanmamızın sebepleri aslında bir önceki kısımda açıklanmıştır. Node.js in sahip olduğu tek çekirdek ve asenkron çalışma mantığı sayesinde ve sunucu tarafında çalışabilmesi, hem front-end hemde back-end tarafının tek dil üzerinden kodlanabilmesi, çok yüksek trafiklerde cevap sürelerinin çok düşük olması gibi birçok avantajdan ötürü Node.js platformunu seçtik. Projemizde bulunan canlı, online test aşaması çok sayıda istemcinin aynı anda bağlanması ve gecikmesiz şekilde hepsine aynı anda ulaşması gerekliliğinden bu iş için en uygun platformlardan birisinin Node.js olacağına karar verdik.

2.6.JAVASCRİPT

2.6.1 JavaScript Nedir?

JavaScript, yaygın olarak web tarayıcılarında kullanılmakta olan bir betik dilidir. Aynı zamanda sunucu tarafında da kullanılabilmektedir JavaScript ile yazılan istemci tarafı kodlar sayesinde tarayıcının kullanıcıyla etkileşimde bulunması, tarayıcının yenilenmesi, asenkron bir şekilde sunucu ile veri alış verişi yapabilmesi ve web sayfası içeriğinin değiştirilmesi gibi özellikler sağlar. Aynı zamanda JavaScript, Node.js gibi platformlar sayesinde sunucu tarafında da etkin şekilde kullanılmaktadır.

İlk olarak bir Netscape çalışanı olan [Brendan Eich](https://tr.wikipedia.org/wiki/Brendan_Eich) tarafından geliştirilen ve 1997 yılından beri bir ECMA standardı olan JavaScript, günümüzde [Mozilla Vakfı](https://tr.wikipedia.org/wiki/Mozilla_Vakf%C4%B1) öncülüğünde özgür yazılım topluluğu tarafından geliştirilmekte ve bakımı yapılmaktadır.

Web sayfalarının görünümünün mükemmelleştirilmesi, işlevselliğinin artırılması ve dinamik web sayfaları tasarlanması amacıyla JavaScript, HTML kodu içinde gömülü olarak (embedded) kullanılabilir. JavaScript, bir programlama dili disiplini ve özelliklerine sahiptir.[2]

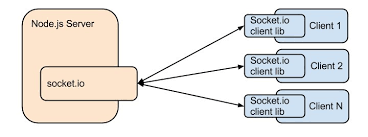
### 2.6.2.JavaScripti Neden Kullandık?

JavaScripti kullanma sebebimiz, Node.Js kullanarak bir socket programlama yapacak olmamızdı. Aynı zamanda front-end kısmında da çok büyük avantajlar sağlamasıyla projemizde yer aldı.

2.7.SOCKET.IO

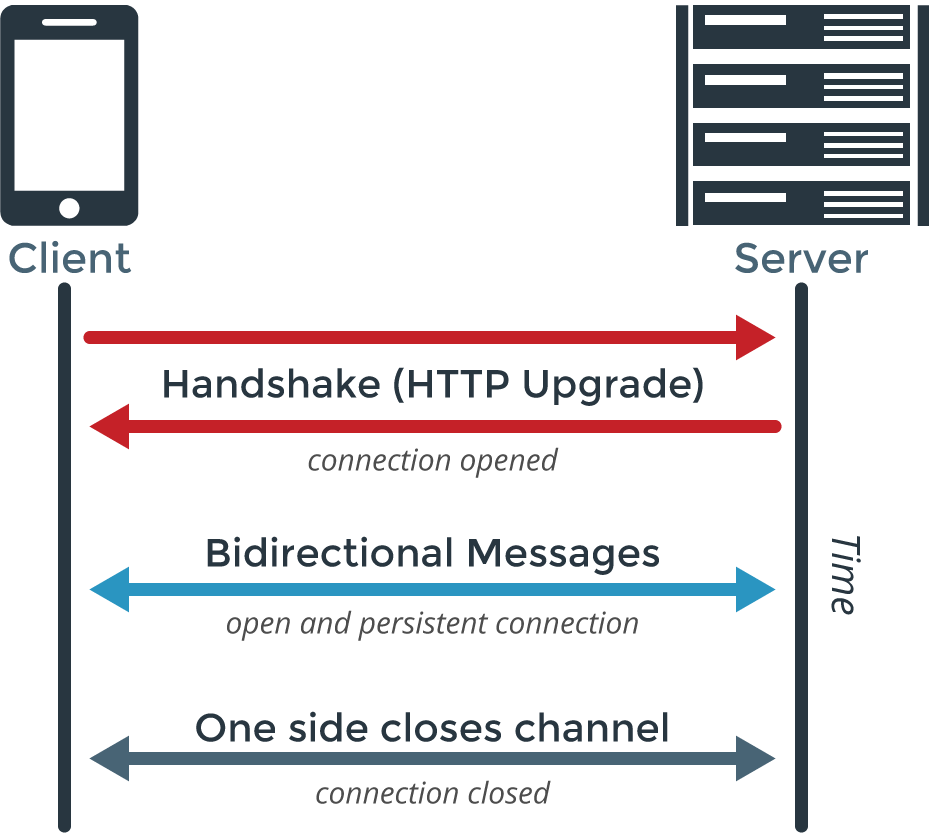
2.7.1 Socket.io Nedir?

Socket.io server ile client arasında gerçek zamanlı iletişimi sağlayan web socket tabanlı bir teknolojidir. Http 1.1 versiyonu ile birlikte request/response arasında tek bir TCP bağlantısı kullanarak veri alışverişi yapar. Socket.io bir NodeJS kütüphanesidir. Bu kütüphane ile gerçek zamanlı programlar yapabiliriz. Socket.io aslında JavaScript temelli bir kütüphanedir. Bu kütüphaneyi kullanarak gerçek zamanlı uygulamalar yapabiliriz.



Şekil 2.7.1.Socket.io ve Node.js Çalışma Yapısı

Standart web teknolojilerinde client server’a istek yapar (request) ve server bu isteğe cevap olarak response verir. Web socket teknolojisinde ise ters taraflı, yani server tarafından client’a bir data gönderilebileceği gibi aynı kanal üzerinden client’ ında data göndermesi mümkündür.



Şekil 2.7. 2.Web soket yapısı

2.7.2 Socket.io u Neden Kullandık?

Socket.io bize gerçek zamanlı uygulamalar yapmamız için çok efektif metotlar sunmaktadır. Bu sayede projemize tam uygunluk sağlamaktadır. Ayrıca Node.js üzerinde çalışması büyük bir hız artısı katmaktadır. Birbiri arasında ki uyum sorunsuzdur. Bu nedenle bizim de projemizin amacına uygun olarak gerçek zamanlı soruların geldiği ve istemcilerin cevapladığı kısımlarda socket.io kütüphanesini kullandık.

2.8 MVC

2.8.1 MVC Nedir?

Mvc yapısı yazdığımız uygulamanın kullanıcı arayüzü ile arka planını birbirlerinden ayıran, uygulamada farklı işleri yapan kısımların birbirleri arasında girmesini engelleyen bir mimaridir. Birçok platform da MVC mimarisi mevcuttur. Bunlardan bazıları .Net MVC, Spring MVC, Php Mvc, IOS MVC, Java MVC, JavaScript MVC gibi birçok platformda MVC mimarisi kullanılmaktadır. MVC mimarisi bir yazılım dili değildir bir geliştirme mimarisidir. Bu mimariden kastımız MVC ‘nin açılımı Model-View-Controller ‘dir. Mvc bir yazılım dili değil Bir mimaridir Bu mimaride 3 ana kısımımız var.

Bunlar Model-View ve Controller dır. Bu katmanlara bakarsak;

* Model: Veritabanına erişim, sınıflar(class),veritabanı ilişkileri gibi data ile ilgili işlemlerin yapıldığı katmandır. Bu katmanda bizim veritabanında ki tablolarımızın class olarak karşılıkları tutulur.
* **View:** Bu katman kullanıcının ekranda gördüğü katman olarak adlandırılır. Bu kısımda Html,Css,Javascript ara yüz teknolojileri kullanılır.
* **Controller:** Kullanıcının View vasıtasıyla verdiği isteklerin, Modele işlenmesini sağlayan katmandır. Kısacası Model ile View arasındaki katmandır. Metotlar, değişkenler, fonksiyonlar  buradan çağrılarak kullanılır. Aslında ana beyin controller dır.

MVC mimarisi ile view ve model katmanlarını birbirlerinden ayırmış oluyoruz. Bu sayede mesela daha sonra uygulamamızın tasarımını değiştirmek istediğimizde yalnızca view tarafında uğraşmak yetecektir.

2.8.2 Neden MVC Kullandık?

MVC kullanmamızın nedeni, MVC mimarisinin bize sağladığı modülerlik ve farklı işleri birbirinden ayrıştırma kolaylığı. Bu sayede html sayfalarımızı tasarlarken arka tarafa yazdığımız kodlarla ilgisi kalmamış oluyor. Daha sonradan html sayfalarımızı değiştirmek yada düzenlemek istediğimizde bu işi sadece gerekli html kısımlarını düzelterek yapabilmekteyiz. Ayrıca MVC mimarisinin bize sağladığı model yapısı ve Entity Framework birleştirilince çok kullanışlı bir yapı ortaya çıkmaktadır.

Buda bize çok fazla zaman ve maliyet kazancı sağlamaktadır. Tüm işlerin tek bir merkezden (controller) yönetilmesi projeyi yönetmek açısından oldukça kolaylık sağlamaktadır.

2.9 MİCROSOFT SQL SERVER

2.9.1 MSSQL Nedir?

MSSQL Microsoft tarafından geliştirilen bir ilişkisel veritabanı sistemidir. Windows platformlar üzerinde [.NET](https://wmaraci.com/nedir/net) veya [ASP](https://wmaraci.com/nedir/asp) [programlama dili](https://wmaraci.com/nedir/programlama-dili) kullanılarak oluşturulan [web](https://wmaraci.com/nedir/web) siteleri ve web yazılımlarda veritabanı görevi görmesi amacıyla kullanılır. Bu [yazılım](https://wmaraci.com/nedir/yazilim) dilleri içerisinde MS [SQL](https://wmaraci.com/nedir/sql) veri tabanına bağlanmak ve bu veri tabanı üzerinde işlemler gerçekleştirmek birçok programlama diline göre çok daha kolaydır. [Windows](https://wmaraci.com/nedir/windows) tabanlı sunucular ve programlama dillerinde MSSQL en çok kullanılan veri tabanı sistemidir. MSSQL ücretsiz bir veritabanı sistemi olmamasına rağmen kullanıcılara gelişmiş özellikler sunması nedeniyle çok tercih edilmektedir.

İlişkisel bir veritabanı sistemi olması sayesinde birçok yapıda kullanılmaktadır. Eğer ilişkiler çok önemliyse, kullanılan veritabanı sistemlerinin başında gelmektedir.

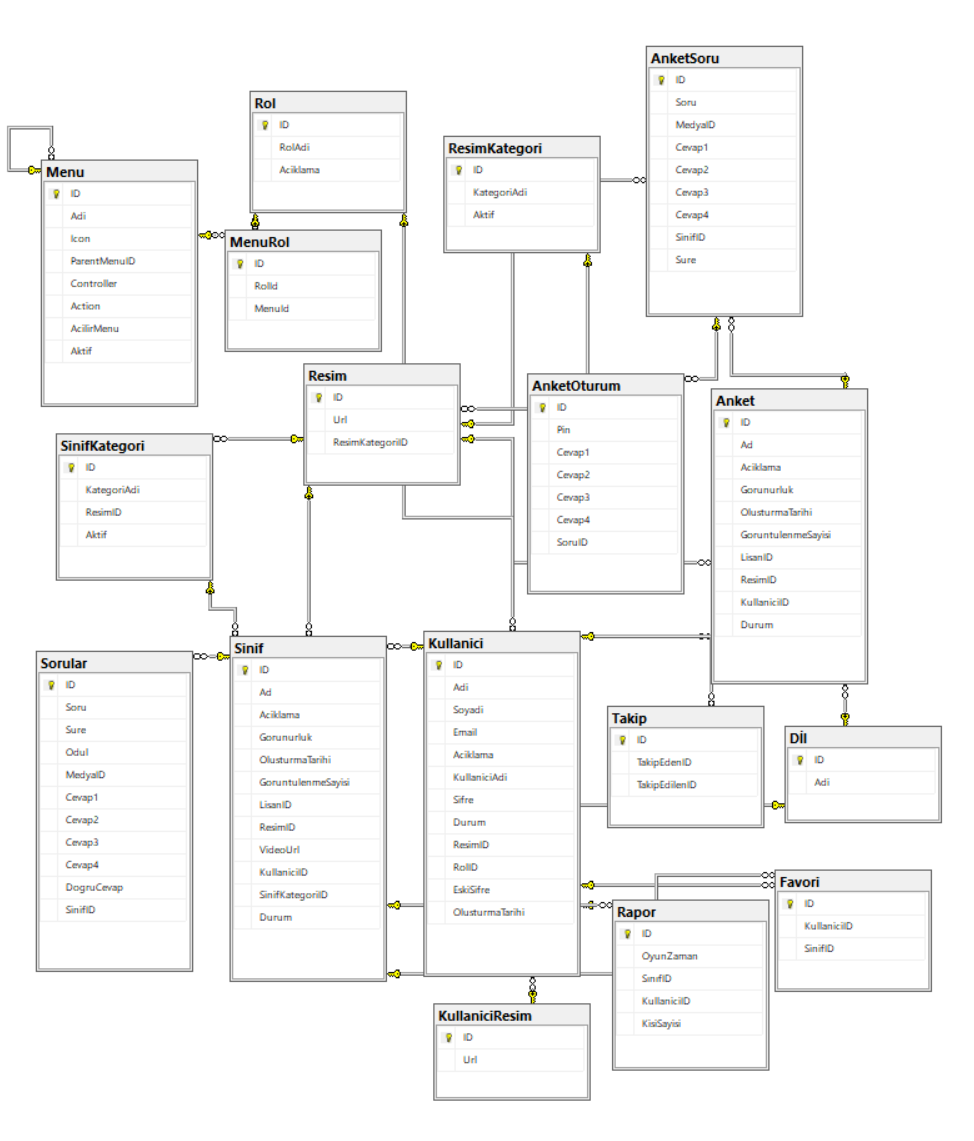
2.9.2 Neden MSSQL Kullandık?

Projemizde ilişkilerin çok önemli olduğu düşünüldüğünde kullanılması en mantıklı veritabanı sistemi MSSQL dir. Bize sunmuş olduğu ayrıntılı GUI ekranı ve birçok yönetim sistemleriyle çok kolaylık sağlamaktadır.

3.VERİTABANI İŞLEMLERİ

3.1.VERİTABANI DİYAGRAMI

Projemizin veri tabanını oluşturmak için Microsoft SQL Server geliştirme ortamını kullandık. SQL Server’ da daha önceden analiz sonucu elde ettiğimiz tabloları oluşturmaya başladık. Tablolar oluşturulduktan sonra tablolar arasındaki bağlantılarda gerçekleştirildi. Bu işlemlerin sonunda oluşturduğumuz diyagram Şekil 3.1.1’de ayrıntılı bir şekilde gösterilmektedir.



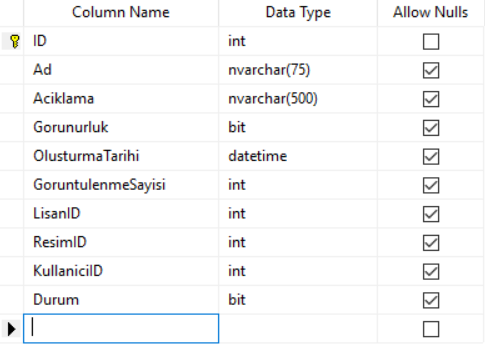
Şekil 3.1. 1. Veritabanı Diyagramı

3.2.TABLOLAR

Projemizin veritabanı tablolarının isimleri alfabetik olarak şu şekildedir; anket, anket oturum, anket soru, dil, favori, iletişim, kullanıcı, kullanıcı resim, menü, menü rol, rapor, resim, resim kategori, rol, sınıf, sınıf kategori, sorular, takip tır. Şimdi bu tabloları daha ayrıntılı bir şekilde inceleyelim.

3.2.1.Anket

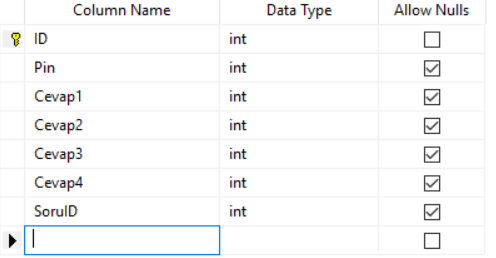
Projemiz ne kadar oyun üzerine tasarlanmış olsa bile anket sınıfları da oluşturulabilmektedir. Önyüzde oluşturulan anket sınıfının ayrıntıları anket tablosunda tutulmaktadır. Anket tablosu Şekil 3.2.1 de gösterilmiştir. Bu tablonun ilk satırında ID tanımlanmıştır. Bu ID değişkeni primary key olarak ve tek tek artan olarak ayarlanmıştır. İkinci satırında ise nvarchar() olarak tanımlanan anket sınıfının adı tutulur. Üçüncü satırda ise anket sınıfına kullanıcının eklediği açıklama bulunmaktadır. Dördüncü satırda ise kullanıcının oluşturduğu anket sınıfının görünürlüğü için bit tipinde değerler tutulmuştur. Bu sütundaki değer 1 ise sınıf bütün kullanıcılara gösterilirken 0 ise sadece o sınıfı oluşturan kullanıcı görüntüleyebilir. Sonraki üç satırda diğer tablolara bağlantısında kullanılacak ID değişkenleri tanımlanmıştır. Bu değişkenler sayesinde bağlı olduğu tablolardaki sınıf özellikleri getirilebilmektedir. En son olarak durum isminde bit tipindeki satır ise silme işlemleri için kullanılmaktadır. Bu değişkenle kullanıcı sildiği sınıfın değer 0 olarak değiştirilip kullanıcıya bir daha gösterilmez. Ancak Bu sınıf anketi admin tarafından görüntülenebilmektedir.



Şekil 3.2.1. Anket Tablosu

3.2.2.Anket Oturum

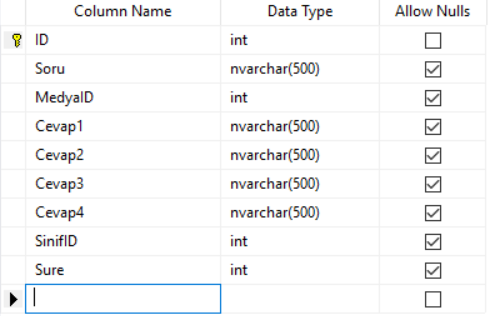
Veritabanı tablolarından bir diğeri de anket oturum tablosudur. Bu tablo Şekil 3.2.2 de gösterilmiştir. Bu tabloda anketlere verilen cevaplar tutulmaktadır. Tablonun ilk satırında ID tanımlanmıştır. Bu ID değişkeni primary key olarak ve tek tek artan olarak ayarlanmıştır. İkinci satırında int tipinde pin değişkeni vardır. Oluşturulan bu değişken sınıf başlatıldığı andaki sınıf pinidir. Bu pin ile veri tabanına kaydedilen cevaplar daha sonradan kolay bir şekilde incelenebilmektedir. Sonraki dört satırda anketin cevap sayılarını tutan int tipinde değişkenler oluşturulmuştur. Eğer Bir kullanıcı cevap 1 tıklar ise bu değişken bir artırılır. Son satırda ise anket soru tablosuna bağlantısında kullanılacak ID değişkeni tanımlanmıştır. Bu değişken sayesinde bağlı olduğu tablodaki sınıf özellikleri getirilebilmektedir.



Şekil 3.2.2.Anket Oturum Tablosu

3.2.3.Anket Soru

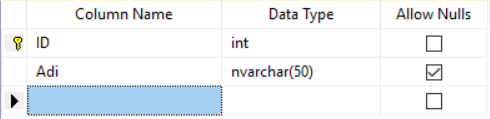
Veritabanı tablolarından üçüncüsü de anket soru tablosudur. Bu tablo Şekil 3.2.3 de gösterilmiştir. Bu tabloda anketlerin soruları, cevapları ve süresi tutulmaktadır. Tablonun ilk satırında ID tanımlanmıştır. Bu ID değişkeni primary key olarak ve tek tek artan olarak ayarlanmıştır. İkinci satırında nvarchar() tipinde soru değişkeni vardır. Oluşturulan bu değişken anket sınıfının sorusu buraya kaydedilmektedir. Nvarchar() olarak tanımlanan cevap 1, cevap 2,cevap 3ve cevap 4 değişkenleri sorunun şıklarının kaydedildiği satırlardır. Diğer bir satırda bulunan int tipindeki değişken anket sorusunun süresini tutmaktadır. En son olarak medya ID ve sınıf ID satırları ise diğer tablolara bağlantıda kullanılacak ID değişkenleridir. Bu değişkenler sayesinde bağlı olduğu tablolardaki sınıf özellikleri getirilebilmektedir.



Şekil 3.2.3.Anket Soru Tablosu

3.2.4.Dil

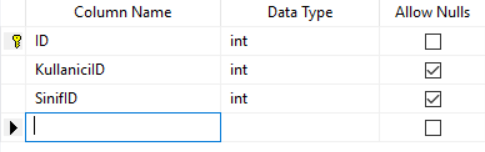
Projemizin ilerde oluşturabileceğimiz dil seçenekleri için oluşturduğumuz veritabanı tablosu Şekil 3.2.4 de gösterilmiştir. Bu tablonun ilk satırında ID tanımlanmıştır. Bu ID değişkeni primary key olarak ve tek tek artan olarak ayarlanmıştır. İkinci satırında ise nvarchar() olarak tanımlanan dilin adı tutulur.



Şekil 3.2.4. Dil Tablosu

3.2.5.Favori

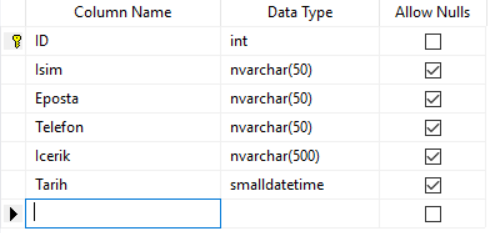
Kullanıcılar üye olduktan sonra daha önceden oluşturulan sınıfları favorilerine eklemesi oluşturduğumuz favori tablosu Şekil 3.2.5 gösterilmiştir. Bu tablonun ilk satırında ID tanımlanmıştır. Bu ID değişkeni primary key olarak ve tek tek artan olarak ayarlanmıştır. Diğer iki satırında ise diğer tablolara bağlanmak için kullanıcı ID ve sınıf ID tanımlanmıştır.



Şekil 3.2.5.Favori Tablosu

3.2.6.İletişim

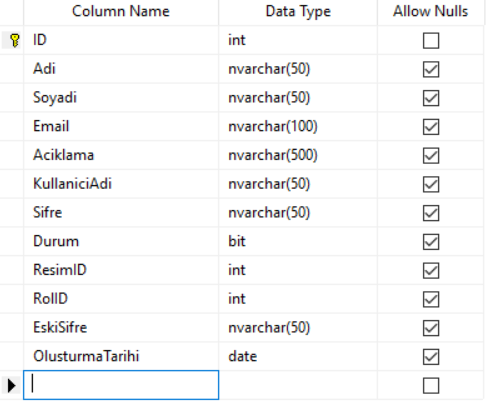
Önyüzde kullanıcılara sunulan iletişim ve herhangi bir şikâyet sekmesi tasarlanmıştır. Bu sekme aracılığı ile admin gönderilen mesajlar bu tabloya kaydedilir. Tablonun ilk satırında ID tanımlanmıştır. Bu ID değişkeni primary key olarak ve tek tek artan olarak ayarlanmıştır. İkinci satırında nvarchar() tipinde isim değişkeni vardır. Oluşturulan bu değişken mesaj atan kullanıcının kaydedildiği satırdır. Daha sonradan nvarchar() olarak tanımlanan e posta, telefon, ve içerik satırları diğer kişisel bilgilerin kaydedildiği satırlardır. En son olarak kullanıcının mesaj attığı tarihin kaydedilmesi için oluşturulan bir satır vardır.



Şekil 3.2 6.iletişim Tablosu

3.2.7.Kullanıcı

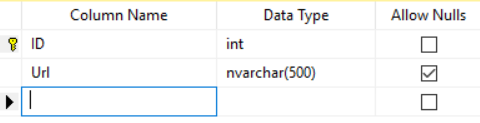
Projemizde kullanıcıların kolayca istediklerine ulaşabilmesi için ve kişisel bilgilerinin korunabilmesi için bilgilerini kaydediyoruz. Dolayısıyla önyüzde oluşturduğumuz kullanıcı kayıt sayfasından gelen verileri kullanıcı tablosuna kaydediyoruz. Kullanıcı tablosu Şekil 3.2.7 de gösterilmiştir. Bu tablonun ilk satırında ID tanımlanmıştır. Bu ID değişkeni primary key olarak ve tek tek artan olarak ayarlanmıştır. İkinci satırında ise nvarchar() olarak tanımlanan kullanıcının adı tutulur. Üçüncü satırda ise kullanıcının soyadı bulunmaktadır. Sırası ile dördüncü ve beşinci satırlarda kullanıcının mail adresi ve açıklaması bulunmaktadır. Kullanıcının kişisel bilgisi olan şifresi ise şifre sütununda tutulmaktadır. Bu sütundaki bilgiler hashlenerek adminnin bile görmesi engellenmiştir. Diğer bir satır olarak durum isminde bit tipindeki satır ise silme işlemleri için kullanılmaktadır. Bu değişkenle kullanıcı hesabını sildiği zaman durum değeri 0 olarak değiştirilip hesabına ulaşamaz. Ancak bu kullanıcı admin tarafından görüntülenebilmektedir. Sonraki iki satırda diğer tablolara bağlantısında kullanılacak ID değişkenleri tanımlanmıştır. Bu değişkenler sayesinde bağlı olduğu tablolardaki kullanıcı özellikleri getirilebilmektedir. En son olarak durum isminde bit tipindeki satır ise silme işlemleri için kullanılmaktadır. En son olarak kullanıcının sisteme kayıt olduğu tarih tutulmaktadır.



Şekil 3.2.7.Kullanıcı Tablosu

3.2.8.Kullanıcı Resim

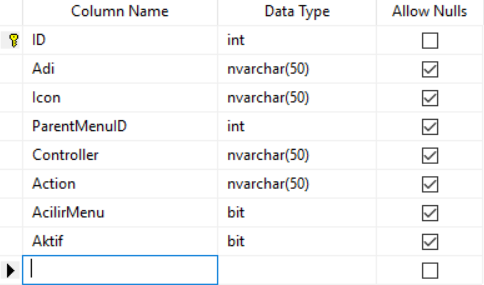
Kullanıcılar sisteme giriş yapabilmek için üye olurken kendi resimlerine de seçerler. Kullanıcı resmini seçerek resim sunucuya kaydedilir ve URL si kullanıcı resim tablosuna kaydedilir. Bu tablo Şekil 3.2.8 de gösterilmiştir. Bu tablonun ilk satırında ID tanımlanmıştır. Bu ID değişkeni primary key olarak ve tek tek artan olarak ayarlanmıştır. Diğer satırında ise kullanıcı resminin tutulduğu URL linki yer almaktadır.



Şekil 3.2.8.Kullanıcı Resim Tablosu

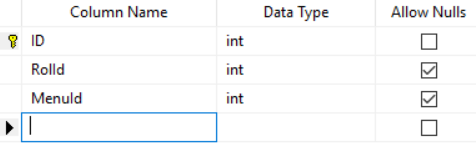
3.2.9.Menü

Projemizde admin arayüzünü yönetmesi kolay olabilmesi için menüler veri tabanından gelmektedir. Daha önceden belirlenen yetkiler sisteme tanıtılarak gerekli sınırlamalar yapılmıştır. Arayüz de bulunan menüler veritabanında menü tablosunda tutulmaktadır. Menü tablosu Şekil 3.2.9 da gösterilmektedir. Bu tablonun ilk satırında ID tanımlanmıştır. Bu ID değişkeni primary key olarak ve tek tek artan olarak ayarlanmıştır. İkinci satırında ise nvarchar() olarak tanımlanan menünün adı tutulmaktadır. Üçüncü satırda ise bu menünün ikon ismi tutulmaktadır. Dördüncü satırda ise menünün var ise üst menüsünün ID tutulmaktadır. Sırası ile dördüncü ce beşinci satırlarda ise nvarchar() olarak tanımlanan menünün controller adı ve action adı tutulmaktadır. Bu değişkenler önyüzde linkleri oluşturup sayfalar arası geçiş yapmaya olanak sağlamaktadır. Diğer bir satır olarak açılır menü isminde bit tipindeki satır ise menünün açılır menü olup olmadığını belirlediğimiz satırdır. En son oluşturduğumuz menünün önyüzde gösterilmesini ayarladığımız satıra geldik. Eğer bu satıra 1 verilirse menü önyüzde gösterilirken 0 verilirse menü önyüzde gösterilmez.



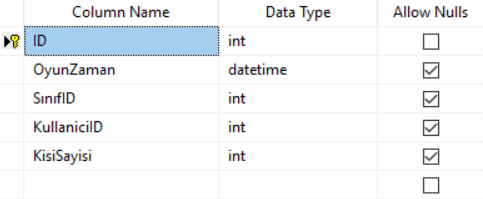
Şekil 3.2. 9.Menü Tablosu

3.2.10.Menü Rol



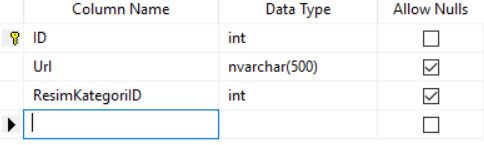
Şekil 3.2.10.Menü Rol Tablosu

3.2.11.Rapor



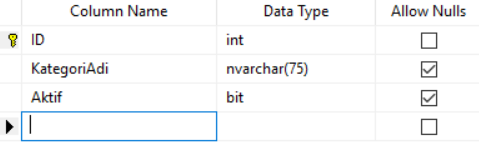
Şekil 3.2.11.Rapor Tablosu

3.2.12.Resim



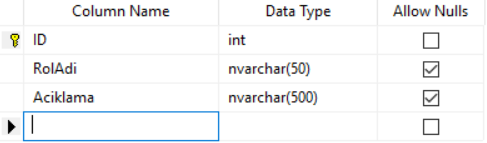
Şekil 3.2.12.Resim Tablosu

3.2.13.Resim Kategori



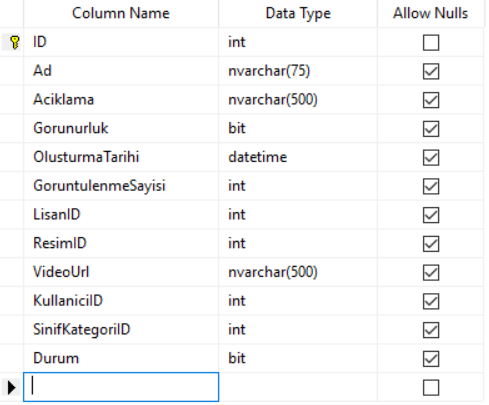
Şekil 3.2.13.Resim Kategori Tablosu

3.2.14.Rol



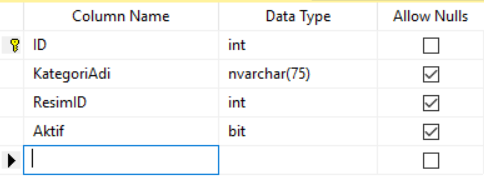
Şekil 3.2.14.Rol Tablosu

3.2.15.Sınıf



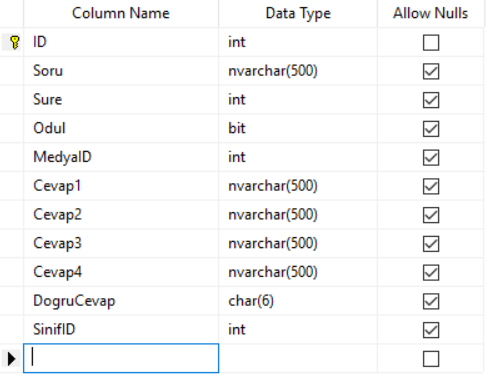
Şekil 3.2.15.Resim Tablosu

3.2.16.Sınıf Kategori



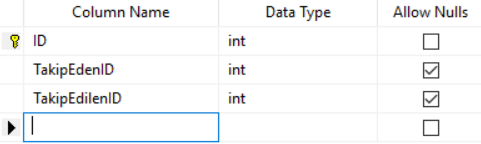
Şekil 3.2.16.Sınıf Kategori Tablosu

3.2.17.Sorular



Şekil 3.2.17.Sorular Tablosu

3.2.18.Takip



Şekil 3.2.18.Takip Tablosu

4.ARAYÜZ İŞLEMLERİ

4.1.KULLANICI ARAYÜZÜ

4.2.ADMİN ARAYÜZÜ

4.3.SINIF ARAYÜZÜ

5.TEST İŞLEMLERİ

6.SONUÇLAR VE ÖNERİLER

[1]BoraKaşmer/Node.js Nedir? http://www.borakasmer.com/nodejs-nedir/

[2]GeleceğiYazanlar/JavaScript Nedir? https://gelecegiyazanlar.turkcell.com.tr/konu/web-programlama/egitim/301-javascript/javascript-nedir

[3] Odabaşı, F., 1998. Bilgisayar Destekli Eğitim, Anadolu Üniversitesi Yayınları No: 1059.

[4] Gürbüz, T., 2001. Değişen Anlamıyla Bilgisayar Okuryazarlığı, Bilişim Teknolojileri Işığında Eğitim (BTIE) Konferans ve Sergisi Bildiriler Kitabı, ODTÜ,Ankara, 3-5 Mayıs 2001, ss.163-168.