

# T.C.

# İSTANBUL AREL ÜNİVERSİTESİ

# MESLEK YÜKSEK OKULU

**İNSAN KAYNAKLARI YÖNETİMİ OTOMASYONU**

**Abdül Samet Başak 201030062**

**Muhammet Emre Aygül 201030068**

**BİLGİSAYAR TEKNOLOJİLERİ BÖLÜMÜ**

**BİLGİSAYAR PROGRAMCILIĞI PROGRAMI**

**MESLEKİ PROJE RAPORU**

**DANIŞMAN**

**Öğr. Gör. Ebru İDMAN**

**İSTANBUL 2022**

# T.C.

# İSTANBUL AREL ÜNİVERSİTESİ

# MESLEK YÜKSEK OKULU

**İNSAN KAYNAKLARI YÖNETİMİ OTOMASYONU**

**Abdül Samet Başak 201030062**

**Muhammet Emre Aygül 201030068**

**BİLGİSAYAR TEKNOLOJİLERİ BÖLÜMÜ**

**BİLGİSAYAR PROGRAMCILIĞI PROGRAMI**

**MESLEKİ PROJE RAPORU**

**DANIŞMAN**

**Öğr. Gör. Ebru İDMAN**

**İSTANBUL 2022**

**ÖNSÖZ**

Bu çalışmanın gerçekleştirilmesinde, iki yıl boyunca değerli bilgilerini bizlerle paylaşan, kullandığı her kelimenin hayatıma kattığı önemini asla unutmayacağım saygıdeğer danışman hocam; Öğr. Gör. Ebru İDMAN, çalışmam boyunca benden bir an olsun yardımlarını ve desteklerini esirgemeyen ailelerimize sonsuz teşekkürlerimizi sunarız.

**Haziran 2022**

**Haziran 2022, İstanbul**

**İÇİNDEKİLER**

**ÖNSÖZ** **i**

**İÇİNDEKİLERii**

**KISALTMALARiii**

**ŞEKİLLERiv**

**BÖLÜM 1. GİRİŞ VE AMAÇ**

* 1. **İnsan kaynakları nedir?**
  2. **İnsan kaynaklarının önemi nedir?**

**BÖLÜM 2. GENEL BİLGİLER**

**2.1 Otomasyon Sitemi Nedir**

**2.1.1 Otomasyon Sistemi Hangi Alanlarda Kullanılır?**

**2.1.2 Otomasyon Yazılımı Türleri**

**2.1.3 Otomasyon Sisteminin Avantajları**

**2.1.4 Otomasyon Sisteminin Dezavantajları**

**2.1.5 Yeni Otomasyon Uygulamaları**

**2.2 C# Nedir?**

**2.2.1 C# Özellikleri**

**2.2.2 C# Avantajları Nelerdir?**

**2.2.3 C# Dezavantajları**

**2.2.4 C# Neden Tercih Edilir?**

**2.3 .NET Vizyonunun Oluşumu**

**2.4 C# için gereken ".NET Framework" nedir?2.3 Visual Studio Nedir?**

**2.5 C# Nerelerde Kullanılır?**

**2.5.1 Masaüstü**

**2.5.2 Web**

**2.5.3 Mobil**

**2.6 Microsoft Visual Studio nedir?**

**2.6.1 Microsoft Visual Studio'nun Avantajları**

**2.6.2 Microsoft Visual Studio Tarihçesi**

**2.6.3 Microsoft Visual Studio ile Neler Yapılır?**

**2.6.4 Microsoft Visual Studio programı kullanıcılarına ne sağlıyor?**

**2.6.5 Microsoft Visual Studio’nun temel özellikleri nelerdir?**

**2.6.5.1 Yüzlerce programlama dili için destek:**

**2.6.5.2 Tasarımcı (Designer):**

**2.6.5.3 Genişletilebilirlik:**

**2.6.6 Microsoft Visual Studio programının desteklediği ürünler nelerdir?**

**2.7 SQL Nedir?**

**2.7.1 SQL Tarihçesi:**

**2.7.2 SQL Yazım Kuralları**

**2.7.3 SQL Dilini Kullanan Veritabanları**

**2.7.4 SQL ile Neler Yapılır?**

**2.7.5 SQL Kullanmanın Avantajları**

**2.7.6 SQL ile modelleme sırasında yapabileceğiniz bazı işlemler:**

**2.7.7 SQL Nerelerde Kullanılır?**

**2.7.8 SQL Hakkındaki Eleştiriler**

**2.7.8.1 Tasarım**

**2.7.8.2 Ortogonaldik ve Tamlık**

**2.7.8.3 NULL**

**2.7.8.4 Mükerrer Kayıtlar**

**2.7.8.5 Empedans Uyumsuzluğu**

**2.8.9 SQL Birlikte Çalışabilirliği ve Standardizasyonu**

**2.8.9.1 Genel Bakış**

**2.8.9.2 Uyumsuzluk Nedenleri**

**2.8.9.3 Standardizasyon Geçmişi**

**2.8.9.4 Mevcut Standart**

**2.8.9.5 Dağıtılmış SQL İşleme**

**2.9 SQL Veri Türleri**

**2.9.1 Önceden tanımlanmış veri türleri**

**2.10 Microsoft SQL Management Nedir?**

**2.10.1 Microsoft SQL Management Gelişimi**

**2.10.2 Microsoft SQL Management Kullanım Alanları Nelerdir?**

**2.10.3 Mevcut Microsoft SQL Management Sürümlerinin Performansı**

**2.10.4 Microsoft SQL Management Sürümleri**

**2.10.5 Microsoft SQL Management Nasıl Kullanılır?**

**2.10.6 Microsoft SQL'in Avantajları ve Dezavantajları**

**2.10.6.1 Avantajları**

**2.10.6.1.1 Kurumsal Düzeyde Yönetim Yazılımı**

**2.10.6.1.2 Mükemmel Veri Kurtarma Desteği**

**2.10.6.2 Dezavantajları**

**2.10.6.2.1 Sınırlı Uyumluluk**

**2.10.6.2.2 Maliyet**

**2.11 Framework Nedir?**

**2.11.1 Framework Kullanım Alanları**

**2.11.2 Framework Kullanmanın Avantajları**

**2.11.3 Framework ile Library ( Kütüphane) Arasındaki Fark**

**2.11.4 Neden Framework kullanırız?**

**2.11.5 UI Framework**

**2.11.6 Guna UI Framework Nedir?**

**BÖLÜM 3. ÇALIŞMALARIMIZ**

**3.1 Platform ve Yazılım Dili Seçimi**

**3.1 Fikir Oluşma Aşaması**

**3.2 Projede Kullanılan Program Dili**

**3.3 Veri Tabanı**

**3.4 Projede yapım aşamaları**

**3.4.1 Giriş ve Kayıt Sayfaları**

**3.4.2 Personel Kayıt Sayfası**

**3.4.3 Personel Listeleme Sayfası**

**3.4.4 Departmanlar**

**3.4.5 Maaş Zamları**

**3.4.6 Mesai Ekle**

**3.4.7 Mesailer**

**3.4.8 İzin Listele**

**3.4.9 Ana sayfa**

**3.4.10 Veri tabanı**

**BÖLÜM 4. SONUÇLAR**

**4.1 Karşılaştığımız Sorunlar**

**KAYNAKLAR**

**ÖZGEÇMİŞ**

**KISALTMALAR**

**SQL:** SQL, verileri yönetmek ve tasarlamak için kullanılan bir dildir.

**MSSQL:** Microsoft sql veritabanı yönetim sistemidir.

**UI:** Kullanıcı arayüzü anlamına gelir.

**C#:** Microsoft tarafından .NET Teknolojisi için geliştirilen modern bir programlama dilidir.

**ŞEKİLLER**

**Şekil 1 / ilk giriş tasarımı**

**Şekil 2/ ilk kayıt sayfası**

**Şekil 3 Hata Mesajı**

**Şekil 4 Hata Mesajı 2**

**Şekil 5 Bildiri Mesajı**

**Şekil 6 Personel Ekleme**

**Şekil 7 Datagridview Kurulum 1**

**Şekil 8 Datagridview Kurulum 2**

**Şekil 9 Datagridview Kurulum 3**

**Şekil 10 Datagridview Kurulum 4**

**Şekil 11 Bildirim Mesajı**

**Şekil 12 Personel İzin Türleri**

**Şekil 13 Anasayfa**

**Şekil 16 deparmanlar**

**Şekil 17 izin**

**Şekil 18 izin türü**

**Şekil 19 Kullanıcı Adı**

**Şekil 20 Mesailer**

**Şekil 21 Personeller**

**BÖLÜM 1**

**GİRİŞ**

**1.1 İnsan kaynakları nedir?**

İnsan kaynakları, hem bir şirket veya kuruluş için çalışan kişileri hem de toplu olarak herhangi bir işletme veya kuruluştaki en değerli kaynaklardan birini temsil eden çalışanlarla ilgili tüm konuları yönetmekten sorumlu departmanı tanımlamak için kullanılır. İnsan kaynakları terimi ilk kez 1960'larda çalışma ilişkilerinin değerinin dikkat çekmeye başladığı ve motivasyon, örgütsel davranış ve seçim değerlendirmeleri gibi kavramların her türlü çalışma ortamında şekillenmeye başladığı yıllarda ortaya çıktı

İnsan kaynakları yönetimi, bir organizasyondaki çalışanların yönetimini ve gelişimini tanımlamak için kullanılan çağdaş, şemsiye bir terimdir. Personel veya yetenek yönetimi olarak da adlandırılır (bu terimler biraz eski olsa da), insan kaynakları yönetimi, bir kuruluşun insan sermayesini yönetmekle ilgili her şeyin denetlenmesini içerir.

**1.2 İnsan kaynaklarının önemi nedir?**

Günümüzde insan kaynakları en küçük şirketten en büyüğüne kadar önemini kanıtlamış bir departman olarak karşımıza çıkmaktadır.  Büyüyen işletmelerde organizasyonlarında büyüme sürecine girmesiyle İK departmanlarının varlığına daha çok ihtiyaç duyulmaya başlanmıştır.

Şirket için de en üst pozisyondan en altta pozisyona kadar herkes kendi özelinde değerlidir. Yönetim ve personel arasındaki iletişimin kuvvetli olması motivasyonu arttıran en önemli öğedir. Kurum içi iletişiminin aktif ve temiz bir şekilde sağlanması, sorunların çözülmesi, gerekli duyuruların yapılmasında insan kaynakları departmanı  en önemli rol aracıdır

**AMAÇ**

Proje konusu olarak insan kaynakları otomasyonu konusu seçtik bunun için internetten gerekli araştırmaları yaparak bilgiler edindik ve konu hakkında fikirler oluşturturduk. Bu proje herhangi bir şirketin insan kaynakları bölümü için tasarlandığından dolayı kolay anlaşılır ve sade bir tasarım tercih edilecektir. Bu otomasyonu kullanan kişi için daha rahat olabileceğini göz önünde bulundurularak projemizin planlamalarını buna göre oluşturduk. Aynı zamanda hedef kitlenin özelliklerini araştırıp buna göre bir proje geliştirmeyi hedefledik.

Projemizin asıl amacı şirketlerde çalışan insanların kişisel bilgilerini, unvan bilgilerini, maaş bilgilerini kayıt altında tutarak bilgi karışıklığını, bilgi kirliliğini, ve verilerinin çalınmasını önlemeyi hedefleyen bir nevi arşiv uygulamasıdır. Bu otomasyon sayesinde insan kaynakları yönetiminde görev alan kişin rahatlıkla hangi çalışanın ne kadar maaş aldığını, ne kadar mesaiye kaldığını, ne zaman hasta olduğunu, ne kadar süreyle izin aldığını istediği zaman görüntüleyebilmektedir

**BÖLÜM 2**

**GENEL BİLGİLER**

**2.1 Otomasyon Sitemi Nedir**

Alışveriş merkezlerinde, mağazalarda, marketlerde ve hızlı satış işlemlerinin yaptığı tüm alanlarda kullanılan otomasyon sistemi, hızlı işlem yaparak müşteri memnuniyetinin sağlanması gerektiği tüm alanlarda kullanılması gereken bir otomasyondur.

Otomasyon sistemi bilgisayar donanımı, mekanik ve yazılımın bir araya getirilerek oluşturulan sisteme verilen genel bir isimdir. Sadece alışveriş merkezlerinde değil, farklı birçok alanda da otomasyon sistemi kullanılır. Mesela kamera kayıtlarının incelenmesi ve işleyişi, araç takip sistemlerinin çalışması gibi sistemlere de otomasyon sistemi denir. Fakat genel olarak alışveriş merkezlerinde müşterilere daha hızlı çözüm sunmak ve mağazanın stok işlemlerini takip etmek amacı ile kullanılmaktadır.

Modern dünyanın insan hayatına sağladığı kolaylıklar bir hayli fazladır. Tüketiciler açısından sunulan kolaylıklar kadar, işletme sahiplerinin de iş yükü hafifletiliyor. Otomasyon sistemi, bu alanda işletmelerin iş yükünü en çok hafifleten teknolojik gelişimdir.

Günlük hayatımızda birçok çalışmanın yapılmasında genellikle insan emeğine ihtiyaç duyulmaktadır. İhtiyaç duyulan bu emek zihinsel olarak harcanan bir emek olabileceği gibi fiziksel olarak da harcanan bir emek olabilir. Bu durumda yapılması gereken işi, insanlar ile makineler arasında paylaştırılmasını sağlayan sistemlere otomasyon sistemleri, bu sistemlerin çalışmasını sağlayan yazılımlara da otomasyon yazılımları denmektedir.

Otomasyonlar da yapılan işinlerin paylaştırılmasına göre insan emeğinin daha fazla harcandığı yarı otomasyonlar ve makine gücünün daha fazla kullanıldığı tam otomasyonlar bulunmaktadır. İnsanların iş hayatlarına çeşitli faydalar sağlayan bu sistemlerin çalıştırılması için otomasyon yazılımı olarak adlandırılan bilişim teknolojisi kullanılır. Bu Teknoloji sayesinde günlük

hayatta birçok işin gerçekleştirilmesinde pratiklik sağlayan ve insan emeğine yardım eden yazılımlar geliştirilmekte ve bu yazılımlarla iş yükü azaltılmaktadır.

Makine ve insan gücünün gerçekleştirdiği işlemleri bir arada toplayarak otomatik bir şekilde işleyen ve sonuçların da insan ve makine gücünün yaptıklarından daha keskin olarak sunan otomasyon, günümüz işletmecilerinin ihtiyacının çok büyük bir bölümünü karşılayabildiği için daha fazla tercih edilmektedir.

**2.1.1 Otomasyon Sistemi Hangi Alanlarda Kullanılır?**

Birçok mağaza sahibi, müşterilerinin memnun kalmasını sağlamak için çeşitli bazı yöntemlere başvurur. Personel sayısını arttırmak bu yöntemlerin ilkidir. Fakat personel sayısını arttırmak, bir mağaza için büyük maliyetlerin ortaya çıkmasına neden olur ve bu da kazancın düşmesini etkiler. Sanayileşmenin hızla arttığı ve teknoloji ile paralel hareket edilen günümüzde, personel sayısını arttırmak yerine otomasyon sistemi kullanmak hem maliyet açısından hem de işlem açısından çok daha etkilidir. Peki, otomasyon hangi alanlarda kullanılır?

* Marketler,
* Mağazalar
* Ayakkabı mağazaları
* Alışveriş merkezleri
* Restoranlar,
* Kafeteryalar,
* Sosyal tesisler,
* Fastfoodlar,
* Şarküteriler,
* Kasaplar.

**2.1.2 Otomasyon Yazılımı Türleri**

Otomasyon yazılımları, birçok sektör sektör ve bu sektörlerde faaliyet gösteren firmaların farklı birimleri için ayrı ayrı tasarlanabilmektedir. İş hayatımızın farklı zamanlarında karşımıza çıkan otomasyon yazılımı çeşitlerinin bazılarını aşağıda maddeler ile belirtmek uygun olacaktır. Unutmadan otomasyon yazılım sistemleri türleri aşağıda kiler ile sınırlı değildir. Aşağıda sıralanan sistemler en çok kullanılan ve bilindik otomasyon sistemi yazılımlarıdır. Hayatımızın her yerinde karşımıza çıkan bu sistemler, ilerleyen teknoloji imkânları sayesinde her gün artarak karşımıza çıkacaktır

* Sanayi Makine otomasyonu yazılımı
* Muhasebe otomasyonu yazılım
* Öğrenci otomasyonu yazılımı
* Fabrika otomasyonu yazılımı
* Bina otomasyon yazılımı
* Ev otomasyonu yazılımı
* Akıllı bina otomasyonu yazılımı
* Akıllı ev otomasyonu yazılımı
* Barkod otomasyon yazılımı
* Depo otomasyon yazılımı
* Kütüphane otomasyonu yazılımı
* Aydınlatma otomasyon yazılımı
* Market otomasyon yazılımı
* Endüstriyel otomasyon yazılımı
* Sera otomasyon yazılımı

**2.1.3 Otomasyon Sisteminin Avantajları**

* Ürün veya hizmette kalite artışı
* Firmanın üretim maliyetinin ciddi oranda azalması
* Pazarda aynı işi yapan diğer firmalara karşı rekabet gücünün artması
* Tehlike sınıfına göre meydana gelen iş kazalarının oranlarının azalması
* Tüm birimlerin çalışmalarında çok büyük oranda zamandan tasarruf sağlama gibi

işletmelere direk hız ve karlılık olarak geri dönecek faydalar sağlamaktadır.

**2.1.4 Otomasyon Sisteminin Dezavantajları**

* Öncelikle otomasyon sisteminin ilk kurulum aşamasında, montaj maliyetlerinin yüksek olması (Çıkarılan maliyetler ancak uzun zaman içerisinde kendini amorti etmektedir.)
* Firmalarda otomasyon sisteminin kullanılmaya başlaması ile, insan gücüne daha az ihtiyaç kalacağından, firmalar daha az personel çalıştırarak personel giderlerini azaltabilirler fakat bu durum iş gücü istihdam oranını düşürmekte ve ekonomiye olumsuz etkide bulunmaktadır.
* Yeni otomasyon sisteminin kurulması ile firmanın yatırım maliyetleri yükselir

Yukarıda belirttiğimiz avantaj ve dezavantajları birbiri ile kıyaslayıp bir analiz yapmak gerekmektedir. Analiz sonucunda firmada otomasyon sistemi kullanımı daha fazla kar getiriyor ve maliyetlere uygun oluyor ise, ona göre karar verilerek otomasyon sistemi kullanılabilir.

**2.1.5 Yeni Otomasyon Uygulamaları**

Otomasyon uygulamaları günlük hayatımızın her yerinde karşımıza çıkmaktadır. Çoğumuz gördüğümüz bu sistemlerin çalışmasını sağlayan bir otomasyon sisteminin olduğunun bile farkına varamayız.

Hemen hemen günü birlik kullandığımız POS cihazları veya para çekmek için kullandığımız ATM makineleri bu otomasyon sistemlerinin en güzel ve en çok kullanılan örnekleri arasında yer almaktadır. Yani otomasyon günlük hayatımızda, nereye gidersek gidelim kullanmak zorunda olduğumuz araçları bir şekilde yönetmektedir. Aşağıda diğer otomasyon uygulamaları ve kullanıldığı alanlar yer almaktadır.

* Çeşitli güvenlik uygulamaları
* Bilimsel çalışma uygulamaları
* İş yerlerimiz (bilgisayar, fax, yazıcı…)
* İmalat uygulamaları
* Mobil telefon (cep telefonu) uygulamaları
* Taşıma işlemleri (lojistik)
* Evlerimiz (buzdolabı, kombi, çamaşır makinesi, bulaşık makinesi, akıllı cihazlar…)
* Sağlık sektörü uygulamaları
* Elektrik sektörü uygulamaları
* Hava tahmin uygulamaları
* İnternet popüler arama motorları, sosyal site uygulamaları

**2.2 C# Nedir?**

C# yani diğer adıyla C Sharp nedir? C#, Microsoft tarafından geliştirilen bir programlama dilidir. Programlama dilleri arasında en çok tercih edilen C ve C++ ile etkileşimli şekilde geliştirilen ve modern bir kodlama sahip olan C#, birçok platform destekleyen Java dili ile de oldukça fazla benzerlik gösteriyor. Bu benzerliklerden en önemlisi ise .NET Framework platformunda nesnesel bir dil olması.

C# her ne kadar önceleri C++ ve Visual Basic dillerindeki tutumsuzluk ve oluşan sorunlar yüzünden geliştirilmiş olsa da şu anki stabilimesin sayesinde, bu iki programlama dilinin önüne geçmiş durumda. Yazılımcıların en çok tercih ettiği dillerden birisi haline gelen C#, sunduğu gelişmiş hata giderme (debugger) özelliği sayesinde de yazılımcıların işlerini bir hayli kolaylaştırıyor.

Debugger özelliği için class yani sınıflandırma yöntemi kullanan C#, "syntax" hataları yani dizinlerdeki hataları yazılımcıya ayrı bir ekranda göstererek yazılımcının bu hataları kolayca tespit etmesini sağlıyor.

**2.2.1 C# Özellikleri**

* Güçlü tipleme kontrolü, dizin sınırları kontrolü ve tanımlanmış değişkenlerin kullanım tespiti gibi özellikler barındırır.
* Programcı portatifliğine sahiptir.
* Sunucu ve gömülü sistemler için tasarlanmıştır.
* C, C++ ve Java dilini kullananların rahatlıkla alışabilecekleri bir kullanım dizaynına sahiptir.
* Esnek yapısı itibariyle C sharp ile geliştirilen bir uygulama masaüstünde, web’de, mobil platformda kullanılabilir.
* C++ ile aynı komut setini kullandığından güçlü bir yapıya sahiptir.
* Göze sade gelen yazılım dilinin sunduğu kullanım kolaylığı ile uygulamalar daha kısa sürede yazılabilir.
* Nesne tabanlı bir programla dili olduğundan soyutlama, paketleme, çok biçimlilik gibi özellikler taşımaktadır.

**2.2.2 C# Avantajları Nelerdir?**

Yapılan araştırmalarda sonucunda en popüler programlama dilleri arasında ilk 10 içerisinde yer almaktadır. Dünya devi bir yazılım şirketi olması da popülerliğini destekliyor.

En önemli avantajı ise kullanım kolaylığıdır. Programlama diline hakim olan kişiler tarafından rahatlıkla kullanılabilirken temeli olmayan kişiler tarafından da hızlı bir şekilde öğrenilebilir.

Nesne tabanlı olan programlama dillerinin sahip olduğu temel dört özelliği C# bünyesinde barındırmaktadır.

Windows uygulamalarını hızlı ve kolay bir şekilde ulaştırabilirsiniz. Kod kolaylığı ile kısa sürede dilediğiniz uygulamayı geliştirebilirsiniz.

Hata giderme ile sunduğu eşsiz deneyim ile C#, kullanıcılarına büyük bir avantaj sağlamakta ve zamandan tasarruf sağlamaktadır. Önemli bir özellik olan hata giderme sayesinde yazılımcıların işleri de oldukça rahatlamaktadır.

C# ile sınırlar kalkıyor diyebiliriz. Bir yazılımcının hayalindeki pek çok tasarımı yapmasına imkan tanıyor. Web uygulamaları, mobil uygulamalar, konsol uygulamaları, web servis uygulamaları, DLL yazımı, Windows Form uygulamaları ve oyun tasarımı bu dil ile mümkün kılınmaktadır.

**2.2.3 C# Dezavantajları**

* Java için söylenilen yavaşlık C# dili içinde geçerli olarak kaynaklarda yer almakta.
* Program üzerinde gerçekleştirdiğimiz her değişiklik için derleme yapmamız gerekmektedir.
* Bağımsız bir dil olmamasından dolayı .Net bağlılığından dolayı esnek bir dil yapısında değildir

**2.2.4 C# Neden Tercih Edilir?**

C sharp, birçok alanda kullanımı yaygın olduğu için ve kolay öğrenilen bir programlama dili olduğu için tercih edilir. C ve C++ etkileşimi ile türemiştir. Java ve diğer programlama dilleriyle birçok benzerlik taşır, bireylerin en sık tercih etmesinin sebeplerinden biri de budur. Bu dilin başka bir özelliği ise nesnelerin önceden sınıflar halinde yazılı olarak bulunmasıdır. Programcıya sadece o nesneyi sürükleyerek ona uygun bir kod yazmak kalır.

**2.3 .NET Vizyonunun Oluşumu**

Microsoft, yazılım geliştiricilere yansıyan sonuçları izliyordu ve yazılım geliştiricilerin bir takım sorunlar ile karşılaştıklarını bu sorunları çözmenin bir yolunu aramaktaydı. Bu sorunları şu şekilde sıralamaktaydı:

Yazılan uygulamalara istedikleri an, kesintisiz ve güvenli bir şekilde ulaşamamak.

Uygulama geliştirirken yazılımcıların, altyapı kodlarını kodlamalarından dolayı geliştirme ve test süresinin uzaması.

Bir uygulamanın farklı platformlarda çalışması için ortak kodların bulunmaması.

Microsoft 1990 yılında bu tür sorunlara çözüm bulmak amacıyla yazılım geliştiricilerin sorunlarını çözecek bir platform geliştirmeye başladı. Bu platformun adı .NET platformuydu. Bu platform her türlü yazılım geliştirme ihtiyacına hazır bir altyapı sunmuştur. Böylece yazılımlar daha kolay geliştirilmeye başlandı.

**2.4 C# için gereken ".NET Framework" nedir?**

C Sharp kodları, C++ ve Visual Basic kodlarından farklı algoritmaya sahip olduğu için yazılımın kodlarını direkt olarak derleyemiyor. C#, kodları yazılım olarak iki aşamada derliyor. MIL derlemesi denilen bu aşamalardan ilkine Assembly deniliyor. İkinci aşamadaki uzantısı ise .EXE oluyor. C# ile geliştirilen bir yazılım bu halde iken sistemler tarafından çalıştırılamadığı için devreye .NET Framework giriyor.

Yine Microsoft tarafından C# için geliştirilen .NET Framework, Microsoft Intermediate Language (MIL) olarak derlenen kodları tekrar derleyerek, uygulamayı sistemlerin yani bilgisayarların çalıştıracağı hale getiriyor. Yani kısaca .NET Framework, bir tür tercümanlık görevi üstleniyor. C# programlama dilini bilgisayarımızın anlayacağı şekilde tercüme ediyor.

Peki neden Java, Visual Basic, C++ gibi programlama dilleri .NET Framework’e ihtiyaç duymuyor? Çünkü bazı diller kodları direkt olarak sistem dillerine çevirebiliyor, bazı dillerin de kendi bünyesinde bu işe yarayacak arayüzleri var. Mesela Java dili kodları önce byte denilen sayılara çeviriyor, bu sayıları da Java Virtual Machine dediğimiz bir sanal makinede derleyip bilgisayarların anlayacağı hale getiriyor.

**2.5 C# Nerelerde Kullanılır?**

**2.5.1 Masaüstü**

Masaüstü uygulamalar 2010 öncesi dönemde çok popülerdi. Çoğu firma kendi uygulamalarını yaptırır, karşılaştıkları problemlere çözümleri masaüstü yazılımlar ile üretirlerdi. Günümüzde bulut sistemlerin yaygınlaşması ve daha kullanışı hale gelmesiyle önemini yitirmiştir. Günümüzde kendi içlerindeki iş akışını masaüstü uygulamalarından takip eden işletme ve mağazalar olsa da sayıları her geçen gün azalmaktadır. Test amaçlı ufak botlar ve yapılan işi kolaylaştıracak ufak araçlar için kullanışlıdır.

**2.5.2 Web**

Günümüzde yaygın olarak MVC (Model-View-Controller) ile birlikte kullanılmaktadır MVC bir programlama dili değil, yazılım mimarisidir. Oluşturulan projenin tasarımıyla kodlarının birbirinden ayrılır. MVC ile profesyonel web uygulamaları, web siteleri yazabilirsiniz. C# ile kendi web servislerinizi de kolayca oluşturabilirsiniz.

**2.5.3 Mobil**

Yaygın olarak Xamarin ile beraber kullanılmaktadır. Microsoft’un cross platform mobil uygulama geliştirme kütüphanesi Xamarin’i almasıyla beraber C# uyumluluğu daha yüksek seviyeye çıkmıştır. Xamarin ile Android, iOS ve Windows Phone sistemlere uygulama geliştirebilirsiniz.

**2.5.4 IoT**

Windows 10 IoT sürümüyle beraber başta Raspberry Pi olmak üzere birçok geliştirici kartına kurulum yapıp Universal C# uygulamalarınızı geliştirebilirsiniz.

**2.6 Microsoft Visual Studio nedir?**

Microsoft Visual Studio, Microsoft tarafından geliştirilen bir tümleşik geliştirme ortamıdır (IDE). Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework ve Microsoft Silverlight tarafından desteklenen tüm platformlar için yönetilen kod ile birlikte yerel kod ve Windows Forms uygulamaları, web siteleri, web uygulamaları ve web servisleri ile birlikte konsol ve grafiksel kullanıcı arayüzü uygulamaları geliştirmek için kullanılır.

Visual Studio IntelliSense'in yanı sıra "code refactoring" destekleyen bir kod editörü içerir. Entegre hata ayıklayıcı, hem kaynak-seviyesinde hem de makine-seviyesinde çalışır. Diğer yerleşik araçlar, GUI uygulamaları, web tasarımcısı, sınıf tasarımcısı ve veritabanı şema tasarımcısı yaratabilmek için bir form tasarımcısı içerir. Hemen hemen her işlevsellik düzeyinde dahil olmak üzere, kaynak kontrol sistemleri için destek (Subversion ve Visual SourceSafe gibi) sunan eklentileri kabul eder.

**2.6.1 Microsoft Visual Studio'nun Avantajları**

Kod yazmak, hata tespit edebilmek veya yazılım geliştirebilmek için açık kaynak kodlu yayınlanarak kullanıcılara sunulan program, kullanıcılarına birçok avantaj sunabiliyor. Programın kullanıcılarına sunduğu faydalardan bazıları şu şekilde:

* Farklı programlama dillerine entegre bir şekilde desteklemesi
* Oldukça kolay ve sade bir kodlama arayüzü
* Microsoft tarafından seçilen eklentiler ve ek paketlerin yüklenebilirliği sayesinde sunulan genişlik

**2.6.2 Microsoft Visual Studio Tarihçesi**

Microsoft Visual Studio’nun ilk sürümü 1995 yılının Nisan ayında 4.0 sürüm numarasına sahip etiketiyle yayınlandı. Kısıtlı altyapı ile yalnızca görsel tasarıma olanak sağlayan bu sürüm, 1997 yılında bir hayli geliştirilerek 5.0 sürüm numarası ve Visual Studio 97 ismiyle güncellendi. Arada geçen 13 farklı sürümün ardından son olarak 2019 yılında 16.0 sürüm numarası ile Visual Studio 2019 yayınlandı. Yayında olan en güncel sürüm olan Visual Studio 2019, 3.5 ile 4.8 arasındaki tüm .NET Framework sürümlerini destekliyor. Kasım ayında yayınlanması beklenen Visual Studio 2022 ise 4.6.2 ile 6 arasındaki tüm .NET Framework sürümlerini destekleyecek.

**2.6.3 Microsoft Visual Studio ile Neler Yapılır?**

Microsoft’un bu programı geliştirilmesindeki amaç yazılım dünyasında yer alan kişilerin işini kolaylaştırmaktır. Yapay zeka destekli olarak geliştirilen bu yazım normalde oldukça vakit alacak bir işi daha kısa sürede tamamlamayı mümkün kılar. Konuyu daha iyi anlayabilmek açısından Visual Studio yapabileceğiniz şeylerin neler olduğunu aşağıdan öğrenebilirsiniz.

* Kod düzenleme ve hata ayıklama
* Diğer programlama dilleri ile entegre çalışma
* Arayüz tasarlama
* Kullanım alanını eklentiler ile genişletme

**2.6.4 Microsoft Visual Studio programı kullanıcılarına ne sağlıyor?**

Entegre geliştirme ortamı(IDE) olan Microsoft Visual Studio kullanıcılara 3 türlü hizmet tanımaktadır; SVsSolution, proje ve çözümleri numaralandırma özelliği sağlar; SVsUIShell, pencereleme ve UI işlevselliği sağlar.Son özellik olan SVsShell ise VSPackages kaydı ile ilgilenir. Programın mimarisi bu şekilde oluşturulmuştur.

Dil hizmetleri adındaki bir VSPackage kullanarak programlama dilleri için destek eklenmiştir. Dil hizmeti, çeşitli arayüzleri tanımlar ve bu sayede VSPackage uygulamaları, çeşitli fonksyonlara destek ekleyebilir.

Bahsedilen şekilde eklenen işlevler; sözdizimi renklendirme, deyim tamamlama, ayraç, eşleşen, parametre bilgileri, üye listeleri ve arka plan derleme ve hata işaretleri olarak bulunmaktadır.

Microsoft Visual Studio, yerleşik herhangi bir kaynak denetimi desteği içermez ancak IDE (Entegre Geliştirme Ortamı) ile entegre etmek için kaynak denetim sistemlerine iki opsiyonlu bir yol sunar. Kaynak Kontrolü VSPackage kendi özelleştirilmiş kullanıcı arayüzünü sağlayabilir.

Buna karşılık, standart bir Visual Studio kullanıcı arayüzü ile MSSCCI (Microsoft Source Code Control Interface) kullanarak bir kaynak denetimi eklentisi, çeşitli kaynak denetimi işlevselliğini uygulamak için kullanılan fonksiyon kümesi sağlar.

**2.6.5 Microsoft Visual Studio’nun temel özellikleri nelerdir?**

Etkili düzenleme ve hata ayıklama (Debugger):

Çeşitli türlere, işlev tanımlarına dayalı bir akıllı kod tamamlamaları olan IntelliSense ile üretkenliği arttırabilme yapabilmekte ve ”Tanıma Git”, “Tüm Başvuruları Bul” gibi özelliklerle de büyük kod tabanlarında rahatlıkla gezinebilme seçeneği düzenlemenin kolaylığı açısından iyi olmakla beraber kesme noktalarını, tam çağrı yığınını ve etkileşimli bir konsol kullandığı için düzenleyeciden kod üzerinde hata ayıklamak son derece basit halde gerçekleşir.

**2.6.5.1 Yüzlerce programlama dili için destek:**

Visual Studio Code, başlıca programlama dillerinin tümünü destekler. JavaScript, TypeScript, CSS ve HTML gibi sık kullanılan web tabanlı diller ve Visual Studio marketinden elde edilen zengin uzantılar diğer yüzlerce programlama dili için tamamlanma, lint uygulama, hata ayıklama ve yeniden düzenleme desteği sunar. Birçok programlama dilini desteklemesiyle de az önce okuduğunuz tüm işlevselliklerini direkt olarak korumaktadır.

**2.6.5.2 Tasarımcı (Designer):**

Visual Studio uygulamaları, kullanıcılarına arayüzlerini geliştirmek ve yardımcı olmak adına birtakım görsel tasarım aracı sağlamaktadır. Bu araçlar şunlardır; Windows Form Tasarımcısı (Windows Form Designer), WPF Tasarımcısı (WPF Designer), Web Tasarımcısı/Geliştiricisi (Web Designer/Development), Sınıf Tasarımcısı (Class Designer), Veri Tasarımcısı (Data Designer), Eşleştirme Tasarımcısı (Mapping Designer).

**2.6.5.3 Genişletilebilirlik:**

Visual Studio, kullanıcılarına programın işlevselliğini arttırmak adına kodlarını Visual Studio uzantılaryla yazmalarına olanak tanır. Bu uzantılar, Visual Studio'ya "takılır" ("plug into") ve onun işlevselliğini genişletmeyi hedefler. Bu uzantıları; makrolar, eklentiler ve paketler şeklinde görebiliriz.

Makrolar, geliştiricilerin kayıt, yeniden oynatma ve dağıtma için programatik olarak kaydetmesi için tekrarlanabilir görev ve işlemleri temsil eder fakat makrolar, yeni komutları kullanamaz ve araç pencereleri oluşturamazlar. Eklentiler, Visual Studio nesne modeline erişim sağlar ve IDE araçları ile etkileşim için kullanılır.

**2.6.6 Microsoft Visual Studio programının desteklediği ürünler nelerdir?**

Visual Studio; .NET tabanını destekler. Hem programlama dilleri ve kodların derlenip ve yorumlandıktan sonra çalıştırılabilir dosya oluşturulması için hem de ASP.NET gibi üst düzey teknolojileri kullanarak projeler üretilmesine olanak sağlar. Bu bağlamda Microsoft Visual Studio kullanıcılarına aşağıdaki dillerde proje üretme imkanı sağlar;

* Microsoft Visual C++
* Microsoft Visual C#
* Microsoft Visual Basic
* Microsoft Visual Web Geliştirici (Microsoft Visual Web Designer)
* Team Foundation Server
* C,
* C++
* C++.NET,
* Visual Basic.NET
* JavaScript
* TypeScript
* CSS
* HTML

**2.7 SQL Nedir?**

Kelime açılımı, “Structered Query Language” SQL, Türkçe ’ye “yapısal sorgulama dili” olarak geçmiştir. Temel olarak SQL’e bir sorgulama dili demek mümkün. Veri tabanı işlemlerini hızlı bir şekilde yapmak için kullanılan bir alt dil diye ifade edebiliriz. SQL ifadesi bir nevi veri tabanı yönetimi kavramını ifade etmektir. SQL ’in burada ki temel amacı verileri doğru yönetmek ve tasarlamaktır. Dünya üzerinde pek çok kişi tarafından programlama dili olarak kabul edilir. Ama bu tam olarak doğru değil. SQL’ in programların içerisinde kullanılıyor olması, onun bir programlama dili olduğu anlamına gelmiyor. ANSI standartlarına dayanarak söylenebilir ki SQL tam olarak bir sorgu dilidir.

**2.7.1 SQL Tarihçesi:**

SQL ilk olarak 1970’lerin başında Edgar F. Codd’dan ilişkisel modeli öğrendikten sonra Donald D. Chamberlin ve Raymond F. Boyce tarafından IBM’de geliştirildi. Başlangıçta SEQUEL (Yapılandırılmış İngilizce Sorgu Dili) olarak adlandırılan bu sürüm, IBM San Jose Araştırma Laboratuvarı’nda bir grubun 1970’lerde geliştirdiği IBM’in orijinal yarı-ilişkisel veri tabanı yönetim sistemi System R’de depolanan verileri işlemek ve almak için tasarlandı.

Chamberlin ve Boyce’un ilişkisel bir veri tabanı dili konusundaki ilk girişimi SQUARE (İlişkisel Bir Ortamda Sorguları Belirtme) idi, ancak alt simge/üst simge gösterimi nedeniyle kullanımı zordu. 1973’te San Jose Araştırma Laboratuvarı’na taşındıktan sonra, SQUARE’in devamı üzerinde çalışmaya başladılar. “SEQUEL” İngiltere merkezli Hawker Siddeley Dynamics Engineering Limited şirketinin ticari markası olduğu için SEQUEL adı daha sonra SQL (sesli harfler düşürülerek) olarak değiştirildi. SQL etiketi daha sonra Yapılandırılmış Sorgu Dili’nin kısaltması oldu.

Sistemin kullanışlılığını ve pratikliğini belirlemek için SQL’i müşteri test sitelerinde test ettikten sonra sırasıyla IBM, 1979, 1981 ve 1983’de piyasada bulunan System/38, SQL/DS ve IBM Db2 dahil olmak üzere System R prototiplerine dayalı ticari ürünler geliştirmeye başladı.

1970’lerin sonlarında, Relational Software, Inc. (şimdi Oracle Corporation), Codd, Chamberlin ve Boyce tarafından tanımlanan kavramların potansiyelini gördü ve ABD Donanması, Merkezi İstihbarat’a satma arzusuyla kendi SQL tabanlı RDBMS’lerini geliştirdi. Ajans ve diğer ABD devlet kurumları. Haziran 1979’da, İlişkisel Yazılım, SQL’in ticari olarak mevcut ilk uygulamalarından birini, VAX bilgisayarlar için Oracle V2’yi (Sürüm2) tanıttı.

1986’da, ANSI ve ISO standart grupları, standart “Veritabanı Dili SQL” dil tanımını resmi olarak kabul etti. Standardın yeni sürümleri 1989, 1992, 1996, 1999, 2003, 2006, 2008, 2011 ve en son 2019’da yayınlandı.

**2.7.2 SQL Yazım Kuralları**

* SQL dili, aşağıdakiler de dahil olmak üzere çeşitli dil öğelerine bölünmüştür:
* İfadelerin ve sorguların kurucu bileşenleri olan tümceler. (Bazı durumlarda bunlar isteğe bağlıdır.)
* Skaler değerler veya sütun ve veri satırlarından oluşan tablolar üretebilen ifadeler
* SQL üç değerli mantık (3VL) (doğru/yanlış/bilinmeyen) veya Boolean doğruluk değerleriyle değerlendirilebilecek koşulları belirten ve deyimlerin ve sorguların etkilerini sınırlamak veya program akışını değiştirmek için kullanılan yüklemler.
* Belirli kriterlere göre verileri alan sorgular. Bu, SQL’in önemli bir unsurudur.
* Şema ve veriler üzerinde kalıcı bir etkiye sahip olabilen veya işlemleri, program akışını, bağlantıları, oturumları veya tanılamayı kontrol edebilen ifadeler.
* SQL ifadeleri ayrıca noktalı virgül (“;”) ifade sonlandırıcısını da içerir. Her platformda gerekli olmasa da SQL gramerinin standart bir parçası olarak tanımlanır.
* Önemsiz boşluklar, SQL deyimlerinde ve sorgularında genellikle yok sayılır, bu da okunabilirlik için SQL kodunu biçimlendirmeyi kolaylaştırır.

**2.7.3 SQL Dilini Kullanan Veritabanları**

* Microsoft Access
* MySQL
* Firebird
* PostgreSQL
* Progress
* Microsoft SQL Server
* IBM Informix
* Oracle
* Sybase
* IBM DB2

**2.7.4 SQL ile Neler Yapılır?**

* Verilerin tamamını güncelleme
* Yeni tablolar oluşturma
* Veri işleme ve depolama
* Veri indeksleme ve sorgulama
* Veri tabanı yönetimi
* Veri akışı optimizasyonu
* Veri raporlama
* İlke tabanlı veri erişimi

**2.7.5 SQL Kullanmanın Avantajları**

* SQL veri tabanı yazılımı sayesinde veriler çok daha kolay bir şekilde kategorize edilebilir. Bu sistem sayesinde gereksiz ve meşakkatli olan veri ortamı dağınıklıktan kurtulur. Kısaca veri dağınıklığı ortadan kalkar. Ayrıca kategorize edilmiş tüm veriler, kullanıcı tekrar ulaşmak istediği zaman SQL sorguları sayesinde kolayca listele SQL veri tabanı yazılımı sayesinde veriler çok daha kolay bir şekilde kategorize edilebilir. Bu sistem sayesinde gereksiz ve meşakkatli olan veri ortamı dağınıklıktan kurtulur. Kısaca veri dağınıklığı ortadan kalkar. Ayrıca kategorize edilmiş tüm veriler, kullanıcı tekrar ulaşmak istediği zaman SQL sorguları sayesinde kolayca listelenebilir. Bu uygulama gelişen dünyada ki en önemli şey olan zamanı tasarruf etmenizi sağlamakta.
* SQL kullanımı web uygulamalarında tekrara düşmekten sizi kurtarır. Birçok kullanıcının eskiden beri yaşadığı bu durum SQL kullanımı ile ortadan kalkar. Bu sebep ile veri tutma ihtiyacı olan her platform muhakkak SQL kullanır.
* Bilgisayar deposunda genellikle dağınık bulunan tüm verilerin klasör haline dönüştürülmesine yardımcı olur. Bu yöntem daha önce gereksiz alan işgali eden tüm dağınık verilerin önüne geçer. Yazılım amacı olarak dosya bütünlüğü sağlayan SQL hem kamu hem de özel kurumlar tarafından sıklıkla tercih edilmektedir.
* Bilişim sektörü hızla büyümekte. Veri tabanı sistemleri de sürekli geliştirilmektedir. SQL, bu sistemin ve bilişim dünyasının temel taşlarından biridir. Bu sebeple neredeyse bütün veri tabanı işlemleri SQL ile yürütülür. SQL sisteminin temel amacı, verilerin ve veri kümelerinin modellenmesini sağlamaktır. SQL tüm bu olanakları size hızlıca sunmakta.

**2.7.6 SQL ile modelleme sırasında yapabileceğiniz bazı işlemler:**

* Tablolar oluşturmak
* Güvenlik önlemleri almak
* Silinen veri kayıtlarını yeniden oluşturabilmek
* Kayıtlı verileri sorgulamak ve listelemek
* Verilerin tamamını güncellemek
* Veri kaydetmek
* Yeni veriler çekerek kaydetmek

**2.7.7 SQL Nerelerde Kullanılır?**

SQL Veri tabanında bulunan verileri kontrol edebildiğimiz bir sistem olduğu için veri tabanına ihtiyaç duyulan her iş ve her sektörde kullanılabilir. Örneğin internet ve ağ üzerine yapılan uygulama geliştirmelerinde, web sitesi geliştirmek isteyenler için kesinlikle kaçınılmaz bir seçenektir.

Örneğin bir web programlaması üzerine çalışıyorsanız mutlaka SQL ile ilişkili olmak zorundasınızdır. Çünkü bir web sitesinde verileri göstermek için temel anlamda 4 şeye ihtiyacınız olur. Bu açıdan bakıldığında aslında SQL nedir nerelerde kullanılır sorgusunun cevabı verilmiş oluyor.

* RDBMS, bir veri tabanı programıdır.
* PHP ve ASP, sunucu tabanlı scriptlerin oluşturduğu dillerdir.
* HTML ve CSS, Veri tabanında bulunan dillerin sitede gösterilmesini sağlayan diller.
* SQL, veri tabanında yapacağınız işlemler için kullanılır.

**2.7.8 SQL Hakkındaki Eleştiriler**

**2.7.8.1 Tasarım**

SQL, teorik temelinden, ilişkisel modelden ve tanımlama grubu hesabından çeşitli şekillerde sapar. Bu modelde, bir tablo bir demetler kümesidir, SQL’de ise tablolar ve sorgu sonuçları satır listeleridir; aynı satır birden çok kez ortaya çıkabilir ve satırların sırası sorgularda kullanılabilir (örneğin, LIMIT yan tümcesinde). Eleştirmenler, SQL’in kesinlikle orijinal temele dönen bir dille değiştirilmesi gerektiğini savunuyorlar: örneğin, Üçüncü Manifesto’ ya bakınız.

**2.7.8.2 Ortogonaldik ve Tamlık**

Erken spesifikasyonlar, birincil anahtarlar gibi ana özellikleri desteklemiyordu. Sonuç kümeleri adlandırılamadı ve alt sorgular tanımlanmadı. Bunlar 1992’de eklendi.

Toplam türlerinin eksikliği, SQL’in kullanıcı tanımlı türlerinin tam olarak kullanılmasına bir engel olarak tanımlanmıştır. Örneğin JSON desteğinin 2016’da yeni bir standart tarafından eklenmesi gerekiyor.

**2.7.8.3 NULL**

Null kavramı bazı tartışmaların konusudur. Null işaretçisi, bir değerin olmadığını gösterir ve bir tamsayı sütunu için 0 değerinden veya bir metin sütunu için boş bir dizeden farklıdır. Null kavramı, SQL’de 3 değerli mantığın genel 3 değerli mantığının somut bir uygulaması olan 3 değerli mantığı zorlar.

**2.7.8.4 Mükerrer Kayıtlar**

Bir başka popüler eleştiri, yinelenen satırlara izin vermesi, veri türleri verileri ayrıştırma açısından ve modülerliğin olmaması nedeniyle doğru bir şekilde temsil etmeyi zorlaştırabilecek Python gibi dillerle entegrasyonu yapmasıdır.

Bu genellikle, tablodaki bir satırı benzersiz şekilde tanımlayan bir veya daha fazla sütunla bir birincil anahtar veya benzersiz bir kısıtlama bildirerek önlenir.

**2.7.8.5 Empedans Uyumsuzluğu**

Nesne-ilişkisel empedans uyumsuzluğuna benzer bir şekilde, bildirimsel SQL dili ile SQL’in tipik olarak gömülü olduğu prosedürel diller arasında bir uyumsuzluk meydana gelir.

**2.8.9 SQL Birlikte Çalışabilirliği ve Standardizasyonu**

**2.8.9.1 Genel Bakış**

SQL uygulamaları satıcılar arasında uyumsuzdur ve standartları tam olarak takip etmesi gerekmez. Özellikle tarih ve saat sözdizimi, dize birleştirme, NULL’lar ve karşılaştırmalı büyük/küçük harf duyarlılığı satıcıdan satıcıya değişir. PostgreSQL her durumda standarda uymasa da, belirli istisnalar, standartlara uygunluk için çaba gösteren PostgreSQL ve Mimer SQL’dir. Örneğin, PostgreSQL’de alıntı yapılmayan isimlerin küçük harfe katlanması, alıntı yapılmayan isimlerin büyük harfe katlanması gerektiğini söyleyen SQL standardı ile uyumlu değildir. Bu nedenle, Foo, standarda göre foo değil, FOO’ya eşdeğer olmalıdır.

SQL’in popüler uygulamaları, genellikle, DATE veya TIME veri türleri gibi Standart SQL’in temel özelliklerini destekler. Bu tür en belirgin örnekler ve tesadüfen en popüler ticari ve tescilli SQL VTYS’leri Oracle (DATE’si DATETIME olarak davranır ve TIME türünden yoksundur) ve MS SQL Server’dır (2008 sürümünden önce). Sonuç olarak, SQL kodu nadiren değişiklik yapılmadan veritabanı sistemleri arasında taşınabilir.

**2.8.9.2 Uyumsuzluk Nedenleri**

Veritabanı sistemleri arasında bu taşınabilirlik eksikliğinin birkaç nedeni şunları içerir:

SQL standardının karmaşıklığı ve boyutu, çoğu uygulayıcının tüm standardı desteklemediği anlamına gelir.

Standart, birkaç önemli alanda (örneğin, dizinler, dosya depolama …) veritabanı davranışını belirtmez ve uygulamaların nasıl davranacağına karar vermesine izin verir.

SQL standardı, uyumlu bir veritabanı sisteminin uygulaması gereken sözdizimini kesin olarak belirtir. Bununla birlikte, standardın dil yapılarının semantiğine ilişkin belirtimi daha az iyi tanımlanmıştır ve bu da belirsizliğe yol açar.

Birçok veritabanı satıcısının mevcut geniş müşteri tabanları vardır; SQL standardının daha yeni sürümünün satıcının veritabanının önceki davranışıyla çakıştığı durumlarda, satıcı geriye dönük uyumluluğu bozmak istemeyebilir.

Veri tabanı tedarikçilerini değiştirmeyi kolaylaştırmak için satıcılar için çok az ticari teşvik mevcuttur (bkz. satıcı kilitleme).

Veritabanı yazılımını değerlendiren kullanıcılar, performans gibi diğer faktörleri standartlara uygunluktan daha yüksek önceliklerine yerleştirme eğilimindedir.

**2.8.9.3 Standardizasyon Geçmişi**

SQL, 1986’da ANSI tarafından SQL-86 ve ISO 1987’de standart olarak kabul edilmiştir. ISO/IEC JTC 1, Bilgi teknolojisi, Alt Komite SC 32, Veri yönetimi ve değişimi tarafından sürdürülür.

1996 yılına kadar, Ulusal Standartlar ve Teknoloji Enstitüsü (NIST) veri yönetimi standartları programı, SQL DBMS’nin SQL standardına uygunluğunu onayladı. Satıcılar artık ürünlerinin uyumluluğunu kendileri onaylıyor.

Orijinal standart, “SQL” için resmi telaffuzun bir başlangıç olduğunu ilan etti: /ˌɛsˌkjuːˈɛl/ (“ess cue el”). Ne olursa olsun, İngilizce konuşan birçok veritabanı uzmanı (Donald Chamberlin’in kendisi dahil), dilin yayın öncesi geliştirme adı olan “SEQUEL”i yansıtan /ˈsiːkwəl/ (“sequel”) kısaltmasına benzer bir telaffuz kullanır.

SQL standardı bir dizi revizyondan geçmiştir:

**2.8.9.4 Mevcut Standart**

Standart genel olarak şu modelle belirtilir: ISO/IEC 9075-n:yyyy Bölüm n: başlık veya kısayol olarak ISO/IEC 9075.

ISO/IEC 9075, ISO/IEC 13249: SQL tabanlı arayüzleri ve video, ses ve uzamsal veriler gibi uygulamaları geniş çapta yaymak için paketleri tanımlayan SQL Multimedya ve Uygulama Paketleri (SQL/MM) ile tamamlanmaktadır. İlgili taraflar, ISO, IEC veya ANSI’den SQL standartları belgeleri satın alabilir. SQL:2008’in bir taslağı zip arşivi olarak ücretsiz olarak mevcuttur.

**2.8.9.5 Dağıtılmış SQL İşleme**

Dağıtılmış İlişkisel Veritabanı Mimarisi (DRDA), 1988’den 1994’e kadar IBM içindeki bir çalışma grubu tarafından tasarlandı. DRDA, ağa bağlı ilişkisel veritabanlarının SQL isteklerini yerine getirmek için işbirliği yapmasını sağlar.

Etkileşimli bir kullanıcı veya program, yerel bir RDB’ye SQL ifadeleri yayınlayabilir ve uzak RDB’lerden yanıt olarak veri tabloları ve durum göstergeleri alabilir. SQL ifadeleri ayrıca paketler olarak uzak RDB’lerde derlenebilir ve saklanabilir ve ardından paket adına göre çağrılabilir. Bu, karmaşık, yüksek frekanslı sorgular yayınlayan uygulama programlarının verimli çalışması için önemlidir. Erişilecek tabloların uzak sistemlerde bulunması özellikle önemlidir.

DRDA’nın mesajları, protokolleri ve yapısal bileşenleri, Dağıtılmış Veri Yönetimi Mimarisi tarafından tanımlanır. Dağıtılmış SQL işleme ala DRDA, çağdaş dağıtılmış SQL veritabanlarından farklıdır.

**2.9 SQL Veri Türleri**

SQL standardı, üç tür veri türü tanımlar ;

* Önceden tanımlanmış veri türleri
* İnşa edilmiş tipler
* Kullanıcı tanımlı türler.

Oluşturulan türler ARRAY, MULTISET, REF(erence) veya ROW’dan biridir. Kullanıcı tanımlı türler, kendi yapıcıları, gözlemcileri, mutatörleri, yöntemleri, kalıtım, aşırı yükleme, üzerine yazma, arabirimler vb. ile nesne yönelimli dildeki sınıflarla karşılaştırılabilir. Önceden tanımlanmış veri türleri, uygulama tarafından özünde desteklenir.

**2.9.1 Önceden tanımlanmış veri türleri**

Karakter Tipleri

* Character -Karakter (CHAR)
* Character varying – Karakter değişen (VARCHAR)
* Character large object – Karakter büyük nesne(CLOB)

Ulusal karakter türleri

* National character – Ulusal karakter (NCHAR)
* National character varying -Ulusal karakter değişen (NCHAR VARYING)
* National character large object – Ulusal karakter büyük nesne (NCLOB)

İkili türler

* Binary – İkili (BINARY)
* Binary varying – İkili değişken (VARBINARY)
* Binary large object – İkili büyük nesne (BLOB)

Sayısal Türler

* Exact numeric types – Tam sayısal türler (NUMERIC, DECIMAL, SMALLINT, INTEGER, BIGINT)
* Approximate numeric types – Yaklaşık sayısal türler (FLOAT, REAL, DOUBLE PRECISION)
* Decimal floating-point type – Ondalık kayan nokta türü (DECFLOAT)
* Datetime types – Tarih saat türleri(DATE, TIME, TIMESTAMP)
* Interval type – Aralık türü(INTERVAL)
* Boolean
* XML
* JSO

**2.10 Microsoft SQL Management Nedir?**

SQL Management, kullanıcıların veri tabanı işlemlerini daha hızlı yapmasını sağlayan bir sistem dilidir. Bu dil, mevcut SQL sürümlerinin standardını temel alır ve bunlara bir dizi ek özellik ekler.

Microsoft SQL Server Management uygulaması verilerin bütünlük ve güven içerisinde depolanmasına ve aynı anda bir veya birden fazla kullanıcı tarafından erişilmesine olanak sunan kurumsal çaplı ve ilişkisel bir veri tabanı yönetimi sistemidir.

Birbirleriyle ilişkili olan verilerin sistematik bir biçimde kaydedilmesine ve bu veriler ile beslenen uygulamalar tarafından gerekli ihtiyaç anında kullanılmasına imkan sağlayarak işletmelerin BT altyapılarında gerçekleşen veri aktarımlarının sorunsuz bir şekilde yönetilmesini sağlar. Bilişim, finans, sağlık gibi birçok çeşitli sektörde önde gelen işletmelerin tercih ettiği SQL Server Management, birbirinden farklı her tip kullanıcının ihtiyaçlarına uyum sağlayabilen ve ölçeklenebilen bir yapıya sahiptir.

Microsoft SQL Management ile yapabileceklerinizin arasında veri işleme ve depolama, veri indeksleme ve sorgulama, veri tabanı yönetimi, veri akışı optimizasyonu, veri raporlama ve ilke tabanlı veri erişimi işlemleri yer almaktadır

MSSQLServer olarak kısaltılan ilişkisel veri tabanı yönetim sistemi Microsoft SQL Server, orijinal olarak Sybase tarafından geliştirilen koddan kaynaklanmaktadır. Ek olarak Microsoft SQL Server, şirketlerde kullanılması amacıyla tasarlanmıştır ve farklı sürümlerde mevcuttur. Finans, ticaret, bilişim gibi sektörlerde profesyonel olarak kullanılmaktadır, tüm özelliklere sahiptir.

Bu sistem dili sayesinde, birbiriyle ilişkili olan veriler bir arada değerlendirilebilmektedir. Veri aktarımı, hızlı ve sorunsuz bir şekilde gerçekleşeceği için çoklu kullanıcılı uygulamalara işlevsellik kazandırmaktadır.

SQL Managemenet sürümleri, işlev yelpazesi, donanım gereksinimleri ve fiyatı bakımından farklılık gösterir. Yazılım, SQL standardını temel alan ancak işlev yelpazesine çok sayıda başka özellik ekleyen SQL varyantı olan Transact-SQL’e (T-SQL) dayanmaktadır. Bu ek özellikler;

* İşlem denetimi,
* Satır işleme,
* Hata işleme,
* İstisna işleme işlevlerini içerir.

Microsoft SQL Server’ın uygulama alanları, özellikle veri depoları ve iş zekâsı uygulamalarıdır.

**Bu alanlar ise şunları içermektedir:**

* Raporlama hizmetleri,
* Analiz hizmetleri,
* Veri tabanı yönetimi,
* Entegrasyon hizmetleri,
* SQL Server Veri Araçları (SSDT) ​​
* Sunucu hizmetleri vb.

Microsoft SQL Server, açık veri tabanı bağlantısı için OLE DB ve ODBC’yi de destekler.

**2.10.1 Microsoft SQL Management Gelişimi**

Microsoft SQL Management, orijinal olarak Sybase tarafından geliştirilen koda dayanmaktadır. OS/2 işletim sistemi için bir veri tabanı yönetim sisteminin ilk versiyonu, 1980’lerin sonunda geliştirilmiştir. Bu ilk versiyon, Sybase, Ashton-Tate ve Microsoft şirketleri arasındaki iş birliğinden ortaya çıktı.

OS/2 için son sürüm olan SQL Server sürüm 4.2, 1992 gibi erken bir tarihte yayınlandı. 1993’ten 4.21 sürümü ile SQL Server, Windows NT işletim sistemi ile birlikte kullanılmıştır.

Sybase ve Microsoft arasındaki iş birliği 1990’larda sona erdiğinden, her iki şirket de kendi SQL Server yazılımı sürümlerini yayınladı. Microsoft, Microsoft SQL adıyla, Sybase ise, Adaptive Server Enterprise (ASE) adı altında yeni sürümleri piyasaya sürdü.

Hem Microsoft SQL Management 2019’da hem de önceki sürümlerde sunucu/CAL ve çekirdek lisanslama arasında bir ayrım yapılmalıdır. Enterprise Edition yalnızca işlemci çekirdeklerine dayalı lisanslamaya izin verirken, Microsoft SQL Server Standard için her iki lisanslama yöntemi de mevcuttur.

Veri tabanı uygulamaları geliştiricilerine yönelik olan ücretsiz Express Edition sürümü ise herhangi bir özel lisans gerektirmez. Ancak bu sürüm, çok sınırlı bir işlev yelpazesine sahiptir.

**2.10.2 Microsoft SQL Management Kullanım Alanları Nelerdir?**

Ağlarda düzenli olarak büyük miktarda verinin işlenmesi, güçlü bir veri tabanı gerektirir. SQL Management, her türlü veriyi depolamak, sorgulamak ve düzenlemek için bir hizmet sağlayan bir yazılımdır.

Microsoft SQL Management, veri işlemek için kullanılan en yaygın ve başarılı yazılım çözümlerinden biridir. Özetle, bu sistem aracılığıyla veriler özel tablolarda depolanır, atanmış sipariş kriterleri ve sonraki işlemler için birbirine bağlanır.

Tüm SQL sunucuları, isimsiz ve standartlaştırılmış Yapılandırılmış Sorgu Dili (SQL) ile adresleyebilir. Microsoft SQL Server ile SQL tabanlı ilişkisel veri tabanı yönetim sistemleri bu şekilde bilişim sistemlerine uygulanabilmektedir.

Sunucu yazılımı, diğer şeylerin yanı sıra finansal muhasebe, süreç analizi, üretim planlaması ve ürün yönetimi alanlarında kullanılmaktadır. Bu nedenle, öncelikle şirketlere ve serbest meslek sahiplerine yöneliktir.

Birçok kurumsal özel ilişki yönetimi yani CRM yazılımları, müşteri verilerini ve ilgili sipariş süreçlerini yönetmek için veri tabanı yönetim sistemini kullanır. Bu nedenle veri tabanı dili SQL, bilgileri yönetmek için her zaman kilit noktadır.

Microsoft SQL Server, diğer şeylerin yanı sıra C#, C++, Delphi, Java, Visual Basic ve Python programlama dillerini destekler. Böylece bilgi yönetimi için yazılım çözümleri geliştirirken maksimum esneklik sağlanır. SQL Management için başka bir uygulama alanı, bir çevrim içi mağaza aracılığıyla malların satışıdır.

**2.10.3 Mevcut Microsoft SQL Management Sürümlerinin Performansı**

Mevcut Microsoft SQL Management sürümleri çok güçlüdür ve bulut depolamada veya büyük veri ortamlarında kullanılmaktadır. Akıllı, iş açısından kritik uygulamalar ve özellikle ölçeklenir bir hibrit veri tabanı platformu, veri tabanı yönetim sistemi ile uygulanmaktadır.

**2.10.4 Microsoft SQL Management Sürümleri**

Gerekli hizmet kapsamına ve kullanım amacına bağlı olarak Microsoft SQL Server’ın farklı sürümleri mevcuttur. Bunlar, Mainstream Editions ve Specialized Edition olmak üzere iki kategoriye ayrılmaktadır. Ana sürümler, örneğin;

* Enterprise Edition,
* Server Standard Edition,
* Web Edition,
* Workgroup Edition,
* Express Edition’ı içerir.
* Özelleştirilmiş sürümler arasında ise:
* Azure SQL Database,
* Compact (SQL CE),
* Developer Edition,
* 2005 Embedded Edition,
* Assessment Edition,
* Fast Track bulunmaktadır

**2.10.5** **Microsoft SQL Management Nasıl Kullanılır?**

SQL server, sunucuya erişim bilgisine sahip olan herkesin ulaşabileceği bir veri tabanı sağlar. Kullanıcılar, istediği yerden bu veri tabanına ulaşır ve işlemlerini hızlı bir şekilde gerçekleştirir.

Bir MSSQL veri tabanı ile herhangi bir işlem yapmak için kullanıcıların bazı bilgilere sahip olması gerekir.

Bunlar örneğin: sunucunun adresi, bir kullanıcı adı ve paroladan oluşan bilgilerdir. Kullanıcı adı ve parola bilgisine sahip olmayanlar, veri tabanına giriş yapamazlar.

SQL Server Management üzerindeki veri tabanına ulaşmak için ise ek özellikler gerekir. Veri tabanını düzenlemek gibi işlemleri yapmak için SQL Server Management Studio (SSMS) programına sahip olmak gerekir.

SQL Managament, çevrim içi veya çevrim dışı olarak kullanabilme özelliğine sahiptir. Programı bilgisayarlara yüklemeden de çalıştırmak mümkündür.

Veri tabanı, bilgisayarda da saklanacaksa, SQL Management cihazlara yüklenerek kullanılmalıdır.

**2.10.6 Microsoft SQL'in Avantajları ve Dezavantajları**

**2.10.6.1 Avantajları**

**2.10.6.1.1 Kurumsal Düzeyde Yönetim Yazılımı**

Microsoft SQL Server, profesyonel, kurumsal düzeyde veritabanı yönetim yazılımı içerir. MySQL gibi birkaç rakip, son yıllarda benzer yazılımlar geliştirdi, ancak Microsoft SQL Server'ın kullanımı daha kolay ve daha fazla özelliğe sahip. Örneğin, tetikleyiciler için tam destek, Microsoft'un ürününde desteklenir. MySQL, yakın zamanda tetikleyicileri tanıttı, ancak bunlar tam olarak desteklenmiyor. Microsoft'un sunduğu yazılım, rakip ürünlerde olmayan .NET çerçevesi ile yakın entegrasyon da sunuyor.

**2.10.6.1.2 Mükemmel Veri Kurtarma Desteği**

Güç kaybı veya uygun olmayan kapanmalar meydana geldiğinde, bozuk veriler her zaman bir endişe kaynağıdır. Microsoft SQL Server, veri geri yükleme ve kurtarmayı destekleyen bir dizi özelliğe sahiptir. Tek tek tablolar yedeklenemese veya geri yüklenemese de, eksiksiz veri tabanı geri yükleme seçenekleri mevcuttur. Microsoft'un ürünü, günlük dosyalarının, önbelleğe almanın ve yedeklemelerin kullanılması yoluyla, olağanüstü durum kurtarma seçeneklerinin bol olduğundan emin olmanızı sağlar.

**2.10.6.2 Dezavantajları**

**2.10.6.2.1 Sınırlı Uyumluluk**

Microsoft SQL Server, yalnızca Windows tabanlı sunucularda çalışacak şekilde tasarlanmıştır. Geliştiriciler, lisans maliyetleri ve güvenlik endişeleri de dahil olmak üzere çeşitli nedenlerle, web sitelerini Unix tabanlı makinelerde barındırmayı tercih edebilir. Bu durumda SQL Server'ı kullanamazlar. Rakip ürünler sıklıkla diğer platformlarda çalışabilir. Microsoft SQL Server'dan farklı olarak MySQL, Windows, Linux, Mac OSX ve diğer Unix türevleri dahil olmak üzere her büyük platformda desteklenir. Windows dışı platformlarda çalıştırılamamanın yanı sıra, diğer platformlarda çalışan uygulamalarla etkileşimle ilgili uyumluluk sorunları da olabilir.

**2.10.6.2.2 Maliyet**

Alternatif bir ilişkisel veritabanı yönetim sistemi yerine Microsoft SQL Server kullanmanın en büyük dezavantajlarından biri, lisanslama seçeneklerinin oldukça pahalı olmasıdır. Yazılımı geliştirme veya eğitim amaçlı kullanmak ücretsiz olmasına rağmen, her türlü iş kullanımı için lisans ücreti alınır. Örneğin SQL Server 2008 için SQL Server Standard Edition, işlemci başına 7,171 ABD dolarıdır. Microsoft web sitesine göre, SQL Server Datacenter sürümü işlemci başına 54,990 ABD dolarıdır. Kar amaçlı web siteleri çalıştıran küçük işletmeler ve bireyler için bu, karşılanamaz. MySQL gibi rakip yazılımların kullanımı genellikle ücretsizdir. Ancak, olmadığı durumlarda, en pahalı MySQL Enterprise paketi her yıl sunucu başına 4,999 dolardır. Bu, Standard Edition Microsoft SQL paketinden bile önemli ölçüde daha ucuzdur.

**2.11 Framework Nedir?**

Frameworkler, başkaları tarafından oluşturulmuş bir nevi şablon diyebileceğimiz kod parçalarının bir arada bulunduğu yapılardır. Türkçeye yazılım iskeleti, yazılım çatısı, yazılım çerçevesi gibi isimler ile çevrilmiş olsa da yaygın olarak İngilizce adıyla bilinir. Dilimizde karşılık geldiği çatı kelimesinden bakacak olursak frameworkün işlevini anlamamız kolaylaşacaktır. Birden fazla kişi tarafından geliştirilen bu platformlar sizi sık kullanılan kodları yeniden yazmaktan kurtarır. Belirli kullanım direktifleri olan frameworkler ile yazmak istediğiniz kodu önceden geliştirilmiş uygun bir çatı üzerine inşa edersiniz. Böylelikle yeni baştan bir kod satırı oluşturmanıza gerek kalmadan hızlı bir biçimde istediğiniz çıktıyı alabilirsiniz. Bu yazılım geliştirme çatısı sayesinde projelerin genelinde bulunan temel kod yapılarını zamandan tasarruf ederek ve güvenlik kaygısı olmadan kullanabilirsiniz. İçerisinde kütüphaneler de bulunduran bu çatı şablonları genelde açık kaynak olarak oluşturulur. Bu özelliği sayesinde başkalarının da katkılarıyla değiştirilebilir, güncellenebilir ve geliştirilebilirler.

**2.11.1 Framework Kullanım Alanları**

Çalışmayı düşündüğünüz her proje için uygun olan bir yazılım geliştirme çatısı bulabilmeniz mümkündür. Neredeyse her yazılım dili için o dille uyumlu çalışan birden çok framework mevcuttur. Bu noktada hangi framework ile çalışacağınız projenizin ihtiyaçlarına ve sizin tercihlerinize göre değişiklik gösterebilir. Dikkate almanız gereken en önemli husus kullanmayı seçtiğiniz dil ile framework birbirine uyumlu olmalıdır. Eğer Java diline hakimseniz Java için geliştirilmiş frameworklerden birisiyle çalışmanız bu aşamada daha doğru olacaktır. Üzerinde çalıştığınız proje hangi alanda olursa olsun size bu süreçte yardımcı olabilecek bir yazılım iskeleti bulabileceğinizden şüpheniz olmasın.

Bir mobil uygulama (app) tasarlıyorsunuz ve bunu hayata geçirme sürenizi olabildiğince kısaltmak istiyorsunuz, framework tam bu noktada en doğru seçeneklerden biri olarak karşınıza çıkacaktır. Bu geliştirme çatısı yalnızca bir app geliştirme sürecinde destek alabileceğiniz bir fonksiyona sahip değildir. Bunun ötesinde ürününüzün arka planda gerçekleşen işlemlerine de katkı sunmaktadır. Arka plan işlemlerinin daha hızlı ve işlevsel ilerlemesine olanak sağlar. Doğru frameworkü seçtiğiniz takdirde arayüz çalışmalarınız için de faydalanabileceğiniz fonksiyonlara sahip olabilirsiniz.

**2.11.2 Framework Kullanmanın Avantajları**

Üzerinde çalıştığınız projede her zaman bir yazılım geliştirme çatısı kullanmak zorunda değilsiniz. Her proje framework kullanmanızı gerektirecek büyüklük ve karmaşıklıkta olmayabilir. Bunun yanı sıra tüm kodlama sürecini baştan sona kendiniz yönetmek istiyor olabilirsiniz. Yine de bir şeye en hızlı ve güvenli yolla ulaşmanın bu kadar değerli olduğu bir zamanda frameworklerin bize sunduğu faydalara göz atmak isteyebilirsiniz. Bir yazılım geliştirme çatısı kullanmanın size sunacağı faydaları birkaç madde ile sıraladık:

* Sık kullanılan ve sürekli olarak yazmanız gereken kod parçalarını yeniden tekrar tekrar yazmanıza gerek kalmaz. Böylelikle zamandan ve harcayacağınız efordan tasarruf edersiniz
* Güvenlik açıklarına karşı korunaklıdır
* Daha temiz ve düzenli bir kod yazmanıza olanak sağlar
* Sürekli olarak geliştirildiğinden dolayı güncelliğini korur
* Ekip olarak eş zamanla çalışmaya imkan verir

**2.11.3 Framework ile Library ( Kütüphane) Arasındaki Fark**

Hem frameworkler hem de kütüphaneler başkaları tarafından geliştirilmiş tekrar kullanılmaya uygun kod parçaları içerir. Framework içerisinde de çeşitli kütüphaneler bulundurabilen yapılardır. Bu iki yapı bazen birbirlerinin yerine kullanılsalar da aslında birbirinden ayrılmalarına olanak sağlayan bazı farkları mevcuttur. Teknik olarak en temel ikisi arasındaki en temel fark kontrolün tersine çevrilmesi ( inversion of control) durumudur. Kütüphane kullanırken gerekli kodu çağıran kişi yazılım geliştiricidir. İhtiyaç duyduğu zamanda ve yerde kütüphaneden kodu çağırabilir. Framework kullanımı bunu yapmanıza izin vermez. Çalışmanızı bir yazılım geliştirme çatısı üzerinde sürdürüyorsanız kullanılan framework gerek gördüğü kodu gerek gördüğü zamanda çağırır.

**2.11.4 Neden Framework kullanırız?**

Yazılım geliştirmek karmaşık bir süreçtir. Kodlama, tasarım ve test dahil olmak üzere çok sayıda görev gerektirmektedir. Yalnızca kodlama kısmı için, programcıların syntax, declarations, garbage collection, statements, exceptions ve daha fazlasıyla ilgilenmesi gerekiyor.

Framework’ler, tüm yazılım geliştirme sürecinin veya çoğunun kontrolünü tek bir platformdan yönetmeye izin vererek geliştiricilerin hayatını kolaylaştırmaktadır.

**2.11.5 UI Framework**

UI Framework(user interface), Türkçe karşılığı ile kullanıcı arayüzü, bir kişinin etkileşime girebileceği cihazlarda (telefonlar, bilgisayarlar, tabletler..) tasarlanan, kullanıcı tarafından yazılan kodu bir arada tutan bir çerçevedir. Bir uygulamanın geliştirilme sürecini daha pratik hale getirir. Telefonlarımızda kullandığımız tuşlar UI Framework için bir örnektir.

Kullanıcı deneyimini iyi yönde etkilemek amacı ile birçok şirket UI Framework kullanmaya başlamıştır.

UI Framework kullanmanın en büyük avantajı, size oldukça zaman kazandırmasıdır. Geliştiriciler, çok fazla tasarım bilgisine sahip olmadan gayet güzel ve hızlı bir şekilde bir cihaz uygulaması oluşturabilir.

Mesela HTML’de çoğu geliştirici tasarım kısmını uzun uzun yazıp uğraşmak istemez. Bunun için de UI Framework siteleri kullanırlar. En fazla kullanılan birkaç UI Framework Sitesine örnek verirsek bunlar; “Guna UI Framework , bunifu framework,Bootstrap, Bulma, SemanticUI” olabilir. Bu frameworklerde yapacağınız tasarım için oldukça fazla taslak bulabilirsiniz.

**2.11.6 Guna UI Framework Nedir?**

Guna UI Framework, harika masaüstü uygulama arayüzleri oluşturmanıza yardımcı olacak DLL tabanlı araçlardır.. Uygulamalarınızda harika bir kullanıcı deneyimini garanti eder ve geliştirme süresini azaltır. DLL'ler .NET ortamınıza kurulur ve WinForms üzerine kuruludur.. Geliştiricilerin sürükleyip bırakmasını sağlar, dolayısıyla daha hızlı kodlama. Desteklenen Platformlar Microsoft Visual Studio, .AĞ, Windows Formları, C# & VB.NET Özel kontroller, veri bağlama özelliklerine sahip VB.NET veya C# WinForms bileşenleri ve çizelgeleri.

Temel ürün özellikleri Yazılım geliştiricinin yaratıcılığını ve üretkenliğini güçlendirir 50+ Kontrol & Şu anda WinForms'u destekleyen bileşenler & uygulamanıza modern bir his verir Etkileşimli modern geçişler Etkileşimli geçişlerle masaüstü uygulamanızı canlandırın

**BÖLÜM 3**

**ÇALIŞMALAR**

**3.1 Fikir Oluşma Aşaması**

Ilk olarak araştırmalarımızın esnasında topladığımız verileri, bulguları, belgeleri, örnekleri derledik ve bir araya getirdik. Araştırmalarımız istenildiği gibi gitmedi ve kendi fikirlerimizi oluşturmaya başladık. Bu fikirler şöyleydi insan kaynakları otomasyonu nedir, nasıl olmalı, ne gibi özelliklere sahip olmalı ve tasarım kısmı nasıl olmalıydı? Önceliklerimizi belirleyebilmek için önceden yapılmış ve internet ortamında bulunan otomasyon örneklerini inceleyip, bu konu hakkında fikirler oluşturup notlar çıkararak başladık. İlk fikrimiz olan yabancı kaynakların içerisinde çok daha fazla örnek bulabileceğimizi düşünürken, tam aksi bir şekilde yabancı kaynaklardan yeterli örnek çeşidi bulamadık. Türkçe kaynaklara dönüp baktığımızda ise daha fazla örnek bulabildiğimizi fark edip o şekilde ilerledik ve sonucunda yaptığımız araştırmalarla birlikte kafamızda bir yol haritası çizebildik. Bu kaynaklardan edindiğimiz bilgilerin bize en çok faydası şöyle dokundu ki; tasarım örnekleri, hataları, eksik olan kısımları veya kendi projemize entegre edebileceğimiz gerekli özellikleri ve tasarımları istediğimiz seviyede olmasa da en azından taslak oluşturabilmek adına yeteri kadarını bulabildik. Bu özelliklerin ve tasarımların göze güzel gelmesi ve kullanışlı olması bizim uyarlayacağımız şeylerin daha göz alıcı ve yaralı olmasına sebep oldu. Biz en başta bu projemizi geliştirirken ilk olarak otomasyonumuzun temelde doğru çalışan bileşenlerinin eksiksiz çalışıp bitmesine odaklandık. Peki bunların sonrasında biz neler eklemeye karar verdik?

Projemizi geliştirirken ilk aşamadan itibaren ortak noktalarda buluşarak devam ettiğimiz için kararlarımızı belirlemek bizim için çok daha kolay oldu. İlk başta karar verdiğimiz şeylerden biri bu programı kullanacak olan kişinin programın içeresine rahatlıkla personel ekleyebilmesi istememiz oldu. Bundan sonra ise otomasyon sistemimize personellerin kişisel bilgileri, departmanları ve maaş bilgilerinin kaydedilmesi bizim ana öncelliklerimiz oldu. Bu ana önceliklerime ek olarak yeni özellikler koymaya başladık genelde bir insan kaynakları otomasyonun ne yapması gerekiyorsa ince ayrıntısına kadar planlarımız tamdı ve programızı geliştirmeye devam ettikçe özellikler eklemeye karar verdik.

**3.2Projede Kullanılan Program Dili**

Öncelikle projemize başlamadan önce hangi yazılım dili üzerinde yazacağımızı düşündük. Önce javafx ile yapmak istedik ama yeterli bir kaynak bulamadığımız için görsel programlama dersinde almış olduğumuz Microsoft c# yazmaya karar verdik. Ardından programımızı hangi ide ile yazacağımızı düşündük önce hafif ve kullanışlı olan Microsoft visual code kullanmak istedik fakat visual studio code da çok fazla hata verdiğinden Windows forms şeklinde proje oluşturmayı başaramadık ve bizde görsel programlama dersinden görmüş olduğumuz Microsoft visual studio 2022 kullanmaya karar verdik. Hem performans açısından Hem de ileride ekleyeceğimiz özelliklerin çalışması açısından en doğru kararı verdik.

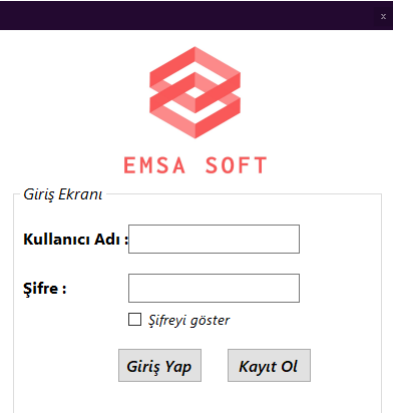
**3.3 Veri Tabanı**

Projemizdeki en önemli kısım olan veri tabanı kısmı bizim ana noktamız olmuştur. Çünkü bizim projemizde bütün işlemler veri tabanı kısmından çalışacaktı, o yüzden kullanacağımız veri tabanında sağlam, sorun çıkartmayan, sorun oluşması halinde gerekli kaynaklara dan bakıp sorunlarımızı çözmemiz gerekiyordu ve bizde veri tabanında Microsoft sql management kullanmaya karar verdik. Zaten daha önceden de veri tabanı dersi almış olduğumuz için programın nasıl çalışacağını ve gerekli işlemlerin nereden yapılacağını biliyorduk.

**3.4 Projede yapım aşamaları**

**3.4.1 Giriş ve Kayıt Sayfaları**

İlk olarak projemizi yazarken programı kullanacak kişinin giriş yapmasını istedik. Bunu yapmamızın nedenini açıklamak isterim. Öncelikle bu programı bir şirket yada bir işletmede kurulumu yapsaydık ve programımızda giriş sayfası olmasaydı şirket yada işletmelerin verileri kötü niyetli kullanıcılar tarafından çalınabilirdi. Bizde bunu önlemek veya engellemek için giriş sayfası koyduk .

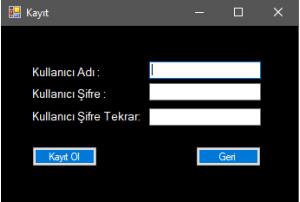


Şekil 1 / ilk giriş tasarımı

Projemize başlarken ilk yapım aşamasından kullandığımız giriş sayfası tasarımı yukarıdaki görseldeki gibidir.

Giriş yap tuşuna bastığınızda arkadaki işlemler şu şekildeydi; textboxtlara veri girilmiş mi, eğer veri girilmemişse lütfen kullanıcı adı ve şifre giriniz şeklinde bir uyarı messagebox tarafından gelir ve kullanıcın giriş bilgileri veri tabanı kısmında doğrulama yapılır ve bu doğrulama yanlış ise kullanıcı adı veya şifreniz yanlış diye messagebox tarafından tekrardan kullanıcıya uyarı verir eğer bütün bilgileriniz doğru ise veri tabanında kullanıcılar tablosundaki giriş bilgilerini textbox’ta girilen bilgileri kontrol eder ve ana sayfa ekranı ortaya çıkar.

Ardından bu programı kullanan kişi giriş yapması için kaydolması gerekiyordu bu sorunu da çözmek için programımıza kayıt sayfası ekledik.



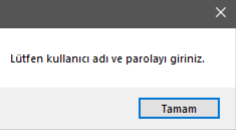
Şekil 2/ ilk kayıt sayfası

Projemize başlarken ilk yapım aşamasından kullandığımız kayıt sayfası tasarımı yukarıdaki görseldeki gibidir.

Kaydol butonuna basınca arkada olan işlemler şu şekildedir; Microsoft sql managment studio ile haberleşir ve bağlantı sağlanır, ardından textboxtlara

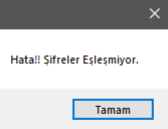
Girilen bilgiler kullanıcılar tablosuna eklenir ve mssql ile haberleşme sona erir.

Kayıt sayfasında kullanıcılar programa kaydolurken şifre bilgilerini doğru yazması ve kullanıcı adı daha önce programda kullanılıyorsa programın kullanıcıya hata vermesini gerekiyordu. Bu problemi c# da ki messagebox parametresiyle kullanıcıya mesaj gönderebildik.



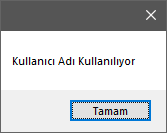
Şekil 3 Hata Mesajı

Eğer kullanıcı kullanıcı adı ve şifre kısımlarını boş girerse yukarıdaki görseldeki gibi kullanıcıya hata verecektir.



Şekil 4 Hata Mesajı 2

Eğer kullanıcı şifre kısımlarını şifre textboxlarına eksik ya da boş şifre girerse görseldeki gibi kullanıcıya hata verecektir.



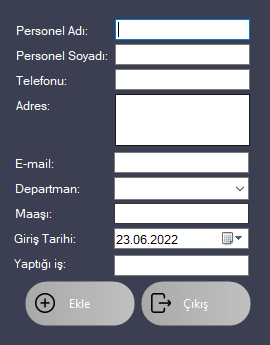
Şekil 5 Bildiri Mesajı

Eğer kullanıcı daha önceden programa kaydolmuş bir kullanıcı adı ile kaydolmaya çalışırsa görseldeki gibi kullanıcıya hata verecektir.

**3.4.2 Personel Kayıt Sayfası**

Projemizde çıkardığımız notları incelediğimizde hep en öne çıkan şey personel kayıt ekleme oldu ve bizde çıkardığımız notu göz önüne alarak personel adı, personel soyadı, telefonu, adresi, e-mail, departmanı, maaşı, işe giriş tarihi, yaptığı iş gibi bilgileri programımıza kaydetmek istedik.

Tasarım olarak da akılda kalıcı, sade ve kullanışlı bir tasarım kullandık. Ekle tuşuna basıldığında ise şu işlemler gerçekleşir;

Microsoft sql management studio ile bağlantı sağlanır eğer bağlantı sağlanmazsa program hata verir eğer bağlantıda bir sorun olmaz ise insan kaynakları veri tabanındaki personeller tablosuna personel adı, personel soyadı, telefonu, adresi, e-mail, departmanı, maaşı, işe giriş tarihi, yaptığı iş gibi bilgilerini veri tabanına ekler ve eklenilen personelle otomatik olarak bir personel id atar ve messagebox tarafından “personel başarıyla eklenildi” şeklinde bir mesaj verir eğer bilgileri doldurmaz boş bırakırsanız da “hata lütfen bilgileri eksiksiz giriniz” şeklinde messagebox tarafından kullanıcıya uyarı verir.

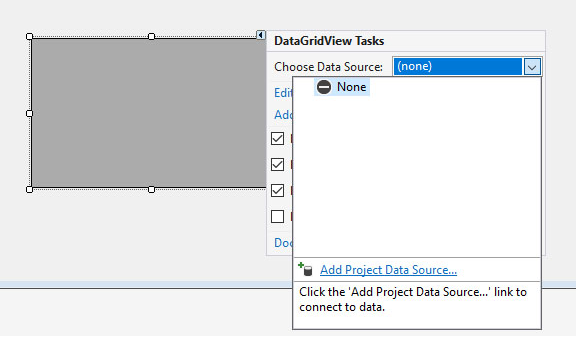
Şekil 6 Personel Ekleme

Yukarıdaki görsel projemizdeki personel ekleme sayfasıdır.

**3.4.3 Personel Listeleme Sayfası**

Projemizde personel ekleme sayfasını eklemiştik şimdi ise bize eklenilen personellerin listeleneceği bir sayfa gerekiyordu ve kafamızda birkaç soru işareti vardı personelleri nasıl listeleyecektik? Bu sorun hakkında düşünürken görsel programlama dersinde öğrendiğimiz datagridview nesnesi aklımıza geldi biz de datagridview ile ilgili araştırma yaptık bu araştırma sonucunda ilk yapacağımız iş datagridview ile microsoft sql managment arasında bağlantı kurmaktı. Bu bağlantı işlemini şöyle yaptık;

Öncelikle form ekranına bir **datagridview** yerleştiriyoruz ve daha önceden oluşturduğumuz veri tabanını bağlıyoruz.

****

Şekil 7 Datagridview Kurulum 1

"Add Project Data Source" a tıklıyoruz ve sırasıyla Database ve Dataset seçeneklerini seçerek ilerliyoruz.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şekil 8 Datagridview Kurulum 2

Sonrasında

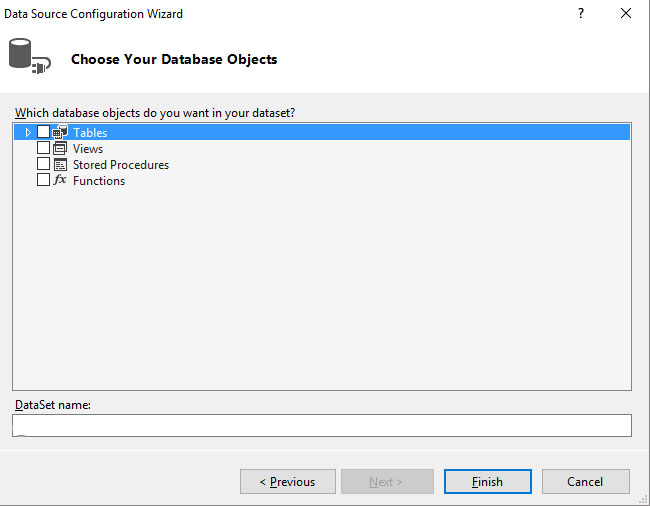
önümüze bağlantı yapmamızı istediğimiz veri tabanını seçme ekranı gelecektir ve bu ekran üzerinde NewConnection butonuna tıklıyoruz.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

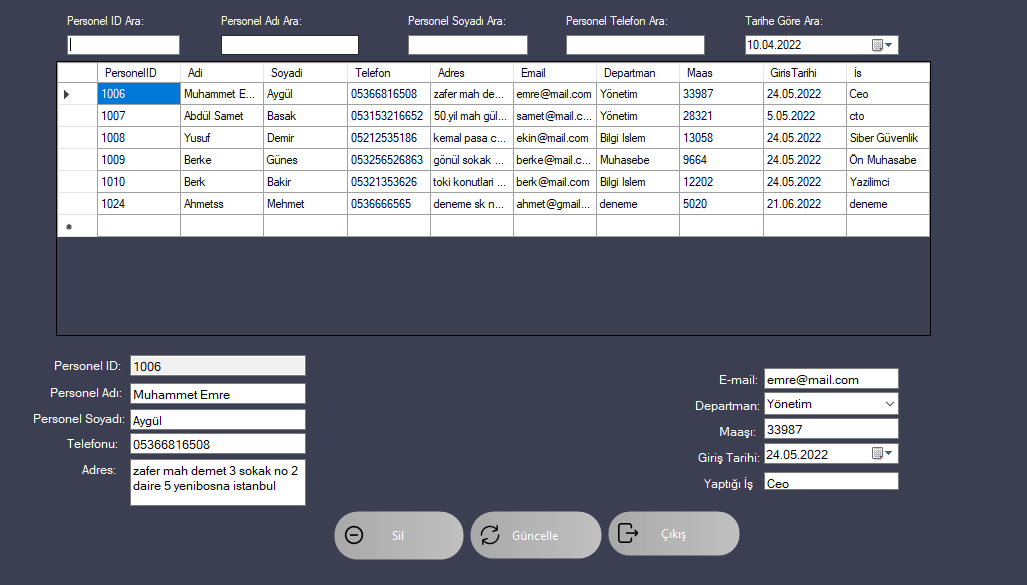
Şekil 9 Datagridview Kurulum 3

Add Connection panelinde bizden server isminizi istiyor Server Name bölümüne server adımızı yazıp Connect to a database kısmında oluşturduğumuz database’i seçip Test Connection butonuna basıp bağlantımızın sağlanıp sağlanmadığını kontrol ediyoruz. Ardından panelimizi kapatıyoruz.



Şekil 10 Datagridview Kurulum 4

Eklemek istediğimiz kısımları seçerek işlemlerimizi sonlandırıyoruz. Yapmış olduğumuz işlemler sayesinde artık datagridview ile veri tabanımızdaki kullanıcıları listeleyebiliriz.



Şekil 11 Personel Listeleme Sayfası

Datagridview işleminden sonra personel listeleme sayfamız yukarıdaki görseldeki haline geldi. Personelleri listeledikten sonra bir sorun daha ortaya çıkmıştı personeller gözüküyor ama aratma gibi işlemlerimiz yoktu. Kaynaklara bakarak datagridview üzerinde daha önce yapılmış filtreleme özelliklerine baktık ve personel listeleme sayfamıza entegre ettik bu arama yapmanın mantığı şu şekildeydi;

Arama textfield’ına bir yazı yazdığımızda gerekli sql komutları veri tabanı ile haberleşip aranan sonucu bulup datagridview’a aktarması şeklinde özetleyebilirim.

Artık personelleri listeleyip listelenen tabloda arama işlemlerini yapabiliyoruz. Sıra geldi güncelleme ve silme işlemlerine. Öncelikle silme ve güncelleme işlemleri yapmadan önce listelenen verileri textbox’a çekip textbox üzerinden silme işlemi yapmamız gerekiyordu bu işlem ise datagridview cell enter özelliği ile yapılacaktı, cell enter bloğunun içerisine gerekli parametreli yazdıktan sonra cell enter ile textbox a veri çekme işlemini yapabiliyoruz güncelleme işlemi ise personel ekleme koduna benzediği için işimiz çok kolay olacaktı. Kısa bir kod değiştirme sürecinden sonra artık güncelleme işlemi yapabiliyorduk silme işlemi ise güncelleme işlemi kadar kolay olmadı silme kısmında tabloda gözüken personelin personel id sine göre silme işlemini yapıncaktı. Biraz sql syntax araştırmasından sonra silme işleminde aktif hale getirip personel listeleme ile ilgili işlemlerimizi tamamlamış olduk.

**3.4.4 Departmanlar**

Departmanlar sayfasının mantığı şu şekilde yapılmıştır şirketlerde kimin nerede çalıştığını görmek için bölümlere ayrılmıştır ve bu bölümlere departmanlar denir. Yapmış olduğumuz araştırmalar sırasında departmanlar sayfasının gerekli olduğunu kanaatine varmış olup departmanlar sayfasını projemize ekledik ve içeresinde şu özellikler bulunmaktadır; Departman adı, departman açıklaması ve bu eklenen departmanların listelenmesi içinde bir datagridview ve gerekli olmayan departmaları silmek için bir silme butonu ve güncelleme butonu bulunmaktadır.Ayriyeten departmanlar kısmına eklediğimiz departmanlar personel ekleme sayfasındaki departmanlar combobox’na otomatik bir şekilde gelmektedir bu eklediğimiz özellikler sayesinde departmanlar sayfamızı oluşturmuş olduk.

**3.4.5 Maaş Zamları**

Maaş zamları sayfası düşünüldüğünde basit ama yapması zor ve uzun süren bir sayfa oldu. Bu sayfayı tasarlarken personellerin maaş zammını iki farklı şekilde yapmak istedik. Bunlar fiyata göre ve yüzdeye göre şu şekilde çalışacaktı combobox’tan seçtiğimiz personellerin maaşının yüzdesine göre zam yapmak oldu. Fiyata göre zam yapıldığında ise personelin maaşına yazılan fiyat kadar maaşına zam yapıldı.

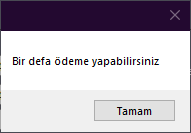
Bu iki özellik için araştırmalar yaptık ve nasıl personellerin maaşlarına zam yapacağımızı sql kodlarıyla gerekli parametleri yazdıktan sonra personel zam sayfamızın özellikleri tamamlandı.

**3.4.6 Mesai Ekle**

Projemizde yapmış olduğumuz mesai sayfamızda personellerin mesai işlemlerini eklediğimiz sayfadır. Bu sayfanın çalışma mantığı şu şekildedir comboboxtan mesai yapacak personel seçilir ardından başlangıç saati datetimepicker dan seçilir ve yine bitiş saatide datetimepicker dan seçilir ardından mesai ücreti yazılır ve başlangıç saati ve bitiş saati arasındaki fark çıkarılıp girilmiş olan mesai saat ücreti kısmı ile çarpılıp tutar kısmına otomatik bir şekilde ne kadar mesai yapılmış ise ona göre ücret yansıtılır ve açıklama kısmına ise mesai yapan personelin ne kadar mesai yapmış olduğunu bilgilendiren bir açıklama yazılır.

**3.4.7 Mesailer**

Mesailer sayfamızda mesai ekle sayfamızla bağlantılıdır. Mesai ekle sayfasındaki eklenen mesailer bu sayfada gözükmektedir. Bu sayfada datagridview’in içeresinde mesai id, personel id, personel ad, soyad, mesai başlangıç tarihi, mesai bitiş tarihi, saatlik mesai ücreti, tutar, açıklama ve ödeme sütunlarımız bulunmaktadır. Burada yeni olarak eklenen ödeme sütünün özelliği mesai yapan personelin mesai ödemesinin yapılıp yapılmadığı gösteren sütündur mesailer sayfasında bulunan mesai ödeme butonu mesaisi ödenmeyen personelin mesaisini ödemeye yaran bir butondur. Eğer bir personelin mesaisi ödenmiş olup bu butona bir kez daha tıklanırsa messagebox tarafından uyarılınır ve mesai yapan personele mesai ücretini maaşına tekrardan eklenmez. Diğer butonlar olarak sil butonu personelin id’sine göre silmeye yaran butondur. Güncelleme butonu ise güncellenecek kişinin id’sine göre güncellemeye yarayan butondur.

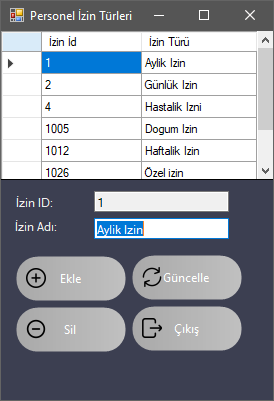


Şekil 11 Bildirim Mesajı

Yukarıdaki görselde mesai ödeme butonuna birden fazla kez tıklarsanız karşılaşacağınız hata mesajı

**3.4.8 İzin Listele**

İzin listeleme sayfamız personellerin yapmış olduğu izinleri takip etmek amacıyla yapılmıştır. Bu sayfadaki datagridview’a eklediklerimiz özellikler şuanlardır; izin hareket id, personel ad soyad, izin türü, izin başlangıç tarihi, izin bitiş tarihi ve açıklama sütunlarımızdan oluşmaktadır. Sayfamızdaki butonlar şunlardır; izin türleri, ekle, güncelle, sil. İzin türleri butonuna bastığımızda ise istediğimiz izinin adını girip ekleyeceğimiz, sileceğimiz, güncelleyebileceğimiz, kısımlardan oluşmaktadır ve eklenen izinler izin türleri adındaki combobox’ımza otomatik bir şekilde listelenmedir. Ekle butonumuza tıklamadan önce izin yapacak personelin adı ve izin türü seçilmelidir bu seçimleri yaptıktan sonra ekle butonumuza tıklarsak datagridview’da izin yapacak olan personel gözükmektedir. Ardından güncelle butonumuz ise önce datagridview’a tıklayıp güncelleme yapacağımız özellikleri tıkladıktan sonra güncelleme yapabiliriz silme butonumuz ise datagridview’a tıklayıp ve sil butonumuza bastıktan sonra kayıtın silindiği kısımdır.



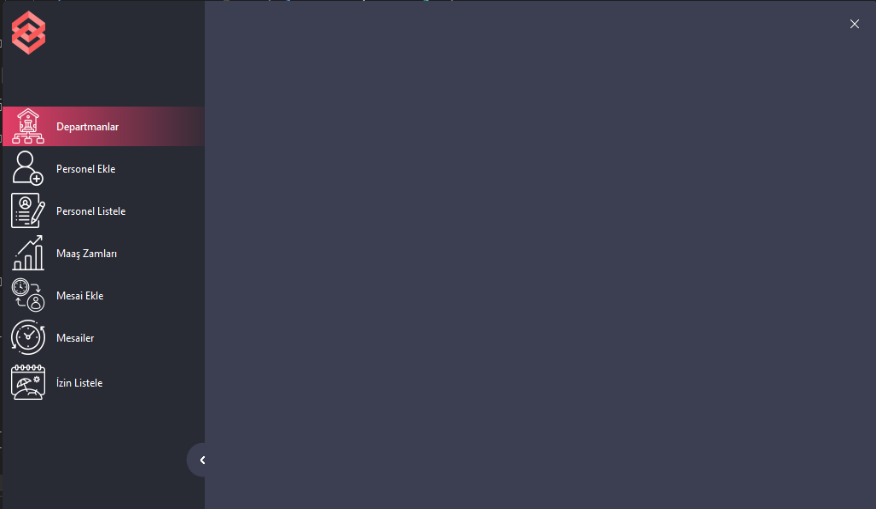
Şekil 12 Personel İzin Türleri

Yukarıdaki görselde izin listele sayfamızdaki izin türleri butonumuza tıkladığınızda karşınıza gelecek sayfadır

**3.4.9 Ana sayfa**

Şekil 13 Anasayfa

Ana sayfa projemiz için kullanışlı ve çok kafa karıştırıcı olmaması gerekiyordu bizde ana sayfa tasarımı için araştırmalar yaparken kullanıcı için en kullanışı tasarımı yukarıdaki görseldeki gibi tasarladık. Tasarımı yaparken visual studio içindeki tasarım araçları yeterli olmadığı için projemize guna ui framework tasarım aracını dahil ettik ve ana sayfamızın sol tarafına açılır kapanır bir menü ve projemizdeki özellikleri kullanabilmek için gerekli butonları ekledikten sonra butonların içine akılda kalıcı ikonlarımızı tasarladık ve sonunda ana sayfa kısmını tamamlamış olduk.

****

Şekil 15 Anasayfa Menü

Yukarıdaki görselde ana sayfa da menüde butonların üzerine gelindiğinde Mouse imlecinin buton üzerinde olduğu belirten bir kırmızı işaret çıkar.

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu**3.4.10 Veri tabanı**

Şekil 16 deparmanlar

tablo içeren bir resim

Açıklama otomatik olarak oluşturulduDepartmanlar sayfamızdaki bilgilerin hangi veri türüne göre bilgilerin tutulacağı kısımlardır departmanid departman eklerken otomatik bir şekilde sayı verdiği kısımdır, departman satırı ise departman adının kayıt edileceği kısımdır, satır açıklama satırı ise departman açıklamasının yapıldığı satırdır.

Şekil 17 izin

İzin tablosu izin sayfasındaki izinlerin bilgilerin tutulacağı kısımdır, id kısmı eklenilen veriye otomatik bir sayı verdiği kısımdır, ad kısmı ise personelin adı ve soyadını tutulacağı kısımdır , izintur satırı izin türlerinin tutulacağı kısımdır, izinbas ve izinbit izin tarihlerinin tutulduğu kısımdır, açıklama ise izin yapan personelin neden izin yaptığı gibi bilgilerin tutulacağı kısımdır.

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şekil 18 izin türü

İzin tür tablosu izin listele sayfasındaki izin türleri butonuna tıkladığımızdaki açılan sayfadaki izin türlerinin saklanacağı tablodur.

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şekil 19 Kullanıcı Adı

Kullanıcılar tablosu ise programımıza kaydolan kullanıcıların bilgilerinin tutulduğu kısımdır. Bu tabloda kullaniciAd kullanıcı adını kullaniciSifre kullanıcı şifresinin bilgilerini tutar.

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şekil 20 Mesailer

Mesailer tablosu mesai sayfasındaki mesai yapan personellerin verilerinin tutulacağı kısımdır. Bu tabloda personel adını, saatbas mesai başlangıç saatini, saatbit mesai bitiş saatini, mesaiucret mesai ücretini, tutar Yapılan mesainin toplam ne kadar tuttuğunu, aciklama mesai yapan personelin açıklamasını, Odenme ise personelin mesai ücretinin ödenip ödenmediği gibi bilgilerin saklanacağı kısımlardır.

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şekil 21 Personeller

Personeller tablomuz ise programımıza eklenen personellerin bilgilerinin tutulduğu kısımdır. Bu tabloda PersonelID personel numarasını, adi personel adını, soyadi personel soyadını ,telefon personel telefonunu, adres personel adresini, email personel emailini, departman personelin kayıt edildiği departmanı ,Maas personelin maaş bilgilerini, GirisTarihi personelin işe giriş tarihini ve İs personelin yaptığı iş gibi bilgilerinin tutulacağı kısımlardır.

**BÖLÜM 4**

**SONUÇLAR**

**4.1 Karşılaştığımız Sorunlar**

İlk olarak karşılaştığımız sorunları çözmek bizim de için hiç de kolay olmadı. Ilk karşılaştığımız sorun olan personel zam sayfasında bir personele zam eklerken zam eklemeyi personel adına göre yazmıştık ve bu sorunu uzun süre görememiştik. Bir gün aynı isimli 2 personel eklerken zam yaptığım personele zam eklenmişti fakat diğer personele de zam eklenmişti. Bu sorunu çözmek için düşündükten sonra şunu uygulamaya karar verdik. Öncelikle seçilen personelin adı ve soyadı kısımları farklı bir combobox’ta olması gerekiyordu adı ve soyadı kısımlarını ayrıdıktan sonra sorun hala devam ediyordu. Biraz daha düşündükten sonra personel idsine göre zam yapmayı düşündük ama bunu nasıl yapacaktık? Bu sorunu da şöyle çözmüştük. Seçilen isime göre bir tane label’a personel idsini aktaracaktık ve bu aktardığımız label’a göre işlemleri yapması gerekiyordu programın gerekli yerlerine gerekli kodları yazdıktan sonra bu sorunu çözmüştük. Artık aynı isimle bir çok farklı kişiye zam yapabiliyoruz.

İkinci karşılaştığımız sorun ise mesai ekleme sayfasında karşılaştık. mesai eklerken sadece personellerin adını gönderebiliyorduk ve mesai listeleme sayfasında sadece mesai yapan personelin adı gözüküyordu. öncelikle bu sorunu çözmek için maaş zam sayfasında yapmış olduğumuz gibi personel adı ve soyadı kısımlarını ayırdık ayırdıktan sonra seçilen personelin adı ve soyadını programı kullanırken gözükmeyen gizli bir label yerleştirdik bu yerleştirdiğimiz label’a personel adını ve soyadını birleştirip mesailer sayfasına yollamayı başard

**Kaynaklar**

**[1]** [www.csharp**nedir**.com](http://www.csharpnedir.com/)

**[2]** [www.webmastersitesi.com](http://www.webmastersitesi.com/)

**[3]** [http://msdn.microsoft.com](http://msdn.microsoft.com/)

**[4]** [http://www.se7ensins.com](http://www.se7ensins.com/)

**[5]** [http://www.redmondpie.com](http://www.redmondpie.com/)

**[6]** <http://stackoverflow.com/>

**[7]** [http://www.c-sharpcorner.com](http://www.c-sharpcorner.com/)

**ÖZGEÇMİŞ**

**Adı Soyadı :** Muhammet Emre Aygül

**Doğum Yeri ve Tarihi :** İstanbul / 2001

**E-Posta :** [muhammetemreaygul20@istanbularel.edu.tr](mailto:muhammetemreaygul20@istanbularel.edu.tr)

**Eğitim :** Yeni Bosna orta okulu , Bahçelievler Türk Telekom Anadolu teknik lisesi

**Staj ve İş Deneyimleri :** Mikron Bilgisayar( Helpdesk)

**Adı Soyadı :** Abdül Samet Başak

**Doğum Yeri ve Tarihi :** İstanbul / 2001

**Eğitim :** Gaziosmanpaşa orta okulu , Gaziosmanpaşa Anadolu Lisesi

**E-Posta :** abdulsametbasak20@istanbularel.edu.tr

**Staj ve İş Deneyimleri :** Yok