

PROJE EKİP

Sümeyye BAYRAKTAR

Deniz KALENDER

Sena ATAK

FIRAT ÜNİVERSİTESİ

Teknoloji Fakültesi

Yazılım Mühendisliği

İLERİ PROGRAMLAMA TEKNİKLERİ



Proje Yürütücüleri

Fatih ÖZKAYNAK

MAYIS 2021

İÇİNDEKİLER

1.Giriş

ÖNSÖZ

1.1 Projenin Tanıtılması

1.2 Projenin Amacı

1.3 Projenin Kapsamı

1.4 Tanımlamalar ve Kısaltmalar

2.Proje Planı

2.1 Giriş

2.2 Projenin Plan Kapsamı

2.3 Proje Zaman-İş Planı

2.4 Proje Ekip Yapısı

2.5 Kullanılan Özel Geliştirme Araçları ve Ortamları

2.6 Proje Standartları, Yöntem ve Metodolojiler

2.7 Kalite Sağlama Planı

2.8 Eğitim Planı

2.9 Test Planı

2.10 Bakım Planı

3.Sistem Çözümleme

3.1 Mevcut Sistemin İncelenmesi	3.1.1 İşlevsel Model
	3.1.2 Veri Modeli
	3.1.3 Varolan Sistemin Değerlendirilmesi
3.2 Gereksenen Sistemin Mantıksal Modeli	3.2.1 Genel Bakış
	3.2.2 Başarım Gerekleri
3.3 Arayüz (Modül) Gerekleri	3.3.1 Yazılım Arayüzü
	3.3.2 Kullanıcı Arayüzü
	3.3.3 İletişim Arayüzü
	3.3.4 Yönetim Arayüzü
3.4 Belgeleme Gerekleri	3.4.1 Geliştirme Sürecinin Belgelenmesi
	3.4.2 Eğitim Belgeleri
	3.4.3 Kullanıcı El Kitapları

4. SİSTEM TASARIMI

4.1 Genel Tasarım Bilgileri	4.1.1 Genel Sistem Tanımı
	4.1.2 Varsayımlar
	4.1.3 Sistem Mimarisi
	4.1.4 Dış Arabirimler
	4.1.5 Veri Modeli
	4.1.6 Testler
4.2 Süreç Tasarımı	4.2.1 Genel Tasarım
	4.2.2 Modüller
4.4 Ortak Alt Sistemlerin Tasarımı	4.4.1 Ortak Alt Sistemler
	4.4.2 Modüller arası Ortak Veriler
	4.4.3 Veri Dağıtım Altsistemi
	4.4.5 Yedekleme ve Arşivleme İşlemleri

5. SİSTEM GERÇEKLEŞTİRİMİ

5.1. Giriş

5.2. Yazılım Geliştirme Ortamları

5.2.1 Programlama Dilleri

5.2.2 Veri Tabanı Yönetim Sistemleri

5.3. Kodlama Stili

5.3.1 Açıklama Satırları

5.3.3 Anlamlı İsimlendirme

5.5. Olağan Dışı Durum Çözümleme

5.5.1 Olağandışı Durum Tanımları

5.5.2 Farklı Olağandışı Durum Çözümleme Yaklaşımları

5.6. Kod Gözden Geçirme

6. DOĞRULAMA VE GEÇERLEME

6.1. Giriş

6.2. Sınama Kavramları

6.3. Doğrulama ve Geçerleme Yaşam Döngüsü

6.5. Sınama ve Bütünleştirme Stratejileri

6.5.1 Yukarıdan Aşağı Sınama ve Bütünleştirme

7. BAKIM

7.1 Giriş

7.2 Kurulum

7.3 Yerinde Destek Organizasyonu

7.4 Yazılım Bakımı

8. Sonuç

9. Kaynaklar

1.GİRİŞ

- **1.1.Projenin Tanıtılması:**

Günümüzde petrolden sonra en değerli şey veridir. Bizde bu cümleden yola çıkarak farklı ülkelerdeki uygulamaları kullanmaktansa kendi yerli instagram benzeri fotoğraf paylaşabileceğimiz bir web sitesi yapmayı amaçladık. Web sitesine eklenen fotoğrafları jpeg algoritmalarıyla boyutlarını küçültüp veri tabanımızda sakladık.

- **1.2.Projenin Amacı:**

Projenin amaç kısmında sorun analizi ve hedef analizi yapılır.
Sorun analizi yaptığımızda;

- Yerli fotoğraf paylaşım sitemizin olmaması
- Fotoğrafların büyük yer kaplaması

Hedef analizi yaptığımızda da;

- Jpeg Algoritması fotoğraf sıkıştırarak kaplanan boyutu küçültür.

- **1.3.Projenin Kapsamı:**

- ✓ Bilgisayar kullanmayı bilen,
- ✓ Fotoğraf paylaşmak isteyen kişiler

- **1.4. Tanımlamalar ve Kısaltmalar**
S= Satır Sayısı(Bin türünden)

2.PROJE PLANI

- **2.1 Giriş:**

Bugs uygulamasıyla büyük boyutlardaki veriler jpeg algoritmalarıyla sıkıştırılarak saklanır. Arkadaş bulma, resim ekleme-silme işlemlerinin yapılmasını sağlar.



Şekil 2.1.Projenin Genel Yapısı

- **2.2 Projenin Plan Kapsamı:**

Projenin planı kapsamında mevcut sistemler incelendiğinde resimlerin büyük boyutlar kapladığı gözükmiştir. Aynı zamanda yerli fotoğraf paylaşma uygulamamız bulunmamaktadır. Yabancı uygulamalar incelendiğinde resimlerin kalitesinin bozulduğu gözlemlenmiştir. Buradan yola çıkarak da jpeg algoritması kullanılan bir sistem tasarlanmıştır.



Resim 2.1. Sistemin Genel İşleyişi

Bugs Web Sitesi Neden Gerekli

- ✚ Fotoğrafların büyük boyutta yer kaplamaması,
- ✚ Verilerin ülkemizde kalması
- ✚ Kolayca arkadaş bulmak, onun resimlerini görmesi için

Araba Kiralama Web Sitesi Alt Yapısı Nasıl Çalışır?

- Üye paneli;
 - ✓ Sistem il açıldığında Giriş Ekranı açılır. Eğer sisteme üye iseniz giriş bilgileri doldurularak giriş yapılır.
 - ✓ Eğer üye değilse aşağıdaki kısımdan üye olunur.
 - ✓ Giriş bilgileri doldurulduktan sonra daha önce paylaşılmış olan resimler listelenir.
 - ✓ Resim ekleme-silme işlemleri yapılabilir.
 - ✓ Sistemden çıkış yapılabilir.
 - ✓ Arkadaş aranabilir.
- Admin paneli;
 - ✓ Resim eklenir-silinir.

Maliyet Kestirim Dökümanı

Cocomo modeline göre hesaplama yapılacaktır.

Projemiz Ayrık Proje Kapsamına girmektedir. O yüzden;

$$\text{İş Gücü (K)} = 2.4 * S^{1.05}$$

$$\text{Zaman (T)} = 2.5 * K^{0.38}$$

Satır Sayısı: 4678

S = 5 olarak alalım;

$$K = 2.4 * 5^{1.05} = 13.005576$$

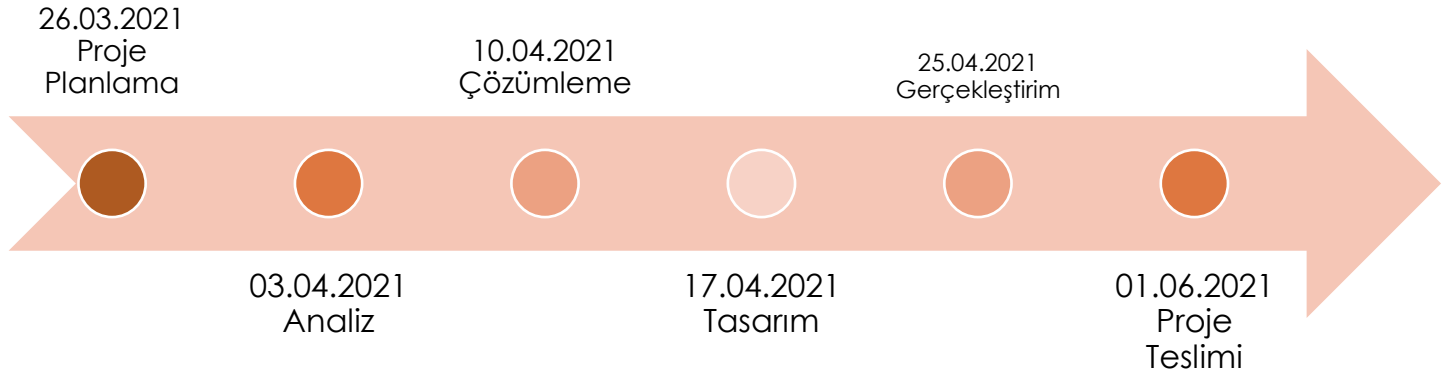
$$T = 2.5 * 13.005576^{0.38} = 6.62675$$

	Effort	13,01 MM
	Duration	6,63 Months
	Team	1,96 People (average)
	Effort	Schedule
Plans & Reqs.	0,8	0,8
Design	2,1	1,3
Development	8,1	3,6
Integration & Test	2,9	1,7
Total	13,8	7,4

Category	Factor	Assessment	Parameter
Product Attributes	Required Software Reliability	Very_High	1,4
	Database Size	Nominal	1
	Product Complexity	Nominal	1
Computer Attributes	Execution Time Constraints	Nominal	1
	Main Storage Constraints	Nominal	1
	Virtual Machine Volatility	High	1,15
Personnel Attributes	Computer Turnaround Time	Nominal	1
	Analyst Capability	Nominal	1
	Applications Experience	High	0,91
	Programmer Capability	Nominal	1
	Virtual Machine Experience	Low	1,1
Project Attributes	Programming Language Experience	Nominal	1
	Use of Modern Programming Practices	Very_High	0,82
	Use of Software Tools	High	0,91
	Required Development Schedule	Nominal	1
	Effort Multiplier		1,20258

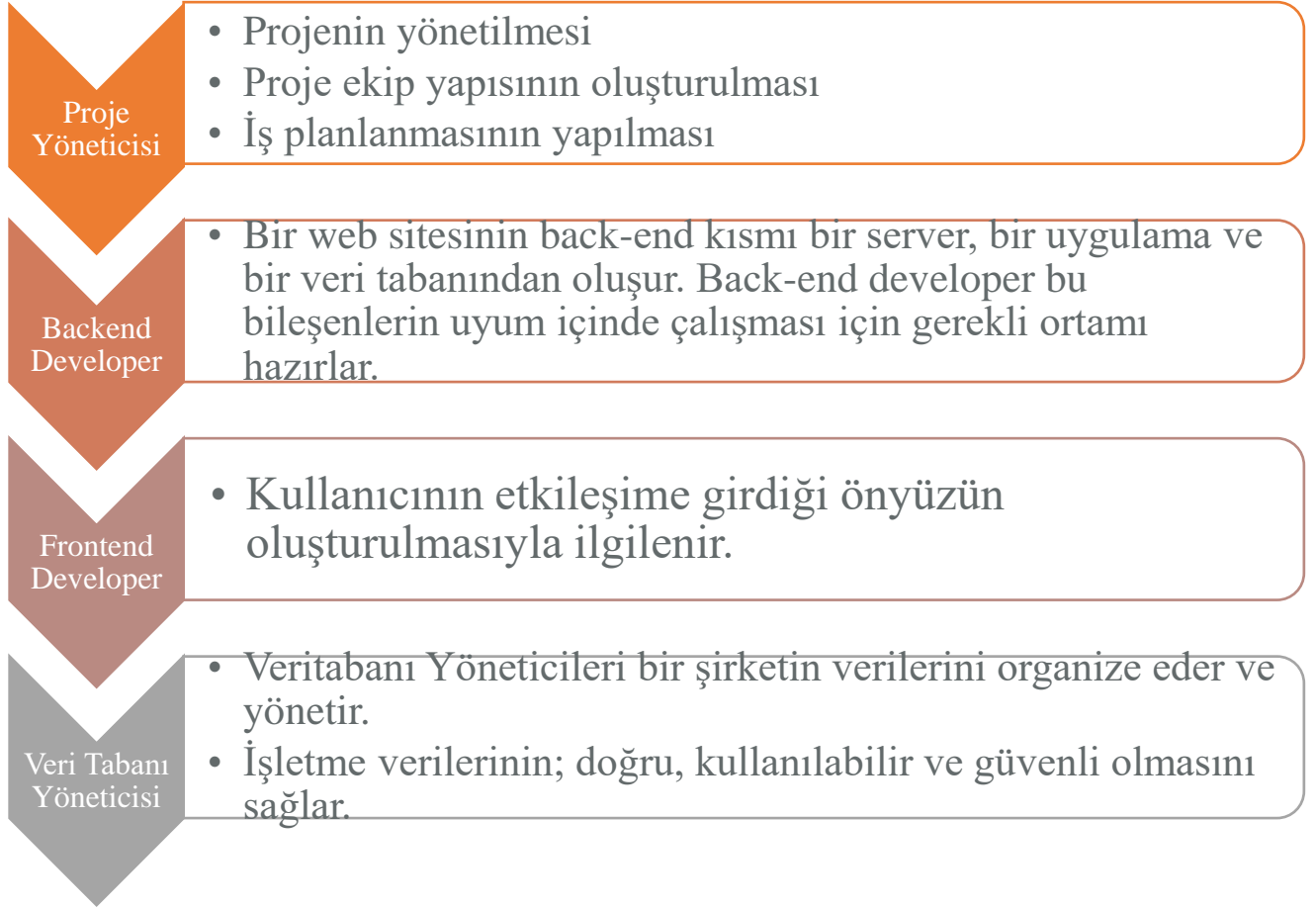
PM_NOM	19,32 MM
PM	23,23 MM
TDEV	6,84 Months
TEAM	3,40 People (average)

2.3 PROJE İŞ ZAMAN PLANI

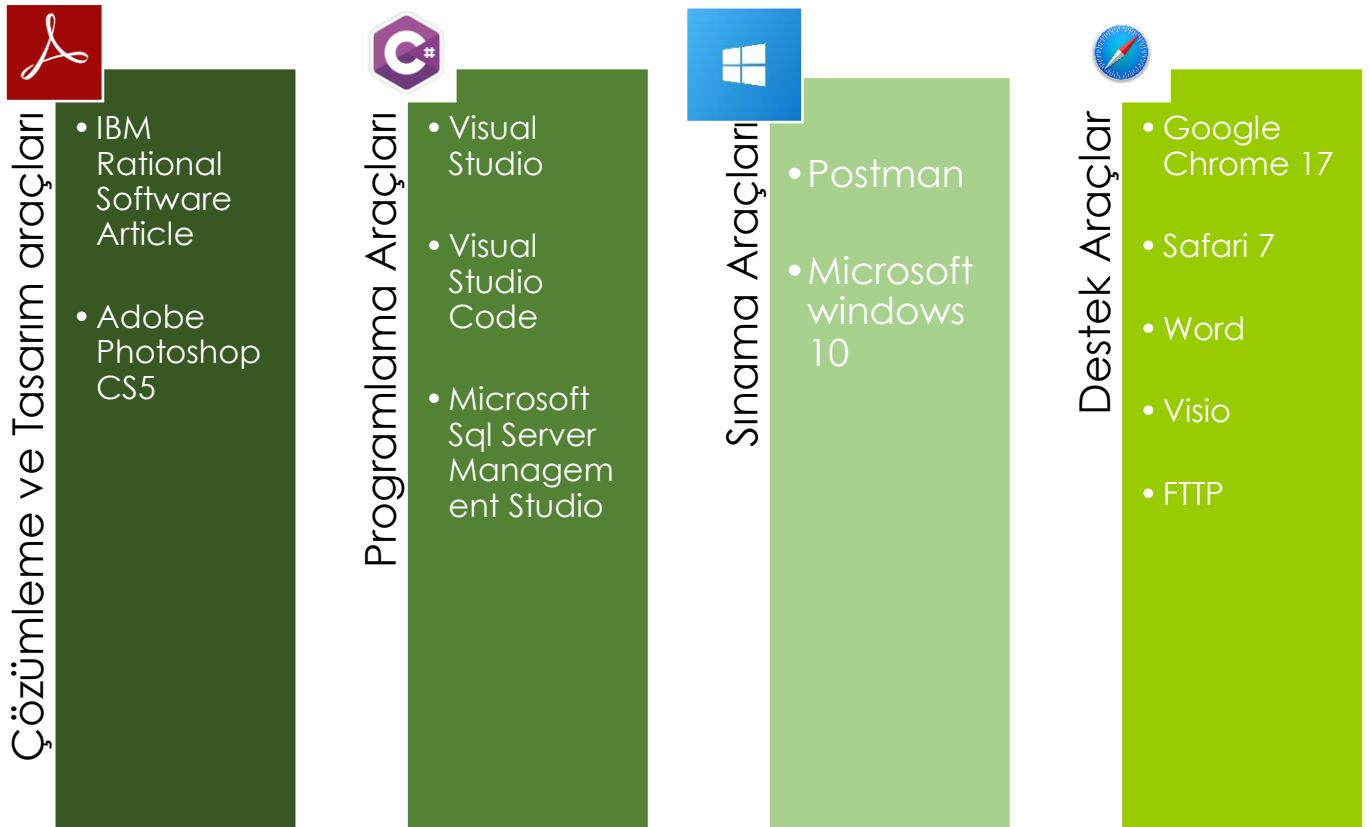


2.4 Proje Ekip Yapısı





2.5 KULLANILAN ÖZEL GELİŞTİRME ARAÇLARI VE ORTAMLARI



2.6 Proje Standartları, Yöntem ve Metodolojiler

Artırımsal Geliştirme Süreç Modeli kullanılacaktır. Bunu kullanmamızın amacı daha çok test imkanı sunarak hataların azaltılması ve bunun yanı sıra da adım adım özellik ekleyerek gidebilmemizdir. Üretilen her yazılım ürünü birbirini kapsayacak ve giderek artan sayıda işlev içerir. Uzun zaman alacak ve sistemin eksik işlevlerle kullanılabileceği projelerde kullanılan bu yöntem bir taraftan kullanım imkanı sunarken diğer taraftan da üretim imkanı sunar. Planlama, analiz ve tasarım aşamalarından sonra bir çekirdek yapı oluşturularak bir sürüm üretilir. Ve sistem artırılması yapılarak devamlı bir şekilde gelişmiş sürümler çıkar. Program hazır olana kadar bu döngü devam eder. Her teslimle birlikte müşteriye bir ürün geldiği için sistem erken aşamalarda işlevsellik kazanır. Bu döngü sayesinde sistem daha fazla test edilme imkânı bulmuş olur.

Böl ve yönet yaklaşım tarzına uygun bir modeldir.



Proje yapılmaya ilk başlandığında sadece resimler listelenmekteydi bu kısım bizim çekirdeğimizi oluşturmaktadır. Daha sonra giriş ekranı oluşturulmuş girilen şifre doğru ise sisteme girişe izin verilip özellik-1 olarak belirlenmiştir. Daha sonra ise üye ol kısmı eklenmiştir. Böyle özellikler eklenerek sistem son halini almıştır.

2.7 Kalite Sağlama Planı

Modülerlik	Yeniden Kullanılabilirlik	Temizlik
<ul style="list-style-type: none">Proje katmanlara ayrılarak oluşturulmuş bu sayede de modülerlik sağlanmıştır.	<ul style="list-style-type: none">Projemizde Core isimli katman yazılmıştır. Bu diğer projelerde de hiçbir kod değiştirilmeden kullanılabilecek katmandır.	<ul style="list-style-type: none">SOLID prensiplerine uyulmuştur.Projemiz Clean Code mantığıyla yazılıp gereksiz hiçbir kod yazılmamıştır.

Güvenilirlik	Anlaşılabilirlik	Belgeleme
<ul style="list-style-type: none"> Uygulamamızda SHA512 algoritması kullanarak şifreleme yapılacaktır. Aynı zamanda Transaction, Authorization ve Validation yönetimi sağlanmıştır. 	<ul style="list-style-type: none"> Standart metot ve değişken isimlendirmeler kullanılmıştır. İsimlendirmedelerde İngilizce isimler tercih edilmiş basit kod parçalarından yazılmıştır. Aynı zamanda yorum satırlarına da yer verilmiştir. 	<ul style="list-style-type: none"> IEEE Srs formatına göre belgeleme yapılmıştır.

2.8 Eğitim Planı

Projeden kazanılacak en önemli olaylardan biride eğitimidir. Kullanılacak dillerin ara yüz editör ve programların kullanımında hakim olunamaması halinde bu program başarıyla neticelendirilemez. Bu yüzden projede bazı eğitimler alınması gereklidir. Proje kapsamında alınacak olan eğitimler;

- C# Dil Eğitimi
- Angular için TypeScript
- SQL Dil Eğitimi gereken eğitimlerdir.

Konu	Süreç
Giriş, açıklamalar, ön bilgi	1.gün
Temel kavramlar ve sistem için gerekli bileşenlerin tanımlanması	2.gün
Gerekli yöntem ve metodolojilerin nasıl kullanılacağı	3.gün
Gerekli hata çözme yöntemlerinin incelenmesi	4.gün
Değerlendirme ve Sonuç	5.gün

Şekil 2.14 Eğitim Planı

Proje teslimi sonrası sistemi kullanacak olan kişiler arasından seçilen yetkili kişilere proje kullanımı için 1 günlük seminer verilecektir. Bu seminerin içeriği şu şekildedir.

- ✓ Sisteme genel bakış,
- ✓ Sistemde anketin nasıl tamamlanacağı,
- ✓ Sistemde yetkili yöneticilerin yetkileri,
- ✓ Sistemin kısıtları açılarak nasıl kullanıldığına dair bilgiler vermektedir.

	Başlama Tarihi	Bitiş Tarihi	Sümeyye Bayraktar
C# Dil Eğitimi	6 Haziran	7 Haziran	+
SQL Dil Eğitim	6 Haziran	7 Haziran	+
Angular + TypeScript	6 Haziran	7 Haziran	+

Şekil 2.15 Eğitim Planı II

Bu şekilde bir eğitim planı uygulanmaktadır. Sistem tam olarak hazırlandığında eğitim planı da garanti olarak ayarlanacaktır ve ona göre eğitim semineri verilecektir. Zira verilmemesi halinde kullanıcılar sistemi tam olarak kullanamayacak ve bazı açıklıklar meydana gelecektir.

2.9 Test Planı

Proje test ekipleri ve görevleri şu şekildedir;

- Şirket bünyesinden seçilmiş bazı kişiler üye olarak resim eklerken, diğer kısım ise üye olma işlemi yaparak sistem kontrol edilir. Veri tabanına kayıt aktarımı yapılmasına dikkat edilir.

2.10 Bakım Planı

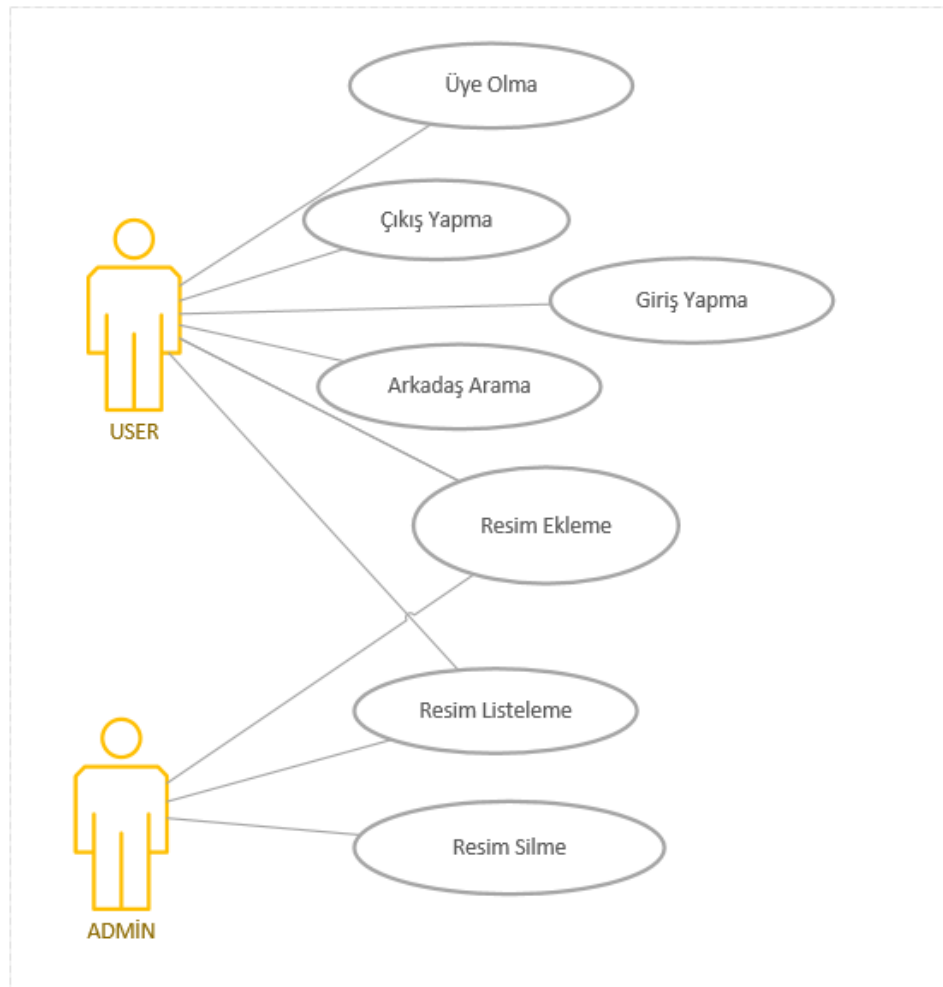
Üye eklenip çıkarılması, resim ekleme silme listeleme gibi tüm bu sistemsel değişiklikler bakım planında yapılacaktır. Sistem bakımı Sümeyye BAYRAKTAR, Sena ATAĞ, Deniz KALENDER tarafından yapılarak her türlü güncelleştirmeye olanak sağlayacaktır.

3.SİSTEM ÇÖZÜMLEME

3.1 Mevcut Sistem İncelemesi

Mevcut sistem incelendiğinde, kalitenin bozularak sisteme yüklendiği, fotoğraf boyutunun büyük olduğu, yine arkadaş arama kısmının olduğu bir sistem gözlemlenmiştir. Kendi sistemimizde ise fotoğraf kalitesinin minimum bozulmaması ve boyutunun küçültülmesi ana amacımızdır.

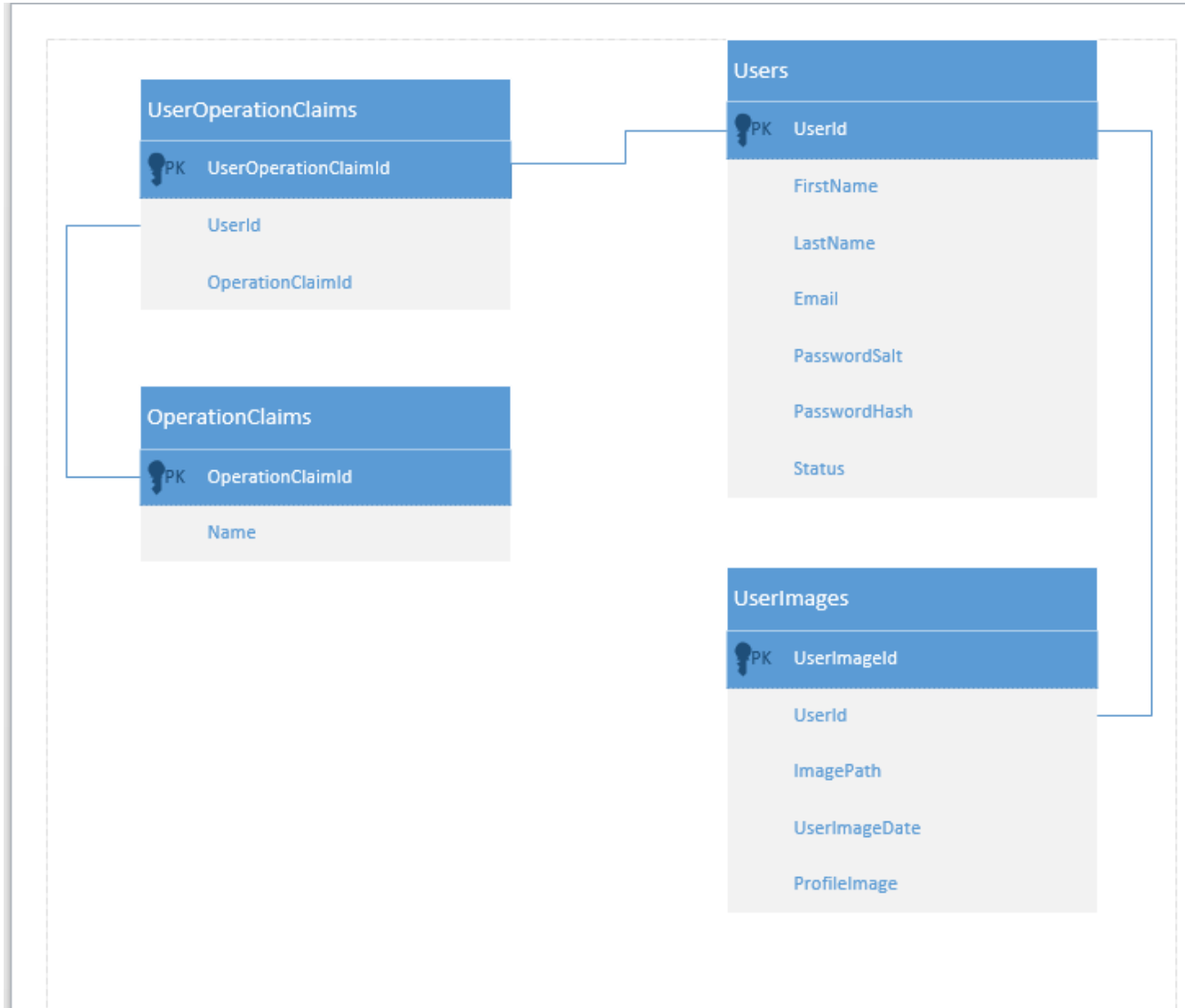
3.1.1 İşlevsel Model

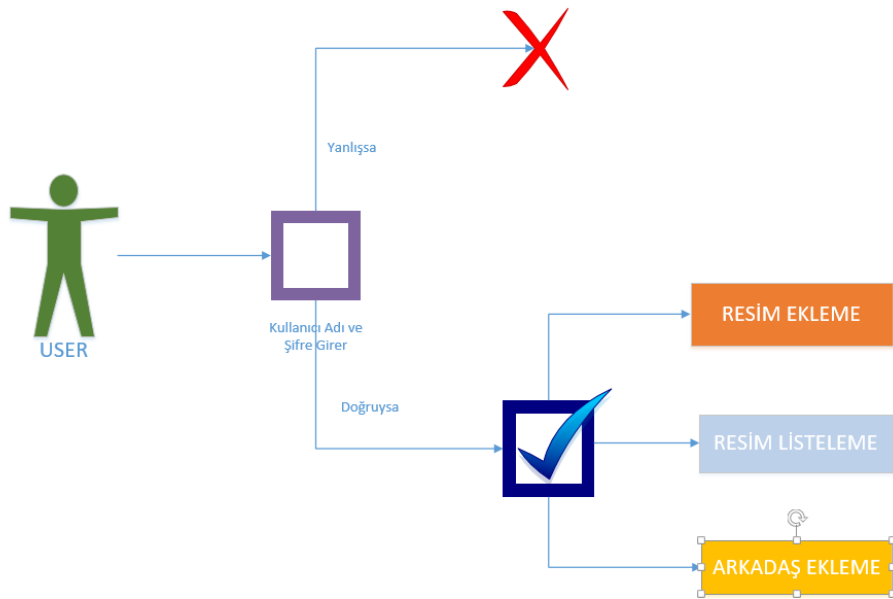
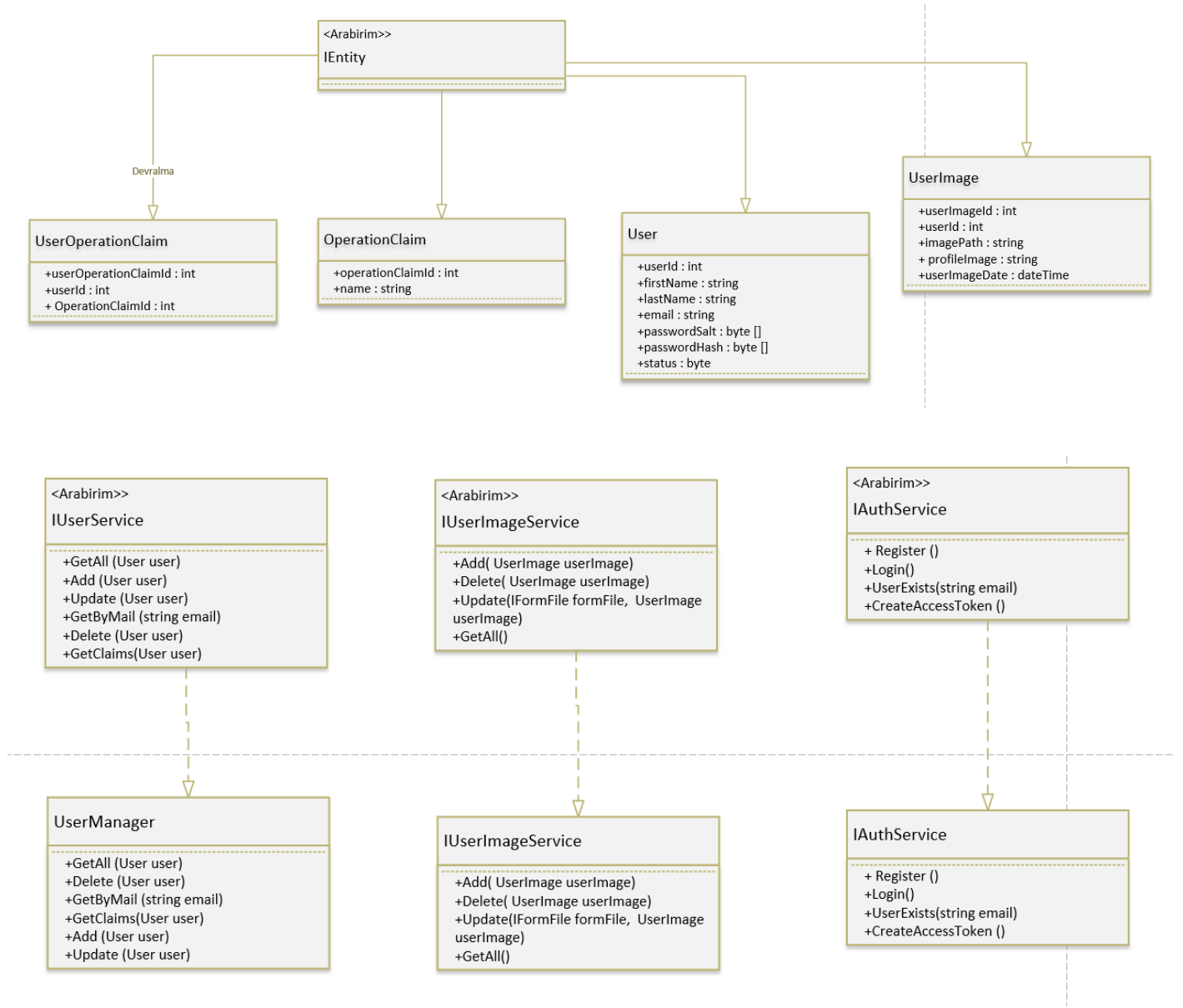


Şekil 3.1 Use Case Diyagramı

3.1.2 Veri Modeli:

Veri tabanı ilişkisel veri modelinde veriler tablolar üzerinden kurulan ilişkiye dayanmaktadır.





Mevcut Sistemin Eksikleri

Mevcutta olan sistemin eksikleri;

- Resimlerin büyük boyutlar kaplaması,
- Resim yüklenirken kalitenin çok fazla bozulması

Çözümü;

- Jpeg algoritması kullanmak

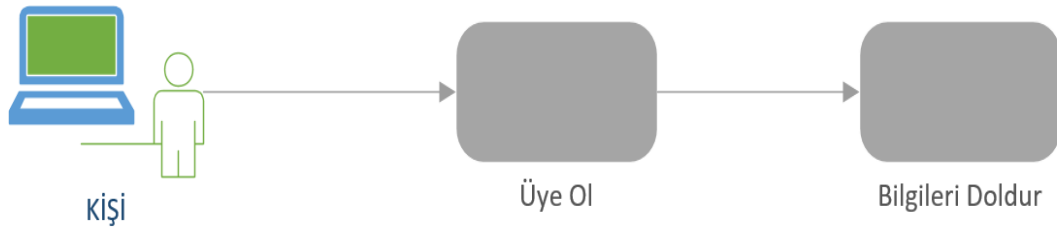
Gerekli Sistem Modeli:

- ✓ Kullanıcı web sitesini açtığında giriş ekranı açılır.
- ✓ Eğer üye değilse kişinin üye olması beklenir.
- ✓ Üye olmak için bilgiler doğru doldurulmalıdır.
- ✓ Üye olduktan sonra giriş yapılır ve eklenen arkadaşların resimleri burada listelenir.
- ✓ Fotoğraf eklenirken jpeg algoritmasından geçer.
- ✓ Arama butonuyla hesaplar aranır.

VERİ SÖZLÜĞÜ

<i>Dosya Adı</i>	<i>Değişken Adı</i>	<i>Veri Tipi</i>	<i>Uzunluk</i>	<i>Açıklama</i>
<i>Users</i>	Id	İnt	4	Kullanıcı numarası
	FirstName	Varchar	50	Kullanıcının Adı
	LastName	Varchar	50	Kullanıcının Soyadı
	Email	Varchar	50	Kullanıcının mail adresi
	PasswordSalt	varbinary	500	Kullanıcı şifresinin saltlanmış hali
	PasswordHash	varbinary	500	Kullanıcı şifresinin hashlenmiş hali
	Status	Bit	1	Kullanıcının aktif olup-olmaması durumu

Dosya Adı	Değişken Adı	Veri Tipi	Uzunluk	Açıklama
UserOperationClaims	Id	İnt	4	Kullanıcı-Yetki numarası
	UserId	İnt	4	Kullanıcı Numarası
	OperationClaimId	İnt	4	Yetki Numarası
OperationClaims	Id	İnt	4	Yetki numarası
	Name	nvarchar	250	Yetki adı
UserImages	Id	İnt	4	Resminin numarası
	UserId	İnt	4	Resmi yükleyen kullanıcı numarası
	ImagePath	nvarchar	50	Yüklenen resminin yolu
	ImageDate	datetime		Resim yüklenme tarihi



3.2 Başarım Gerekleri

Mevcut sistemler incelendi ve mevcut sistemin eksiklerinden yola çıkılarak, sistemin başarımı için

- Kolay üye olma imkanı,
- Kolay giriş,
- Güvenilirlik,
- Kaliteli ve düşük boyutlu fotoğraf paylaşımı,

temel gereklilikler olarak tespit edilmiştir.

3.3 Arayüz (Modül) Gereklere

3.3.1 Yazılım Arayüzü

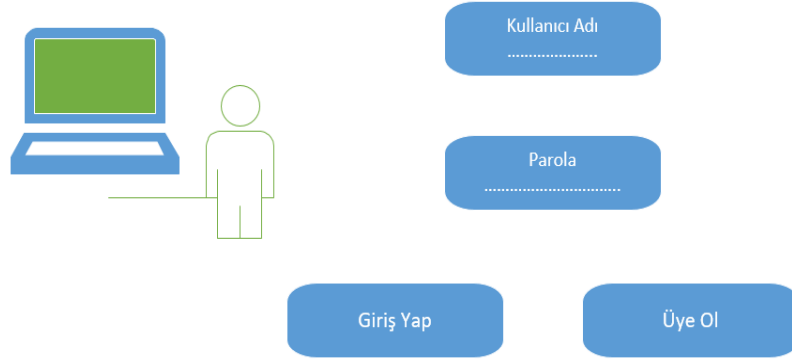
Projenin çalışması esnasında açık verilmemesine özen gösterildi. Gerekli olan her türlü değişiklik C# kodları ve postman üzerinden yapılarak tekrar derlenecek. Kodlanması sürecinde kullanılacak olan Katmanlı yapı ve standart kodlarla hata bulma işlemi de kontrol altına alınmıştır.

3.3.2 Kullanıcı Arayüzü

Projede kullanıcının arayüzü tasarlanırken göze hitap edecek bir şekilde renkler seçilerek tarafsız rahat büyük puntolu yazılı bir arayüz tasarlanacaktır. Sade, kolay anlaşılabilir olmalıdır.

3.3.3 Yönetim Arayüzü

Proje boyunca kullanıcı veya admin arayüzü kullanılmıştır.



Şekil 3.50

3.4 Belgeleme Gereklere

3.4.1 Geliştirme Sürecinin Belgelenmesi

Geliştirme sürecinin belgelendirilmesi nerede eksiklikler olduğunu genel hatlarıyla görmek, proje ekibine sonradan dahil olan kişiler için hazırlanmıştır. Ek doküman da yer almaktadır

3.4.2 Eğitim Belgeleri

Mevcut bir belgemiz bulunmamaktadır.

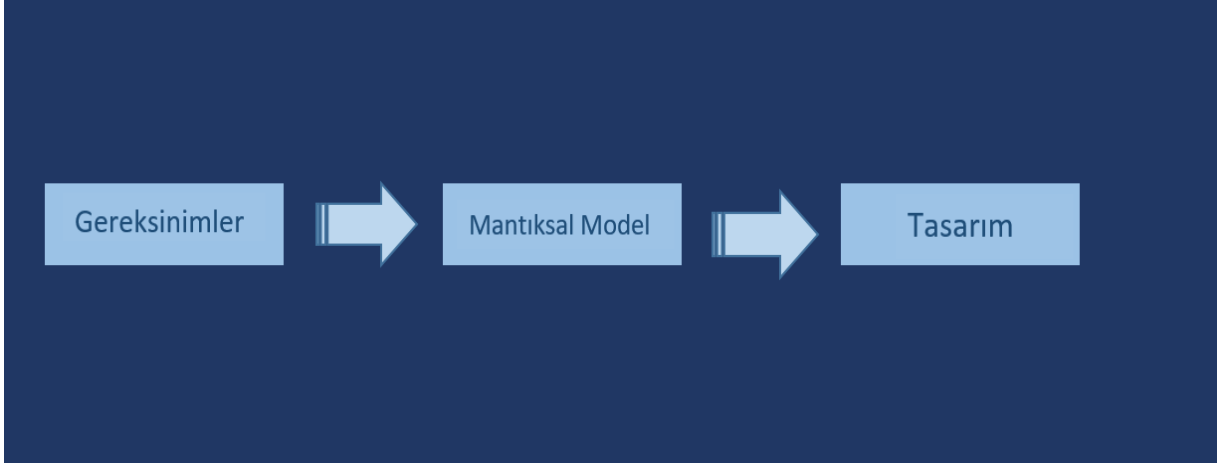
3.4.3 Kullanıcı El Kitapları

Proje sonunda rahat ve kolay kullanımdan dolayı bir eğitim semineri ve bir kullanım kitapçığı hazırlanacaktır. Ek doküman içerisinde detaylı yer almaktadır.

4.SİSTEM TASARIMI

4.1 Genel Tasarım Bilgileri

4.1.1 Genel Sistem Tanımı



Şekil 4.1 Genel Sistem Tanımı

- Gereksinimler

Belirli şehir ve semtlerde anket yapılacak gereksinimler böylelikle kullanıcı isteklerine uygun olarak belirlenecektir.

- İşlevsel Belirtilimler

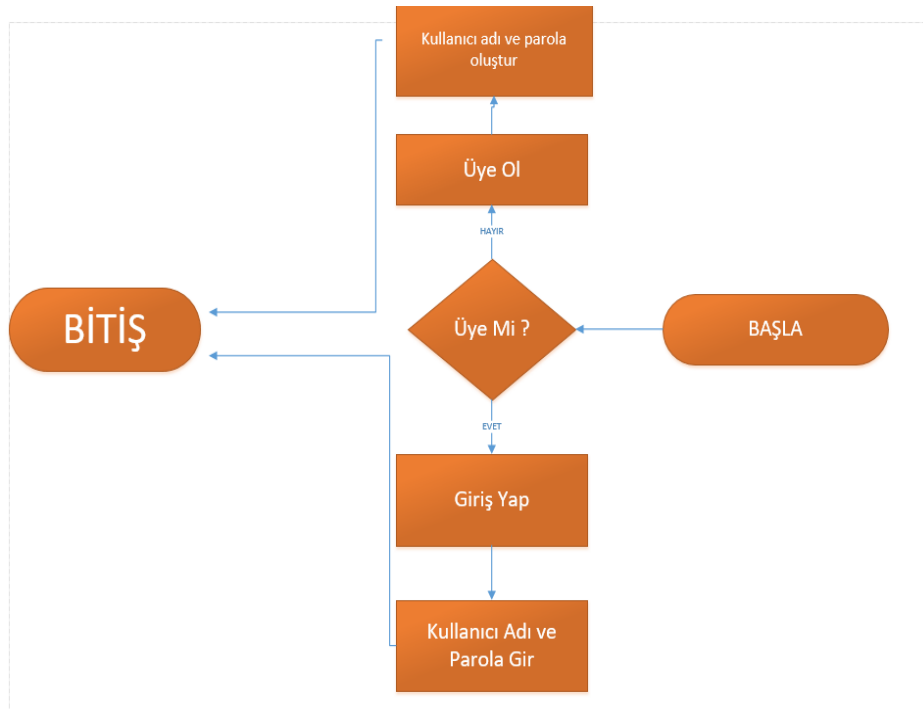
Sistem ne yapacak?

Sisteme üye olduktan sonra giriş yapılarak kalitesi düşmeden fotoğraf paylaşılacak, arkadaşlar aranabilecek, arkadaş resimleri görülebilecek.

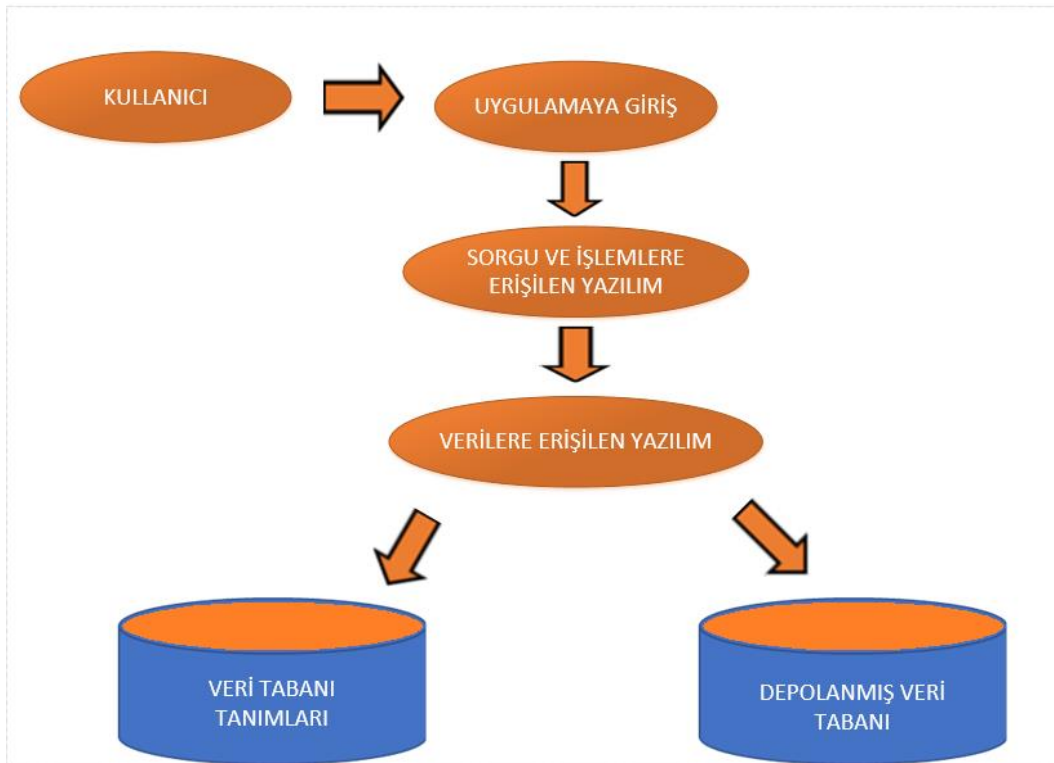
4.1.2 Varsayımlar

- İnternete bağlı olunması,
- Sisteme üye olması,
- Bilgisayar kullanmasını bilmesi.

4.1.3 Sistem Mimarisi



Şekil 4.7



Şekil 4.8

Sistemin mimarisinin akış diyagramı şeklinde verilmesinin temel nedeni sistemin işleyiş mantığının nasıl olduğu ve nasıl bir yol çizileceğinin bilinmesidir. Akış diyagramı sistemin temel mantığı hakkında bize fikir verecektir.

4.1.4 Dış Arabirimler

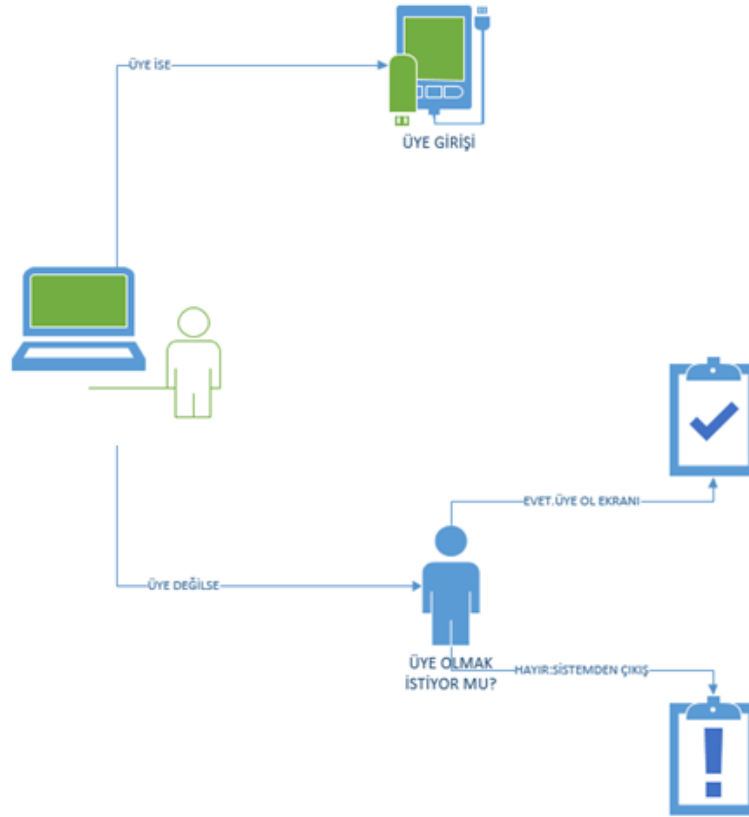
4.1.4.1 Kullanıcı Arabirimleri



Kullanıcı arabirimlerin ilk başında sistem giriş ekranı bulunacak. Ve her birimin kendine ait ekranları olacak. Kullanıcı girişi bu arabirimde amaçlanmıştır. Kullanıcı bilgilerini girerek bu sisteme giriş yapmış olacaktır.

4.1.4.2 Veri Arabirimleri

Veri arabirimlerinde sistem flash ile çalışacağından veri tabanından kayıt yazıp okurken arada XML ve api arabirimleri olacaktır. Veriler, dosyalar, evraklar bunlar veri halinde iletim sağlanacaktır. Eskiden var olan dosya ve evraklarda optik olarak okutulup arşivde tutulacaktır.



Şekil 4.10

4.1.5 Veri Modeli :

Sistem çözümlemede 3.1.2 de verilmiştir.

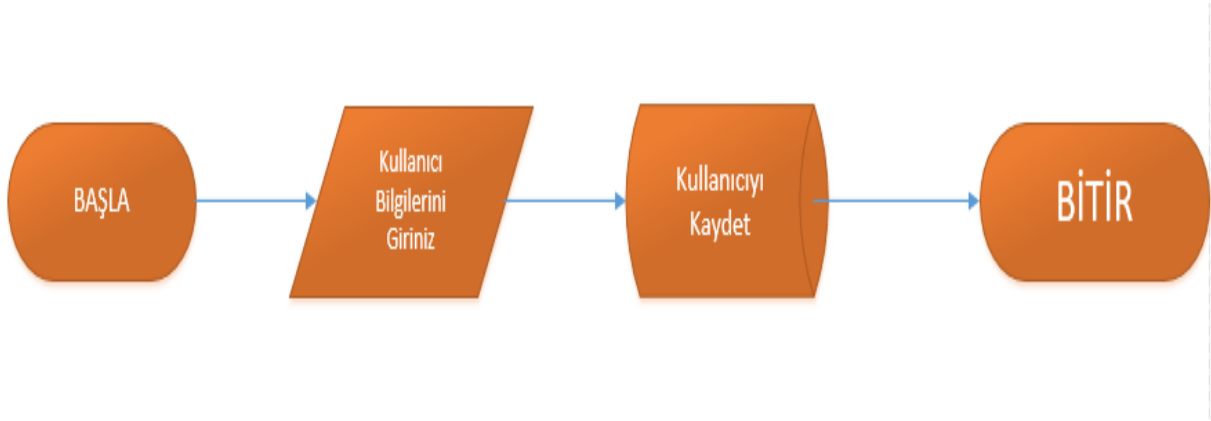
4.1.6 Testler

Projemiz arttırımsal geliştirme modeliyle tasarlandığı için birçok deneme yapılacaktır. İlk olarak Ana Ekran tasarlanmış Ana Ekran kullanımdayken Üye Ol Ekranı oluşturulmuştur. Böylelikle Üye Ekranı oluşturulurken Üye Ol Ekranı birçok kez denenmiştir. Sisteme eklenen her unsurda bir önceki faaliyetler de denenmiş, test edilmiştir.

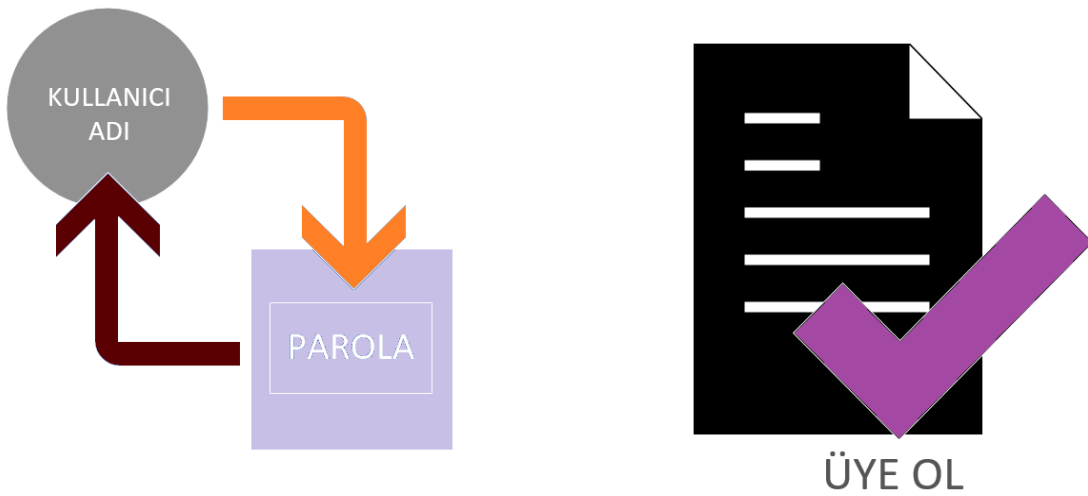
4.3 Süreç Tasarımı

4.3.1 Genel Tasarım

Genel olarak tasarımda ilk önce veri tabanı modeli oluşturuldu. Ardından giriş modülü onun ardından admin modülleri ve en sonda kullanıcı arayüzü oluşturduk.

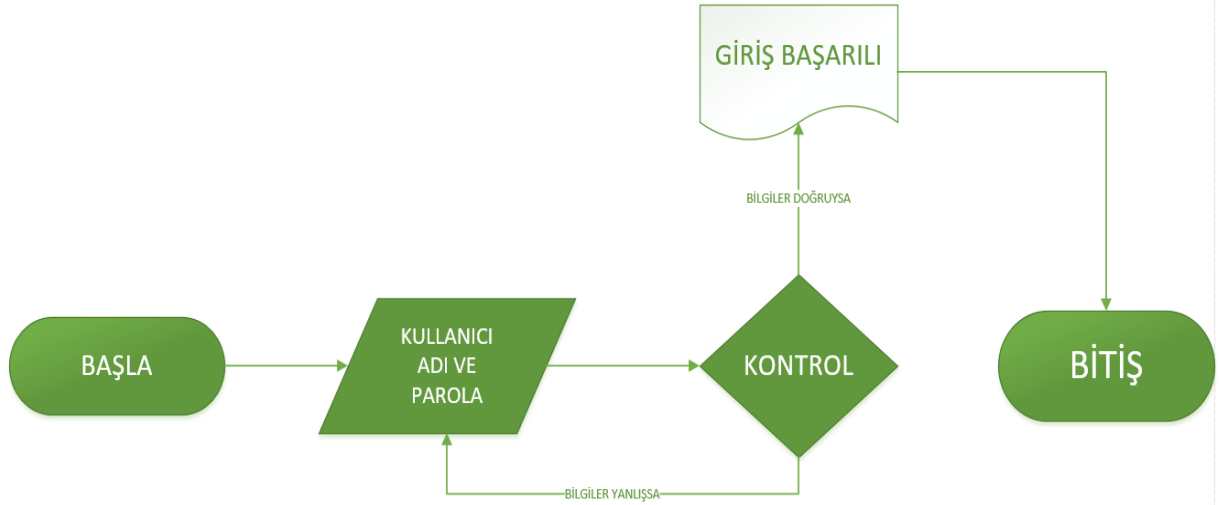


Şekil 4.16 Kullanıcı Kaydetme



Şekil 4.17

4.3.2.1.4 Modül iç Tasarımı



Şekil 4.21

4.4 Ortak Alt Sistemlerin Tasarımı

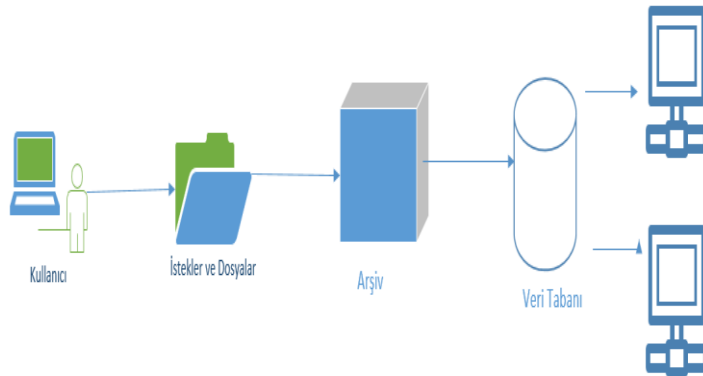
4.4.1 Ortak Alt Sistemler

Ortak alt sistem bulunmamaktadır.

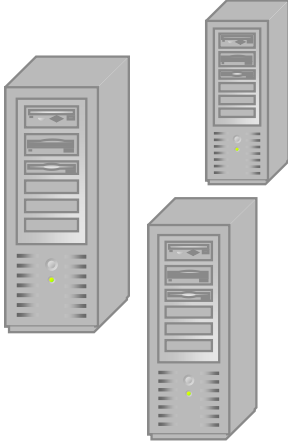
4.4.2 Modüller arası Ortak Veriler

Modüller arasında ilişkili veri modelinde olduğu üzere ortak veriler mevcuttur. Bunlar kullanıcı adı, ve parola birbirleriyle ilişkili ortak verilerdir. Yetkilendirme tablomuzdaki yetki id si ve user id de ilişkilidir. UserImages tablosundaki userId ile Users tablosu ilişkilidir.

4.4.5 Veri Dağıtım Altsistemi

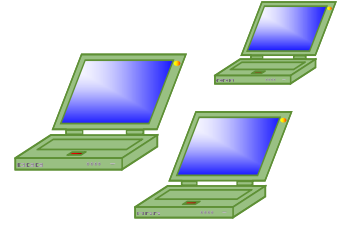


4.4.6 Yedekleme ve Arşivleme İşlemleri



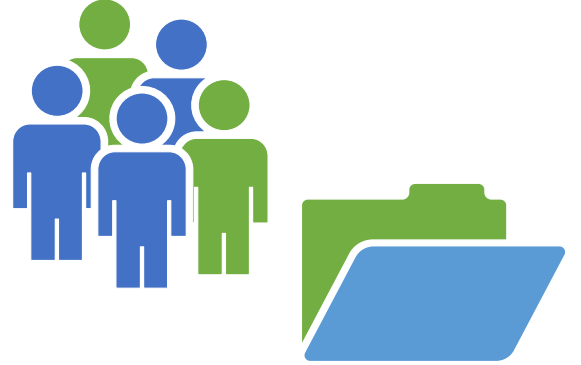
Sunucular

Depolanan verilerin, herhangi bir nedenle zarar görmesi, sistemin çalışma süreçlerinde ciddi zararlar oluşturabilir. Yaşanabilecek bir felaket durumu sonrasında, depolanan verilerin geri yüklenememesi, sistemin sağlandığı kullanıcılara veya kurumlara çok ciddi zararlar verebilir.



Diszibler

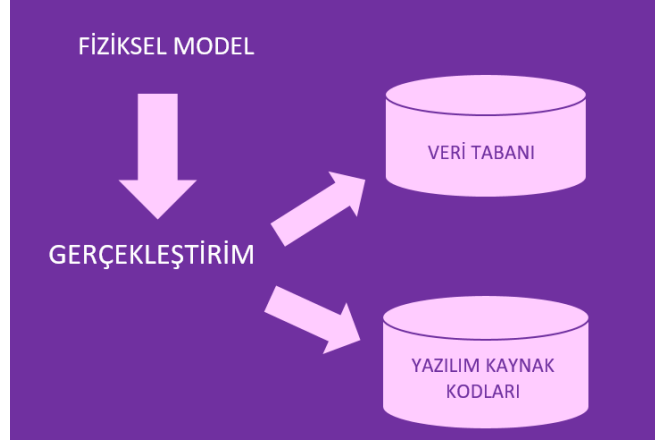
Bu nedenle sistemin çalışma süreçlerine bağlı olarak, yedekleme sistemleri kurulmalı ve yedekleme işlemleri günlük olarak takip edilmelidir. Yedekleme sistemlerinin kurulumu; yedeklenecek veri miktarı, yedekleme sıklığı, yedeklenen verinin zaman içerisinde değişme oranı ve maksimum veri kaybı gibi parametrelere bağlıdır. Sistemin birden fazla sunucusunun eş zamanlı yedekleme işlemini yapabilmesi, işletim sistemlerinin kayıt dosyalarını tam ve eş zamanlı olarak yedekleyebilmesi ve işletim sistemleri üzerinde çalışan veri tabanı uygulamasının yedeklerini sistem kapatılmadan alabilmesi gerekmektedir.



5.SİSTEM GERÇEKLEŞTİRİMİ

5.1 Giriş

Gerçekleştirim çalışması, tasarım sonucu üretilen süreç ve veri tabanının fiziksel yapısını içeren fiziksel modelin bilgisayar ortamında çalışan yazılım biçimine dönüştürülmesi çalışmalarını içerir. Yazılımın geliştirilmesi için her şeyden önce belirli bir yazılım geliştirme ortamının seçilmesi gerekmektedir.



Şekil 5.1

5.1 Yazılım Geliştirme Ortamları

Yazılım geliştirme ortamı, tasarım sonunda üretilen fiziksel modelin, bilgisayar ortamında çalıştırılabilmesi için gerekli olan:

- Programlama Dili
- Veri Tabanı Yönetim Sistemi
- Hazır Program Kitapçıkları

CASE Araçları belirlendi ve yazılım geliştirme ortamı hazırlandı.

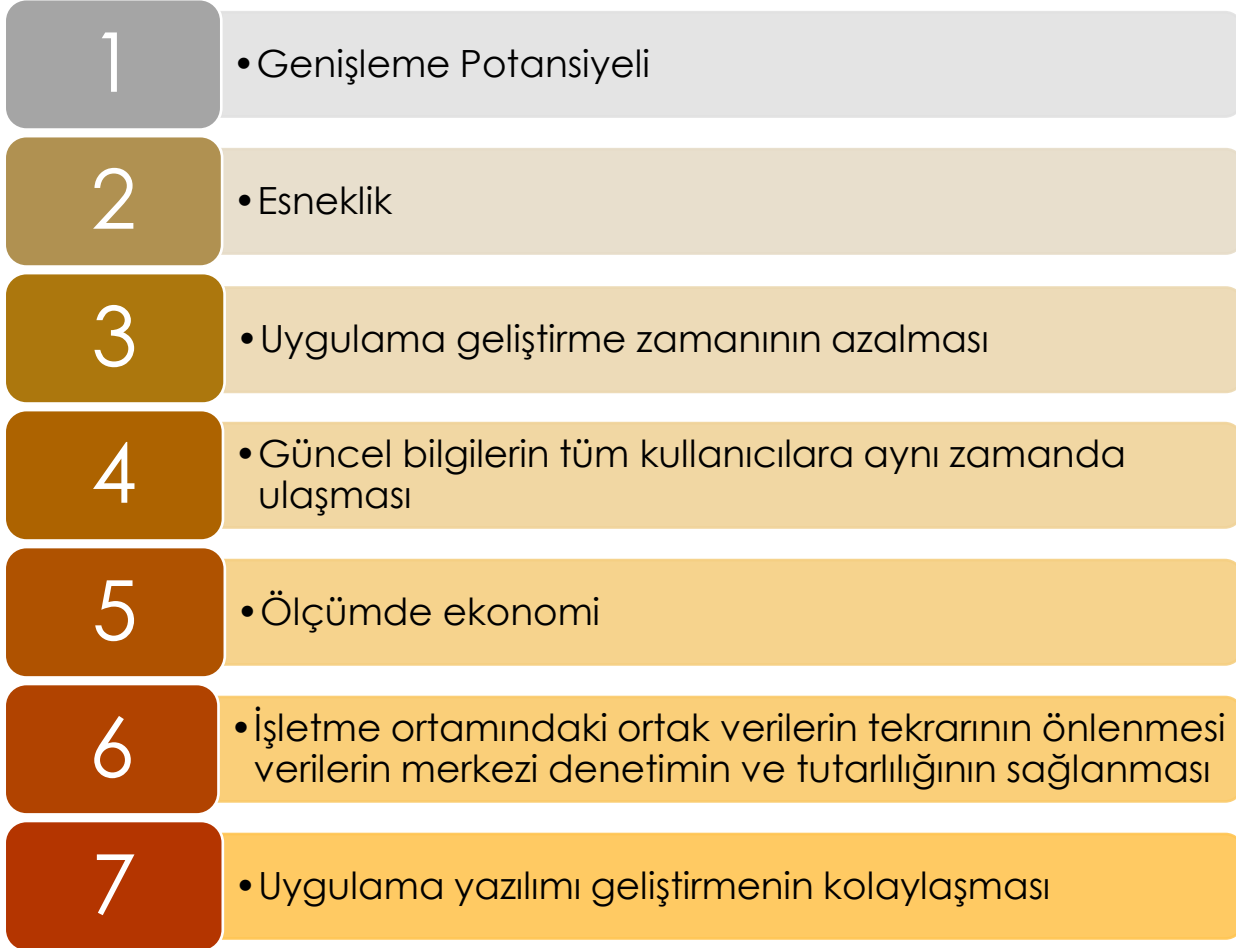
5.1.1 Programlama Dilleri

Projemizde kullandığımız programlama dilleri aşağıda listelenmiştir.

- C #
- JavaScript
- HTML & CSS

5.1.2 Veri Tabanı Yönetim Sistemleri

- Microsoft SQL Server



Şekil 5.4

1.2 Kodlama Stili

Bakım programcısıyla beraber metotlar, sınıflar, interfacerler de PascalCase; değişken isimlendirmede de camelCase isimlendirme kullanılmıştır. Business katmanının soyut sınıfları Service, somut sınıfları Manager olarak isimlendirilmiştir. DataAccess katmanı ise Dal olarak isimlendirilmiştir. Aynı zamanda İngilizce isimler tercih edilmiştir. Eğer bir gün ekip arkadaşlarımızda olası bir değişiklikte genel olarak bilinen bir şekil olduğu için tercih edilmiştir.

5.3.1 Açıklama Satırları

Yine herhangi ekip değişikliğine karşı veya yazan kişinin sonradan unutabilme durumuna karşı programda sonradan bakıldığında anlaşılmayacak kod parçalarına açıklama satırı eklenmiştir. Ama standart isimlendirme kurallarına uyulduğu için minimum yorum satırı içermektedir.

5.3.3 Anlamlı İsimlendirme

İki kelimeden oluşan değişkenler isimlendirilirken ilk harf küçük ikinci harf büyük olacak şekilde isimlendirilmiştir. Aynı zamanda veri tabanında da bulunuyorsa aynı ismin çoğulunu almasına dikkat ettik. Bunların yanı sıra Türkçe karakter de kullanılmamaya dikkat edilmiştir.

1.1 Olağan Dışı Durum Çözümleme

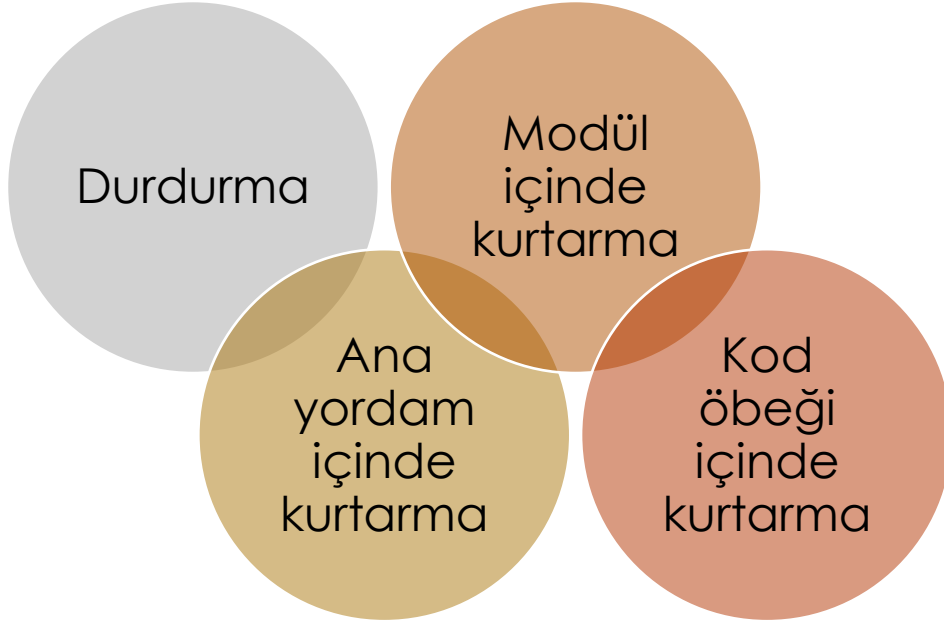
Olağan dışı durum, programın sonsuza kadar çalışması, yanlış sonuç üretmesi veya hata ekranı çıkması şeklinde olabilir.

Olağandışı Durum Tanımları

Olağan dışı durumlara karşı Exception Sınıfını kullanarak try-catch blokları oluşturduk. Böylelikle olağan dışı durumlarda programın çökmesi engellenmiş oldu. Aynı zamanda if- else lerle de kontrol edilmiştir.

Farklı Olağandışı Durum Çözümleme Yaklaşımları

Olağandışı durumlarda programın çökmesi engellemek için hata mesajlarıyla önlem aldık.



Şekil 5.12

1.5 Kod Gözden Geçirme

Programı oluşturan kodları insanlar yazdığı için hata çıkması olasıdır. Bu yüzden kodlar yazıldıktan sonra kod gözden geçirme işlemi yapılmalıdır. Yazarken fark edilmeyen veya o an düşünülmeyen herhangi bir şeyi böylelikle ekleyebilir veya düzeltebiliriz. Kod gözden geçirme sayesinde %5 e yakın düzeltme sağlanabilir.

6. DOĞRULAMA VE GEÇERLEME

6.1. Giriş

Projemizin doğrulama ve geçerlemesi aslında üretim süreci boyunca devam eder.



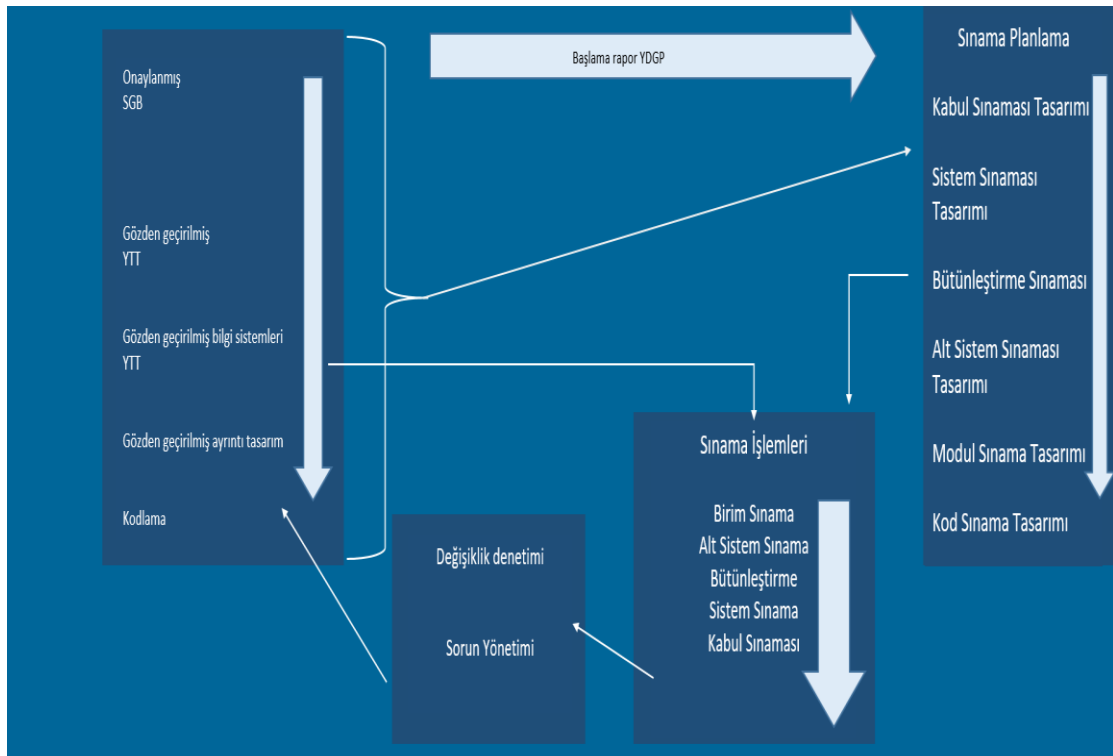
Şekil 6.2

6.2. Sınama Kavramları

Sistem Sınama: Sistemin bütün olarak sınanması yapıldı ve programın eksiksiz olduğu onaylandı.

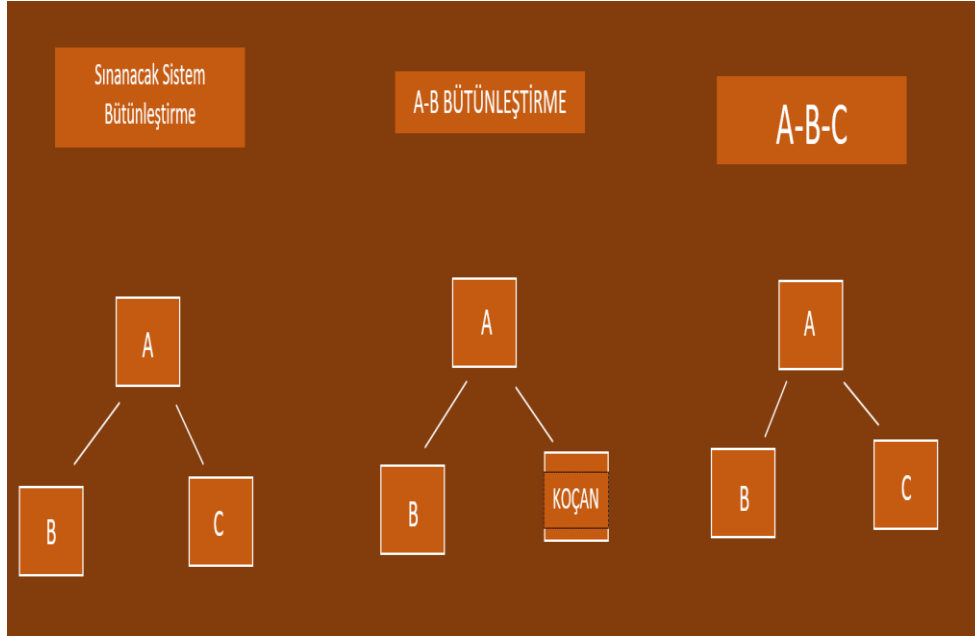
Kabul Sınama: Sistem prototipten çıkartılıp gerçek veriler girildi ve sorunsuz olduğu bir kez daha onaylandı.

6.3. Doğrulama ve Geçerleme Yaşam Döngüsü

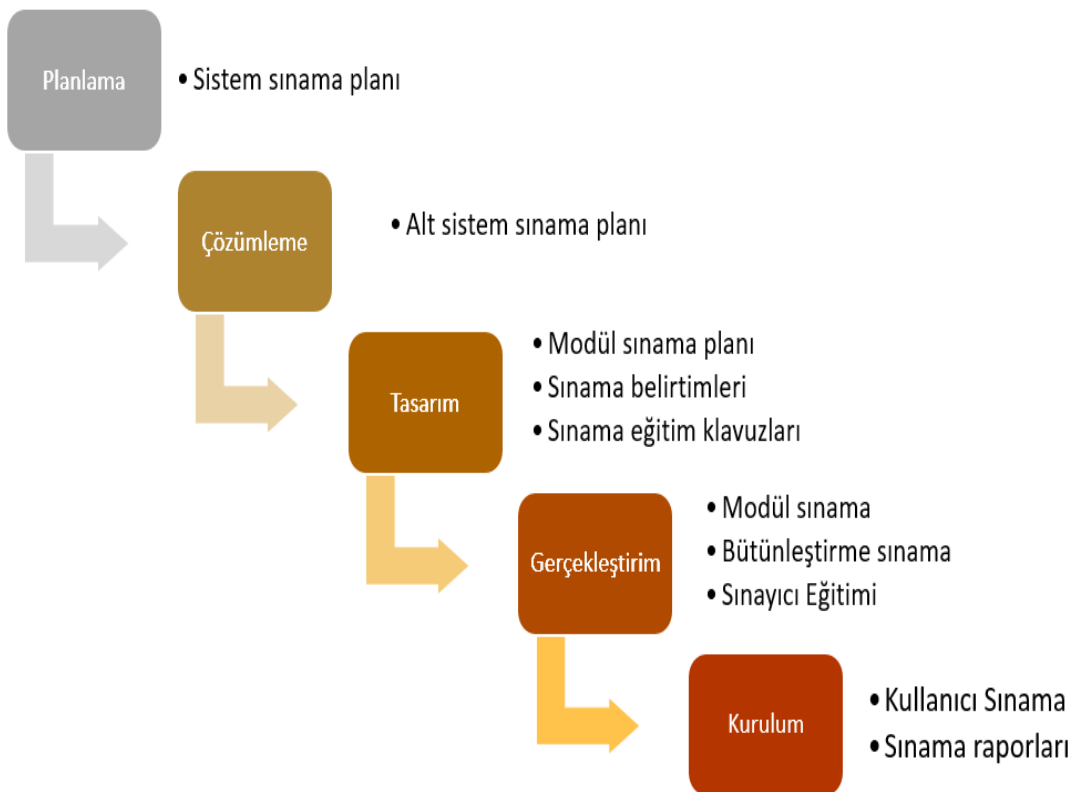


6.5 Yukarıdan Aşağı Sınama ve Bütünleştirme

Yukarıdan aşağı bütünleştirmede, önce sistemin en üst düzeylerinin sınanması ve sonra aşağıya doğru olan düzeyleri, ilgili modüllerin takılarak sınanmaları söz konusudur. En üst noktadaki bileşen, bir birim/modül/alt sistem olarak sılandıktan sonra alt düzeye geçilmelidir. Ancak bu en üstteki bileşenin tam olarak sınanması için alttaki bileşenlerle olan bağlantılarının da çalışması gerekir. Genel hatlarıyla özetlemek gerekirse şu mantıkla sistem sınaması yapıldı.



Şekil 6.8



Sınama sırasında bulunan her hata için, deęişiklik kontrol sistemine (DKS), "Yazılım Deęişiklik İsteęi" türünde bir kayıt girilir.

GERÇEKLEŐTİRİM VE DOĞRULAMA-ZAMAN DİYAGRAMI			
	7.HAFTA	8.HAFTA	9.HAFTA
GERÇEKLEŐTİRME	┐	┐	+
DOĞRULAMA			+

Őekil 6.15

7.BAKIM

7.1 Giriş

Sistemin tasarımı bittikten sonra bakıma sokulması gerekir. Bakım bölümüne ilişkin yapılan açıklamalarda IEEE 1219-1998 standardı örnek olarak alınmıştır.

7.2 Kurulum

Sistem kurulumuna değinmek gerekirse sistem yönetim güvencesinde verilecek olan serverlara yüklenecek olan sistemimizde FTP arayüzü ile dosyaları servera aktaracağız ve internet explorer olan tüm cihazlarda çalışacak.

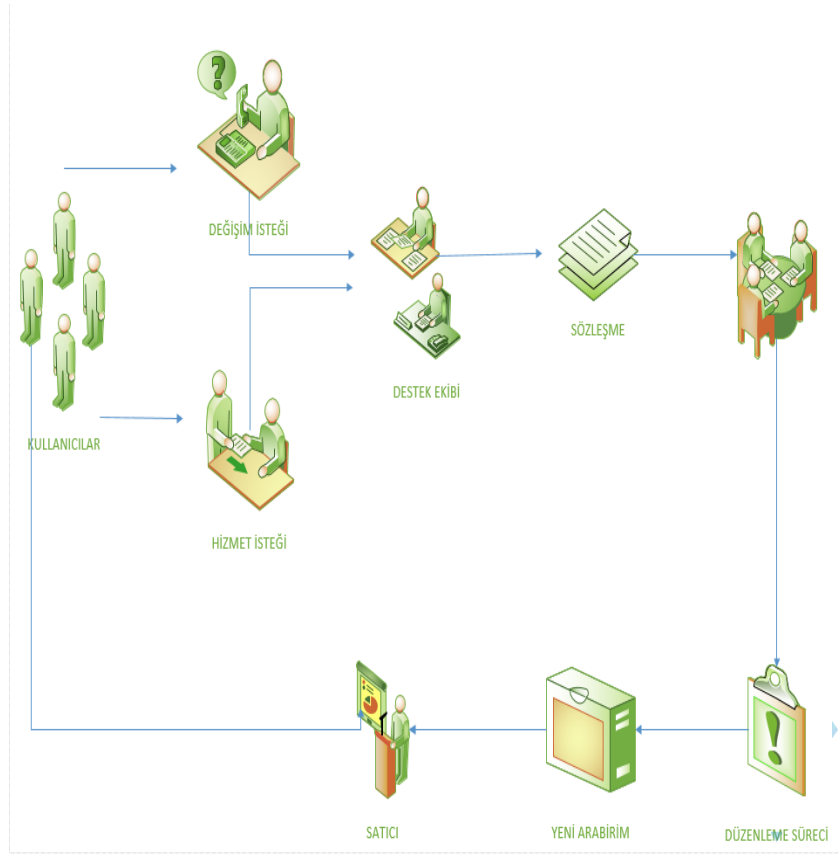


7.3 Yerinde Destek Organizasyonu

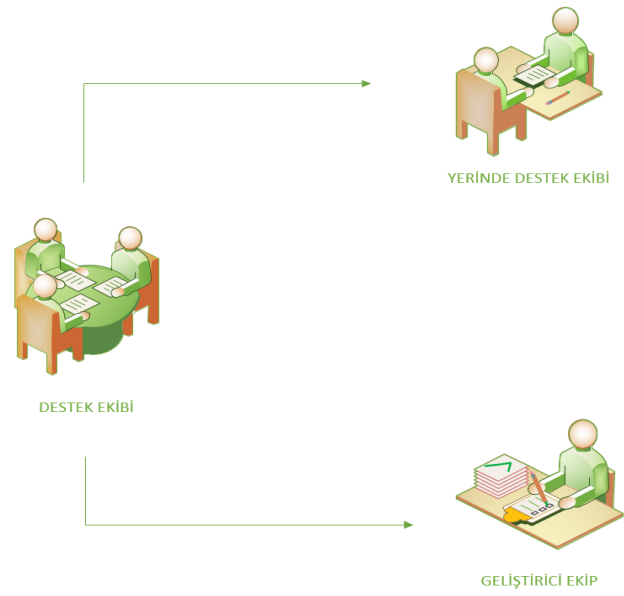


Sorun ile Ekip Sorumlusu Sümeyye Bayraktar ilgilenecektir. Sistem üzerinden tüm kontroller sürekli olarak yapılacaktır.

7.4 Yazılım Bakımı



7.4.2 Bakım Süreç Modeli



Şekil 7.4

1. Adım: Sorunu Tanımlama Süreci
2. Adım: Çözümleme Süreci
3. Adım: Tasarım Süreci
4. Adım: Gerçekleştirim Süreci
5. Adım: Sistem Sınama Süreci
6. Adım: Kabul Sınaması Süreci
7. Adım: Kurulum Süreci



BAKIM İSTEĞİNİN TEK OLDUĞUNU
BELİRLE



BAKIM İSTEĞİ



SORUN TANIMLAMA



KARAR



ÖLÇÜT

8.Sonuç

Sonuç olarak geliştirdiğimiz Bugs Web Sitesiyle;

- ✓ Kolay üyelik ve sisteme giriş,
- ✓ Kaliteli ve düşük boyutlu fotoğraf paylaşma imkanı,
- ✓ Arkadaş arama,
- ✓ Arkadaş resimleri görme,

İşlemleri sağlanmıştır.

9.KAYNAKLAR

- Yazılım kodlama kısmında çıkan hatalarda stackoverflow sitesi kullanılmıştır.
- C# için ‘Engin Demiroğ’ videolarından yardım alınmıştır.
- Döükanlama içinse IEEE standartları kaynak alınmıştır