



**Yazılım Tasarım Tanımlama Dokümanı (SDD)**

**GAN'lar ile Yüz Yaşlandırma**

**Cansu AYTEN – 171180010**

**Gamze AKSU – 171180005**

**BM496 BİLGİSAYAR PROJESİ II**

**MAYIS 2022**

# İÇİNDEKİLER

## Sayfa

İÇİNDEKİLER.....	i
KISALTMALAR .....	ii
1. KAPSAM.....	1
1.1 Tanım.....	1
1.2 Sisteme Genel Bakış .....	2
1.3 Dokümana Genel Bakış .....	2
2. İLGİLİ DOKÜMANLAR.....	3
3. SİSTEM MİMARİ TASARIMI.....	3
3.1 Görüntüden Görüntüye Çevirme.....	3
3.2 Model Oluşturma .....	4
3.2.1 Generative Adversarial Network.....	4
3.2.1.2 CycleGAN .....	6
3.2.1.3 Resnet Blokları .....	7
4. BİLEŞENLERİN DETAYLI ANLATIMI .....	7
4.1 Veri Seti .....	7
4.2 Veri İşleme .....	8
4.3 Kodlama.....	8
4.4 Genel Çalıştırma (Execution) Kavramı .....	8
5. KULLANICI ARA YÜZ TASARIMI.....	9
6. GEREKSİNİMLERİN İZLENEBİLİRLİĞİ.....	10
EKLER .....	11

**KISALTMALAR**

<b>Kısaltmalar</b>	<b>Açıklamalar</b>
<b>CNN</b>	Convolutional Neural Network
<b>CycleGAN</b>	Cycle-Consistent Adversarial Network
<b>GAN</b>	Generative Adversarial Networks
<b>GPU</b>	Graphics Processing Unit
<b>GUI</b>	Graphical User Interface
<b>SDD</b>	Software Design Description
<b>SPMP</b>	Software Project Management Plan
<b>SRS</b>	Software Requirement Specifications
<b>STD</b>	Software Test Description
<b>UAG</b>	Unsupervised Attention-Guided

## 1. KAPSAM

### 1.1 Tanım

GAN'lar ile Yüz Yaşlandırma, sürüm 1.0 (beta)

İnternetin son trendi olan efektler ile yaş fark etmeksizin neredeyse tüm insanlar yüzlerini bambaşka şekillere büründürebilmektedir. Bu akım ise hız kesmeden büyümeye devam etmektedir. Çünkü insanlar yüzlerinin özellikle gelecekte nasıl bir şekil alacağını merak etmektedir. Sosyal medya aracılığıyla başlamış olan bu akımlar ile çekilen bir fotoğraf yaşlandırılabilen, gençleştirilebilen, kedi ya da köpek gibi hayvanlara uyarlanabilmektedir. Ayrıca bir fotoğraf üzerine eklenen uzun saç, makyaj, sakal, bıyık vb. gibi değişikliklerle bir fotoğraftaki kişiye ait yüzün kadın ya da erkeğe benzemesi sağlanabilmektedir. Bunlar için özel olarak geliştirilen uygulamalar insanlar arasında popülerleşmektedir. Yüz yaşlandırmaya ya da gençleştirmeye yardımcı olacak birçok uygulama mevcuttur. Bu uygulamalar yüz yaşı dönüştürme işlemi yaparak bir yüzdeki karakteristik özellikleri koruyarak yüze yaşlanma ya da gençleşme etkisi vermeye yararlar. Bu etkinin olabildiğince doğal olması değişimin göze batmaması beklenmektedir. Bu işlem GAN yöntemi ile sağlanabilir. GAN'lar derin öğrenmeye dayalı üretken modelleme yapan bir yaklaşımdır. GAN'lar ile Yüz Yaşlandırma konulu projenin temel amacı insanlara ait yüz görüntülerinin belirli bir zaman öncesi ya da sonrasına ait durumlarını yüz yaşı dönüştürme işlemi ile elde etmektir. Bu proje eğlence alanı dışında da birçok alanda kullanılabilmektedir. Örneğin bir çocuk kaçırdıktan sonra yıllar içerisinde fiziksel açıdan birçok değişime uğrayabilir. Eldeki son fotoğraf ile kayıp çocuğu bulmak neredeyse imkânsız olabilir. Sonuç olarak eldeki son fotoğrafla bulma işlemi başarısızlıkla sonuçlanacaktır. Bu nedenle yüz yaşlandırma içeren bir program ile çocuğun son mevcut fotoğrafının kullanılmasıyla şu anki yüzüne yakın bir yüz elde edilebilir. Bu şekilde de kayıp çocuğun tespit edilebilmesi için büyük ve önemli bir adım atılmış olur. Kısaca dijital eğlenceden kayıp ilanlarına kadar kullanılabilen geniş bir kullanım alanına sahiptir.

Yüz yaşlandırma, insanların gelecekteki yüzlerinin görünümünü tahmin etme işlemi için bu proje kapsamında GAN teknolojilerinden faydalanılacaktır. CycleGAN ya da ConditionalGAN gibi GAN türlerinden yararlanılacaktır.

## 1.2 Sisteme Genel Bakış

GAN'lar ile Yüz Yaşlandırma konulu projede insanlara ait yüzleri içeren .jpg ya da .png uzantılı fotoğraflar kullanılarak yüzler üzerinde yaşlandırma çalışması yapılmaktadır. Yüz görüntülerinin kullanılmasıyla bu görüntüler üzerinde doğal bir şekilde yaşlandırma işlemi gerçekleştirilecektir. Bu proje kapsamında derin öğrenme modeli olan GAN'lar ve Python programlama dili kullanılacaktır. Projenin mevcut sürümü Beta durumunda olacağı için genel sürüm için uygun değildir.

Yüz yaşlandırma uygulamalarının genelinde olduğu gibi bu projenin de kullanıcı kitlesi yüzünün yıllar sonraki halini merak eden yaş fark etmeksizin tüm insanları kapsamaktadır. Bunun dışında kayıp ilanları konusunda arama-kurtarma çalışmalarında fayda sağlayabildiği için bu alanda da kullanılabilir. Ek olarak kaçak suçluların yakalanmasına yardımcı olmak için de kullanılabilir. Yıllar boyunca kaçak durumda olan suçlular için yüz yaşlandırma işlemi ile polislere yardımcı olunabilir.

GAN'lar ile Yüz Yaşlandırma konulu proje Gazi Üniversitesi Mühendislik Fakültesi'nin Bilgisayar Mühendisliği bölümünde 4. sınıfta öğrenim gören iki kişilik bir öğrenci grubu tarafından gerçekleştirecektir. Projenin dokümantasyon takvimi şu şekildedir:

1. 25 Mart 2022: Yazılım Proje Yönetim Planı (SPMP)
2. 22 Nisan 2022: Yazılım Gereksinim Belirtim Dokümanı (SRS)
3. 20 Mayıs 2022: Yazılım Tasarım Tanımlama Dokümanı (SDD)
4. 27 Mayıs 2022: Yazılım Test Tanımlama Dokümanı (STD)
5. 02-03 Haziran 2022: Proje Sergisi
6. 26 Haziran 2022: Proje Rapor Teslimi ve Sunumu

## 1.3 Dokümana Genel Bakış

Bu Yazılım Tasarım Dokümanı (SDD -Software Design Description) tasarımının detaylı açıklanmasıdır. Burada her bir yazılım biriminin açıklaması yapılır. SRS (Software Requirement Specification) raporundaki her bir gereksinim için nasıl bir yol izlenmesi gerektiğine dair bilgiler içermektedir. Ayrıntılı bilgilerin daha açıklayıcı ve anlaşılır olması için use case, veri akış gibi bazı diyagramlardan yararlanılmıştır.

## 2. İLGİLİ DOKÜMANLAR

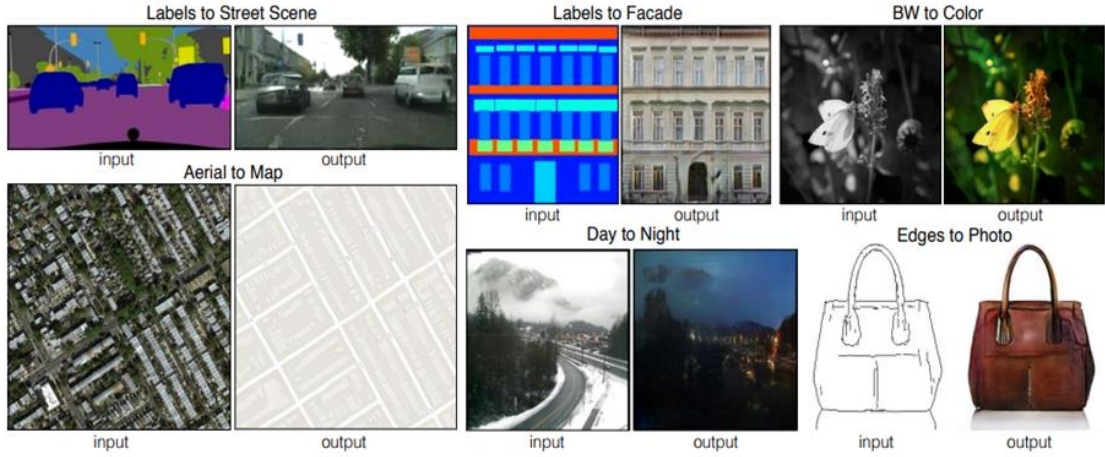
1. Buyukkinaci, M. (2017). Generative Adversarial Networks — GAN nedir ? ( Türkçe )  
<https://medium.com/@muhammedbuyukkinaci/generative-adversarial-networks-gan-nedir-t%C3%BCrk%C3%A7e-5819fe9c1fa7>
2. Brownlee, J. (2019). A Gentle Introduction to Generative Adversarial Networks (GANs)  
<https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/>
3. MathWorks. (2022). Get Started with GANs for Image-to-Image Translation.  
<https://www.mathworks.com/help/images/get-started-with-gans-for-image-to-image-translation.html>
4. Matcha, A. (2019).Implementing CycleGAN for Age Conversion.  
<https://blog.paperspace.com/use-cyclegan-age-conversion-keras-python/>
5. Zhu J Y, Park T, Isola P and Efros A A 2017 Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In: *Proceedings of the IEEE International Conference on Computer Vision: 2242–2251* <https://doi.org/10.1109/ICCV.2017.244>
6. <https://susanqq.github.io/UTKFace/>

## 3. SİSTEM MİMARİ TASARIMI

### 3.1 Görüntüden Görüntüye Çevirme

Görüntüden görüntüye dönüşüm bir görüntünün belirli değişiklikler yapılarak yeni bir sentetik versiyonunun üretilmesi işlemidir. Bu işlemde görüntü içerisindeki stiller ve özellikler gibi bazı alanlar korunarak hedef alana aktarılmaktadır. Görüntüden görüntüye dönüşüm için bir model oluşturularak bu modele büyük bir veri kümesi eğitim için girdi olarak verilmektedir.

Görüntüden görüntüye dönüşüm günden güne popülerleşmeye başlamıştır. Bu nedenle birçok uygulama alanına sahiptir. Örnek olarak bölütleme (segmentation), stil transferi (style transfer), restorasyon (restoration), nesne dönüştürme (object transfiguration), sezon aktarımı (season transfer), fotoğraf geliştirme (photo enhancement) ve poz tahmini (pose estimation) gibi bilgisayarla görme ve görüntü işleme alanları verilebilir. Gündüz çekilen görüntüler kullanılarak gece görüntülerinin elde edilmesi işlemi gündüzden geceye stil dönüştürme ya da düşük çözünürlükteki görüntülerden yüksek çözünürlüğe sahip görüntülerin elde edilmesi süper çözünürlük olarak isimlendirilebilir.



Şekil 1: GAN Uygulama Örnekleri [1]

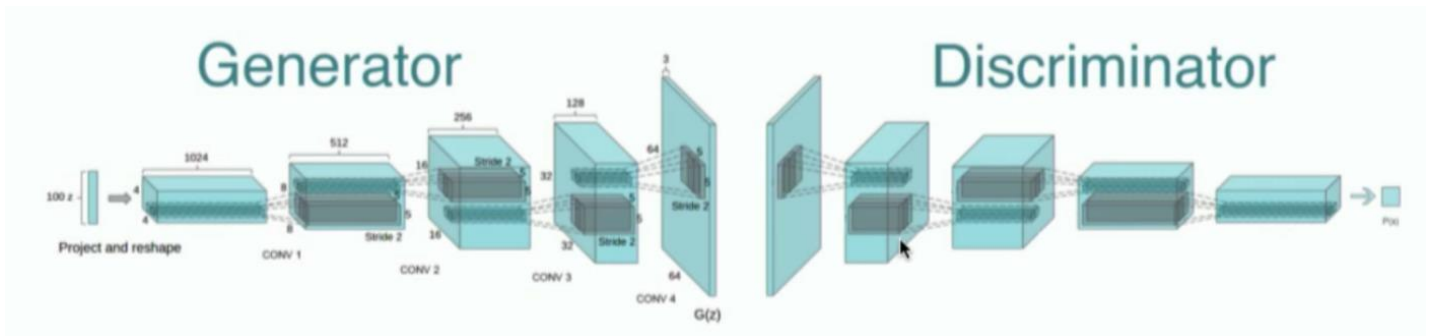
Görüntüden görüntüye dönüşüm işleminde GAN'lar kullanılmaktadır. GAN'lar kendi arasında birçok türe sahiptir ve denetimli - denetimsiz öğrenme olarak iki farklı şekilde yapılabilmektedir. Denetimli öğrenme içerisinde örnek olarak Pix2Pix, BicycleGAN, denetimsiz öğrenme için ise DiscoGAN, CycleGAN, UAG-GAN ve StarGAN gibi örnekler verilebilir.

GAN türü eğitim için kullanılacak olan verilere göre seçilmelidir. Denetimli de kaynak olarak verilen alanlar ile hedef alanlar birebir eşleşmeye sahipken denetimsiz birebir eşleşmeye sahip değildir. Denetimliye örnek olarak kaynak için kullanılan sokak görsellerinden hedef olarak elde edilmesi beklenen bölütleme haritalarını temsil eden görüntülerin oluşturulması verilebilir. Her görüntü için bir segmentation map sağlanır. Denetimsizye örnek olarak ise bahsedilen gündüz çekilen görsellerden gece görsellerinin elde edilmesi verilebilir. Çünkü gündüz ile gece görüntülerinin sahne içerikleri farklıdır [3].

### 3.2 Model Oluşturma

Model oluşturma Ek-A'da use - case diyagramı ile gösterilmiştir.

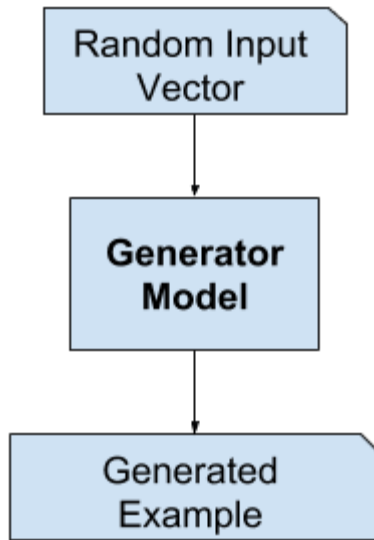
#### 3.2.1 Generative Adversarial Network



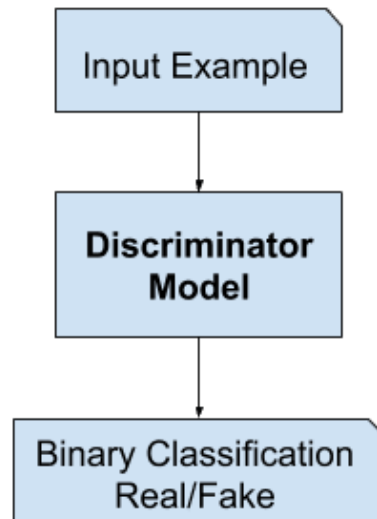
Şekil 2: Generative Adversarial Networks [1]

GAN'lar 2014 yılında bir Google araştırmacısı olan Ian Goodfellow tarafından geliştirilmiştir. GAN'lar aracılığıyla gerçek ile çok farkı bulunmayan yeni görüntüler oluşturulabilmektedir. GAN çalışırken iki ana yapıyı kullanmaktadır. Bunlar Generator (Üretici) ve Discriminator (Ayırt Edici) şeklinde isimlendirilmiştir. Generator ve Discriminator iki yapay sinir ağıdır [1].

GAN'lar genel görüntülerle çalıştığı için CNN'leri kullanmaktadır. Bunun nedeni ise CNN'lerin son zamanlarda bilgisayarlı görü alanında yapılan nesne algılama ya da yüz tanıma benzeri işlemlerde dikkate değer bir ilerleme kaydetmesidir. CNN'lerin kullanılmasıyla birlikte GAN'ların sonuçları da daha iyi bir hale gelmiş, kapasiteleri artmıştır.



Şekil 3: GAN Generator Modeli [2]

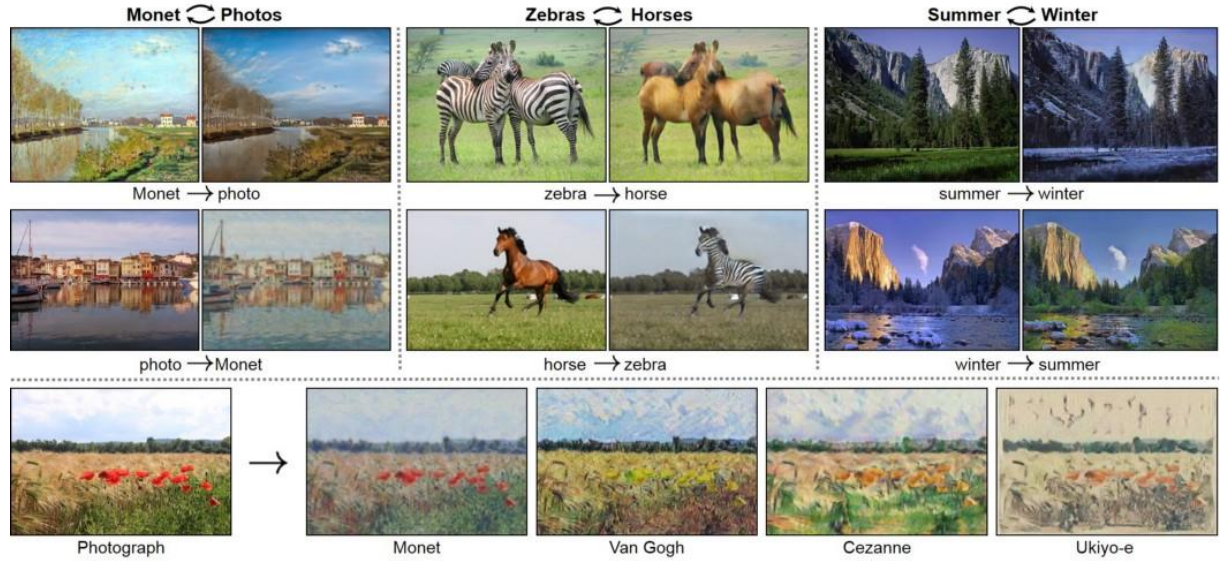


Şekil 4: GAN Discriminator Modeli [2]

Generator rastgele sayıları kullanarak bir görüntü oluşturmaya çalışmaktadır. Daha sonra Discriminator Generator'ün ürettiği görselin gerçek ya da sahte olup olduğunu tahmin etmeye çalışmaktadır. Bunu yaparken ise gerçek görüntülerden faydalanmaktadır. Generator ise görüntü üretimi yaparken asla gerçek görüntülerden yararlanmaz. Discriminator yaptığı karşılaştırma sonrası Generator'e geri bildirim göndermektedir. Generator ise bu bildirim ile kendini geliştirip güncelleyerek gerçeğe daha yakın görseller oluşturmaktadır. Discriminator artık Generator'ün ürettiği görüntüleri gerçeklerden ayırt edemez hale gelene kadar bu döngü içerisinde eğitim devam eder. Discriminator Generator'ün ürettiği görseli gerçeğinden ayırt edemediğinde model dengeye ulaşmış olur. Kısaca Generator modeli ile Discriminator modeli birbirleri ile rekabet içerisinde denilebilir. Ek -B de GAN'ların çalışması bir akış diyagramı ile gösterilmiştir.



### 3.2.1.2 CycleGAN



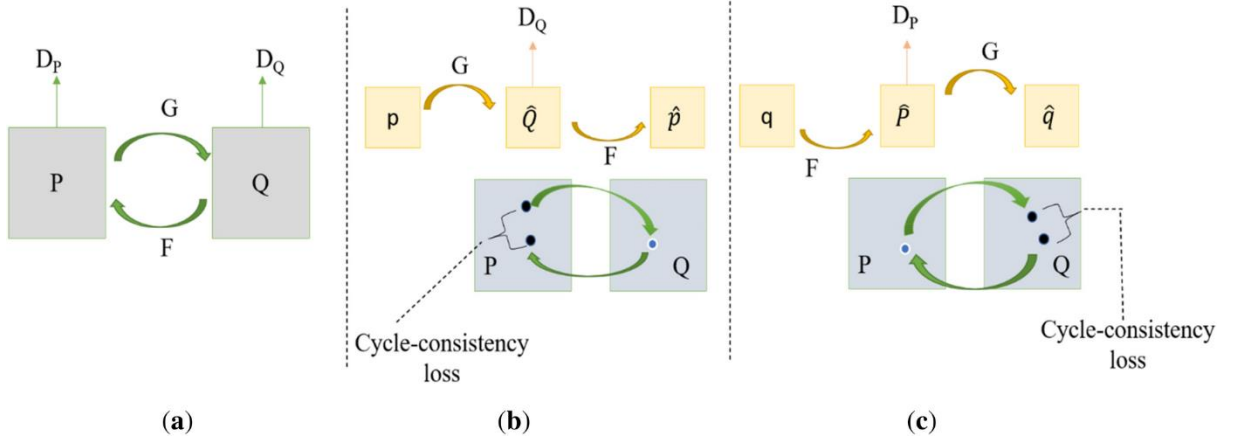
Şekil 5: CycleGAN Uygulama Örnekleri [4]

CycleGAN denetimsiz öğrenme içerisinde yer alan GAN türlerinden biridir. CycleGAN'ın farkı iki Generator ve Discriminator çiftine sahip olmasıdır. CycleGAN Zhu ve arkadaşları [5] tarafından 2017'de tanıtılmıştır.

GAN'larda eğitim için veri sayısının büyük olması gerekmektedir. Bu soruna bir çözüm olarak CycleGAN geliştirilmiştir. Örnek olarak GAN eğitiminde zebra – at dönüşümü yapılacak olursa eğitim için bir sürü etiketli at ve buna benzer etiketli zebra görüntüsüne (pair) ihtiyaç duyulmaktadır. CycleGAN da ise buna gerek yoktur. CycleGAN' tek ağ yerine iki ağ kullanarak bu sorunu çözmektedir.

Burada CycleGAN çalışma yapısını daha iyi anlatabilmek için at-zebra dönüşümü ağ çiftlerini G1-D1 ve G2-D2 olarak adlandırılarak anlatılacaktır. G1 girdi olarak bir at görüntüsü alıp onu bir zebraya dönüştürür. D1 ise G1 çıktısını yani sahte zebrayı alarak bu görüntünün gerçek bir zebra mı yoksa bir Generator tarafından üretilmiş sahte bir zebra mı olup olmadığını tespit eder. Daha sonra G1'in oluşturduğu çıktı G2 ağına gönderilir. G2 oluşturulmuş sahte olan zebra görüntüsünü tekrar at görüntüsüne dönüştürür. Daha sonra D2 de G2 çıktısının gerçek bir at görüntüsü mü yoksa sahte mi olduğunu ayırt eder. Bu şekilde bir döngü oluşturulmuş olur.

İki GAN yapısının kayıplarına ek olarak döngüsel kayıp (cycle consistency loss) da eklenir. Bu kayıp at görüntüsünün tekrar bir at görüntüsüne döndürülmesinde kullanılır. Son at görüntüsü ile orijinal at görüntüsü birbirine benzer olmalıdır.



Şekil 6: CycleGAN Çalışması

### 3.2.1.3 Resnet Blokları

CycleGAN içerisinde kullanılan Resnet Blokları görüntü girdisinin çıktıya dönüşene kadar girdinin karakteristik özelliklerinin korunmasını sağlar. Bir önceki katmanların girdi özelliklerinin sonraki katmanlara aktarımını yapar. Karakteristik özellikler aktarıldığından son çıktılar orijinal görüntüden çok fazla sapmaz. Resnet blokları olmadığında karakteristik özellikler korunmadığından çıktıların tutarlılığı bozulur. Görüntüden görüntüye dönüşüm ve yüz yaşlandırma da görüntülerin karakteristik özelliklerinin korunması en önemli amaçlardan biridir. Bu yüzden Resnet Blokları CycleGAN içinde önemlidir.

## 4. BİLEŞENLERİN DETAYLI ANLATIMI

### 4.1 Veri Seti

Proje içerisinde UTKFace veri seti kullanılmıştır. Bu veri seti içerisinde 20000'den fazla yüz görüntüsü bulunmaktadır. Bu görüntüler 0 ile 116 yaş aralığına sahip insanlara aittir. Görüntüler yaş, cinsiyet ve etnik kökene göre etiketlenmiştir. Görüntülerin boyutları 200x200'dür [6].

Her görüntü yaş\_cinsiyet\_ırk\_tarih&saat.jpg şeklinde isimlendirilmiştir.

- Yaş 0 ile 116 arasında bulunan bir tamsayıdır. Fotoğraftaki kişinin yaşını belirtir.
- Cinsiyet 0 ya da 1 şeklinde belirtilmiştir. 0 fotoğraftaki kişinin erkek olduğunu 1 ise kadın olduğunu temsil eder.
- ırk 0'dan 4'e kadar bir tamsayıdır. Bu sayı Beyaz, Siyahi, Hint ve Diğerlerini (İspanyol, Latin, Orta Doğu gibi) temsil eder.
- Tarih & saat ymmddHHMMSSFFF formatında tutulmaktadır. Görüntünün veri setine toplandığı tarih ve saati belirtmektedir.



Şekil 7: UTKFace Örnek Görüntüler [6]

## 4.2 Veri İşleme

UTKFace veri kümesinde görüntülerin adlarında görüntünün yaş bilgisi de bulunmaktadır. İlk olarak görüntülerin yaş bilgisi çekilmiştir. Daha sonra bu görüntüler yaşlarına göre sınıflandırılmıştır. 0-10 yaş aralığı için 0, 10-20 yaş aralığı için 1, 20-30 yaş aralığı 2, 30-40 yaş aralığı 3, 40-50 yaş aralığı 4, 50-60 yaş aralığı 5, 60-116 yaş aralığı 6 olacak şekilde sınıflandırılmıştır. Sonrasında veri kümesinde bulunan görüntüler RGB dönüştürülerek kullanılmıştır.

Modelde eğitim için insanlar yaşlı ve genç olarak iki sınıfa ayrılmıştır. 20-30 yaş aralığı genç olarak kabul edilmiştir. Bu sınıfta 7784 görüntü bulunmaktadır. 50-116 yaş aralığı ise yaşlı olarak kabul edilmiştir. Bu sınıfta da 4608 görüntü bulunmaktadır.

## 4.3 Kodlama

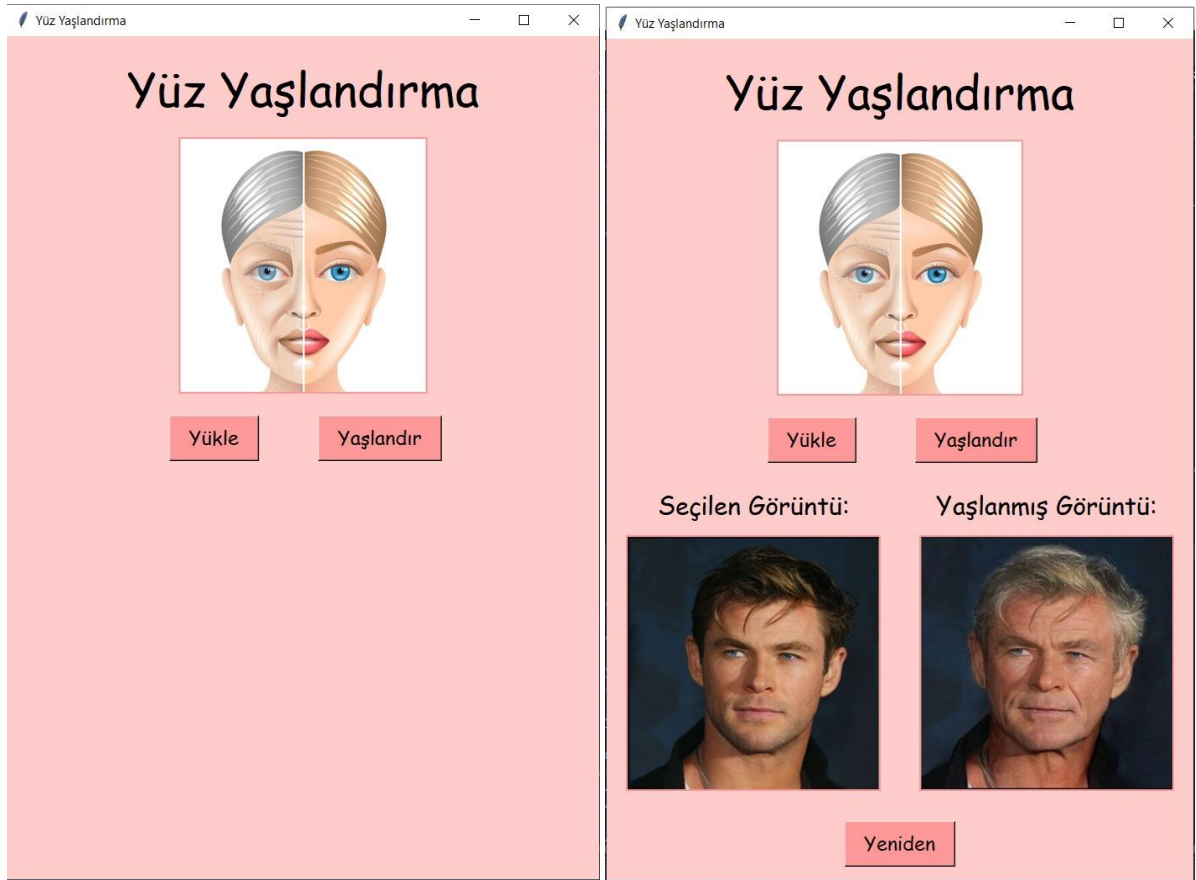
Projenin kodlama işlemleri Google Colab üzerinden gerçekleştirilmiştir. Çalışma zamanı türü olarak GPU seçilmiştir. Gerekli olan pandas, numpy, matplotlib, seaborn, tensorflow, keras, PIL kütüphaneleri kullanılmıştır. Ara yüz gerçekleştirimi işlemleri Spyder IDE'si üzerinden yapılmıştır. Ara yüz oluşturulurken tkinter kütüphanesinden yararlanılmıştır.

## 4.4 Genel Çalıştırma (Execution) Kavramı

Uygulamanın genel çalışması ile ilgili veri akış diyagramları Ek-C'de gösterilmiştir.

## 5. KULLANICI ARA YÜZ TASARIMI

Ara yüzün kullanıcı dostu olması amacıyla ara yüz içerisinde çok fazla karmaşık işleme yer verilmemiştir. Yükle butonu ile kullanıcılar bilgisayarından seçtiği bir görseli yükler. Yüklenen görsel butonların altında kullanıcıya gösterilir. Yaşlandır butonuna tıklandığında seçilen görsel yaşlandırılarak kaydedileceği dizin ve görselin adını kullanıcı tarafından değiştirilebilme yeteneğiyle kaydedilir. Ek olarak görüntü seçilen görüntünün sağında gösterilir. Yeniden butonu ile sayfa ilk haline döner ve yeni bir işlem için hazır hale gelir. Ara yüz için Use Case Diyagramı EK-D’de bulunmaktadır



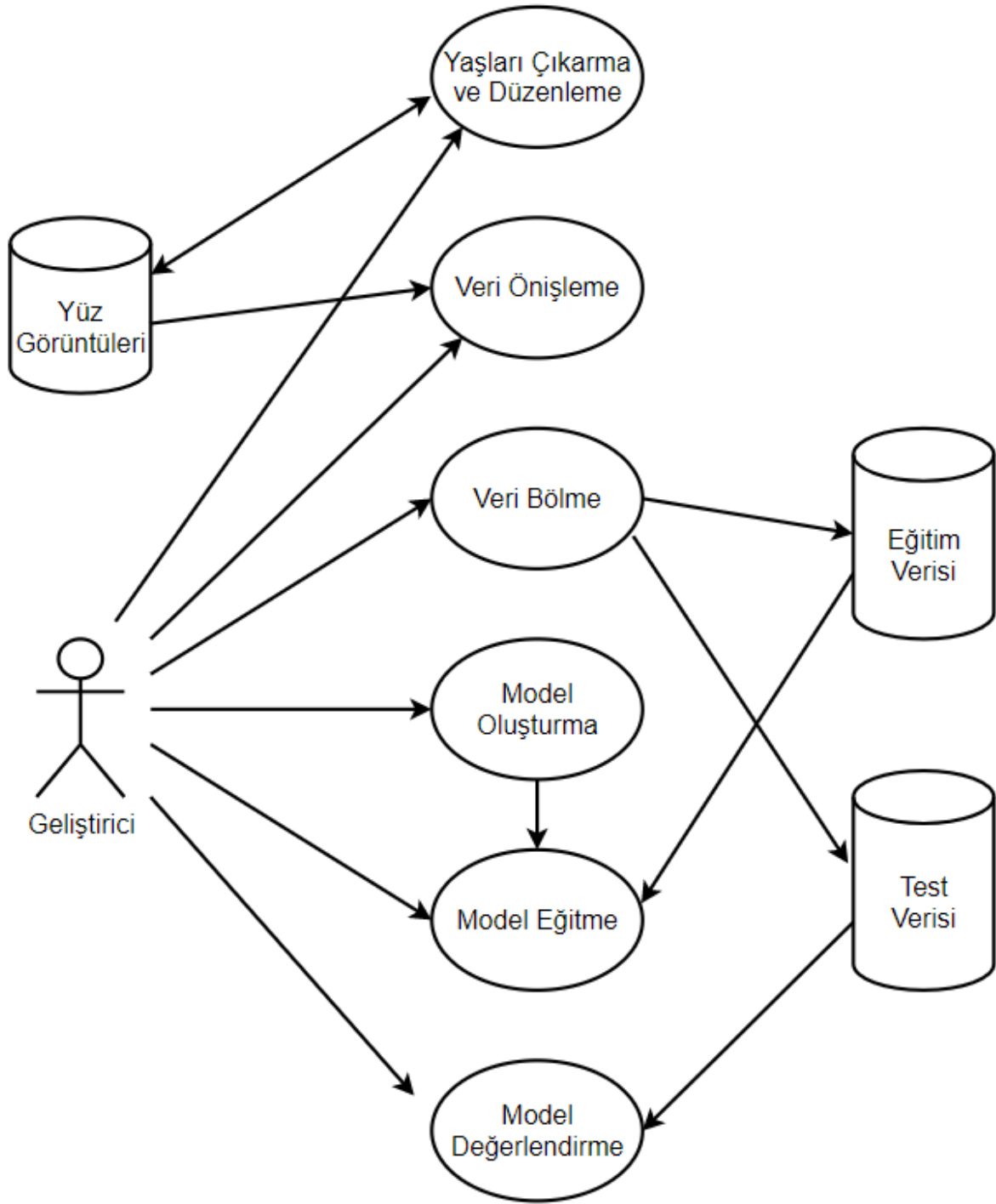
Şekil 8: Proje Ara Yüzü

## 6. GEREKSİNİMLERİN İZLENEBİLİRLİĞİ

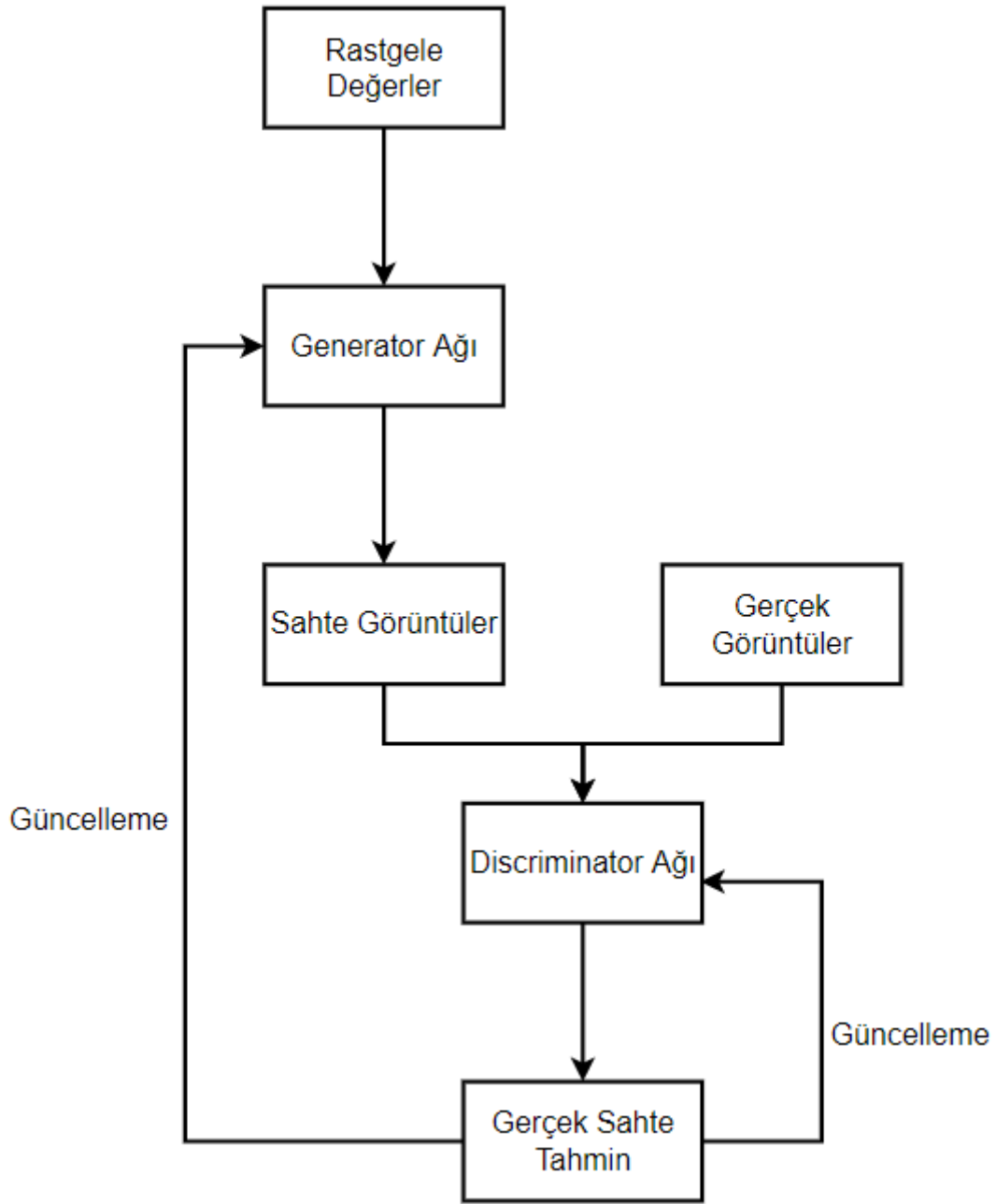
ID	SRS Başlık No	Gereksinim	SDD Başlık No
1	3.2.1.1	Görüntü bir insan yüzü içeriyor olmalıdır.	4.1
2	3.2.1.2	. Alınan görüntüler modele girmesi için gerekli boyutlara dönüştürülmelidir.	4.1
3	3.2.1.3	Uygulamaya yüklenecek görüntülerin .jpg uzantılı olması gerekmektedir.	4.1
4	3.2.2.1	Çıktı bir görüntü olmalıdır.	4.4
5	3.2.2.2	. Görüntünün karakteristik özellikleri değiştirilmeden yaşlandırma yapılmalıdır.	3.2.1.3
6	3.5.1	Yaş gruplarının yaklaşık düzgün bir dağılıma sahip olmalıdır.	4.2
7	3.5.2	Veri sayısı yeterli olmalıdır.	4.2
8	3.5.3	Görüntülerin kalitesi eğitim için yeterli olmalıdır.	4.2
9	3.9.1	Colab servisinin çalışması için bir tarayıcı açılmalıdır.	4.3
10	3.10.1.1 3.10.3.1	Görüntü işleme yapılacağı için Colab üzerinden Tesla K80 GPU kullanılmalıdır	4.3
11	3.10.3.2	Verilere dair işlemler yapılırken Pandas, Numpy, Matplotlib, Seaborn, Tensorflow Keras ve PIL gibi kütüphaneler kullanılacaktır.	4.3
12	3.12.1	Uygulama Python programlama dili kullanılarak geliştirilmelidir.	4.3
13	3.14.1 3.14.2 3.14.3 3.14.4	Eğitimle ilgili gereksinimler	4.2



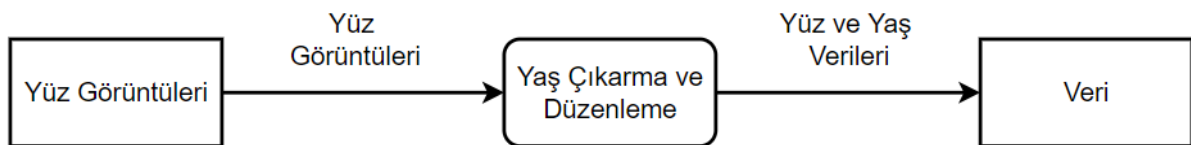
**EKLER**  
**EK-A**



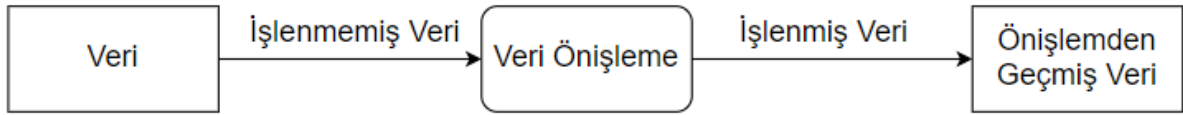
Diyagram 1: Model Oluşturma Use Case Diyagramı

**EK- B**

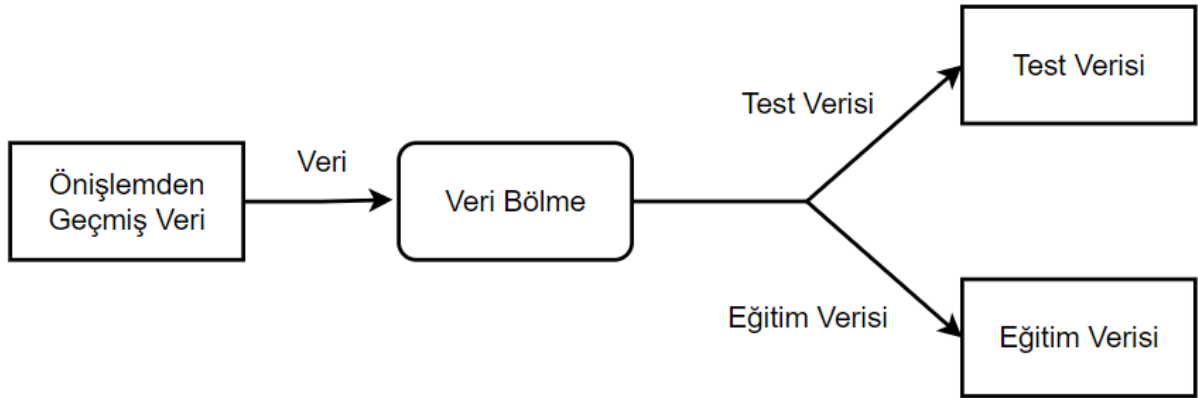
Diyagram 2: GAN modelinin çalışmasının Akış Diyagramı

**EK-C**

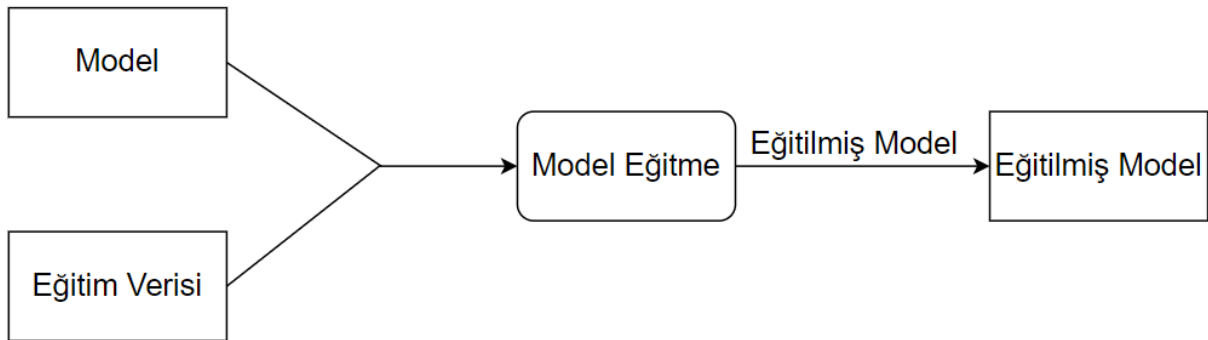
Diyagram 3: Yaş Çıkarma ve Düzenleme Veri Akış Diyagramı



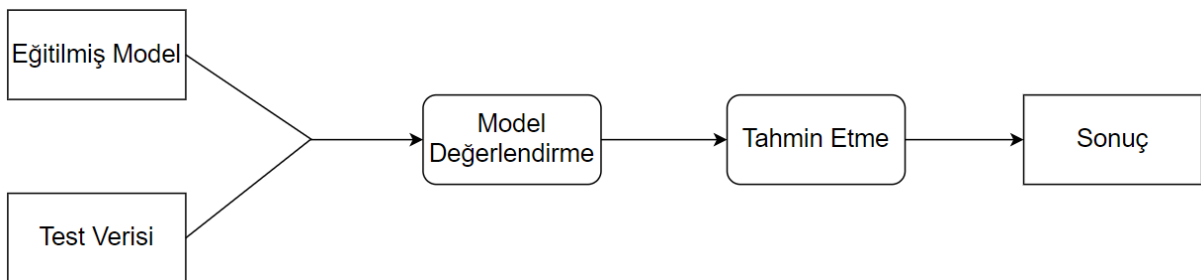
Diyagram 4: Veri Önışleme Veri Akış Diyagramı



Diyagram 5: Veri Bölme İşlemi Veri Akış Diyagramı

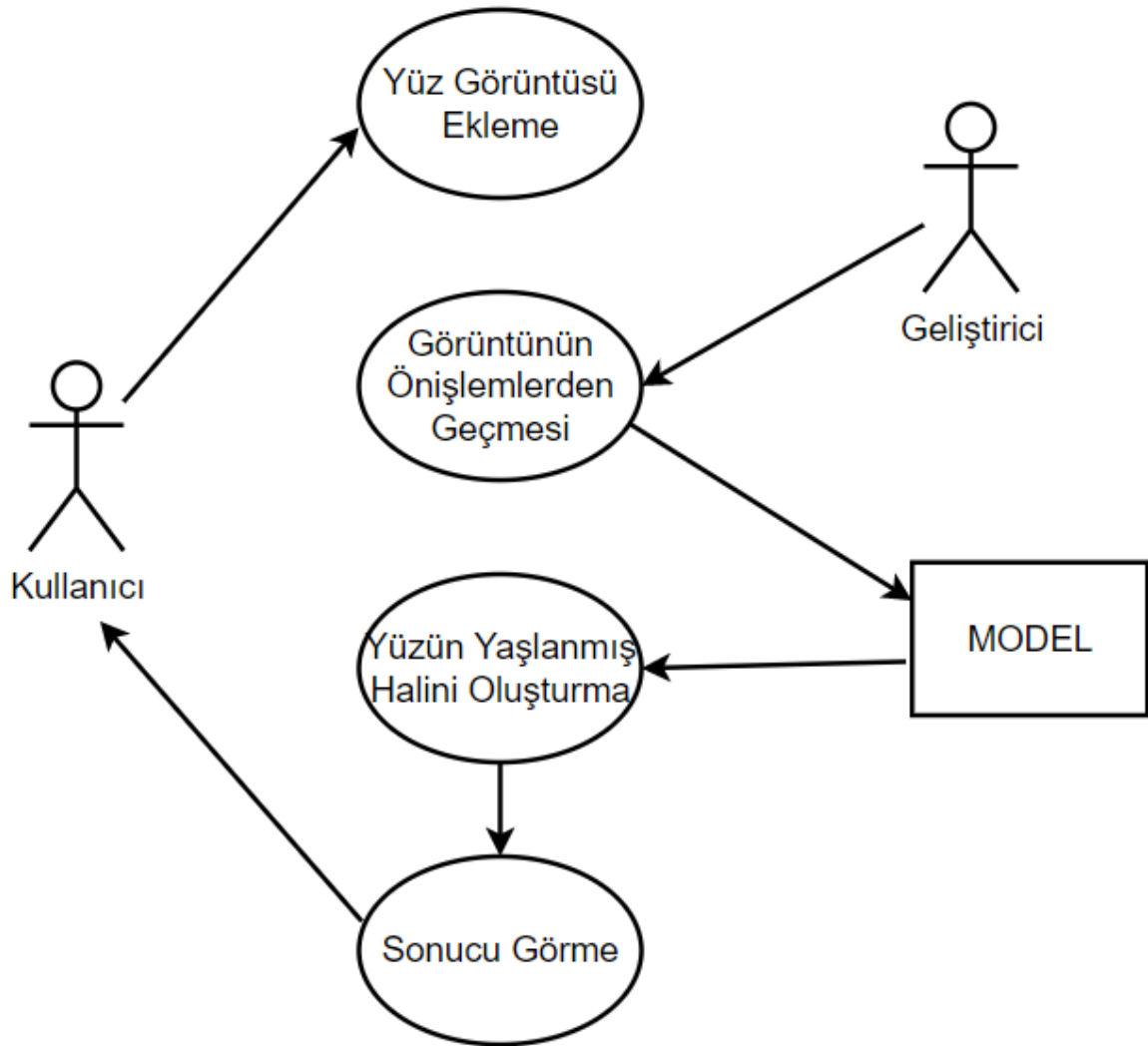


Diyagram 6: Model Eğitim Veri Akış Diyagramı



Diyagram 7: Model Değerlendirme Veri Akış Diyagramı



**EK- D**

Diyagram 8: Ara Yüz için Use Case Diyagramı