



Oyunlarda procedural generation oyuncu memnuniyetini nasıl etkiler

Author: Samet Zengin⁽¹⁾ Mail: smt.zngn02@gmail.com

Abstract

Procedural generation (PCG), oyun dünyalarının, seviyelerinin ve içeriklerinin algoritmik olarak oluşturulmasını sağlayarak oyuncu deneyimini kişiselleştiren ve tekrar oynanabilirliği artıran güçlü bir tekniktir. Bu çalışma, PCG'nin oyuncu memnuniyeti üzerindeki etkilerini araştırmayı amaçlamaktadır. PCG teknikleri kullanılarak oluşturulmuş haritalar ile manuel tasarılanmış haritaların oyuncular üzerindeki etkileri karşılaştırılmıştır. Gönüllü 50 oyuncudan toplanan veriler ele alınarak, PCG teknikleri kullanılarak oluşturulan haritaların oyuncu memnuniyeti ve oyun süresindeki olumlu etkileri istatistiksel olarak analiz edilmiştir. Çalışmanın sonuçları, PCG tekniklerinin oyuncuların ilgisini arturdığı, tekrar oynanabilirliği desteklediği ve oyun tasarımları süreçlerinde geliştiricilere önemli avantajlar sağladığını göstermiştir.

Keywords: Procedural Generation (PCG), oyun geliştirme, oyuncu memnuniyeti, hırcesel otomata, oyun tasarımları, tekrar oynanabilirlik

1. Introduction

Oyun endüstrisi, oyuncuların ilgisini çekmek ve değişen bekłentilerine yanıt verebilmek için sürekli bir ara yıl içerisindedir. Dijital oyunlar, eğlencenin ötesine geçerek birçok oyuncu için sosyal bir platform, meydan okuma, keşif duygusunu ve yaratıcılığı destekleyen bir araç haline gelmiştir. Oyun endüstrisinin son gelişim dönemlerinde, yıllık milyarlarca dolarlık bir gelirle eğlence endüstrisinin en büyük kollarından biri haline gelmiştir. Bu büyümeye, sadece teknolojik gelişmelere değil, aynı zamanda oyuncuların deneyimlerinin daha derin ve tatmin edici hale getirilmesine yönelik yenilikçi yaklaşımlara da dayanmaktadır.

Oyunların başarısı, büyük ölçüde oyunculara sundukları deneyimlerin kalitesine ve sundukları gorselliğin kalitesine bağlıdır. Oyuncular, tekrar oynamaya isteği uyandıran, zengin ve çeşitli oyun içerikleri ile oynlara karşı ilgilerini uzun süre sürdürmektektir. Bu açıdan PCG, oyun dünyalarının, seviyelerin ve diğer oyundaki bulunan içeriklerin algoritmalar kullanılarak otomatik olarak oluşturulmasını sağlayan bir teknoloji olarak dikkat çekmektedir. PCG, sabit ve durağan içeriklerin yanı sıra her oyun aşamasında benzersiz bir deneyim sunar. Bu durum, özellikle oyuncuların sürekli olarak yeni oyun dünyaları içerisinde keşif duygusunu artırrarak oyunda kalmasını sağlar. PCG, bu etki sayesinde tekrar oynanabilirliği artırmak için güçlü bir araç haline gelmiştir.

PCG'nin temel mantığı, oyun içeriklerini rastgele ve algoritmik olarak oluşturma yeteneğidir. Örneğin, bir açık dünya oyununda oyuncunun karşılaşacağı manzaralar, zorluklar veya ödüller farklı olabilir. Bu durum, hem oyunculara her yeni oturumda farklı bir deneyim sunar hem de oyunların uzun ömürlü olmasını sağlar [1]. Ayrıca, geliştiriciler için büyük ölçekli oyun dünyalarını manuel olarak tasarlama zorluğunu ortadan kaldırarak üretim maliyetlerini düşürür ve yaratıcılık için sınırsız olanaklar sunar [2].

Son yıllarda, PCG teknikleri yalnızca oyun dünyalarını oluşturanın ötesine geçerek, oyuncu tercihlerine uyarlanabilir içerikler oluşturmak için de kullanılmaktadır. Örneğin, roguelike türünde bir oyun olan Rogue, her oynanışta rastgele oluşturulan haritalarıyla oyuncuların uygulayabilecekleri strateji yelpazesini geliştirmekte ve ilgiyi canlı tutmaktadır [3].

Bu çalışmada, PCG'nin oyunların tekrar oynanabilirliğine olan etkisi ele alınacak ve bu teknolojinin oyuncu deneyimindeki önemi analiz edilecektir. Literatürdeki çalışmalar ve yapılan analizler sayesinde, PCG'nin oyuncu bekłentilerini nasıl karşıladığı ve oyun tasarımlına sağladığı katkı değerlendirilecektir.

2. Related Work

PCG, oyun geliştirmede dinamik ve kişiselleştirilmiş içerikler oluşturma konusunda dikkat çekmekte ve oyuncu memnuniyetini artırmada önemli bir araç olarak görülmektedir. Bu bölümde, PCG'nin oyuncu davranışlarına etkisi, uyarlanabilir içerik oluşturma ve kişiselleştirme konularındaki çalışmalara yer verilmektedir.

PCG teknolojisi, oyuncuya farklı içerikler sunarak oyuncunun sürekli aynı bölümleri görmesini engellemektedir. "**Tanagra: Reactive Planning and Constraint Solving for Mixed-Initiative Level Design**" adlı çalışma, insan yaratıcılığını algoritmik süreçlerle birleştirerek ritim ve hız tabanlı yapılarla oyun seviyelerini optimize etmektedir. Bu sistem, oyuncular için daha ilgi çekici ve dengeli içerikler sunmayı hedeflemektedir [4]. Benzer şekilde, "Procedural Level Generation in Educational Games from Natural Language Instruction" adlı çalışma, doğal dil işleme (NLP) teknikleri kullanarak, tasarımcıların sezgisel bir şekilde oyun seviyelerini oluşturmalarına olanak sağlamakta ve oyuncu odaklı içerik üretimini genişletmektedir [5].

Oyuncu davranışlarına dayalı içerik üretimi, PCG'nin önemli bir boyutunu oluşturmaktadır. "**Automatic Content Generation in the Galactic Arms Race Video Game**" çalışmada kullanılan **cgNEAT** algoritması, oyuncuların tercihlerini analiz ederek gerçek zamanlı içerik üretmektedir. Bu sistem, hem oyuncu memnuniyetini artırmak hem de tekrar oynanabilirliği desteklemek için dinamik içerik geliştirme süreçlerini optimize etmektedir [6]. Ayrıca, "**A Generic Approach to Challenge Modeling for the Procedural Creation of Video Game Levels**" adlı çalışma, dinamik zorluk eğrileri kullanarak oyun seviyelerinin eğlenceliliğini artırmayı ve oyuncuların memnuniyetini optimize etmeyi amaçlamaktadır [7]. Benzer şekilde, "**An Approach to Level Design Using Procedural Content Generation and Difficulty Curves**" adlı çalışma, genetik algoritmalar aracılığıyla oyuncu geri bildirimlerine dayalı olarak içeriklerin kişiselleştirilmesini ve zorluk dengelerinin sağlanmasıını hedeflemektedir [8].

PCG'nin bilgi transferi ve yeniden kullanım yoluyla içerik oluşturma süreçlerindeki etkisi de dikkate değerdir. "**Procedural Content Generation via Knowledge Transformation (PCG-KT)**" çalışması, oyunlar arasında bilgi transferini ele almaktır ve mevcut içeriklerin yaratıcı bir şekilde yeniden kullanımıyla yenilikçi içerikler oluşturulmasını sağlamaktadır [9]. Bu yöntem, hem içerik üretimini hızlandırmakta hem de oyuncuların bekłentilerine uygun içerikler sunmaktadır.

Oyuncu davranışlarını anlamak ve kişiselleştirilmiş içerikler oluşturmak, PCG'nin oyuncu memnuniyetine olan etkisini artırmaktadır. "**Modeling Player Personality Factors from In-Game Behavior and Affective Expression**" adlı çalışma, oyuncu davranışları ve kişilik özellikleri arasındaki ilişkiyi inceleyerek, kişilik temelli içerik üretiminin oyuncu memnuniyetini nasıl artırabileceğini göstermektedir [10]. Benzer şekilde, "**Personality and Behavior in Role-Based Online Games**" çalışması, oyuncu davranışlarının demografik değişkenler ve oyun tarzları ile ilişkisini analiz ederek, daha kişiselleştirilmiş oyun deneyimlerinin nasıl yaratılabileceğini ele almaktadır [11].

Diğer araştırmalar, prosedürel içerik üretimi süreçlerinde pekiştirmeli öğrenmenin (Reinforcement Learning) entegrasyonuna dikkat çekmektedir. "**Deep Reinforcement Learning for Procedural Content Generation of 3D Virtual Environments**" adlı çalışma, DRL ajanları kullanarak oyun seviyelerini dinamik bir şekilde oluşturmayı ve oyuncu bekłentileriyle uyumlu hale getirmeyi hedeflemektedir [9]. Son olarak, "**Search-Based Procedural Content Generation: A Taxonomy and Survey**" çalışması, içerik üretiminde optimizasyon algoritmalarını kategorize ederek, oyuncu tercihleriyle uyumlu içeriklerin nasıl geliştirilebileceğini tartızmaktadır [12].

Bu çalışmalar, prosedürel içerik üretiminin oyuncu memnuniyetini artırmadaki potansiyelini vurgulamaktadır. Dinamik algoritmalar, oyuncu davranışlarını dikkate alan yaklaşımalar ve yaratıcı framework'ler sayesinde PCG, oyun tasarımindan hem eğlenceliliği artırmakta hem de tekrar oynanabilirliği desteklemektedir.

3. Materials And Method

Bu bölümde, PCG tekniklerini uygulayabileceğimiz algoritmaların biri olan hücresel otomata (Cellular Automata) algoritmasını kullanacağız. Hücresel otomata sayesinde oyun haritalarını oluşturarak kullanıcılar memnuniyet anketi yapacakız.

Hücresel Otomata: 1970’lerde bir matematikçi olan John Conway, *The Game Of Life* adında bir simülasyon tanımlamıştır. Bu simülasyon aslında bir oyun değil, daha çok ızgaradaki hücrelerin (ya yaşayan ya da ölü) basit kurallar çerçevesinde değişimini gösterir.

3.1. Hücresel Otomata ile Prosedürel İçerik Üretilimi

Hücresel otomata, her bir hücrenin komşu hücrelerin durumuna göre değiştiği ve geliştiği bir matematiksel modeldir. Bu çalışmada, oyun dünyalarının oluşturulması için rastgele başlatılmış bir ızgara üzerinde hücresel otomata kuralları uygulanmıştır.

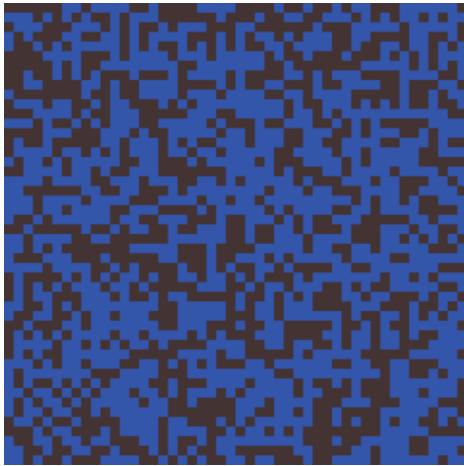
1. Başlangıç Izgarasının oluşturulması:

- Boyutları tanımlanmış bir ızgara (width x height) oluşturulmuştur.
- Her hücre, belirli bir olasılıkla (chanceToStartAlive) "dolu" veya "boş" olarak rastgele atanmıştır.
- Başlangıç durumu şu şekilde tanımlanmıştır:
- "Dolu" hücreler, oyuncu tarafından geçilemez katı yapıları temsil eder.
- "Boş" hücreler, oyuncunun hareket edebileceği alanları ifade eder.

```
Algorithm InitialiseGrid(width, height, chanceToStartAlive)
    Input: width (int), height (int), chanceToStartAlive (float)
    Output: grid (2D Boolean array)

    // Izgarayı oluştur ve başlat
    grid ← Create 2D array of size [width][height]
    for x from 0 to width-1 do
        for y from 0 to height-1 do
            randomValue ← GenerateRandomFloat(0, 1)
            if randomValue < chanceToStartAlive then
                grid[x][y] ← true // Hücre dolu
            else
                grid[x][y] ← false // Hücre boş
            end if
        end for
    end for
    return grid
End Algorithm
```

Görsel 1: Başlangıç izgarasının oluşturulma algoritması



Görsel 2: Simülasyon adımından önceki rastgele mağaramız.

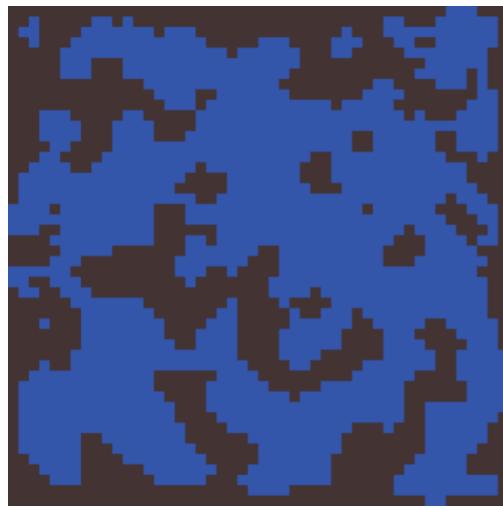
2. Simülasyon Adımları:

- Rastgele oluşturulan ızgaraya belirli kurallar uygulanmıştır:
 - Bir hücre, yeterli komşusu yoksa ölür (boş hale gelir).
 - Boş bir hücre, yeterli komşusu varsa "doğar" (dolu hale gelir).
- Simülasyon, oyuncu deneyimi için daha düzgün ve ilgi çekici haritalar oluşturana kadar belirli sayıda tekrarlanmıştır.

```
Algorithm doSimulationStep(oldGrid, deathLimit, birthLimit)
    Input: oldGrid (2D Boolean array), deathLimit (int), birthLimit (int)
    Output: newGrid (2D Boolean array)

    newGrid ← Create 2D array of size [width][height]
    for x from 0 to oldGrid.width-1 do
        for y from 0 to oldGrid.height-1 do
            aliveNeighbours ← countAliveNeighbours(oldGrid, x, y)
            if oldGrid[x][y] == true then
                if aliveNeighbours < deathLimit then
                    newGrid[x][y] ← false // Hücre ölü
                else
                    newGrid[x][y] ← true // Hücre hayatı kalır
                end if
            else
                if aliveNeighbours > birthLimit then
                    newGrid[x][y] ← true // Hücre doğar
                else
                    newGrid[x][y] ← false // Hücre ölü kalır
                end if
            end if
        end for
    end for
    return newGrid
End Algorithm
```

Görsel 3: Simülasyon adımlarının uygulanma algoritması



Görsel 4: Hücresel Otomata iki kez uygulandıktan sonra oluşan mağara

Görsel 2 ve 4 karşılaştırıldığında, başlangıçta hiçbir anlam ifade etmeyen noktalar birleşerek bir harita oluşumunu ortaya çıkartmıştır. Basit bir ön izleme ile kahverengi noktalar sınırları, mavi noktalar ise oyuncuların erişebileceğini noktaları betimlemektedir. Kullanılan teknik sayesinde, sadece harita oluşumunu değil içerikleri de yerleştirebiliriz.

3.2. Oyuncu Verilerinin Toplanması ve Analizi

Player ID	PCG	Game Time (minutes)	Satisfaction Score (1-10)
Player_1	Evet	95	9
Player_2	Hayır	45	5
Player_3	Evet	110	10
Player_4	Evet	85	8
Player_5	Hayır	30	4
Player_6	Evet	100	9
Player_7	Hayır	50	6
Player_8	Evet	90	8
Player_9	Hayır	40	5
Player_10	Evet	105	9
Player_11	Evet	115	10
Player_12	Hayır	20	3
Player_13	Evet	80	7
Player_14	Evet	95	9
Player_15	Hayır	35	4
Player_16	Evet	120	10
Player_17	Evet	90	8
Player_18	Hayır	55	5
Player_19	Evet	100	9

Player_20	Hayır	40	4
Player_21	Evet	85	7
Player_22	Evet	110	9
Player_23	Hayır	25	3
Player_24	Evet	120	10
Player_25	Hayır	50	6
Player_26	Evet	100	9
Player_27	Evet	95	8
Player_28	Hayır	35	4
Player_29	Evet	115	9
Player_30	Hayır	30	2
Player_31	Evet	110	10
Player_32	Hayır	45	5
Player_33	Evet	105	9
Player_34	Hayır	40	4
Player_35	Evet	90	8
Player_36	Hayır	55	6
Player_37	Evet	120	10
Player_38	Hayır	20	3
Player_39	Evet	115	9
Player_40	Hayır	25	3
Player_41	Evet	100	8
Player_42	Hayır	50	5
Player_43	Evet	110	9
Player_44	Hayır	35	4
Player_45	Evet	105	10
Player_46	Hayır	30	3
Player_47	Evet	95	8
Player_48	Hayır	55	6
Player_49	Evet	120	10
Player_50	Hayır	40	4

Tablo 1: Kullanıcı memnuniyet tablosu

Tablo 1'e bakıldığından, PCG'nin oyuncu memnuniyeti üzerindeki etkisini incelemek amacıyla 50 gönüllü oyuncudan veri toplanmıştır. Oyuncular, dinamik olarak oluşturulan ve sabit tasarlanmış oyun haritalarını oynayarak değerlendirmelerde bulunmuşlardır. Veriler, oyun süresi ve memnuniyet puanı gibi kritik metriklere dayanmaktadır.

Oyuncuların değerlendirmeleri aşağıdaki üç temel veri kategorisine ayrılarak incelenmiştir:

- 1. Harita Türü:** Haritaların PCG teknikleri kullanılarak veya manuel olarak tasarlanmış olduğu belirtilmiştir
- 2. Oyun Süresi (dk):** Oyuncuların bir gün içerisinde belirli bir haritada geçirdikleri toplam süre kaydedilmiştir. PCG teknikleri ile oluşturulan haritalar, oyuncuların oyun süresini artırarak daha fazla ilgi çekmiştir.
- 3. Memnuniyet Puanı (1-10):** Oyuncular, oynadıkları haritalarla ilgili genel memnuniyetlerini bir ölçek üzerinde değerlendirmiştir. PCG teknikleri ile oluşturulan haritalar, manuel tasarımlarla oluşturulmuş haritalara kıyasla daha yüksek memnuniyet puanlarıyla sonuçlanmıştır.

References

- [1] Hendrikx, M., Meijer, S., van der Velden, J., & Iosup, A. (2013). Procedural content generation for games: A survey. *ACM Transactions on Multimedia Computing, Communications and Applications*, 9(1). <https://doi.org/10.1145/2422956.2422957>
- [2] Gao, T., Zhang, J., & Mi, Q. (2022). Procedural Generation of Game Levels and Maps: A Review. *4th International Conference on Artificial Intelligence in Information and Communication, ICAIIC 2022 - Proceedings*, 50–55. <https://doi.org/10.1109/ICAIIC54071.2022.9722624>
- [3] Risi, S., & Togelius, J. (2019). Increasing Generality in Machine Learning through Procedural Content Generation. *Nature Machine Intelligence*, 2(8), 428–436. <https://doi.org/10.1038/s42256-020-0208-z>
- [4] Smith, G., Whitehead, J., & Mateas, M. (2011). Tanagra: Reactive Planning and Constraint Solving for Mixed-Initiative Level Design. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(3), 201–215. <https://doi.org/10.1109/TCIAIG.2011.2159716>
- [5] Kumaran, V., Carpenter, D., Rowe, J., Mott, B., & Lester, J. (2024). Procedural Level Generation in Educational Games from Natural Language Instruction. *IEEE Transactions on Games*. <https://doi.org/10.1109/TG.2024.3392670>
- [6] Hastings, E. J., Guha, R. K., & Stanley, K. O. (2009). Automatic Content Generation in the Galactic Arms Race Video Game. *IEEE Transactions on Computational Intelligence and AI in Games*, 1(4), 245–263. <https://doi.org/10.1109/TCIAIG.2009.2038365>
- [7] Sorenson, N., Pasquier, P., & DiPaola, S. (2011). A Generic Approach to Challenge Modeling for the Procedural Creation of Video Game Levels. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(3), 229–244. <https://doi.org/10.1109/TCIAIG.2011.2161310>
- [8] Adrian, D. F. H., & Ana Luisa, S. G. C. (2013). An Approach to Level Design Using Procedural Content Generation and Difficulty Curves. *IEEE Conference on Computational Intelligence and Games*. <https://doi.org/10.1109/CIG.2013.6633640>
- [9] Sarkar, A., Guzdial, M., Snodgrass, S., Summerville, A., Machado, T., & Smith, G. (2024). Procedural Content Generation via Knowledge Transformation (PCG-KT). *IEEE Transactions on Games*, 16(1), 36–50. <https://doi.org/10.1109/TG.2023.3270422>
- [10] Habibi, R., Pfau, J., & El-Nasr, M. S. (2023). Modeling Player Personality Factors from In-Game Behavior and Affective Expression. <https://arxiv.org/abs/2308.14224v1>
- [11] Wang, Z., Sapienza, A., Culotta, A., & Ferrara, E. (2019). Personality and Behavior in Role-Based Online Games. *IEEE Conference on Computational Intelligence and Games*. <https://doi.org/10.1109/CIG.2019.8848027>
- [9] López, C. E., Cunningham, J., Ashour, O., & Tucker, C. S. (2020). Deep Reinforcement Learning for Procedural Content Generation of 3D Virtual Environments. *Journal of Computing and Information Science in Engineering*, 20(5). <https://doi.org/10.1115/1.4046293/1074423>
- [12] Togelius, J., Yannakakis, G. N., Stanley, K. O., & Browne, C. (2011). Search-Based Procedural Content Generation: A Taxonomy and Survey. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(3), 172–186. <https://doi.org/10.1109/TCIAIG.2011.2148116>



! Bu çıktı resmi belge değildir.
Resmi işlemlerde kullanılmaz.

**Balıkesir Su ve Kanalizasyon İdaresi Genel Müdürlüğü****Abonelik Sözleşme Feshi Başvurusu****Abonelik Bilgileri**

Abone Numarası	1400052737
Adresi	SUNULLAH MAH. HACI CAFER YAŞBEK SOK. NO:70 D:1 BANDIRMA/BALIKESİR B.K.=2558340
Sayaç Numarası	11338803

Su Abonelik Sözleşme Feshi Başvuru Bilgileri

Başvuru Numarası	48502
Başvuru Tarihi	16/11/2024 07:45:57
Başvuru Durumu	4-BAŞVURU ONAYLANDI
Başvuru Durum Açıklaması	BAŞVURU ONAYLANDI
Cep Telefonu Numaranız	5414121714
E-Posta Adresiniz	smt.zngn02@gmail.com
İrtibat Adresi	SUNULLAH MAH. HACI CAFER YAŞBEK SK. NO: 110 İÇ KAPI NO: 1 BANDIRMA / BALIKESİR
Suyun En Erken Kapatılacağı Tarih	16/11/2024
Güvence (Teminat) Bedelinin Tarafınıza İade Şekli	Kendi banka hesabımı yatırılmasını istiyorum.
Banka Hesabının Sahibi	SAMET ZENGİN
Banka Hesabınızın IBAN Numarası	TR19000670100000055945961



e-Devlet Kapısı'nın kurulması ve yönetilmesi görevi T.C. Cumhurbaşkanlığı Dijital Dönüşüm Ofisi Başkanlığı tarafından yürütülmekte olup, sistemin geliştirilmesi ve işletilmesi Türksat A.Ş.

<u>İş Yerinde Çalışılan Birim:</u> Marketing ve Game Developer	Tarih: 02/08/2024
--	-------------------

Yapılan İş: Mücevher patlatma oyunu geliştirmeye başladım ve temel sahneyi oluşturdum.

Bugün stajimin ilk gününde evden çalışmaya başladım. Mücevher patlatma oyunu üzerinde çalıştım ve oyunun temel sahnesini oluşturdum. Oyunum, Candy Crush gibi olacak, yani yan yana 3 veya 4 mücevher geldiğinde patlayacak. Temel işleyishi bu şekilde olacak.

Unity'de bir oyun tahtası (board) tasarladım. Bu tahtada mücevherler yer alacak.

Mücevherlerin konumlanacağı hücreler için bir grid sistemi hazırladım.

DuzenleFNC() adında bir fonksiyon yazdım. Bu fonksiyon, tahtaya mücevherleri yerleştiriyor: Gridin her hücresına gidip rastgele bir mücevher yerleştiriyorum.

Eğer yerleştirdiğim mücevher daha önceki mücevherlerle eşleşiyorsa, yeni bir tane seçiyorum. Bunu yapmak için bir while döngüsü kullandım. Bu sayede oyunun başında üçlü mücevherler olusmaz.

Sonunda, uygun pozisyonda mücevherleri oluşturdum.

Kısaca gün boyu temel işler yaptım. Bugün ek olarak hangi projeye başlayacağımı araştırdım ve gün içerisinde fikir araştırmalarında bulundum



<u>Stajyerin imzası</u>	<u>İş yeri yetkilisinin imzası</u>	<u>Sayfa no</u>
		1

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 05/08/2024

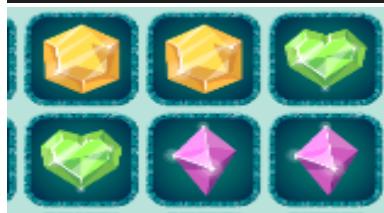
Yapılan iş: Mücevherlerin rastgele oyun tahtasına yerleştirilmesini ve pozisyonlarının iki boyutlu bir dizide saklamasını sağladım.

Bugün, mücevher patlatma oyunum için farklı renklerde ve şekillerde mücevher sprite'larını oluşturdum. Oluşturduğum bu mücevherleri rastgele oyun tahtası üzerine yerleştirdim ve her bir mücevheri iki boyutlu bir diziye aktararak pozisyonlarını kaydettim.

Bunun için MucevherOlustur adında bir fonksiyon geliştirdim. Bu fonksiyon, pozisyon bilgisi ve oluşturulacak mücevher tipini alarak mücevherin oyun tahtasında doğru bir şekilde yer olmasını sağlıyor. Eğer rastgele bir bomba çıkma şansı tutarsa, o mücevher bomba olarak değişiyor. Ayrıca, mücevherlerin yukarıdan gelmesini sağlamak için pozisyon ayarlamaları yaptım.

Yazdığım kod, her mücevherin pozisyonunu tumMucevherler adlı iki boyutlu dizide saklamayı mümkün kılıyor. Bu adımlarla oyunun temel mekaniği için önemli bir ilerlemek kaydettim.

```
c.boyutlu  
void MucevherOlustur(Vector2Int pos, Mucevher olusacakMucevher) // normalde Vector2'di ama tumMucevherler[pos.x, pos.y] = olusacakMucevher;  
{  
    if(Random.Range(0f, 100f)<bombaCikmaSansi)  
    {  
        olusacakMucevher = bomba;  
    }  
  
    Mucevher mucevher = Instantiate(ulusacakMucevher, new Vector3(pos.x, pos.y+yukseklik, 0f), Quaternion.identity); /  
    mucevher.transform.parent = this.transform;  
    mucevher.name = "Mucevher - " + pos.x + ", " + pos.y;  
    tumMucevherler[pos.x, pos.y] = mucevher;  
    mucevher.MucevheriDuzenle(pos, this); // Mucevher script'den nesne ile pozisyonu eşleştirme kısmını alıyoruz.  
}
```



Stajyerin imzası

İş yeri yetkilisinin imzası

Sayfa no

<u>İş Yerinde Çalışılan Birim:</u> Marketing ve Game Developer	Tarih: 06/08/2024
<u>Yapılan İş:</u> Mücevherlerin sürüklelenmesi ve hareket ettirilmesi işlemleri kodlanarak, üçlü eşleşmelerin tespiti ve listeye alınması üzerinde çalıştım	
<p>Bugün mücevherlerin sürüklelenmesi ve hareket ettirilmesi üzerine yoğunlaştım. İlk olarak, oyuncuların mücevherleri sürükleyerek hareket ettirebilmesi için dokunma ve sürükleme işlevlerini kodladım. Mücevherin sürükleneceğini hesaplayarak hangi yöne doğru hareket edeceğini belirledim. Ayrıca, sürüklenen mücevherin, grid üzerindeki hücrelerle doğru bir şekilde yer değiştirmesini sağladım.</p> <p>Projenin bu aşamasında, oyun tahtasında mücevher eşleşmelerini kontrol eden bir fonksiyon oluşturdum. Bu fonksiyon, yatay ve dikey sıralardaki üçlü mücevher eşleşmelerini kontrol ediyor ve eşleşen mücevherleri bir listeye alıyor. Bunun için çift döngü (for loop) kullanarak her bir mücevheri kontrol ettim. Kodda önce mücevherin sol ve sağındaki hücreleri (x koordinatında) kontrol ederek, aynı tipteki mücevherlerin yan yana gelip gelmediğini test ettim. Eğer eşleşme varsa, bu mücevherleri bir listeye ekleyip “eşleşti” olarak işaretledim. Aynı işlemi yukarı ve aşağı (y koordinatında) yöndeki mücevherler için de yaptım. Bu sayede hem yatay hem dikey üçlü eşleşmeler tespit edilmiş oldu.</p> <p>Kod içerisindeki şu satırla mücevher eşleşmelerini buluyorum:</p> <pre> if(solMucevher.tipi == gecerliMucevher.tipi && sagMucevher.tipi == gecerliMucevher.tipi) { gecerliMucevher.eslestimi = true; solMucevher.eslestimi = true; sagMucevher.eslestimi = true; BulunanMucevherlerListe.Add(gecerliMucevher); BulunanMucevherlerListe.Add(solMucevher); BulunanMucevherlerListe.Add(sagMucevher); } </pre>	
<u>Stajyerin imzası</u>	<u>İş yeri yetkilisinin imzası</u>
	3

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 06/08/2024

Yapılan iş: Mücevherlerin sürüklelenmesi ve hareket ettirilmesi işlemleri kodlanarak, üçlü eşleşmelerin tespiti ve listeye alınması üzerinde çalıştım

Bu fonksiyon mücevherlerin eşleşme olup olmadığını kontrol ediyor ve eğer varsa, onları bir listeye ekliyor. Aynı işlemi dikey eşleşmeler için de uyguladım. Daha sonra, bu listeyi Distinct() fonksiyonu ile filtreleyerek aynı mücevherin birden fazla kez eklenmesini önledim ve tespit edilen tüm eşleşmeler silinmeye hazır hale getirdi.

Ek olarak, BulunanMucevherlerListe adında bir liste oluşturarak, üçlü eşleşme gerçekleştiğinde bu mücevherleri listeye alıp, silinmeye uygun hale getirdim. Bu listeyi temizlemek ve yeni eşleşmeleri bulmak için her kontrol işlemi başında listeyi sıfırlıyorum. Bu sürecin sonunda, patlayan mücevherler için de bir güncelleme yaparak, patlama animasyonlarını eklemeyi planlıyorum.

```
Board board;
public List<Mucevher> BulunanMucevherlerListe = new List<Mucevher>(); // 3 tane ya
@ Unity İletisi | 0 başvuru
private void Awake()
{
    board = Object.FindObjectOfType<Board>();
}
2 başvuru
public void EslesmeleriBulFNC()
{
    BulunanMucevherlerListe.Clear();

    for (int x = 0; x < board.genislik; x++)
    {
        for (int y = 0; y < board.yukseklik; y++)
        {
            Mucevher gecerliMucevher = board.tumMucevherler[x, y];

            if(gecerliMucevher != null)
            {
                // x sırasındaki eşleşenleri kontrol ediyoruz
                if (x>0 && x<board.genislik-1)
                {
                    Mucevher solMucevher = board.tumMucevherler[x - 1, y];
                    Mucevher sagMucevher = board.tumMucevherler[x + 1, y];

                    if(solMucevher != null && sagMucevher !=null)
                    {
                        if(solMucevher.eslesme == sagMucevher.eslesme)
                        {
                            BulunanMucevherlerListe.Add(sagMucevher);
                        }
                    }
                }
            }
        }
    }
}
```

Stajyerin imzası

İş yeri yetkilisinin imzası

Sayfa no

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 07/08/2024

Yapılan iş: Mücevherlerin yatay ve dikey eşleşmelerini kontrol eden eşleşenleri bir listeye ekleyen ve performans sorunlarını önlemek için kontrol mekanizması ekleyen bir fonksiyon üzerinde çalıştım.

Bugün, mücevher patlatma oyunumda eşleşen mücevherleri tespit etmek ve yönetmek için bir fonksiyon geliştirdim. Mücevherlerin yer değiştirmesinin ardından yatay ve dikeyde üçlü ya da daha fazla mücevherin eşleşip eşleşmediğini kontrol eden bu fonksiyon, eşleşen mücevherleri bir listeye ekleyerek geçici olarak saklıyor. Listeyi, aynı mücevherin birden fazla kez eklenmesini önlemek amacıyla Distinct() fonksiyonu ile filtreledim ve tespit edilen tüm eşleşmeleri silinmeye hazır hale getirdim. Ayrıca, sonsuz döngü oluşumunu engellemek ve performans problemlerinin önüne geçmek için bir kontrol sayacı ekleyerek kodu optimize ettim.

Yazdığım fonksiyon, mücevherlerin hareketini takip edip, eşleşme olup olmadığını belirliyor. Eşleşme olmadığı durumlarda mücevherler eski pozisyonlarına geri dönüyor. Kodun hatasız çalışmasını sağlamak için "HareketiKontrolEtRouitne" adında bir Coroutine fonksiyonu ekledim, böylece mücevherlerin hareketi sırasında gereken kontroller yapılmış oluyor. Bu sayede, eşleşen mücevherler listesi doğru bir şekilde güncelleniyor ve eşleşme bulunamayan durumlarda gereksiz kontrollerden kaçınılarak performans artırılıyor.

Evden çalışarak, mücevherlerin yer değişimlerine ve eşleşmelerine dayalı oyun mekaniklerinin temelini oluşturan bu adımları başarıyla tamamladım.

```
    İbayuru
    public IEnumarator HareketiKontrolEtRouitne()
    {
        board.gecerliDurum = Board.BoardDurum.beKLiyor;

        yield return new WaitForSeconds(.5f); //Mücevherlerin geri dönerken hareketlerini kontrol ettirir.

        board.eslesmeController.EslesmeleriBulFNC();

        if (digermucevher != null)
        {
            if (!eslestimi && !digermucevher.eslestimi)
            {
                digermucevher.posIndex = posIndex; // pozisyon index'si eşlestimi ilk pozisyon'a dönsün
                posIndex = ilkPos;
            }
        }

        board.tumMucevherler[posIndex.x, posIndex.y] = this;
        board.tumMucevherler[digermucevher.posIndex.x, digermucevher.posIndex.y] = digermucevher;
    }
```

Stajyerin imzası

İş yeri yetkilisinin imzası

Sayfa no

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 08/08/2024

Yapılan İş: Üçlü eşleşmeleri engelleyen algoritma geliştirdim ve eşleşen mücevherleri yok eden, ardından boş alanlara yeni mücevherler yerleştirilen fonksiyonlar üzerinde çalıştım

Bugün mücevher patlatma oyunumda, oyun tahtasının başlangıçta üçlü eşleşmeleri engelleyecek şekilde oluşturulmasını sağlamak ve mücevherleri yok eden mekanizmayı geliştirmek üzerine çalıştım. Bu süreçte farklı algoritmalar ve fonksiyonlar geliştirdim.

Başlangıçta Üçlü Eşleşmeleri Engelleyen Algoritma: Oyun tahtası ilk defa oluşturulurken, mücevherlerin yerleştirilmesi sırasında üçlü eşleşmelerin olmasını önlemek için bir kontrol mekanizması ekledim. Eğer mücevherler yerleştirildiğinde yan yana ya da üst üste üçlü bir eşleşme oluyorsa, bu mücevherler yeniden yerleştiriliyor. Böylece oyunun başında bir avantajlı durum oluşmaması sağlanıyor. Bu kontrol, YanlisYerlestirmeleriKontrolEtFNC() fonksiyonu içerisinde gerçekleşiyor ve her bir mücevher yerleştirildiğinde devreye giriyor.

Eşleşen Mücevherleri Yok Etme Fonksiyonu: Mücevherlerin eşleşme durumunu kontrol eden ve eşleşen mücevherleri yok eden bir fonksiyon yazdım. Bu fonksiyon, EslesenMucevheriYokEt() adında bir fonksiyon olup, parametre olarak mücevherin pozisyonunu alıyor. Eşleşen mücevherleri yok ederken ayrıca ses efektleri ve patlama efektleri de çalıştırılıyor. Eğer eşleşen mücevher bir bomba ise, özel bir patlama sesi ve efekt devreye giriyor.

```
1 basvuru
void EslesenMucevheriYokEt(Vector2Int pos) // yan yana 3 tane mücevher olunca işlevini yerine getirir.
{
    if (tumMucevherler[pos.x, pos.y] != null)
    {
        if (tumMucevherler[pos.x, pos.y].eslestimi)
        {
            // patlama sesi için bir alt satır kullanıyoruz
            if (tumMucevherler[pos.x, pos.y].tipi == Mucevher.MucevherTipi.bomba)
            {
                SoundManager.instance.PatlamaSesiCikar(); // bomba patlayınca bomba sesi patlama için
            }
            else
            {
                SoundManager.instance.MucevherSesiCikar();
            }
        }

        Instantiate(tumMucevherler[pos.x, pos.y].mucevherEfekt, new Vector2(pos.x, pos.y), Quaternion.identity);
    }
}
```

Stajyerin imzası

İş yeri yetkilisinin imzası

Sayfa no

<u>İş Yerinde Çalışılan Birim:</u> Marketing ve Game Developer	<u>Tarih:</u> 08/08/2024
--	--------------------------

Yapılan İş: Üçlü eşleşmeleri engelleyen algoritma geliştirdim ve eşleşen mücevherleri yok eden, ardından boş alanlara yeni mücevherler yerleştiren fonksiyonlar üzerinde çalıştım

Yok Edilen Mücevherlerin Yerini Doldurma: Eşleşen mücevherler yok edildikten sonra, boşalan alanların yukarıdan yeni mücevherlerle doldurulması gerekiyor. Bunun için `BoardYenidenDoldurRouitine()` isimli bir fonksiyon yazdım. Bu fonksiyon, eşleşen mücevherler yok edildikten sonra tahtanın üst kısmındaki boşlukları doldurarak oyunun devamlılığını sağlıyor. Aynı zamanda, yeni mücevherler yerleştirilirken eşleşmelerin tekrar olup olmadığını kontrol etmek için bir zaman gecikmesi ekledim. Eğer yeni eşleşmeler varsa, tekrar yok ediliyor.

Boş Alanları Doldurma ve Rastgele Mücevher Oluşturma: Mücevherler yok edildikten sonra, boş kalan alanları doldurmak için `UstBosluklariDoldurFNC()` isimli fonksiyonu yazdım. Bu fonksiyon, oyun tahtasında boş kalan hücreleri rastgele mücevherlerle dolduruyor. Aynı zamanda, yanlış yerleşimlerin engellenmesi için oluşturulan mücevherlerin üçlü eşleşme yapmadığından emin olunuyor

Yoğun bir gündü, aslında bugün bir önceki günün gecesinde çalıştığım çalışmalarda bulunmakta.

```

1 başvuru
IEnumerator BoardYenidenDoldurRouitine() // Yok ettiğimiz eşleşmeleri doldur
{
    yield return new WaitForSeconds(.5f);
    UstBosluklariDoldurFNC();

    yield return new WaitForSeconds(0.5f);

    eslesmeController.EslesmeleriBulFNC(); // mücevherler arasında eşleşmeler varsa

    if(eslesmeController.BulunanMuçevherlerListe.Count>0)
    {
        yield return new WaitForSeconds(0.5f); // tüm eşleşmeleri tamamlandıktan sonra
        TumEslesenleriYokEt();
    } else
    {
        yield return new WaitForSeconds(.5f);
        gecerliDurum = BoardDurum.hareketEdiyor;
    }
}

```

<u>Stajyerin imzası</u>	<u>İş yeri yetkilisinin imzası</u>	<u>Sayfa no</u>
		7

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 09/08/2024

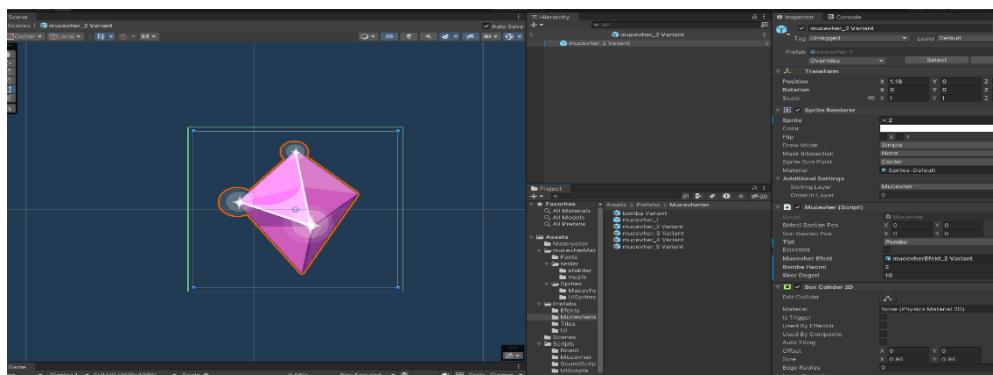
Yapılan İş: Mücevher Yok Edilme Animasyonları ve Görsel Efektlerin Geliştirilmesi

Bugün, mücevherlerin yok edilmesi sırasında oluşan boşluklara animasyonlarla yeni mücevherlerin düşmesini sağladım. Bu, oyuncuya görsel bir tatmin sağlarken, oyunun dinamikliğini artırdı. Mücevherler yok olduğunda, patlama efektleri ve parçacık efektleri ekleyerek görsel zenginliği artırdım. Bu efektleri sahneye entegre etmek, oyuncunun oyun deneyimini daha akıcı ve çekici hale getirdi.

Unity'de efekt sistemleri, oyunun atmosferini oluşturmak ve oyuncunun deneyimini geliştirmek için kritik öneme sahiptir. Unity'nin **Particle System** aracı sayesinde, çeşitli efektler (patlama, duman, ışılıtı vb.) oluşturmak mümkündür. Bu sistem, oyun nesnelerinin belirli olaylara tepki vermesini sağlar ve görsel olarak etkileyici bir atmosfer yaratır. Yok edilen mücevherlerin patlama efektlerini çalıştırırmak için şu kodu kullandım:

```
Instantiate(tumMucevherler[pos.x, pos.y].muceverEfekt, new Vector2(pos.x, pos.y), Quaternion.identity); // Patlama efektlerini çalıştırıyoruz.
```

Bu kod, belirli bir pozisyonda mücevherin efektini oluşturarak, oyuncunun dikkatini çeken bir görsel geri bildirim sağlar. **Instantiate** fonksiyonu, belirtilen efekt prefab'ını oyun sahnesinde yaratır. Pozisyon ve dönüşüm (Quaternion.identity) ayarları, efektin tam olarak hedef konumda görünmesini sağlar.



Stajyerin imzası

İş yetkiyetlisinin
imzası

Sayfa no

8

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 12/08/2024

Yapılan İş: Oyuna rastgele bombalar ekleyerek patlama sistemi geliştirdim ve kullanıcı isteğine bağlı olarak sahnedeki mücevherleri karıştıran bir fonksiyon oluşturduğum.

Bugün, oyuna rastgele bombalar ekleyerek patlama sistemi geliştirdim. Patlama mekanizması, bombanın çevresindeki mücevherlerin yok olmasını sağlıyor ve oyunculara daha stratejik bir oyun deneyimi sunuyor. Bunun için, bombanın çıkma olasılığını belirlemek üzere bir **bombaCikmaSansi** değişkeni tanımladım



Ayrıca, **BombaBolgesiniIsaretle** fonksiyonu ile patlama alanındaki mücevherleri işaretleyerek yok olmalarını sağlıyorum

Ve son olarak kullanıcının isteğine bağlı olarak sahnedeki tüm mücevherlerin karıştırılmasını sağlayan bir fonksiyon geliştirdim. **BoardKaristirFNC** isimli bu fonksiyon, mevcut mücevherleri karıştırarak yeni kombinasyonlar oluşturuyor. Bu fonksiyon, sahnedeki mücevherleri rastgele yerleştirerek oyunculara yeni kombinasyonlar deneme fırsatı sunuyor. P tuşuna atamasını yaptım. İleride yeniden düzenleyip menü üzerinde çalışmasını sağlamayı amaçlıyorum

Stajyerin imzası

İş yeri yetkilisinin imzası

Sayfa no

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 13/08/2024

Yapılan iş: Skor sistemi ve Zamanlama üzerinde çalışma gerçekleştirdim

Bugün, oyun içi skor ve zamanlama sistemlerini geliştirmeye odaklandım. Evden çalışarak yürüttüğüm bu projede, oyuncunun puan kazanmasını sağlayan bir skor tablosu oluşturdum. Oyundaki mücevher eşleşmeleri sonucunda oyuncuya puan kazandırmak amacıyla **PuanıArtırFNC()** adlı fonksiyonu kodladım. Bu fonksiyon, eşleşme olduğunda ilgili puanı hesaplayarak skoru güncelliyor ve ekranda gösteriyor.

Ayrıca, oyunda kalan süreyi gösteren bir sayaç sistemi geliştirdim. Bunun için **GeriSayRoutine()** adlı bir korutin oluşturduk. Her saniyede bir süreyi azaltan bu sayaç, süre sıfıra ulaştığında oyundaki hareketleri durduracak şekilde yapılandırıldı. Bu sayaç sayesinde oyuncular, kalan zamanı ekranda görebiliyor ve süre dolduğunda oyun duruyor. Kodlar üzerinde, her saniyede geri sayımın güncellenmesini sağlayan mekanizmaları **kalanZamanTxt** ve **kalanZaman** değişkenleriyle oluşturup, bunları oyuna entegre ettim.

`UIManager.instance.PuanıArtırFNC(eslesmeController.BulunanMucevherlerListe[i].skorDegeri);`



Stajyerin imzası

İş veri yetkilisinin imzası

Sayfa no

10

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 14/08/2024

Yapılan İş: Ana Menü Unity üzerinde tasarlardım ve çalışmasını sağladım.

Bugün oyunun ana menüsünü geliştirdim ve menüdeki butonların işlevsellliğini sağladım. Ana menüye "Başla" ve "Çıkış" butonları ekledim. Butonlar, kullanıcıya oyun başlangıcında yönlendirme yaparak, oyunu başlatma ve çıkış yapma işlemlerini mümkün kılmakta.

Ana menüyü kodlarken, `MainMenuController` sınıfını oluşturdum ve butonların işlevlerini bu sınıf üzerinden yönettim. "Başla" butonu ile oyunu başlatmak için `OynaBasla()` fonksiyonunu yazdım. Bu fonksiyon, oyuncunun belirli bir seviyeye başlamasını sağlayacak şekilde `SceneManager.LoadScene(levelAdı)` komutuyla sahneler arasında geçiş yapıyor. Ayrıca "Çıkış" butonunu, uygulamadan çıkış yapacak şekilde kodladım. Bunun için `Application.Quit()` fonksiyonunu kullanarak, Unity'de build alındığında oyunu kapatacak bir yapı kurdum.

Aynı zamanda, genel kullanıcı arayüzü (UI) üzerinde çalışarak oyunun ilk seviyesi olan Level 1'e giriş yapabilmeyi sağladım. Arayüz elemanları üzerinde gerekli düzenlemeleri yaparak, oyuncunun menüden direkt olarak oyuna başlamasını ve sezgisel bir kullanıcı deneyimi yaşamamasını hedefledim.



Stajyerin imzası

İş veri yetkilisinin imzası

Sayfa no

11

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 15/08/2024

Yapılan İş: Duraklatma (Pause), Ses Efektlerini oyun ekledim ve Oyun Bitti

Bugün oyunun önemli fonksiyonlarından biri olan duraklatma (pause) sistemi, ses efektlerini geliştirdim.

Öncelikle ses efektlerini oyuna entegre ettim. Farklı oyun durumlarına (mücevher toplama, patlama, oyun bitisi) uygun ses efektleri oluşturarak bu sesleri SoundManager adlı bir sınıf aracılığıyla yönettim. Seslerin daha doğal bir şekilde çıkması için rastgele pitch değerleri ayarlayarak ses seviyelerinde küçük oynamalar yaptım. Bu, oyun sırasında ses efektlerinin monoton olmamasını sağladı.

```
using UnityEngine;
using System.Collections;

public class SoundManager : MonoBehaviour
{
    public static SoundManager instance; // her yerden erişebilmek için

    public AudioSource mucevherSesi, patlamaSesi, oyunBittiSesi;

    @ Unity iletişimi | 0 başvuru
    public void Awake()
    {
        instance = this;
    }

    1 başvuru
    public void MucevherSesiCikar()
    {
        mucevherSesi.Stop();

        mucevherSesi.pitch = Random.Range(0.8f, 1.2f); // Sese ahenk katmak için random bir şekilde sesle düz

        mucevherSesi.Play(); // Mücevher sesi bu şekilde çıkacak
    }
}
```

Stajyerin imzası

İş yeri yetkilisinin
imzası

Sayfa no

12

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 15/08/2024

Yapılan İş: Duraklatma (Pause), Ses Efektlerini oyun ekledim ve Oyun Bitti

Ayrıca, oyunu duraklatma ve devam ettirme işlevlerini gerçekleştiren bir Pause paneli tasarlardım. Paneli açıp kapatmak için bir fonksiyon yazdım ve oyun duraklılığında Time.timeScale değerini 0 yaparak tüm hareketleri durdurduğum. Aynı şekilde oyuna devam edildiğinde ise Time.timeScale'i 1 yaparak oyunun normal akışına geri dönmesini sağladım. Bu sayede oyuncular oyunu istedikleri zaman duraklatıp devam edebilecekler.



Son küçük düzeltmelerden sonra oyunumuz hazır durumda şu an sadece bir tane level bulunuyor. İstediğimiz gibi level'ları artırıp oyunun oynanış süresini değiştirebiliriz.

Stajyerin imzası

İş yeri yetkilisinin
imzası

Sayfa no

13

İş Yerinde Çalışılan Birim: Marketing ve Game Developer	Tarih: 16/08/2024	
Yapılan İş: Şirkete sunulacak fikir araştırması		
Bugün, şirket bünyesinde yeni oyun projeleri geliştirmek amacıyla farklı simülasyon oyun fikirleri üzerine arkadaşım ile birlikte araştırmalar yaptık. Gün boyu bu projelerin içeriklerini oluşturarak, araştırmalarımızı dokümant edip şirket yetkililerine sunmak üzere hazırlıklar yaptı. Araştırdığımız oyun projeleri şunlardı:		
1.	Arkeoloji Sim: Oyuncuların kazı çalışmaları yaparak nadir bulunan eserleri keşfettiği bir simülasyon oyunu. Bu eserler açık artırmalarla satılarak kazanç elde ediliyor. Oyun, tarih ve ticareti bir araya getiren bir deneyim sunmayı hedefliyor.	
2.	At Eğitimi Sim: Oyuncuların sıfırdan bir at çiftliği kurduğu, at eğitimi ve yetiştiriciliğiyle ilgilentiği bir simülasyon. Oyunda atları besleme, bakma ve çeşitli yarışmalara hazırlama gibi detaylı mekanikler bulunuyor.	
3.	Film Yapımcısı Sim: Sıfırdan bir film stüdyosu kurarak, film yapım sürecine dahil olduğumuz bir simülasyon oyunu. Oyuncular filmleri IMDB tarzı bir değerlendirme sistemi ile puanlayacak ve bu filmlerden kazanç sağlayacaklar.	
4.	Kovboy Sim: İki farklı ilerleyiş sunan bir simülasyon. Oyuncular ya kasabayı haydutlardan koruyan bir kovboy olacak ya da haydut olup kasabaları soyabilecek. Seçimlere dayalı ilerleyiş mekanikleri ile farklı deneyimler sunmayı amaçlayan bir oyun fikri.	
Bu projelerin araştırılması, içerik oluşturulması ve oyun mekaniklerinin geliştirilmesi üzerine yoğunlaştık. Oyunların hem eğlence hem de yönetim dinamiklerini içerecek şekilde nasıl daha ilgi çekici hale getirilebileceği üzerine öneriler geliştirdik. Araştırılan oyunların dokümanlara aşağıdaki linklerden ulaşabilirsiniz:		
Arkeoloji Sim / Film Yapımcısı Sim / At Eğitimi / Kovboy Sim		
Stajyerin imzası	İş veri yetkilisinin imzası	Sayfa no
		14

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 19/08/2024

Yapılan iş: Şirkete sunulacak fikir araştırması

Bugün, simülasyon oyunları fikirlerinin geri kalanını araştırmaya ve dokümanete etmeye devam ettik. Yine aynı şekilde, projelerin temel mekaniklerini ve içeriklerini oluşturarak şirket yetkililerine sunulmak üzere hazırlık yaptık. Araştırdığımız projeler şunlardı:

1. Saat Shop Sim: Oyuncuların saat ürünleri alıp sattığı, ardından bu saatleri tamir edebildiği bir simülasyon oyunu. Zaman yönetimi ve ticaret mekanikleri üzerine kurulu bir oyun tasarlandı.
2. Optik Shop Sim: Oyuncuların gözlük ürünlerini alıp sattığı ve gözlüklerin tamirini yaptığı bir simülasyon oyunu. Müşteri memnuniyeti ve envanter yönetimi gibi detaylı mekanikler içeriyor.
3. Elektronics Shop Sim: Teknoloji ürünlerinin satıldığı ve tamir edilebildiği bir simülasyon oyunu. Oyun, mağaza yönetimi ve elektronik cihazların bakımı üzerine odaklanıyor.
4. Belediye İşçisi Sim: Belediye işçi olarak parkları ve sokakları temizlediğimiz, parkları dizayn ettiğimiz bir simülasyon oyunu. Oyuncular çevre düzenlemeleri yaparak şehirdeki parkları güzelleştirmeye çalışacaklar.
5. Müze Sim: Arkeolojik eserleri açık artırma yoluyla satın alıp, müze kurarak bu eserleri sergilediğimiz bir simülasyon oyunu. Oyuncular, müzeyi genişletip ziyaretçi çekerek gelir elde edecekler.
6. Gelecek Dünya Sim: İleri bir tarihte geçen, oyuncuların açık dünyada hayatı kalmaya çalıştığı bir simülasyon oyunu. Gelecekteki teknolojilerin ve yaşama şartlarının yer aldığı bir dünya tasarlanyor.

Bu projelerin mekaniklerini detaylandırdıktan sonra, oyunların nasıl daha geniş bir kitleye hitap edebileceği üzerine sunumlar准备了。Gün sonunda Cubecube games yetkilileri ile bir online toplantı ile bunları arkadaşım ile birlikte şirkete sunduk。

<u>Stajyerin imzası</u>	<u>İş veri yetkilisinin imzası</u>	<u>Sayfa no</u>
		15

İş Yerinde Çalışılan Birim: Marketing ve Game Developer	Tarih: 20/08/2024	
Yapılan İş: Seçilen Projenin Temel işleyişinin anlatıldığı Game Dizayn Dokümanını hazırladım		
<p>Bugün, şirket yetkilileriyle gerçekleştirdiğimiz toplantıının ardından üzerinde çalıştığımız oyun projesi olan Teknoloji Shop için detaylı bir oyun dökümanı (GDD) hazırladım. Toplantıda alınan kararlar doğrultusunda, oyunun temel mekanikleri ve platformları belirlendi ve bu doğrultuda bir demo versiyonu oluşturmak için gerekli adımları planladım. Bugünkü çalışmalarım, oyun mekaniklerinin detaylandırılması ve proje açıklamasının düzenlenmesi üzerine yoğunlaştı.</p>		
Araştırma ve Düzenleme: Öncelikle, benzer simülasyon oyunları üzerine araştırmalar yaparak satış, tamirat, ve tedarik süreçlerinin nasıl işlediği konusunda bilgi edindim. Bu bilgiler ışığında, hazırladığım GDD belgesini revize ettim. Ayrıca, oyuncu deneyimini daha ilgi çekici kılmak adına yapay zeka destekli NPC'lerin mağaza yönetimine nasıl entegre edileceği üzerine çalıştım.		
Hazırladığım GDD'nin Özeti:		
<ul style="list-style-type: none">Tema: Market ve SimülasyonTemel Oyun Mekanikleri:<ul style="list-style-type: none">Satış: Ürünlerin satışı ve para üstü işlemleri.Tamirat: Arızalı ürünlerin onarılması.Tedarik: Ürün lisanslarının açılması ve sipariş verilmesi.Market Düzenlemesi: Raf ve vitrinlerin yerlesimi.AI Desteği: Temizlikçi, kasier, ve usta tamirci gibi NPC'lerin oyuncuya destek olması.Hedef Platformlar: PCPara Kazanma Modeli: Oyun, ücretli olacak ve demo sürümü oyuncuların test etmesi için sunulacak.Proje Açıklaması: Elektronik Shop Simülasyonu, oyuncuların modern bir teknoloji mağazasını yönetmesini sağlayan, detaylı ve kapsamlı bir mağaza yönetim oyunudur. Oyuncular, ürün satışı, tamirat, tedarik ve mağaza düzenlemesi gibi birçok işlemi kontrol ederek mağazalarını büyütmeye çalışır.		
Stajyerin imzası	İş yeri yetkilisinin imzası	Sayfa no
		16

<u>İş Yerinde Çalışılan Birim:</u> Marketing ve Game Developer	Tarih: 20/08/2024
--	-------------------

Yapılan İş: Seçilen Projenin Temel işleyişinin anlatıldığı Game Dizayn Dokümanını hazırladım

Oynanış: Küçük bir elektronik mağazası işletecek başlanan oyunda, oyuncular çeşitli ürünler satışı sunar, müşteri taleplerine yanıt verir ve mağazalarını büyütmek için stratejik kararlar alırlar. Yapay zeka destekli NPC'ler de bu süreçte oyuncuya destek olur.

Hikaye: Modernleştirilen bir mağazanın eski popüleritesini geri kazanma hikayesi. Girişimcinin başarıları ve mağazayı bir teknoloji merkezine dönüştürme süreci anlatılıyor.

Bu çalışmayla birlikte, projenin temel mekanikleri ve ilerleyişine dair detayları oluşturmuş oldum. Örnek aldığım bir kaç mağaza oyunu görseli :



<u>Stajyerin imzası</u>	<u>İş yeri yetkilisinin imzası</u>	<u>Sayfa no</u>
-------------------------	------------------------------------	-----------------

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 21/08/2024

Yapılan İş: Hikâye ve Oyun Mekanikleri üzerinde çalışma

Bugün, daha önce hazırladığım Teknoloji Shop oyununun tasarım dökümanına ek olarak, oyuna dahil edeceğimiz eğlence unsurları ve oyunun hikayesi üzerine çalıştım. Hedefimiz, oyuncu etkileşimi ve oyunun keyif verici unsurlarını artırmak. Bunun için temel hikaye geliştirmeleri ve yeni oyun mekanikleri üzerine yoğunlaştım.

Geliştirilen Hikaye ve Mekanikler: Ana hikaye olarak, oyuncunun geçmişte çalıştığı ve kendisini kovmuş olan eski patronuna rakip olarak yeni bir elektronik mağaza açmasını ve bu mağazayı büyütme mücadelemini belirledim. Bu hikaye etrafında, rakip dükkan ile etkileşimi artıracak rekabet ve sabotaj mekaniklerini ekledim.

- **Hikaye:** Oyuncu, bir zamanlar karşısındaki Elektro Shop'ta çırak olarak çalışırken kovalmuş ve intikam almak için kendi dükkanını açmış bir girişimcidir. Hedefi, rakip dükkanın daha başarılı bir elektronik mağazası kurmak ve kasabanın teknoloji merkezi olmaktır.
- **Rakip Dükkan ile Etkileşim:**
 - **Rekabet ve İddialar:** Rakip dükkan sahibi, oyuncuya belli görevler (örneğin, 3 gün içinde 100 ürün satma) teklif eder. Görev kabul edilirse ve başarıyla tamamlanırsa ödül kazanılır; aksi halde para kaybedilir.
 - **Sabotaj Mekanığı:** Oyuncu rakip dükkanın malzeme çalabilir veya rakip dükkanın zarar verebilir (örneğin, camları kırmak veya müşterileri korkutmak). Ancak bu eylemler polis tarafından fark edilirse oyuncu mağazasında ceza ile karşılaşabilir.
 - **Rakip Dükkanın Saldırıları:** Rakip dükkan sahibi de oyuncunun mağazasına zarar vermeye çalışabilir, bu da oyuncunun mağazasını korumasını gerektirecektir.
- **Kampanyalar ve İndirim Günleri:**
 - Oyuncu, belirli günlerde indirim kampanyaları düzenleyebilir ve reklam vererek müşteri sayısını artırabilir. Kampanya dönemlerinde mağazaya daha fazla müşteri çekmek için kasierleri ve diğer mağaza içi düzenlemeleri optimize etmek gerekecektir.

Stajyerin imzası

İş veri yetkilisinin imzası

Sayfa no

<u>İş Yerinde Çalışılan Birim:</u> Marketing ve Game Developer	<u>Tarih:</u> 21/08/2024
<u>Yapılan iş:</u> Hikâye ve Oyun Mekanikleri üzerinde çalışma	
<p>Eğlence Unsurları: Oyuncunun mağaza yönetim sürecinde sıkılmaması ve eğlenceli görevlerle meşgul olmasını sağlamak adına bazı mini oyunlar ve ilginç mekanikler ekledim:</p> <p>Telefon Kılıfı Tasarımı: Oyuncu, müşterilere özel tasarım telefon kılıfları yaparak daha fazla gelir elde edebilir.</p> <p>Hırsız Kovma Mekaniği: Dükkanını hırsızlardan korumak için oyuncu, eğlenceli bir şekilde Nokia 3310 telefon fırlatabilir. Bu küçük mizahi dokunuş, oyuna eğlenceli bir atmosfer katacaktır.</p> <p>Tamirat Süreçleri: Tamir süreçleri, seviyeye bağlı olarak karmaşıklAŞacak ve yeni araçlar eklenerek oyuncunun tamirat becerilerini geliştirmesine olanak tanıyacak şekilde tasarlandı. Örneğin:</p> <p>Kırık Ekran Tamiri: Oyuncu, kırık cam parçalarını temizleyip, ekranı değiştirir.</p> <p>Su Kaçması: Su kaçan cihazı kurutma ve temizleme işlemi.</p> <p>Kablo Bağlama: Karışık kabloları doğru şekilde bağlamak.</p> <p>Ürün ve Stand Tasarımı: Mağazada satılacak ürünler için özel standlar ve dolap tasarımları üzerine çalıştım. Her ürün tipi için ayrı bir sergileme düzeni olacak şekilde raf türleri oluşturulacak. Bu düzenlemeler mağaza içi dekorasyon ve satışları doğrudan etkileyecək.</p> <p>Yeni Ürün Lisansları: Oyuncu, yeni ürünler mağazasında satmak için lisans açabilir ve daha geniş bir ürün yelpazesine sahip olabilir. Örneğin: Lisans 1: 8 Adet Akıllı Telefon, 8 Adet Akıllı Telefon Kılıfı, 4 Adet Kulaklık (Bluetooth / Kablolu)</p> <p>Bu çalışmalar sayesinde, Teknoloji Shop oyununun temel hikayesini ve eğlence unsurlarını daha ayrıntılı hale getirdim. Bu eklemeler, oyunun hem stratejik hem de mizahi yönünü güçlendirecek ve oyunculara daha keyifli bir deneyim sunacaktır</p>	
<u>Stajyerin imzası</u>	<u>İş veri yetkilisinin imzası</u>
	19

<u>İş Yerinde Çalışılan Birim:</u> Marketing ve Game Developer	Tarih: 22/08/2024
--	-------------------

Yapılan İş: İstanbul Ziyareti ve Oyun Geliştirme Süreci

Bugün, şirketle anlaşma sağladığımız oyun üzerine sözleşme imzalamak için İstanbul'a gittim. Günüm büyük ölçüde yolda geçtiği için staj süresince herhangi bir iş gerçekleştirmeye fırsatım olmadı. Oyun, Teknoloji Mağazası işlettigimiz bir simülasyon oyunudur. Bu süreçte, oyunun geliştirilmesi GDD (Game Design Document) belgesine uygun olarak gerçekleştirilecektir. GDD, oyunun tüm tasarım unsurlarını ve mekaniklerini içeren önemli bir dökümandır ve bu doğrultuda sürecimizi yönlendirecek. Oyun hakkında daha fazla bilgi vermek gereklirse, Teknoloji Mağazası simülasyonu; oyuncuların bir teknoloji mağazası işletmelerine olanak tanıyacak, ürün yönetimi, satış, müşteri ilişkileri ve mağaza düzeni gibi çeşitli mekanikleri içerecektir. Bunların yanında iki el telefon alışverişesi, telefon kılıflarına çizim yapma gibi ek mekanikler ile oyun zengileştirmeyi amaçlıyoruz.

<u>Stajyerin imzası</u>	<u>İş veri yetkilisinin imzası</u>	<u>Sayfa no</u>
		20

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 23/08/2024

Yapılan İş: Teknoloji Shop oyunun başlanması

Çalışan arkadaşım ile birlikte, bugün bize şirket tarafından sağlanan sistemleri incelemeye başladık. İlk olarak LocalizationManager sınıfını ele aldık. Bu sınıf, oyundaki metinlerin farklı dillere çevrilmesi ve oyunun farklı dillerde oynanabilmesi için kullanılıyor. Özellikle oyunda yer alan öğelerin isimlerinin ve açıklamalarının yerelleştirilmesini sağlayan mekanizmayı anlamaya çalıştık. Şu an için bu sistemin nasıl çalıştığını çözmeye odaklandık, ancak ileride oyumuzda yerelleştirme desteği eklemeyi düşündüğümüzde bu kod bizim için çok değerli olacak gibi görünüyor.

Bir sonraki adımda InteractionController sınıfına göz attık. Bu sınıf, oyuncunun etkileşime gelebileceği nesneleri ve bu etkileşimlerin nasıl yönetildiğini kontrol ediyor. Örneğin, oyuncu bir nesneye yaklaşlığında o nesneyle etkileşim kurup kuramayacağını belirleyen temel sistemi içermektedir. Bu sınıfı detaylıca inceledik ama şu aşamada sadece nasıl çalıştığını anlamaya çalıştık.

```
1 bayuru
public void MoveToTarget(InteractableItem item, Transform pickupParent)
{
    Vector3 currentPos = item.transform.position;
    Vector3 targetPos = pickupParent.position;
    Vector3 direction = (targetPos - currentPos).normalized;

    RaycastHit hit;
    if (Physics.Raycast(currentPos, direction, out hit, Vector3.Distance(currentPos, targetPos)))
    {
        Collider hitCollider = hit.collider;
        float objectTop = hitCollider.bounds.max.y;
        float itemHalfHeight = item.GetComponent<Collider>().bounds.max.y;

        Vector3 hitPoint = hit.point;
        float heightAbove = objectTop + (itemHalfHeight * 1.5f);
        Vector3 overObjectPoint = new Vector3(hitPoint.x, heightAbove, hitPoint.z);
        item.transform.DOLocalRotate(Vector3.zero, 0.4f, RotateMode.Fast).SetEase(Ease.Linear);
        item.transform.DOMove(overObjectPoint, 0.1f).OnComplete(() =>
        {
            item.transform.SetParent(pickupParent);
            item.transform.DOLocalMove(Vector3.zero, 0.4f);
        });
    }
    else
    {
        item.transform.SetParent(pickupParent);

        item.transform.DOLocalMove(Vector3.zero, 0.3f).SetEase(Ease.Linear);
        item.transform.DOLocalRotate(Vector3.zero, 0.3f, RotateMode.Fast);
    }
}
```

Stajyerin imzası

İş yeri yetkilisinin imzası

Sayfa no

21

<u>İş Yerinde Çalışılan Birim:</u> Marketing ve Game Developer	Tarih: 26/08/2024
<u>Yapılan iş:</u> Firmanın sunduğu sistemlerin incelenmesi	
<p>Bugünkü incelemelerimizde envanter sistemine odaklandık. İlk olarak InteractableItem sınıfını ele aldık. Bu sınıf, oyunda oyuncunun etkileşime gecebileceği nesneleri tanımlıyor. Interaktif nesnelerin oyunla nasıl entegre edildiğini ve hangi özelliklerle donatıldığını inceleyerek, bu yapıların oyunlarımızda nasıl kullanılabileceğini anlamaya çalıştık. Henüz uygulamaya geçmedik, ama etkileşimli nesnelerin yönetimi konusunda birçok fikir edindik.</p> <p>Ardından InventoryItem sınıfına geçtik. Bu sınıf, oyundaki her bir öğenin envanterde nasıl tutulduğunu ve hangi özelliklere sahip olduğunu tanımlıyor. Özellikle eşyaların istatistiklerini (örneğin, saldırısı hızı, dayanıklılık gibi) yöneten yapıları inceledik. ItemStat sistemi, her bir eşyanın statlarını dinamik olarak yönetmek için kullanılıyor ve biz de bu yapıyı nasıl daha iyi anlayabileceğimizi tartıştık.</p> <p>Son olarak ItemData sınıfını inceleyerek günümüzü tamamladık. Bu sınıf, oyundaki eşyaların temel verilerini (örneğin, isim, görsel, statlar) içeriyor. Aynı zamanda yerelleştirme desteği sayesinde farklı dillere çevrilebiliyor. Bu sınıfın, oyunumuzda kullanacağımız eşyaların temel yapı taşlarını oluşturduğunu fark ettik. Şimdilik sadece bu sistemlerin nasıl çalıştığını anlamaya odaklandık, ilerleyen günlerde daha detaylı çalışmalar yapmayı planlıyoruz.</p> <pre>[CreateAssetMenu(fileName = "New Item", menuName = "CubeCubeGames/Item")] @UnityBetiği 2 bayuru public class ItemData : ScriptableObject { [OnInspectorInit("UpdateItemQualities")] public string identifierName; [Space] public ItemCategory itemType; [ShowIf("@itemType==ItemCategory.WeaponBase itemType==ItemCategory.Hilt")] public ItemCategory craftItemType; [BoxGroup("Equipment Data")] [ShowIf("@itemType==ItemCategory.Equipments")] public GameObject equipVisual; public ItemSource[] itemSource; [PreviewField(50)] public Sprite itemImage; [BoxGroup("Default Item Datas")] [SerializeField] private List<ItemStat> defaultItemStats; }</pre>	
<u>Stajyerin imzası</u>	<u>İş yeri yetkilisinin imzası</u>
	22

İş Yerinde Çalışılan Birim: Marketing ve Game Developer	Tarih: 27/08/2024
Yapılan iş: Marketing için çalışmalar ve Kamera Kontroller	
<p>Bugün, oyun için CameraController kodları yazıldı. Bu sınıf, sinematik kameraların kontrolünü sağlamak, kameralar arasında geçiş yapmak ve çeşitli efektleri yönetmek amacıyla geliştirildi.</p> <p>Kod içerisinde Cinemachine kullanılarak, kameralar arasındaki geçiş yöneten SetCamera ve ResetCamera fonksiyonları eklendi. Bu fonksiyonlar sayesinde aktif kamera belirlenip önceliklendirilerek geçişler yapılabiliyor. Böylece, oyun içinde farklı açılardan dinamik sahneler elde edilebiliyor.</p> <p>Ayrıca, kamera sarsıntısı efektini oluşturmak için DOTween kütüphanesi kullanıldı. ShakeCamera fonksiyonu, belirli bir süre ve şiddette kamera sarsıntı yaratarak aksiyon anlarını daha dramatik hale getiriyor. Bu fonksiyonla birlikte, kamera hareketlerinin oyuncu deneyimine etkisi ölçüde artırılmış oldu.</p> <p>Son olarak, UpdatePostProcess fonksiyonu sayesinde oyun sırasında post-processing efektleri dinamik olarak değiştirilebiliyor. Color Adjustments bileşeni ile oyunun atmosferine göre pozlama değerleri ayarlanarak sahneye uygun görsel efektler eklenebiliyor.</p> <p>Bugün ek olarak, şirket için marketing yönünde yayıncı analizi ve haber bülteni sayfaları üzerinde araştırmalar gerçekleştirildi. 50 civarında haber bülteni sayfası ve editörlere ulaşılacak iletişim bilgileri derlendi. Bu çalışmalar, oyunun tanıtımı ve lansman sürecinde hedef kitleye ulaşmak için büyük önem taşıyor.</p> <pre> private void Awake() { _cinemachineBasicMultiChannelPerlin = firstPersonCamera.GetCinemachineComponent< CinemachineBasicMultiChannelPerlin>(); } 0 başvuru public void SetCamera(CinemachineVirtualCamera newCamera) { _currentCamera = newCamera; newCamera.Priority = _mainCameraPriority; firstPersonCamera.Priority = _secondaryCameraPriority; _cinemachineBasicMultiChannelPerlin = _currentCamera.GetCinemachineComponent< CinemachineBasicMultiChannelPerlin>(); } </pre>	
Stajyerin imzası	İş yeri yetkilisinin imzası
	Sayfa no
	23

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 28/08/2024

Yapılan iş: Marketing için çalışmalar ve Item Sistemleri

Bugün, oyunumuzun ürün etkileşim sistemlerini geliştiren SellableItem ve PlacableItem kodlarını inceledim. Bu kodlar, oyuncuların mağaza içinde ürünler alıp taşımaları, raflara dizmesi ve kutulara yerleştirmesini düzenliyor. SellableItem, oyuncunun bir ürünü alıp taşıırken ya da kutulara dizmesine yönelik mekanikleri yönetirken, PlacableItem ise bu ürünlerin raflara yerleştirilmesini ve düzenli şekilde yerleştirilmelerini sağlıyor. Her iki kod bloğu da oyunun mağaza yönetimi mekaniklerinde önemli bir yer tutuyor.

Bunun yanında, marketing çalışmaları kapsamında yeni haber bülteni ve editöre ulaşarak, toplamda 140'a yakın iletişim bilgisi tamamlandı. Bu editörler ve bültenler üzerinden ilerleyen günlerde tanıtımlar yapılacak. Ayrıca, sosyal medya platformları için paylaşmak üzere içerikler hazırladım.

```
        else if (pickedItem.interactableItemType == InteractableType.Box)
    {
        for (int i = pickedItem.GetComponent<PlacableItem>().slotTransforms.Count; i > 0; i--)
        {
            if (!pickedItem.GetComponent<PlacableItem>().slotTransforms[i - 1].gameObject.GetComponent<SlotProps>().isAvailable)
            {
                InteractableItem stackedObject = pickedItem.GetComponent<PlacableItem>().slotTransforms[i - 1].GetChild(0).GetComponent<InteractableItem>();
                if (placableIndex == stackedObject.GetComponent<SellableItem>().itemIndex)
                {
                    pickedItem.GetComponent<PlacableItem>().slotTransforms[i - 1].gameObject.GetComponent<SlotProps>().isAvailable = true;
                    SetStackParent(stackedObject);
                    StackToTarget(this, stackedObject);
                    stackedObject.GetComponent<BoxCollider>().enabled = true;
                }
                return;
            }
        }
    }
```

```
// Elde kutu varken Ürün olan bir objeyi kutuya dizme
if (InteractionController.instance.currentPickedItem && InteractionController.instance.currentPickedItem.interactableItemType == InteractableType.Box)
{
    InteractableItem pickedItem = InteractionController.instance.currentPickedItem.GetComponent<InteractableItem>();
    if (itemIndex == pickedItem.GetComponent<PlacableItem>().placableIndex)
    {
        ToggleColliders(false);
        ToggleRigidbody(false);
        SetStackParent(pickedItem);
        StackToTarget(pickedItem, InteractionController.instance.CurrentInteractableItem);
        if (slot != null)
        {
            slot.isAvailable = true;
        }
    }
}
```

Stajyerin imzası

İş yeri yetkilisinin imzası

Sayfa no

24

İş Yerinde Çalışılan Birim: Marketing ve Game Developer

Tarih: 29/08/2024

Yapılan iş: Simülasyon oyunu için kasa sistemi

Bu kod, bir mağaza ortamında kasıyer ve müşteri etkileşimi simüle eden bir sistemdir. Temel bileşenleri kasıyer, müşteri ve ürünler olarak tanımlanmıştır. Başlangıç fonksiyonu ile birlikte bir coroutine olan `CustomerApproach()` başlatılır; bu, müşterinin kasaya yaklaşmasını simüle eder ve ardından müşteri ile kasıyer arasında bir iletişim başlatır. `Update()` fonksiyonu, müşteri kasada iken ürünlerin taranmasını sağlar; bu süreç, boşluk (Space) tuşuna basılarak gerçekleştirilir. Ürünlerin sabit fiyatları, `productPrices` dizisinden alınarak toplam fiyat hesaplanır ve ürünlerin taranması tamamlandığında enter (Return) tuşuna basılması ile toplam fiyat kullanıcıya bildirilir. Ödeme işlemi, `ProcessPayment()` metodu ile simüle edilir; burada müşteri ödemesi, ürün fiyatından fazla olabiliyor. Ödeme yeterliyse, kasıyer para üstünü hesaplar ve işlemi tamamlar; yetersiz ödeme durumunda müşteri bilgilendirilir. İşlem tamamlandığında `EndTransaction()` metodu devreye girer ve işlemin sona erdiğini bildirir. Bu sistem, oyuncuların etkileşime geçebileceğine basit bir kasıyer-satış mekanizmasını simüle ederek oyuncuların mağaza yönetimi ve satış süreçlerini anlamalarına yardımcı olur. Ayrıca, gerçekçi bir simülasyon sunarak oyuncuların mağaza yönetimi becerilerini geliştirmelerine olanak tanır.

```
    I bagıru
void ProcessPayment()
{
    // Müşteri ödeme yapıyor
    if (Input.GetKeyDown(KeyCode.M))
    {
        Debug.Log("Lütfen ödeme miktarını giriniz:");
        customerPayment = totalProductPrice + Random.Range(0f, 50f); // Ödeme, ürün fiyatından fazla olabilir
        Debug.Log("Müşteri: Size " + customerPayment + " TL ödüyorum.");

        if (customerPayment >= totalProductPrice)
        {
            changeToGive = customerPayment - totalProductPrice;
            Debug.Log("Kasıyer: Ödeme alındı. Para üstü: " + changeToGive + " TL.");
            transactionComplete = true;
            EndTransaction();
        }
        else
        {
            float remainingAmount = totalProductPrice - customerPayment;
            Debug.Log("Kasıyer: Yetersiz ödeme. Lütfen " + remainingAmount + " TL daha ödeyin.");
        }
    }
}
```

Stajyerin imzası

İş yerinin yetkilisinin imzası

Sayfa no

25

Rahim Ağrı Kanseri Teşhisinde Sınıflandırma Yöntemlerinin Etkisi

Samet Zengin^{1*} (Orcid ID: 0009-0004-9235-7521)

¹*Yazılım Mühendisliği / Fen Bilimleri Enstitüsü, Bandırma Onyedi Eylül Üniversitesi, Türkiye*

^{*}(smt.zngn02@gmail.com)

Geliş Tarihi (Received):

Kabul Tarihi(Accepted):

Özet – Bu çalışmada, rahim ağrı kanseri teşhisinde sınıflandırma yöntemlerinin etkisini incelemek amacıyla üç farklı makine öğrenmesi modeli olan Destek Vektör Makineleri, Yapay Sinir Ağları ve Random Forest karşılaştırılmıştır. Çalışmanın veri seti, rahim ağrı kanserini etkileyebilecek klinik testler ve demografik faktörlerden oluşmuştur. Bu veri setinde Schiller, Hinselmann ve Sitoloji test sonuçları gibi özellikler, modelin performansını analiz etmek için kullanılmıştır. Öncelikle, modellerin performansı Biopsy, DxCancer ve Dx:HPV hedef değişkenleri üzerinde değerlendirilmiştir. Performans ölçütü olarak ROC eğrisi, Precision-Recall eğrisi ve Karışıklık Matrisleri kullanılmıştır. Sonuçlarına göre, Random Forest modeli, özellikle Biopsy hedef değişkeninde %96 doğruluk, 0.69 F1 skoru ve 0.98 AUC değeri ile en iyi performansı göstermiştir. Destek Vektör Makineleri ve Yapay Sinir Ağları modelleri, bazı hedef değişkenlerde kabul edilebilir performans sunsa da, Random Forest modelinin doğruluk ve genelleme yeteneği üstün bulunmuştur. Ayrıca, Grid Search yöntemi ile Random Forest modelinin hiperparametre optimizasyonu gerçekleştirilmiştir ve en iyi parametreler belirlenmiştir. Modelin sunduğu özellik önem sıralaması, Schiller Testi'nin sınıflandırmada en önemli özellik olduğunu, bunu Hinselmann Testi ve Sitoloji'nin takip ettiğini göstermiştir. Bu bulgular, klinik testlerin makine öğrenmesi yöntemleriyle entegre edilmesinin teşhis sürecindeki etkinliği artırabileceğini ortaya koymuştur. Sonuç olarak, makine öğrenmesi modellerinin rahim ağrı kanseri teşhisinde geleneksel yöntemlere kıyasla önemli avantajlar sağladığı görülmüştür. Özellikle Random Forest modeli, özelliklerin önem derecelerini belirlemesiyile klinik karar destek sistemlerinin geliştirilmesine katkı sunabilecek potansiyele sahiptir.

Anahtar Kelimeler – Rahim ağrı kanseri, destek vektör makineleri, random forest, yapay sinir ağları, hiperparametre optimizasyonu, özellik çıkarımı

1. GİRİŞ

Rahim ağızı kanseri, dünya genelinde kadınlar arasında en yaygın görülen ve önemli bir halk sağlığı olan kanser türlerinden biridir. Bu kanser türü, çoğunlukla insan papilloma virüsü (HPV) enfeksiyonu ile ilişkili olup, düzgün tarama ve erken teşhis yöntemleri kullanılmadığı durumlarda ciddi mortalite oranlarına yol açmaktadır (Devi ve ark., 2016).

Dünya Sağlık Örgütü'ne (WHO) göre, rahim ağızı kanseri her yıl yaklaşık 570 bin yeni vaka ve 311 bin ölümle sonuçlanmaktadır, bu da kadınlar arasında dünyada dördüncü en sık görülen kanser türünü oluşturmaktadır (Bosch ve De Sanjosé, 2007). HPV, Rahim ağızı kanserinin ana etkeni olarak tanımlanmakta ve 16 ile 18 türleri bu hastalığın %70'inden sorumlu tutulmaktadır (Saslow ve ark., 2012).

Rahim ağızı kanserinin erken evrelerinde, Schiller testi, Sitoloji (Pap smear) ve Hinselmann gibi klinik değerlendirmelerle tespit edilebilmektedir. Ancak, bu geleneksel yöntemler; uygulamacının deneyimine bağlı hata riski, zaman alıcı olması ve maliyet etkinliğinin düşük olması gibi sorunları beraberinde getirmektedir (Devi ve ark., 2016). Bu durum, daha hassas ve otomatik karar destek sistemlerinin geliştirilmesine olan ihtiyacı göstermektedir.

Makine öğrenimi, klinik verilerin hızlı ve doğru analiz edilmesi konusunda etkili bir yöntem olarak öne çıkmaktadır. Bu çalışmanın amacı, Rahim ağızı kanserinin erken teşhisinde Destek Vektör Makineleri (SVM), Yapay Sinir Ağları (ANN) ve Random Forest (RF) gibi makine öğrenimi algoritmalarını karşılaştırmak ve bu yöntemlerin etkinliğini değerlendirmektedir. RF algoritması, özellikle özellik önem sıralaması ve öngörü performansı konularında öne çıkmaktır ve Schiller, Sitoloji ve Hinselmann gibi testlerin klinik değerini göstermektedir.

Bu çalışma, makine öğrenimi algoritmalarının Rahim ağızı kanseri teşhisindeki potansiyelini göstermeyi ve bu yöntemlerin geleneksel yöntemlere alternatif olarak klinik uygulamalarda nasıl kullanılabileceğini incelemeyi hedeflemektedir.

2. LİTERATÜR TARAMASI

Rahim ağızı kanseri teşhisinde makine öğrenimi yöntemlerinin etkili olduğunu çeşitli çalışmalar göstermiştir. Devi ve ark. [1], yapay sinir ağlarını kullanarak %94.5 doğruluk oranı ile rahim ağızı kanseri sınıflandırmasında yüksek performans elde etmişlerdir. Bosh ve De Sanjosé [2], HPV'nin Rahim ağızı kanseri üzerindeki nedensel etkisini epidemiyolojik verilerle açıklamış, HPV DNA'sının tanışsal kesinliğini artırmada PCR tabanlı yöntemlerin önemini vurgulamışlardır. Saslow ve ark. [3], HPV'nin yüksek riskli türlerini tanımlayarak, tarama rehberlerini güncellemiş ve tarama ile mortalite oranlarının %50 oranında düşürülebileceğini belirtmişlerdir. Wu ve ark.[4], derin öğrenme tabanlı algoritmaların sitoloji görüntüleri

üzerinde insan düzeyinde doğruluk sağlayarak kanser taramasını daha hızlı ve hassas hale getirdiğini göstermiştir (%96.8 doğruluk, %0.92 AUC). Khan ve ark [5]. sınıf dengeleme teknikleriyle desteklenen SVM ve KNN algoritmaları ile %95'in üzerinde AUC değerlerine ulaşmışlardır. Li ve ark. [6], STRIDE adlı büyük ölçekli bir yapay zeka sistemiyle 341,889 slaytta %98'in üzerinde doğruluk ve patalog düzeyinde açıklanabilirlik sağlamıştır. Rahaman ve ark. [7], DeepCervix adlı hibrit derin öğrenme tekniği ile iki sınıflı sınıflandırmada %99.85, beş sınıflı sınıflandırmada ise %99.14 doğruluk elde ederek mevcut yöntemleri geride bırakmıştır. Sarhangi ve ark.[8], kolposkopi görüntülerinde segmentasyon ve sınıflandırma için derin öğrenme kullanmış, %93 doğrulukla hassas bi tarama sağlamıştır. Poudel ve ark. [9], düşük kaynaklı ülkelerdeki görsel muayene testlerinde, akıllı telefon destekli yapay zeka sistemleri ile hassasiyetin artırılabilceğini göstermiştir. Son olarak, Rahimi ve ark. [10], rahim ağzı kanserinde hayatı kalma tahmini için makine öğrenimi algoritmalarının kullanıldığını ve AUC değerlerinin %0.40 ile %0.99 arasında değiştiğini rapor etmiştir.

Bu çalışmalar, rahim ağzı kanserinin erken teşhisinde makine öğrenimi veya yapay zeka tabanlı sistemlerin potansiyelini ve bu yöntemlerin klinik uygulamalarda nasıl faydalayabileceğini göstermektedir.

3. MATERİYAL VE YÖNTEM

3.1 Veri Seti

Bu çalışmada kullanılan veri seti, rahim ağzı kanseri teşhisinde etkili olan kinik ve demografik bilgileri içermektedir. Veri seti, toplamda 36 özelliği ve 835 gözlemi içerir. Bu özellikler, rahim ağzı kanserine yönelik erken teşhis ve risk faktörlerinin analizine olanak tanımaktadır. Tablo 1'de veri setindeki seçilen önemli özellikler ve bunların açıklamaları verilmiştir:

Özellik	Açıklama
Age	Hastanın yaşı.
Number of sexual partners	Hastanın cinsel partner sayısı. HPV enfeksiyonu riskini değerlendirmek için önemlidir.
First sexual intercourse	İlk cinsel ilişkinin gerçekleştiği yaş. Erken yaşta cinsel aktivite HPV enfeksiyonu riskini artırır.
Num of pregnancies	Hastanın geçirdiği toplam gebelik sayısı. Hormonal değişiklikler risk faktörlerini etkileyebilir.

Smokes	Sigara içme durumu. Sigara, bağılıklık sistemini baskılıayarak HPV'nin ilerlemesini kolaylaştırabilir.
Hormonal Contraceptives	Hormonal doğum kontrol yöntemlerinin kullanım süresi. Uzun süreli kullanım bazı kanser türleriyle ilişkilidir.
STDs:HPV	Hastanın HPV ile ilişkili cinsel yolla bulaşan hastalık geçmişi.
Dx:Cancer	Kanser teşhisini gösteren bir özellik.
Biopsy	Hastanın biyopsi sonuçlarını ifade eden değişken. Rahim ağzı kanseri teşhisinin temel çıktısidır.

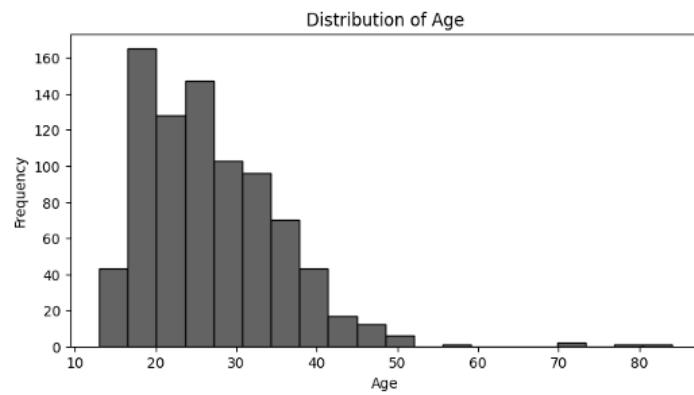
Tablo 1. Veri setinde seçilen önemli özellikler ve açıklamaları

Veri setinde eksik değerler medyan ile doldurulmuş ve tüm veriler analize uygun hale getirilmiştir. Ayrıca, aşağıda veri setinden 5 örnek gözlem sunulmaktadır.

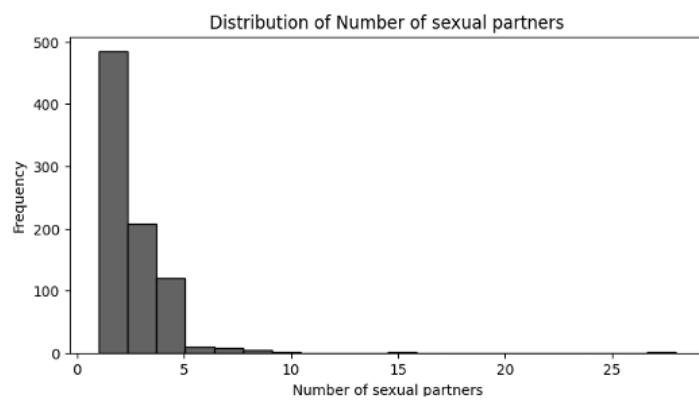
Number Age of sexual partners	First sexual intercourse	Num of pregnancies	Smokes	Hormonal Contraceptives	STDs:HPV	Dx:Cancer	Biopsy
25	1	18	0	0	0	0	0
30	2	16	1	1	1	0	1
35	3	21	2	0	1	1	0
40	2	19	3	1	0	1	1
28	1	18	0	0	0	0	0

Tablo 2. Veri seti örnekleri

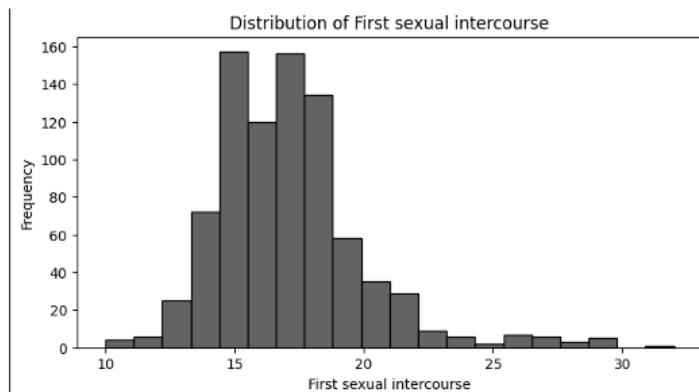
Bu özelliklerin seçiminde, rahim ağzı kanseriyle ilgili literatürden elde edilen bilgiler ve model tarafından belirlenen özellik önem dereceleri dikkate alınmıştır. Ayrıca, bu özelliklerin dağılımları histogramlar yardımıyla görselleştirilmiş ve veri setinin genel yapısı incelenmiştir. Aşağıda özelliklerin histogram dağılımları listelenmektedir.



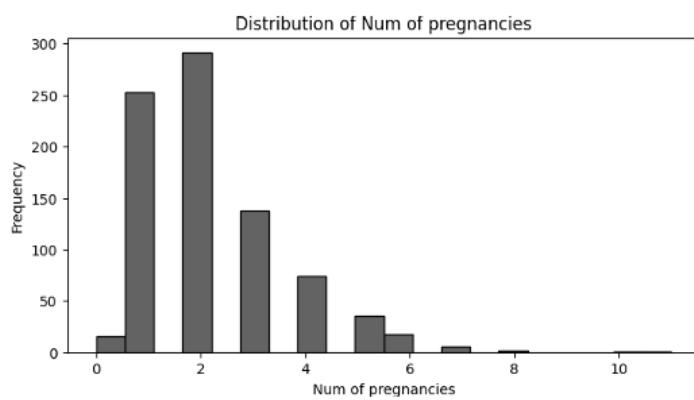
Şekil 1: Yaş özelliğine göre dağılım grafiği.



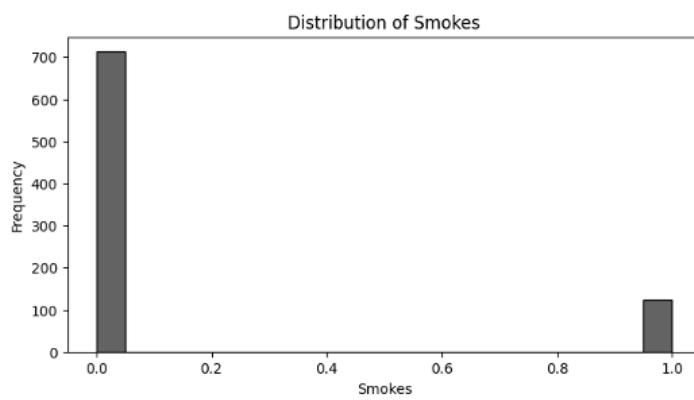
Şekil 2: Hastanın cinsel partner sayısına göre dağılım grafiği.



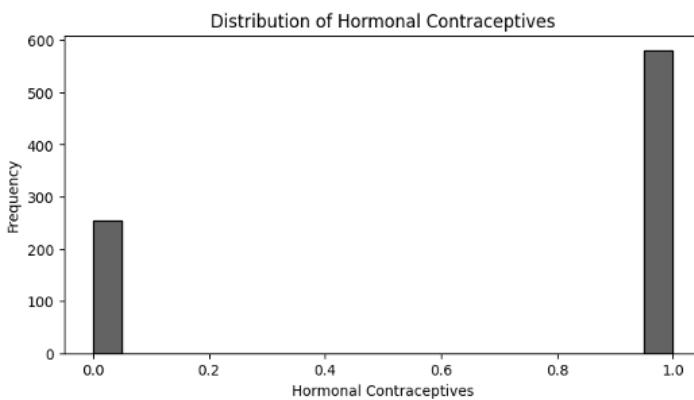
Şekil 3: İlk cinsel ilişkinin gerçekleştiği yaşın dağılım grafiği.



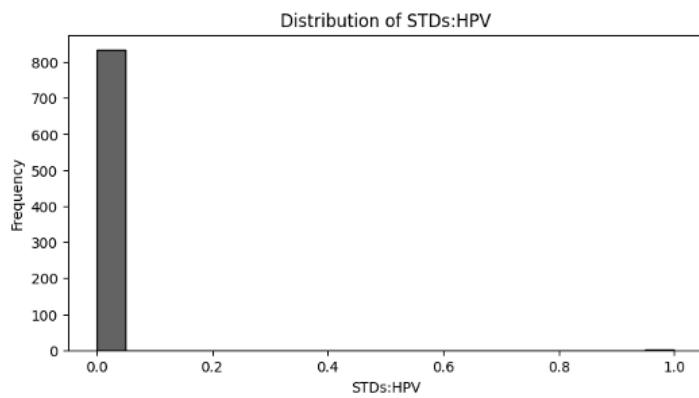
Şekil 4: Geçirilen toplam gebelik sayısının dağılım grafiği.



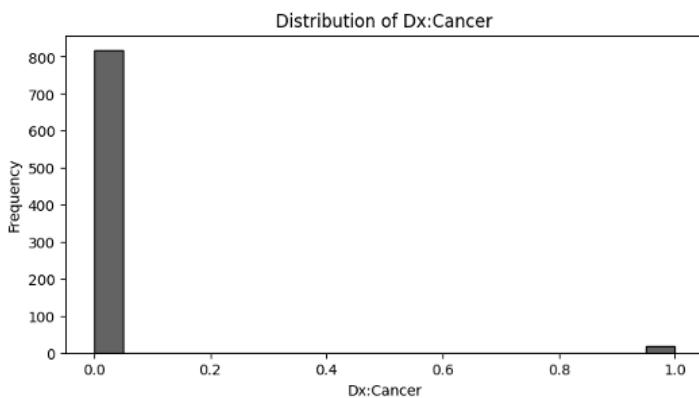
Şekil 5: Sigara içen sayısının dağılım grafiği.



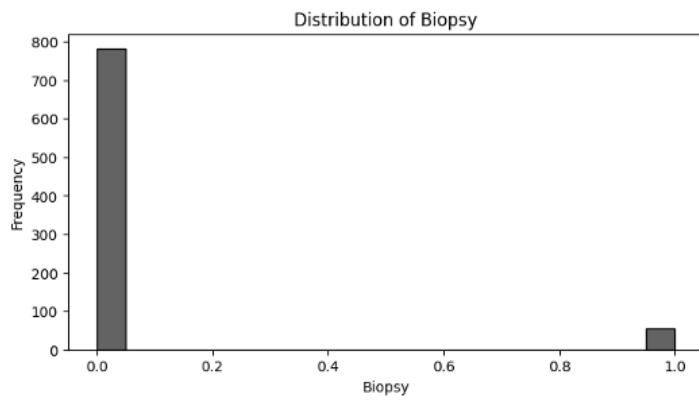
Şekil 6: Doğum kontrol yöntemlerini kullananların dağılım grafiği.



Şekil 7:HPV ile ilişkili geçen hastalık geçmişinin dağılım grafiği.



Şekil 8:DxCancer özelliğinin dağılım grafiği.



Şekil 9: Biyopsi sonuçlarının dağılım grafiği.

3.2 Hiperparametre Optimizasyonu ve Özellik Çıkarımı

Hiperparametre Optimizasyonu: Hiperparametre optimizasyonu, bir makine öğrenmesi modelinin performansını en iyi hale getirmek için modelin eğitilmeden önce ayarlanması gereken parametrelerin (hiperparametrelerin) optimize edilme sürecidir [18]. Makine öğrenmesi algoritmalarının performansı, doğrudan doğru hiperparametre ayarına bağlıdır.

- Hiperparametreler, modelin yapısını ve eğitim sürecini kontrol eder. Örneğin, RF modelinde `ağaç sayısı` (`n_estimators`) ve maksimum derinlik (`max_depth`) gibi parametreler hiperparametrelerdir.
- Hiperparametre optimizasyonunda, yaygın olarak Grid Search ve Random Search gibi yöntemler kullanılır [18] :
 - **Grid Search:** Tüm olası hiperparametre kombinasyonlarını dener ve en iyi sonucu veren kombinasyonu seçer.
 - **Random Search:** Rastgele hiperparametre kombinasyonları arasından denemeler yapar ve en iyisini bulur.

Özellik Çıkarımı: Özellik çıkartımı, bir veri setindeki ham verilerden modelin öğrenmesini kolaylaştıracak daha anlamlı ve özet özelliklerin seçilmesi veya türetilmesi işlemidir. Modelin performansını artırmak için kullanılan bu yöntem gereksiz özellikleri azaltarak modelin daha hızlı çalışmasını sağlar.

Bu çalışmada ise, karşılaştırılan makine öğrenmesi modelleri arasında en iyi performansı veren model seçilerek hiperparametre optimizasyonu ve özellik çıkartımı yöntemleri uygulanacaktır.

3.3 Makine Öğrenmesi Yöntemleri

Makine öğrenmesi (ML), bilgisayarların veri kullanarak belirli bir görevi yerine getirmeyi öğrenmesini sağlayan bir yapay zeka alt alanıdır. Bu yöntemler, geçmiş verilere dayalı olarak tahminlerde bulunma, sınıflandırma yapma ve örüntüleri belirleme gibi görevlerde kullanılır [11][12].

Bu çalışmada üç farklı makine öğrenimi yöntemi kullanılmıştır. SVM, RF ve ANN. Bu algoritmaların çalışma prensipleri aşağıda açıklanmıştır.

Destek Vektör Makineleri (SVM)

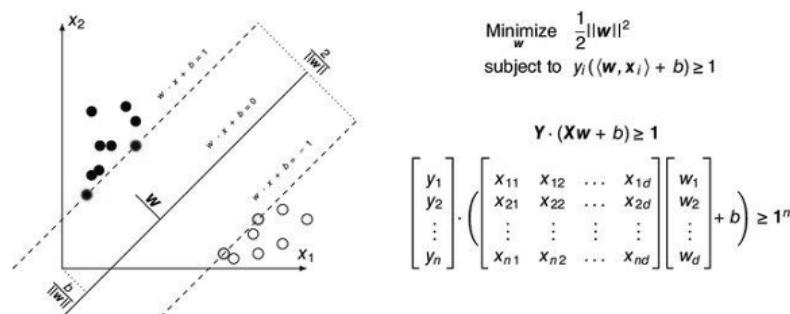
SVM, sınıflandırma ve regresyon analizinde kullanılan güçlü bir algoritmadır. Sınıflandırma problemlerinde SVM, farklı sınıfları ayıran en iyi hiper düzlemi bulmayı hedefler [13]. Karar sınırını optimize etmek için şu matematiksel formül kullanılır [14]:

$$\max_{\mathbf{w}} \frac{2}{|\mathbf{w}|}$$

subject to $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, \forall i$

- \mathbf{w} : Hiper düzlemin ağırlık vektörü,
- b : Bias terimi,
- \mathbf{x}_i : Gözlem noktası,
- y_i : Sınıf etiketi.

SVM, doğrusal olmayan durumlarda da çekirdek fonksiyonları (Kernel functions) kullanarak sınıfları ayırbilir. Çekirdek fonksiyonları, veri noktalarını daha yüksek boyutlu bir uzaya dönüştürür.

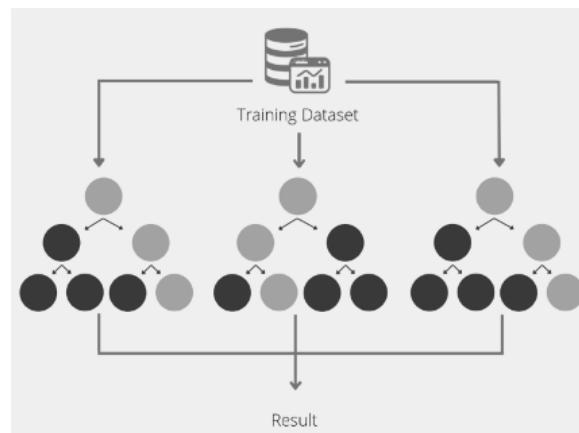


Şekil 10: SVM formülü ve matrix vektör çarpımı (Albert Cano, 2018)

Random Forest (RF)

RF, birden fazla karar ağacından oluşan bir topluluk yöntemidir. Bu algoritma, her bir ağacın bağımsız olarak eğitilmesi ve nihai sınıflandırmanın çoğuluk oyu prensibi ile belirlenmesiyle çalışır. Modelin temel prensipleri şunlardır:

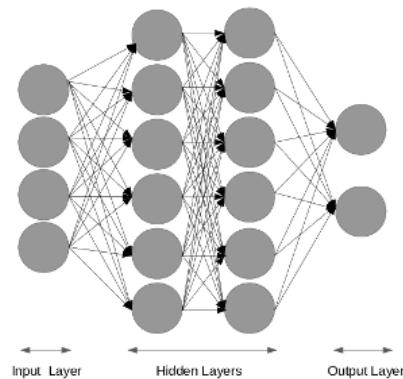
1. Eğitim veri setinden rastgele örneklemle alt veri setleri oluşturulur.
2. Her bir alt veri seti ile bir karar ağacı eğitilir.
3. Nihai sınıflandırma, tüm ağaçların tahminlerinin ortalaması veya çoğuluk oyu ile belirlenir.



Şekil 11: RF çalışma mantığı (What is a Random Forest?, 2022 Database Camp)

Yapay Sinir Ağları (ANN)

ANN, veriler arasındaki karmaşık ilişkileri öğrenmek için kullanılan çok katmanlı bir yapay zeka modelidir. ANN, tablosal veri üzerinde çalışmak için idealdir ve genellikle sınıflandırma, regresyon ve örüntü tanıma gibi görevlerde kullanılır [11][17].



Şekil 12: ANN katmanları

Bir ANN şu temel bileşenlerden oluşur:

1. **Giriş Katmanı:** ANN'in başlangıç noktasıdır ve verileri ağa iletten katmandır.
2. **Gizli Katmanlar:** Veriler arasındaki ilişkileri öğrenir ve ara işlemlerin gerçekleştirildiği katmandır. Bir veya birden daha fazla katman bulunabilir.
3. **Çıkış Katmanı:** Modelin tahminlerini ve sınıflandırma sonuçlarının alındığı katmandır.

Matematiksel olarak bir ANN'in aktivasyonu şu şekilde ifade edilir:

$$z = \sigma(Wx + b)$$

- z: Aktivasyon çıktısı,
- W: Ağırlık matrisi,
- x: Giriş verisi,
- b: Bias terimi,
- σ : Aktivasyon fonksiyonu

4. DENEYSEL SONUÇLAR

Bu çalışmada, rahim ağzi kanseri teşhisinde kullanılan üç farklı makine öğrenmesi yöntemi (SVM, ANN ve RF) değerlendirilmiş ve karşılaştırılmıştır. Modellerin performansı doğruluk (Accuracy), F1 skoru ve ROC-AUC gibi metriklerle analiz edilmiştir. Ayrıca, her bir modelin farklı hedef değişkenler üzerindeki etkisi incelenmiş ve görselleştirilmiştir. Deneysel sürec ve sonuçlar aşağıdaki gibi sunulmaktadır:

4.1 Performans Karşılaştırılması

Modeller, biyopsi(Biyopsy), kanser teşhisini (DxCancer) ve HPV teşhisini (DxHPV) hedef değişkenlerine göre değerlendirilmiştir. Aşağıda her bir modelin metriklerinin yer aldığı tablo bulunmaktadır:

Model	Hedef Değişken	Doğruluk (Accuracy)	F1 Skoru	AUC
SVM	Biopsy	93.63%	0.00	0.97
	Dx:Cancer	97.61%	0.00	0.97
	Dx:HPV	98.01%	0.00	0.05
RF	Biopsy	96.02%	0.64	0.98
	Dx:Cancer	98.01%	0.29	0.98
	Dx:HPV	98.01%	0.00	0.98
ANN	Biopsy	94.42%	0.50	0.90
	Dx:Cancer	97.61%	0.00	0.45
	Dx:HPV	98.01%	0.00	0.50

Tablo 3. Makine öğrenmesi modellerinin hedef değişkenleri üzerinde karşılaştırılması

Tablo 2'ye göre, SVM modeli, genel olarak yüksek doğruluk oranı vermiştir fakat hiçbir F1 skor vermemiştir bunun yanı sıra Biopsy ve DxCancer hedef değişkenlerinde iyi sonuçlar verirken Dx:HPV hedef değişkeninde rastgele tahminlerde bulunmuştur ve AUC değeri çok düşük gelmiştir. RF modeli ise, SVM'e benzer şekilde doğruluk oranları vermiştir SVM'den farklı olarak hedef değişkenleri sırasıyla Biopsy, DxCancer ve DxHPV olmak üzere 0.64, 0.29 ve 0 F1 skorları üretmiştir. Ayrıca, 0.98 AUC değeri ile istikrarını korumuştur. Son olarak ANN modeli ise, tipki SVM ve RF gibi yüksek doğruluk değerleri elde etmiştir. SVM'e farklı olarak Biopsy hedef değişkeninde F1 skor üretetmiştir. Ayrıca, istikrarlı olmasa da 0.90, 0.45 ve 0.50 AUC değerleri üretmiştir.

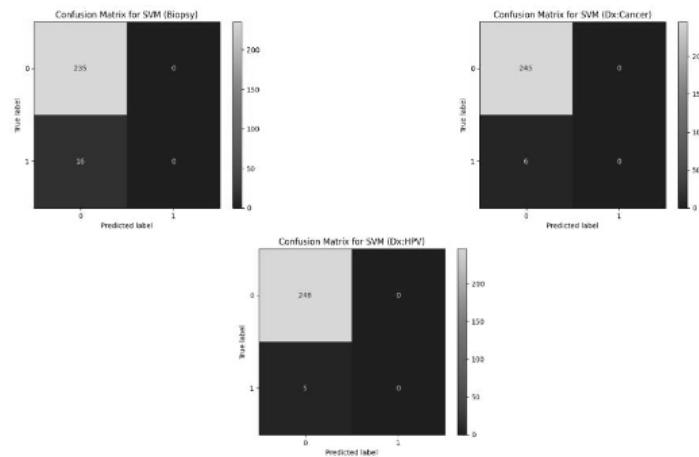
Çalışmanın devamında Tablo 2'deki veriler grafiklerle daha detaylı açıklanmıştır.

4.2 Confusion Matrix (Karışıklık Matrisi)

Modellerin karışıklık matrisleri, farklı hedef değişkenleri üzerinde değerlendirilmiştir. Karışıklık matrisi, bir sınıflandırma modelinin performansını değerlendirmek için kullanılır. Bu matris, modelin tahmin ettiği sınıflar ile gerçek sınıflar arasındaki ilişkiyi görselleştirir ve aşağıdaki dört değeri içerir:

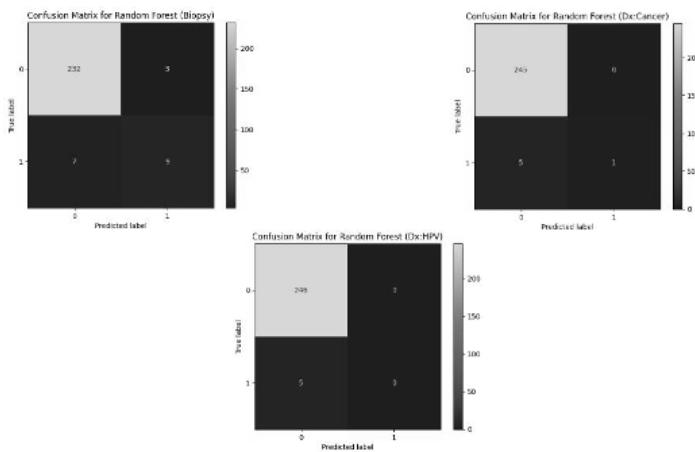
1. **True Positives (TP):** Doğru bir şekilde pozitif olarak tahmin edilen örnekler.
2. **True Negatives (TN):** Doğru bir şekilde negatif olarak tahmin edilen örnekler.
3. **False Positives (FP):** Gerçekte negatif olan ancak pozitif olarak tahmin edilen örnekler (Type I Error).
4. **False Negatives (FN):** Gerçekte pozitif olan ancak negatif olarak tahmin edilen örnekler (Type II Error).

Modellerin hedef değişken üzerindeki karışıklık matrisleri bu başlık altındaki şekillerde açıklamalarıyla birlikte gösterilmektedir.



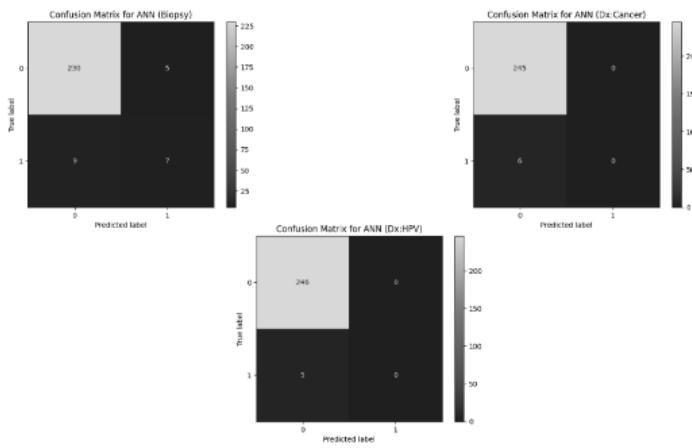
Şekil 13: SVM için karışıklık matris

Şekil 13'de SVM için oluşturulan karışıklık matrisleri, tüm hedef değişkenlerde modelin pozitif sınıfı doğru bir şekilde tahmin edemediğini göstermektedir. Biopsy, Dx:Cancer ve Dx:HPV hedef değişkenlerinde tüm pozitif örnekler yanlış negatif olarak tahmin edilmiş, yalnızca negatif sınıf üzerinde yüksek doğruluk elde edilmiştir. Bu durum, sınıf dengesizliği ve modelin pozitif sınıfı öğrenememe problemine işaret etmektedir.



Şekil 14: RF için karışıklık matris

Şekil 14'de RF, Biopsy hedef değişkeninde genel olarak iyi bir performans göstermiştir. Ancak Dx:Cancer ve Dx:HPV hedef değişkenlerinde pozitif sınıfta tahminler sınırlı kalmıştır. Bu sonuçlar, modelin pozitif sınıfı tanımda zorlandığını ve veri setindeki dengesizliğin pozitif tahmin performansını etkileyebileceğini göstermektedir. Özellikle Dx:HPV için, modelin pozitif sınıfta hiç doğru tahmin yapamaması önemli bir eksikliktir.



Şekil 15: ANN için karışıklık matrisi

Şekil 15'de ANN, Biopsy hedef değişkeninde pozitif ve negatif sınıflar arasında kısmen dengeli bir performans göstermiştir, ancak Dx:Cancer ve Dx:HPV hedef değişkenlerinde pozitif sınıfta tamamen başarısız olmuştur. Bu durum, ANN'in dengesiz veri setinde pozitif sınıfı öğrenme zorluğunu yaşadığını göstermektedir.

Bu sonuçlar, sınıf dengesizliği gibi problemler karşısında RF'in daha dayanıklı olduğunu ve rahim ağzı kanseri teşhisinde en uygun model olarak tercih edilmesi gerektiğini göstermektedir.

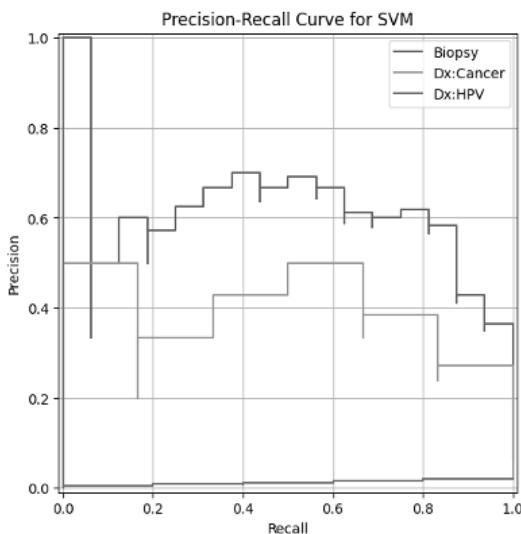
4.3 Precision-Recall Eğrileri

Modellerin, hassasiyet (Precision) ve duyarlılık (Recall) arasındaki ilişkiler Şekil 16,17 ve 18'de gösterilmiştir. Tablo 2.'de sunulan F1 Skoru hassasiyet ve recall'un dengeli bir ölçütüdür. Hassasiyet, duyarlılık ve F1 skor metriklerinin formülleri aşağıdaki gibi hesaplanmaktadır:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

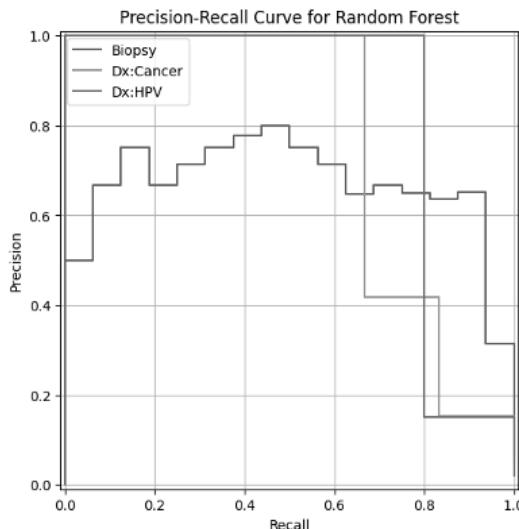
$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}$$

$$F1 \text{ Skoru} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$



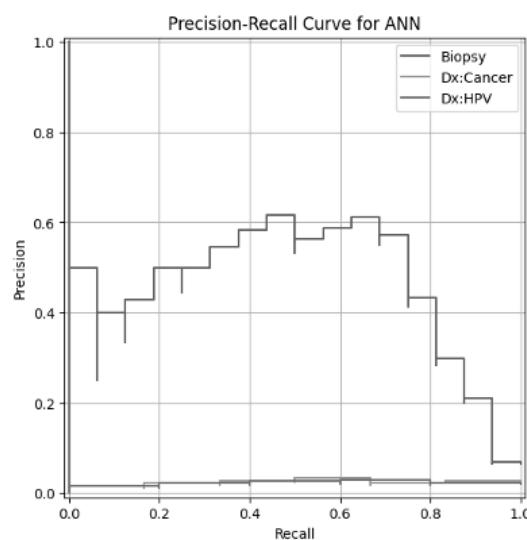
Şekil 16: SVM için Precision-Recall Eğrisi

Şekil 16'de sunulan SVM için **Precision-Recall eğrisi**, Biopsy hedef değişkeninde Precision ve Recall arasında orta seviyede bir ilişki sergilemiştir. Eğri, belirli noktalarda Precision'ın düşüğünü göstermesine rağmen genel olarak dengeli bir performans sunmaktadır. Dx:Cancer hedef değişkeninde ise Precision ve Recall değerleri oldukça düşük seviyelerde kalmış, modelin bu değişken için sınırlı bir ayırtırma yeteneğine sahip olduğunu göstermiştir. Dx:HPV hedef değişkeninde ise Precision ve Recall değerleri neredeyse sıfır seviyesindedir. Bu durum, SVM'in HPV teşhisinde başarısız olduğunu ve tahminlerinin rastgele olduğunu ortaya koymaktadır. Genel olarak, SVM, Biopsy ve Dx:Cancer değişkenlerinde kısmen etkili bir performans sergilerken, Dx:HPV için tatmin edici bir sonuç üretmemiştir.



Şekil 17: RF için Precision-Recall Eğrisi

Şekil 17'de sunulan RF için Precision-Recall eğrisi, tüm hedef değişkenlerde güçlü bir ilişki göstermiştir. Biopsy hedef değişkeninde Precision ve Recall yüksek seviyelerde dengeli bir şekilde ilerlemiş, modelin güçlü bir sınıflandırma performansı sunduğunu göstermiştir. Dx:Cancer hedef değişkeninde de benzer bir başarı gözlemlenmiştir. Dx:HPV hedef değişkeni için eğri, modelin bu hedefte de oldukça yüksek bir performansa sahip olduğunu ortaya koymuştur.



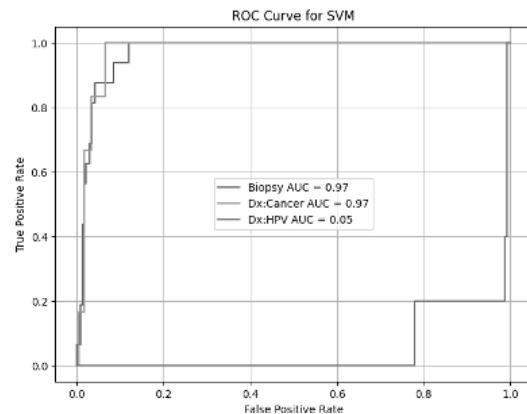
Şekil 18: ANN için Precision-Recall Eğrisi

Şekil 18'de sunulan ANN için Precision-Recall eğrisi, Biopsy hedef değişkeninde orta seviyede bir performans göstermiştir. Precision ve Recall değerleri bazı aralıklarda

istikrarsızlık göstermiştir, ancak genel olarak kabul edilebilir bir sonuç sunmuştur. Dx:Cancer hedef değişkeninde ise Precision ve Recall oldukça düşük seviyelerde kalmış ve modelin sınırlı bir başarısı olduğunu göstermiştir. Dx:HPV hedef değişkeninde Precision ve Recall değerleri sıfır seviyesine yakın olup, ANN'in bu hedef değişkeni üzerinde başarısız olduğunu göstermiştir. ANN, yalnızca Biopsy hedef değişkeninde makul bir performans sunarken diğer değişkenlerde yetersiz kalmıştır.

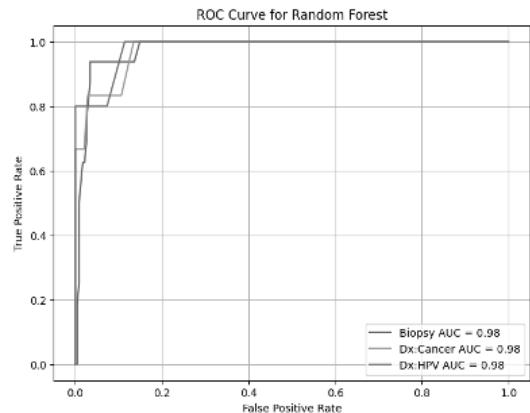
4.4 ROC Eğrileri

Modellerin ROC eğrileri, farklı hedef değişkenler için aşağıdaki Şekil 19, 20 ve 21'de sunulmaktadır. Bu eğriler, modelin pozitif sınıfı doğru bir şekilde ayırmayı performansını görselleştirmektedir.



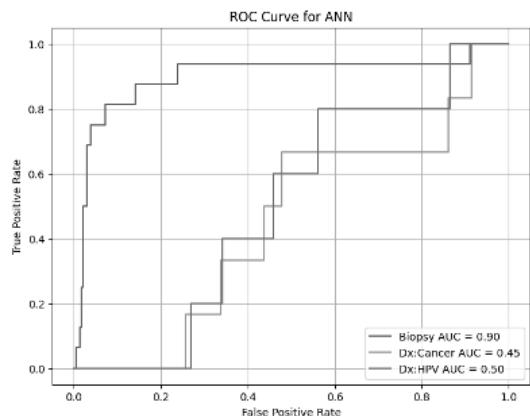
Şekil 19 SVM için ROC eğrisi

Şekil 19'de sunulan SVM için ROC eğrisi, Biopsy hedef değişkeninde AUC (Alan Altında Kalan Alan) değeri 0.97 ile oldukça yüksektir. Bu SVM'in biyopsi sınıflandırmasında güçlü bir ayırtma yeteneğine sahip olduğunu göstermektedir. Dx:Cancer hedef değişkeninde yine 0.97'dir. Model, kanser teşhisinde biyopsiye benzer bir doğruluk göstermiştir. Dx:HPV hedef değişkeninde ise AUC değeri 0.05 gibi çok düşük bir değerde kalmıştır. Bu SVM'in HPV teşhisinde oldukça düşük bir performans gösterdiğini ve rastgele bir tahminde bulunduğunu göstermiştir. SVM, biyopsi ve kanser teşhisinde etkili ancak HPV teşhisinde başarısız performans sergilemiştir.



Şekil 20: RF için ROC eğrisi

Şekil 20'de sunulan RF için ROC eğrisi, Biopsy hedef değişkeninde 0.98 ile çok yüksektir. Bu RF'in biopsy sınıflandırmasında oldukça başarılı olduğunu göstermektedir. Dx:Cancer hedef değişkeninde ise AUC değeri yine 0.98'dir. Model, kanser teşhisinde oldukça başarılıdır. Dx:HPV hedef değişkeninde ise AUC değeri 0.98 olarak görülmektedir ve bu modelin, HPV teşhisinde de tutarlı ve yüksek bir performans gösterdiğini göstermektedir. RF, tüm hedef değişkenlerde oldukça yüksek bir doğrulukla çalışmıştır ve en dengeli performansı sergilemiştir.



Şekil 21: ANN için ROC eğrisi

Şekil 21'de sunulan ANN için ROC eğrisi, Biopsy hedef değişkeninde AUC değeri 0.90 ile kabul edilebilir bir düzeydedir. Bu ANN'in, biyopsi sınıflandırmasında başarılı olduğunu göstermektedir. Dx:Cancer hedef değişkeninde AUC değeri 0.45 gibi düşük bir seviyededir. Bu ANN'in kanser teşhisinde çok sınırlı bir performansa sahip olduğunu işaret etmektedir. Dx:HPV hedef değişkeninin ise AUC değeri 0.50'dir. Bu değer, modelin rastgele bir tahmin

seviyesinde olduğunu göstermektedir. ANN, biyopsi için yeterince iyi bir performans sergilemiş olsa da kanser ve HPV teşhisinde düşük bir başarı göstermiştir.

4.5 Random Forest Hiperparametre Optimizasyonu ve Özellik Önem Sıralaması Sonuçları

Bu başlıklarda karşılaştırılan modeller arasında en iyi performansı veren RF modeli kullanılarak hiperparametre optimizasyonu ve özellik çıkarımı yöntemleri uygulanacaktır. RF modeli için hedef değişken Biopsy olarak belirlenmiştir.

Hiperparametre Optimizasyonu

Hiperparametre optimizasyonunun iki yaygın çeşidi vardır. Bunlar Grid Search ve Random Search'dur. Hiperparametre Optimizasyonu ve Özellik Çıkarımı başlığı altında bu iki çeşidin kullanım amacı açıklanmıştır. Çalışmanın devamında, Grid Search optimizasyon yöntemi kullanılacaktır.

Grid Search yöntemi sonucunda, RF modeli için en iyi hiperparametre kombinasyonu şu şekilde belirlenmiştir:

Parametre	Değer
max_depth	None
min_samples_leaf	1
min_samples_split	10
n_estimators	50

Tablo 4. Seçilen hiperparametreler ve değerleri

Tablo 3'de verilen parametreler, modelin sınıflandırma performansını optimize etmek için seçilmiş ve bu parametrelerle yeniden eğitilen model, sınıflandırma görevinde en yüksek performansı göstermiştir.

Random Forest Performans Metrikleri

Çalışmanın sonuç kısımlarında belirleyici olan RF modelinin hiperparametre optimizasyonunda hedef değişken olan Biopsy değişkeni üzerindeki performans metrikleri Tablo 5'de sunulmaktadır. Tablo 5 incelendiğinde, RF modeli Biopsy değişkeni üzerinde 0.96 doğruluk, 0.69 F1 skor ve 0.98 AUC değerlerini elde etmiştir. Elde edilen bu sonuçlar

neticesinde özellik çıkarımı ve özellik önem sıralaması belirlenirken ciddi katkıda bulunacaktır.

Performans Metrikleri	Değer
Doğruluk	None
F1 Skor	1
AUC	10

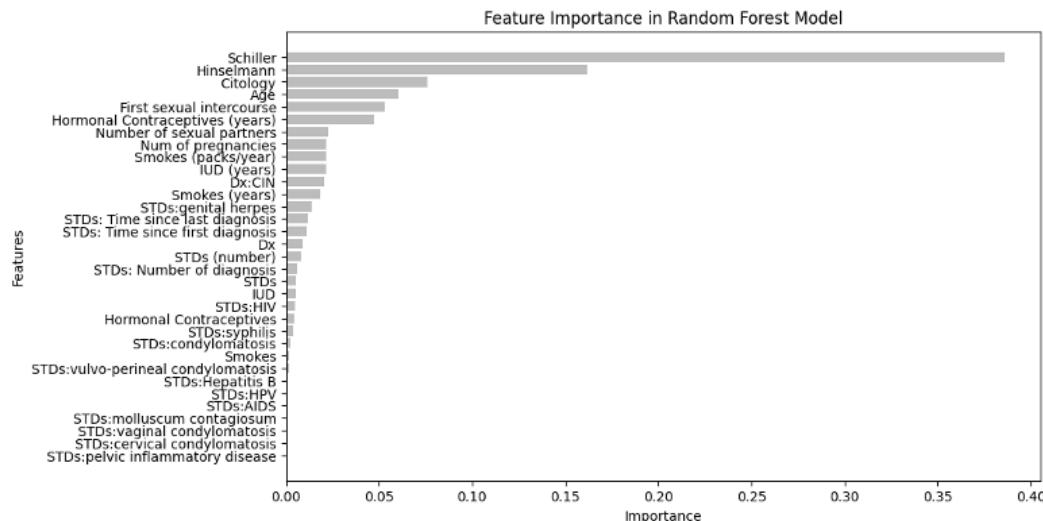
Tablo 5. Seçilen hiperparametreler ve değerleri

Özellik Çıkarımı ve Özellik Önem Sıralaması

RF modeli, hedef değişken olan Biopsy için tahmin yaparken kullandığı özelliklerin önem derecelerini belirlemiştir. Tablo 5'de en önemli 10 özellik ve önem skorları sunulmuştur:

Özellik	Önem Skoru
Schiller Testi	0.385655
Hinselmann Testi	0.161460
Sitoloji (Cytology)	0.075954
Yaş (Age)	0.060403
İlk Cinsel Deneyim (First sexual intercourse)	0.053016
Hormonal Kontraseptif Kullanımı (Hormonal Contraceptives - years)	0.047056
Cinsel Partner Sayısı (Number of sexual partners)	0.022869
Hamilelik Sayısı (Num of pregnancies)	0.021784
Sigara Kullanımı (Smokes - packs/year)	0.021683
Rahim İçi Araç Kullanım Süresi (IUD - years)	0.021577

Tablo 6. Biopsy hedef değişkenine göre özellik önem sıralaması.



Şekil 22: Random Forest Özellik Önem Sıralaması

RF modeline göre, Tablo 5'e ve Şekil 22'ye bakıldığından sınıflandırmada en etkili özellik Schiller Testi olmuştur. Bu özellik, rahim ağzı kanseri teşhisinde en güçlü belirleyici faktör olarak öne çıkmıştır. Hinselmann Testi ve Sitoloji de model için önemli özellikler olarak belirlenmiştir. Yaş, ilk cinsel deneyim yaşı ve hormonal kontraseptif kullanımı gibi özellikler de modelde belirli bir etkiye sahiptir. Ancak, klinik testlerinden sonra ikinci derecede önemlidir.

5. SONUÇLAR

Bu çalışmada rahim ağzı kanseri teşhisinde SVM, RF ve ANN modelleri kullanılarak performans karşılaştırımları gerçekleştirilmiştir. Hedef değişken olarak Biopsy, DxCancer ve DxHPV belirlenmiş, modellerin sınıflandırma performansları çeşitli metrikler ve görsellerle değerlendirilmiştir. Son olarak, hiperparametre optimizasyonu sonrası RF modeli seçilmiş ve özellik önem sıralaması analiz edilmiştir. Bu analiz, klinik testlerin (Schiller, Hinselmann ve Sitoloji) rahim ağzı kanseri teşhisinde belirleyici faktörler olduğunu ortaya koymaktadır. Diğer yandan, demografik faktörler ve hormonel etkiler de belirli ölçüde katkı sağlamaktadır. RF modeline göre elde edilen sonuçlar, modelin yüksek doğruluk ve ayırtırma performansı sergilediğini göstermektedir. Özellikle AUC değerinin 0.98 olması, modelin pozitif ve negatif sınıfları başarıyla ayırdığını kanıtlamaktadır. Ancak F1 skorunun 0.69 seviyesinde kalması, modelin küçük veri setindeki sınıf dengesizliğinden etkilendiğini göstermektedir.

KAYNAKLAR

- [1] M. A. Devi, S. Ravi, J. Vaishnavi, and S. Punitha, “Classification of Cervical Cancer Using Artificial Neural Networks,” *Procedia Computer Science*, vol. 89, pp. 465–472, 2016, doi: 10.1016/J.PROCS.2016.06.105.
- [2] F. X. Bosch and S. de Sanjosé, “The epidemiology of human papillomavirus infection and cervical cancer,” *Disease Markers*, vol. 23, no. 4, pp. 213–227, 2007, doi: 10.1155/2007/914823.
- [3] D. Saslow, D. Solomon, H. W. Lawson, M. Killackey, et al., “American Cancer Society, American Society for Colposcopy and Cervical Pathology, and American Society for Clinical Pathology screening guidelines for the prevention and early detection of cervical cancer,” *CA: A Cancer Journal for Clinicians*, vol. 62, no. 3, pp. 147–172, 2012, doi: 10.3322/CAAC.21139.
- [4] T. Wu, E. Lucas, F. Zhao, P. Basu, and Y. Qiao, “Cancer Biology & Medicine,” *Cancer Biology & Medicine*, Sep. 2024, doi: 10.20892/j.issn.2095-3941.2024.0198.
- [5] M. Glučina, A. Lorencin, N. Andelić, and I. Lorencin, “Cervical Cancer Diagnostics Using Machine Learning Algorithms and Class Balancing Techniques,” *Applied Sciences*, vol. 13, no. 2, p. 1061, 2023, doi: 10.3390/APP13021061.
- [6] H. Li et al., “Large-scale cervical precancerous screening via AI-assisted cytology whole slide image analysis,” *arXiv preprint arXiv:2407.19512v1*, 2024.
- [7] M. M. Rahaman et al., “DeepCervix: A Deep Learning-based Framework for the Classification of Cervical Cells Using Hybrid Deep Feature Fusion Techniques,” *Computers in Biology and Medicine*, vol. 136, 2021, doi: 10.1016/j.combiomed.2021.104649.
- [8] H. A. Sarhangi, D. Beigifard, E. Farmani, and H. Bolhasani, “Deep Learning Techniques for Cervical Cancer Diagnosis based on Pathology and Colposcopy Images,” *arXiv preprint arXiv:2310.16662v1*, 2023.
- [9] K. Poudel et al., “AI-Assisted Cervical Cancer Screening,” *arXiv preprint arXiv:2403.11936v2*, 2024.

- [10] M. Rahimi, A. Akbari, F. Asadi, and H. Emami, “Cervical cancer survival prediction by machine learning algorithms: a systematic review,” *BMC Cancer*, vol. 23, no. 1, pp. 1–10, 2023, doi: 10.1186/S12885-023-10808-3.
- [11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
- [12] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, 2009.
- [13] C. Cortes and V. Vapnik, “Support-Vector Networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995, doi: 10.1007/BF00994018.
- [14] C. J. C. Burges, “A Tutorial on Support Vector Machines for Pattern Recognition,” *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.
- [15] L. Breiman, “Random Forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
- [16] D. R. Cutler et al., “Random Forests for Classification in Ecology,” *Ecology*, vol. 88, no. 11, pp. 2783–2792, 2007.
- [17] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.
- [18] J. Bergstra and Y. Bengio, “Random search for hyper-parameter optimization,” *Journal of Machine Learning Research*, vol. 13, pp. 281–305, 2012.



**PARK
YAPILMAZ**