



ONDOKUZ MAYIS ÜNİVERSİTESİ

EĞİTİM BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ

ANABİLİM DALI

**5.SINIF ÖĞRENCİLERİNE PROGRAMLAMA ÖĞRETİMİNDE
HEDEFE DAYALI SENARYO KULLANIMININ ETKİSİ ve
ÖĞRENCİ GÖRÜŞLERİ**

Emine KANDİN

Danışman

Dr. Öğr. Üyesi Emine ŞENDURUR

YÜKSEK LİSANS TEZİ

Temmuz, 2019

TELİF HAKKI

2547 Sayılı Yükseköğretim Kanunu Ek Madde 40 hükümleri çerçevesinde (Ek:22/2/2018-7100/10 md.) “*Lisansüstü tezler yetkili kurum ve kuruluşlar tarafından gizlilik kararı alınmadıkça, bilime katkı sağlamak amacıyla Yükseköğretim Kurulu Ulusal Tez Merkezi tarafından elektronik ortamda erişime açılır.*”

Araştırmacılar tezlerin tamamı veya bir bölümünü yazarın izni olmadan ticari veya mali kazanç amaçlı kullanamaz, yayınlamaz, dağıtamaz ve kopyalayamaz. Ulusal Tez Merkezi Web Sayfasını kullanan araştırmacılar, tezlerden bilimsel etik ve atıf kuralları çerçevesinde yararlanırlar.

YAZARIN

Adı : Emine

Soyadı : KANDİN

Bölümü : Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı

İmza :

Teslim Tarihi : 28.08.2019

TEZİN

Türkçe Adı : 5.Sınıf Öğrencilerine Programlama Öğretiminde Hedefe Dayalı Senaryo Kullanımının Etkisi ve Öğrenci Görüşleri

İngilizce Adı : The Effect Of Goal-Based Scenario Used for Programming Education of 5th Graders and Students’ Opinions

ETİK İLKELERE UYGUNLUK BEYANI

Tez yazma sürecinde bilimsel ve etik ilkelere uyduđumu, yararlandıđım tüm kaynakları kaynak gösterme ilkelerine uygun olarak kaynakçada belirttiđimi ve bu bölümler dışındaki tüm ifadelerin şahsıma ait olduđunu beyan ederim.

Yazar Adı Soyadı: Emine KANDİN

İmza:

KABUL VE ONAY

Emine KANDİN tarafından hazırlanan “**5.Sınıf Öğrencilerine Programlama Öğretiminde Hedefe Dayalı Senaryo Kullanımının Etkisi ve Öğrenci Görüşleri**” adlı tez çalışması aşağıdaki jüri tarafından oy birliği/ oy çokluğu ile Ondokuz Mayıs Üniversitesi **Bilgisayar ve Öğretim Teknolojileri Eğitimi** Anabilim Dalı’nda Yüksek Lisans tezi olarak kabul edilmiştir.

Danışman: Dr. Öğr. Üyesi Emine ŞENDURUR

Bilgisayar ve Öğretim Teknolojileri Eğitimi, Ondokuz Mayıs Üniversitesi.....

Başkan: Dr. Öğr. Üyesi Ömer Faruk İSLİM

Bilgisayar ve Öğretim Teknolojileri Eğitimi, Ahi Evran Üniversitesi

Üye: Dr. Öğr. Üyesi Rezan YILMAZ

Ortaöğretim Fen ve Matematik Alan Eğitimi, Ondokuz Mayıs Üniversitesi.....

Üye: Dr. Öğr. Üyesi Emine ŞENDURUR

Bilgisayar ve Öğretim Teknolojileri Eğitimi, Ondokuz Mayıs Üniversitesi.....

Bu tezin **Bilgisayar ve Öğretim Teknolojileri Eğitimi** Anabilim Dalı’nda Yüksek Lisans tezi olması için şartları yerine getirdiğini onaylıyorum.

Tarihi: __/__/____

Prof. Dr. Ali ERASLAN

Eğitim Bilimleri Enstitüsü Müdürü

(İmza ve Mühür)



Canım Aileme

TEŞEKKÜRLER

Yüksek Lisans Eğitimim ve Tez araştırma çalışmalarım sürecinde derin bilgi, birikim ve tecrübeleri ile bana yol gösterici olan, çalışma disiplini ve özverisiyle çalışmalarım da her daim destek olan ve her daim ilgilenen, akademik çalışmalarıyla bana model olan her zaman kendime örnek aldığım çok değerli ve benim için çok kıymetli olan Tez Danışmanım Dr. Öğr. Üyesi Emine ŞENDURUR'a (Ondokuz Mayıs Üniversitesi) sonsuz teşekkürlerimi ve saygılarımı sunuyorum.

Çalışmalarım boyunca yardımlarını esirgemeyen, yüksek lisans eğitimim boyunca bilgi ve deneyimleriyle model olan ve her daim destek olan saygıdeğer hocam Dr. Öğr. Üyesi Polat ŞENDURUR'a (Ondokuz Mayıs Üniversitesi) ve yanımda olan çok değerli hocalarıma ve değerli arkadaşlarıma ayrı ayrı teşekkür ediyorum. Benim bugünlere gelmemde büyük pay sahibi olan, maddi ve manevi desteklerini hiçbir zaman esirgemeyen, her daim yanımda olan biricik ailem, çok değerli Annem, Babam, Kardeşlerime ve her zaman yanımda olan ve manevi desteklerini hiçbir zaman esirgemeyen Uygur GÖK'e sonsuz saygı, sevgi ve teşekkürlerimi sunuyorum. Ayrıca çalışmalarım sürecinde bana destek olan arkadaşım Elvan DÖNMEZ'e teşekkür ediyorum. Son olarak tezimin uygulama aşamasında önemli katkıları bulunan, araştırma sürecinde istekli katılarak büyük fedakârlıkla yaptığı uygulamalarla ve yapılan görüşmelerle çalışmama destek veren değerli öğrencilerime de çok teşekkür ediyorum.

**5.SINIF ÖĞRENCİLERİNE PROGRAMLAMA ÖĞRETİMİNDE
HEDEFE DAYALI SENARYO KULLANIMININ ETKİSİ ve
ÖĞRENCİ GÖRÜŞLERİ**

Yüksek Lisans Tezi

Emine KANDİN

ONDOKUZ MAYIS ÜNİVERSİTESİ

EĞİTİM BİLİMLERİ ENSTİTÜSÜ

Temmuz, 2019

ÖZ

Bu çalışmanın amacı, Senaryo tabanlı öğrenme yaklaşımının ilköğretim 5. Sınıf öğrencilerinin programlama performanslarının gelişimi üzerindeki etkisini belirlemektir. Bu çalışmada senaryo tabanlı öğrenme modelinin, hiç programlama becerisi olmayan ve motivasyonu düşük olan öğrencilerin programlama becerilerinin geliştirilmesi üzerinde etkisini belirlemek üzere Scratch görsel programlama aracı kullanılmıştır. Bu çalışmada, Scratch ile kodlama öğretimi yapıldıktan sonra uygulama esnasında hedefe dayalı senaryolar kullanılarak öğrencilerin haftalık ürünlerinin gelişimi incelenmiştir. Katılımcılar, daha önce kodlamayla tanışmamış iki gruptan oluşmaktadır. Her iki grubun ders anlatım süreci aynıken, uygulamada farklılaşmaktadırlar. Her haftanın senaryo akışı hayal ürünü bir karakter üzerinden gitmektedir. Uygulamanın ilk 2 haftalık bölümünde algoritmik düşünme becerilerinin geliştirilmesi, programlama mantığının kazandırılması ve temel programlama yapılarının öğretilmesi amaçlanmıştır. 10 haftalık uygulamada öğrencilerden, “animasyonlu davetiye, piyano oyunu, labirent oyunu, hediye toplama oyunu, bilgi yarışması oyunu, balon patlatma oyunu, matematik 4 işlem resim bulmaca oyunu, tuğla kırma oyunu, trafik kuralları oyunu, mevsimler oyunu” tasarımları istenmiştir. Bu ürünler haftalık ürünlere ve kazanımlara göre oluşturulan kriterler doğrultusunda değerlendirilmiştir. Ayrıca, her uygulama geliştirme aşaması araştırmacılardan biri tarafından gözlemlenmiş ve alan notları alınmıştır. Nitel ve nicel araştırma

VII

metotlarının birlikte kullanıldığı bu çalışmada, veri toplama aracı olarak performanslarına yönelik programlama ile ilgili temel becerileri yoklayan ön-test ve son-test uygulanmıştır. Bunun yanı sıra, öğrencilerin yaptığı tasarım ve kodlamalarını gösteren ekran kayıtları alınmıştır. Ayrıca öğrencilerin performanslarına yönelik yarı yapılandırılmış görüşmeler yapılmıştır. Çalışmanın sonuçlarına göre; öğrencilerle yapılan görüşmeler, içerik analizi yöntemiyle analiz edilmiş olup, bu görüşmeler Scratch programlamaya yönelik algıları, hedefe dayalı senaryoya ve çalışma kâğıtlarına yönelik algıları ve öz değerlendirme temaları altında kategorize edilmiştir. Bu analiz sonuçlarına göre; Scratch programlamaya yönelik öğrencilerin düşüncelerinde Scratch programının kolaylık yada zorluk algısı; karakterler, karakterlerin hareket etme özelliği, dinamik (hareketli) öğeler gibi görsellikle ilgili faktörler, mutluluk, hayal gücünü destekleme, özgüven, tasarlama boyutlarında motivasyon edici faktörler, sürükle bırak yönteminin kolaylaştırıcı etkisi ve kodlama için ayrılan süreden kazandırması, oyunu tasarladıktan sonra etkileşimli oyun ortamını sağlaması ve oyun imkanı sağlaması gibi kodlamayla ilgili boyutlarda olumlu (hoşa giden boyutlar), bilgisayarla ilgili teknik problemler, bilgisayarın donması, yada programın kasma gibi donanımsal hatalar, kodlamanın kompleks ve zor olarak algılanması, kod bloklarının karıştırılması, yapamadığında olumsuz duygular hissetme, uzun zaman gerektiren kodlama yapısına sahip olması ve tasarımda zorlanma gibi olumsuz (hoşa gitmeyen boyutlar) ele alınmıştır. Hedefe dayalı senaryoya yada çalışma kağıdına yönelik algılarında ise; senaryonun içeriği ve senaryonun adım adım yönergelerle yol gösterici olması gibi anlama boyutuna etki eden faktörler; senaryonun devamlılık göstermesi, senaryoya aidiyet duygusu, başarı duygusu ve motivasyon gibi duyuşsal boyuta etki eden faktörler; esneklik boyutunda öğrencilerin kendi düşüncelerine göre tasarım yapabilmesi, çalışma kağıtlarının öğrencinin tasarımını kolaylaştırıcı yada sınırlandırıcı etkisi gibi faktörler analiz edilmiştir. Ayrıca öz değerlendirmeye yönelik algılarında ise, sürenin değerlendirilmesi, öğrencilerin performanslarının değerlendirilmesi, motivasyon, özgüven, başarı ve mutluluk gibi duyuşsal faktörler, kendi değerlendirmelerine göre Scratch performans puanları ve verilen senaryoya bağlılık gibi boyutların her iki grupta farklı etkileri ortaya çıkmıştır.

Anahtar Kelimeler : Durum tabanlı öğrenme, Senaryo tabanlı öğrenme, Hedefe dayalı senaryolar, kodlama, kodlama eğitimi, kodlayarak öğrenme, Scratch programlama, görsel programlama, geri dönüt, motivasyon

Sayfa Sayısı : 225

Danışman : Dr. Öğr. Üyesi Emine ŞENDURUR

THE EFFECT OF GOAL-BASED SCENARIO USED FOR PROGRAMMING EDUCATION OF 5TH GRADERS & STUDENTS' OPINIONS

MS Thesis

Emine KANDİN

ONDOKUZ MAYIS UNIVERSITY

GRADUATE SCHOOL OF EDUCATIONAL SCIENCES

July, 2019

ABSTRACT

The aim of this study is to find out the effects of scenario-based learning approach on the development of fifth grade students' programming performance. In this study, Scratch visual programming tool was used to find out the effect of scenario-based learning approach on programming skills of students who did not have any programming experience and who had low motivation. After coding was taught, goal based scenarios were used during practice and the development in students' weekly products was examined. The participants consisted of two groups, which never met coding before. While the lecturing process of both groups was the same, practice process was different. The scenario flow of each week occurred through an imaginary character. In the first two weeks of practice, the aim was to develop algorithmic thinking skills, to teach the logic of programming and to teach the basic programming structures. Within the 10-week long practice, the students were asked to design "animated invitation, piano game, labyrinth game, gift collection game, quiz show, balloon blasting game, mathematics 4 operations picture finding game, brick breaking game, traffic rules game and seasons game". These products were assessed in line with criteria formed according to weekly products and attainments. In addition, each practice development stage was observed by one of the researchers and field notes were taken. In the present study in which qualitative and quantitative research methods

were used together, pre-test and post-test were given to test the basic skills of students about programming as data collection tools. In addition, screenshot images of the design and codes students' made were taken. Semi-structured interviews were made to assess students' performances. According to the results of the study, interviews made with the students were analyzed by using content analysis method and these interviews were categorized under the themes of students' perceptions about Scratch programming, goal-based scenario or worksheets, self-assessment. According to these results, positive (liked by students) dimensions such as perceptions of whether Scratch program is easy or difficult; factors such as characters, characters' movement characteristics, factors related with visuality such as dynamic (moving) element; motivating factors such as happiness, supporting imagination, self-confidence and designing dimensions; code-related dimensions such as the facilitation effect of drag and drop method and saving from the time for coding, interactive game environment after designing the game and providing opportunity to play and negative (not liked by students) dimensions such as technical problems about the computer, perceiving coding as complex and difficult, confusion about code blocks, negative feelings when one cannot succeed, coding structure taking too long and having difficulties in design were discussed. In terms of perceptions about goal-based scenario or worksheet, factors influencing the comprehension dimension such as content of the scenario and the scenario guiding with step by step instructions; factors influencing the cognitive dimension such as the scenario showing continuity, feelings of belonging for the scenario, feelings of accomplishment and motivation; factors such as students' being able to design according to their own thoughts, and worksheets' easing or limiting effect according to students' thoughts were discussed. In terms of students' perceptions about self-assessment, different effects were found in both groups in terms of the assessment of time, students' performances, cognitive factors such as motivation, self-confidence, success and happiness, Scratch performance scores according to students' own assessments and being loyal to given scenario.

Key Words : Case-based learning, scenario-based learning, goal-based scenarios, coding, coding education, learning coding, Scratch programming, visual programming, feedback, motivation

Number of Pages : 225

Advisor : Asst. Dr. Prof. Emine SENDURUR

İÇİNDEKİLER

TELİF HAKKI.....	II
ETİK İLKELERE UYGUNLUK BEYANI.....	III
KABUL VE ONAY	IV
TEŞEKKÜRLER	VI
ÖZ.....	VII
ABSTRACT	IX
İÇİNDEKİLER	XI
BİRİNCİ BÖLÜM.....	1
I. GİRİŞ.....	1
1.1 Araştırmanın Konusu, Amacı ve Önemi.....	8
1.2 Araştırma Soruları.....	10
1.3 Terimler.....	10
İKİNCİ BÖLÜM	12
II. KURAMSAL ÇERÇEVE.....	12
2.1 Programlama	13
2.2 Metin Temelli Programlama	14
2.3 Blok Temelli Programlama	16
2.3.1 Programlama Öğretiminde Kullanılan Araçlar	18
2.3.2 Scratch Programlama	25
2.3.3 Scratch Programlama ile İlgili Yapılmış Çalışmalar	26
2.4 Çocuklara Yönelik Programlama (4P).....	27
2.4.1 Project (Proje).....	28
2.4.2 Peers (Akranlar)	28
2.4.3 Passion (İlgi ve Motivasyon)	29
2.4.4 Play (Oyun).....	30
2.5 Programlama Öğretiminin Faydaları/Avantajları.....	31
2.6 Eğitim Müfredatlarında Kodlamanın Yeri.....	33
2.7 Senaryo Tabanlı Öğrenme (STÖ).....	35
2.8 Senaryo Tabanlı Öğrenme (STÖ) Aşamaları	37
2.8.1 Öğrenme Ürünlerinin Tanımlanması	37
2.8.2 Biçimine Karar Verilmesi	38
2.8.3 Uygun Konunun Seçilmesi	39

2.8.4 Gdleyici (Harekete Geirici) Olay ya da Durumun Belirlenmesi .	39
2.8.5 Senaryonun Akran Deęerlendirilmesi ve Gzden Geirilmesi.....	40
2.9 Sanal Senaryo Tabanlı Öğrenme Araları.....	40
2.10 Hedefe Dayalı Senaryolar (Goal-Based Scenarios).....	42
2.10.1 Hedefe Dayalı Senaryo (HDS) Ögeleri.....	44
ÜÇÜNCÜ BÖLÜM	47
III. YÖNTEM.....	47
3.1 Araştırma Yöntemi.....	47
3.2 Katılımcılar	48
3.3 Prosedürler.....	49
3.3.1 Hazırlık Aşaması.....	49
3.3.2 Uygulama Aşaması	52
3.4 Hedefe Dayalı Senaryolar	56
3.5 Veri Toplama Araları.....	60
3.5.1 Ön Test- Son Test.....	60
3.5.2 Mlakat (Grşmeler).....	61
3.5.3 Gzlem Formları.....	62
3.5.4 Rubrik (Scratch Performans Deęerlendirme Aracı).....	62
3.5.5 Scratch Grsel Programlama Aracı (Scratch 3.0 Beta Srm)	63
3.6 Veri Analizi Sreci	69
3.7 Geerlik ve Gvenirlik.....	79
DRDNC BLM	80
IV. BULGULAR VE YORUM.....	80
4.1 HDS'lerin Scratch Kullanma Srecine Etkisi	80
4.1.1 Grsel Tasarım.....	81
4.1.2 Kod Bloklarının Doęruluęu	86
4.1.3 rn Tamamlama Durumu	92
4.2 HDS'lerin Bilgi Dzeyindeki Etkisi	96
4.3 HDS'lerin Performans Puanlarına Etkisi	99
4.4 Katılımcı Grşleri.....	100
BEŞİNCİ BLM	106
V. SONUÇ, TARTIŞMA VE ÖNERİLER.....	106
KAYNAKÇA	114
EKLER.....	119

TABLolar LİSTESİ

Tablo 1: Katılımcılar	49
Tablo 2: Pilot Uygulama – HDS	50
Tablo 3: Uygulama Süreci	53
Tablo 4: Hafta2- Etkileşimli Piyano Oyunu Uygulaması	57
Tablo 5: Ön Test-Son Test Hedef Kazanımlar Belirtke Tablosu	61
Tablo 6: Öğrencilerin HDS/ Çalışma kağıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Görsel Tasarıma ilişkin Görüşleri.....	70
Tablo 7: Öğrencilerin HDS/ Çalışma kağıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Kodlama ve Kod Bloklarına ilişkin Görüşleri.....	71
Tablo 8: Öğrenci (5) ile gerçekleştirilen Görüşme Kod Analizi Örneği.....	72
Tablo 9: Hafta 2- Etkileşimli Piyano Oyunu Rubrik Kod Analizi.....	74
Tablo 10: Hafta 2- Etkileşimli Piyano Oyunu Gözlem Formu Kod Analizi.....	78
Tablo 11: Görsel Tasarım Ortalama Puanları	81
Tablo 12: Öğrencilerin HDS/ Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Görsel Tasarıma ilişkin Görüş Frekansları (DG-KG)	85
Tablo 13: Kod Bloklarının Doğruluğu Ortalama Puanları.....	86
Tablo 14: Öğrencilerin HDS/ Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Kod bloklarının doğruluğu üzerine Görüşleri (DG-KG).....	90
Tablo 15: Ürün Tamamlama Durumları Ortalama Puanları	92
Tablo 16: Öğrencilerin HDS/ Çalışma kağıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Ürün Tamamlama durumları üzerine ilişkin Görüşleri (DG-KG)	95
Tablo 17: Betimsel İstatistikler (DG-KG Ön test-Son test).....	97
Tablo 18: Normallik Testleri.....	98
Tablo 19: Bağımsız Örneklem T-testi Sonuçları (Ön-Test ve Son-Test)	99
Tablo 20: Performans Değerleri.....	100
Tablo 21: Bağımsız Örneklem T-Testi Sonuçları (Performans Puanları).....	100
Tablo 22: Hafta 1: Animasyonlu Davetiye	127
Tablo 23: Hafta 2: Etkileşimli Piyano Oyunu.....	130
Tablo 24: Hafta 3: Labirent Oyunu	134
Tablo 25: Hafta 4: Hediye Toplama Oyunu.....	139
Tablo 26: Hafta 5: Bilgi Yarışması Oyunu	144
Tablo 27: Hafta 6: Balon Patlatma Oyunu	148
Tablo 28: Hafta 7: Matematik 4 İşlem Resimli Bulmaca	151
Tablo 29: Hafta 8: Tuğla Kırma Oyunu	154
Tablo 30: Hafta 9:Trafik Kuralları Oyunu	158
Tablo 31: Hafta 10: Mevsimler Oyunu	162
Tablo 32: Hafta1: Animasyonlu Davetiye Tasarlama.....	166
Tablo 33: Hafta2: Etkileşimli Piyano Oyunu Uygulaması	169
Tablo 34: Hafta3: Labirent Oyunu Uygulaması.....	173
Tablo 35: Hafta4: Hediye Toplama Oyunu Uygulaması	176
Tablo 36: Hafta5: Bilgi Yarışması Oyunu Uygulaması.....	179
Tablo 37: Hafta6: Balon Patlatma Oyunu Uygulaması	182
Tablo 38: Hafta7: Matematik 4 İşlem Resim Bulmaca Oyunu Uygulaması	185
Tablo 39: Hafta8: Tuğla Kırma Oyunu Uygulaması.....	188
Tablo 40: Hafta9: Trafik Kuralları Oyunu Uygulaması.....	191

Tablo 41: Hafta 10: Mevsimler Oyunu	195
Tablo 42: Hafta 1: Animasyonlu Davetiye Tasarlama Gözlem Formu	197
Tablo 43: Hafta 2: Etkileşimli Pişano Oyunu Gözlem Formu	197
Tablo 44: Hafta 3: Labirent Oyunu Gözlem Formu.....	198
Tablo 45: Hafta 4: Hediye Paketi Toplama Oyunu Gözlem Formu	199
Tablo 46: Hafta 5: Bilgi Yarışması Oyunu Gözlem Formu.....	200
Tablo 47: Hafta 6: Balon Patlatma Oyunu Gözlem Formu	200
Tablo 48: Hafta 7: Matematik 4 İşlem Resim Bulmaca Oyunu Gözlem Formu	201
Tablo 49: Hafta 8: Tuğla Kırma Oyunu Gözlem Formu.....	202
Tablo 50: Hafta 9: Trafik Kuralları Oyunu Gözlem Formu.....	203
Tablo 51: Hafta 10: Mevsimler Oyunu Gözlem Formu.....	204
Tablo 52: 5. Sınıf Güncellenen Bilişim Teknolojileri ve Yazılım Dersi müfredatı kazanımları (http://www.e-müfredat.meb.gov.tr).....	207



ŞEKİLLER LİSTESİ

Şekil 1: Alice Arayüzü	18
Şekil 2: Blockly Arayüzü	19
Şekil 3: Kodu Game Lab Arayüzü	20
Şekil 4: MIT App Inventor Arayüzü	20
Şekil 5: Code Org Arayüzü	21
Şekil 7: Tynker Arayüzü	22
Şekil 8: Kodable Arayüzü	23
Şekil 9: Scratch Jr. Arayüzü	23
Şekil 10: Scratch Arayüzü-1	24
Şekil 11: Scratch Arayüzü-2	25
Şekil 12: Creative Learning (Yaratıcı Öğrenme-4P İlkesi)	28
Şekil 13: Senaryo Tabanlı Öğrenme (STÖ) Aşamaları	37
Şekil 14: Senaryo Tabanlı Öğrenme Ortamı Örneği (Mariappan, Shih ve Schrader'den (2004) uyarlanmıştır)	42
Şekil 15: Hedefe dayalı Senaryo (HDS) Öğeleri (Schank'den (1992) adapte edilmiştir.)	44
Şekil 16: Gömülü Karma Araştırma Desen (Cresswell'den (2012) uyarlanmıştır) ..	47
Şekil 17: Hazırlık aşaması	54
Şekil 18: Uygulama aşaması	55
Şekil 19: Hafta 2- Etkileşimli Piyano Oyunu HDS Örneği	56
Şekil 20: Scratch Ara yüzü-3	63
Şekil 21: Scratch Ara yüzü-4	64
Şekil 22: Scratch programı kod blokları bölümü	65
Şekil 23: Scratch programı kodlama paneli bölümü	66
Şekil 24: Scratch programı proje ekranı bölümü	67
Şekil 25: Scratch programı araç çubuğu bölümü	67
Şekil 26: Scratch programı örnek kod bloğu	68
Şekil 27: Görsel Tasarım Ortalama Puanları	82
Şekil 28: Kod Bloklarının Doğruluğu Ortalama Puanları	87
Şekil 29: Ürün Tamamlama Durumları Ortalama Puanları	94
Şekil 30: Hafta 1- Animasyonlu Davetiye Tasarlama	166
Şekil 31: Hafta 2- Etkileşimli Piyano Oyunu	169
Şekil 32: Hafta 3- Labirent Oyunu	172
Şekil 33: Hafta 4- Hediye Toplama Oyunu	175
Şekil 34: Hafta 5- Bilgi Yarışması Oyunu	178
Şekil 35: Hafta 6- Balon Patlatma Oyunu	182
Şekil 36: Hafta 7- Matematik 4 İşlem Resim Bulmaca Oyunu	185
Şekil 37: Hafta 8- Tuğla Kırma Oyunu	188
Şekil 38: Hafta 9- Trafik Kuralları Oyunu	191
Şekil 39: Hafta 10-Mevsimler Oyunu	194

SİMGELER VE KISALTMALAR

MEB: Milli Eğitim Bakanlığı

MIT: Massachusetts Teknoloji Enstitüsü

STÖ: Senaryo Tabanlı Öğrenme

HDS: Hedefe Dayalı Senaryolar

BT: Bilişim Teknolojileri



BİRİNCİ BÖLÜM

I. GİRİŞ

Öğrenme-öğretme sürecinin iyi planlanması ve etkileşimli olması anlamlı öğrenmenin gerekliliklerindedir. Günümüzde yaygın olan yapılandırmacı yaklaşıma göre öğrencilerin birebir uygulama yapmalarına imkân verilmesi öğrencilerin becerilerinin geliştirilmesine katkı sağlayabilir. Öğretim sürecinde öğrencilerin motivasyonunu istediğimiz hedefe yönlendirmek için onların aktif katılımını sağlamak, öğrenme çıktılarını olumlu yönde etkileyebilir (Batista, Connolly ve Angotti, 2016). Farklı Öğretim materyallerinin öğretim ortamlarında entegre edilmesi, öğrenmeyi kolaylaştırmasının yanında anlamlı öğrenmenin oluşmasında da etkili olmaktadır. Dale'nin (1946) yaşantı konisi modeline göre, eğitim ortamının birden fazla duyuya hitap edilecek şekilde sunulması, bireylerin aktif katılımının artması, bireylerin kendi kendine yaparak yaşayarak öğrenmesi öğrenmenin etkili ve kalıcı hale gelmesinde önemli rol oynamaktadır. Dale (1946), öğrenme sürecinde birden fazla duyu organına hitap edildiğinde öğrenmenin de o kadar kalıcı olacağını öne sürmektedir. Soyut içerikli öğrenmelerin, anlaşılması zor ve karmaşık konuların bile somut yaşantılarla, birebir uygulamalarla kolaylaşacağı, öğrenmelerin anlamlı ve kalıcı hale geleceği söylenebilir.

İnsan davranışının her açısında hedefleri elde etme ve hedeflere ulaşma çabası vardır. Yapılandırmacı öğrenme yaklaşımı, öğrencilerin öğretim sürecine aktif olarak katılmasını sağlamak ve motivasyonlarını arttırmak amacıyla eğitim öğretim sisteminde yerini almıştır. Bu öğrenme sürecinde öğrencinin etkileşimli etkinlikler yoluyla bilgiyi yapılandırması hedeflenir (Batista, Connolly ve Angotti, 2016).

Yapılandırmacı yaklaşım çatısı altında değerlendirilebilecek öğretim yöntemlerinden biri olan Hedefe Dayalı Senaryoda (HDS), öğrenciler senaryonun gerçek zamanlı katılımcıdır (Driscoll, 2005). HDS sürecinde öğrenci, kazanımların bir senaryo çerçevesinde kurgulanmasıyla kendisini olayın içinde bulmakta ve olayın bir karakteri

olarak karşılaştığı problemi çözme sürecine dâhil olmaktadır. Çağdaş eğitimin gerekli kıldığı biçimde öğrenci merkezli uygulamaların kullanılması etkili ve kalıcı bir öğrenme sağlamaktadır (Yaman ve Süğümlü, 2009). Alan yazındaki birçok çalışma, öğrencilerin doğrudan öğretim sürecine katıldığı problem-tabanlı öğrenme (Akınoğlu ve Tandoğan, 2007; Kilroy, 2004; Wood, 2003), işbirlikli öğrenme (Kreijns, Kirschner ve Jochems, 2003; Van den Bossche, Gijsselaers, Segers ve Kirschner, 2006), vb. yaklaşımlarda öğrenmenin olumlu yönde etkilendiğini göstermiştir. Bu tarz modern yaklaşımların örnekleri, biyolojiden sosyal bilgilere birçok konu alanında görülmektedir. Matematik ve fen bilimleri gibi soyut alanlarda da HDS çalışılmış ve öğrencileri daha fazla araştırma sürecine teşvik ederek aktif katılımının sağlanması ve böylece kavram yanlışları varsa bunların giderilmesi, kalıtım, genetik gibi soyut kavram ve konuların daha somutlaştırılarak anlamlı öğrenmenin arttırıldığı gibi sonuçlar ortaya çıkmıştır (Kocadağ ve Özsevgeç, 2013). Programlama veya kodlama öğretimi henüz müfredatlarda yeni yeni kendini göstermeye başlayan bir başka soyut alandır.

Programlama karmaşık soyut kavramlar içeren, mantıksal ve algoritmik düşünme becerilerinin yanı sıra çaba ve motivasyon gerektiren bir beceridir (Allsopp ve Ejsing-Duun, 2016). Programlamada bir öğrenci günlük hayatta karşılaştığı herhangi bir problem durumunda problem çözme becerileri, bilişsel veya matematiksel beceriler gibi farklı üst düzey düşünme becerilerini aktif olarak kullanmak durumundadır. Programlama öğretiminde algoritmalar, programlama mantığı, veri yapıları, kodların dizimi ve anlam bilgisi bakımından programlamayla ilk karşılaşan acemi bireylere başlangıçta zor ve karmaşık gelebilir. Bununla birlikte öğrencilerin yeni karşılaştıkları problem durumuna karşı çözümler üretmeleri, problemi çözebilecek algoritmaları oluşturmaları onlar için soyut gelebilir. Bu nedenle öğrencilerin programlamaya karşı motivasyonunun düşük olma olasılığı ve çabuk bırakma eğilimi yüksek olabilir (Angotti, Batista ve Connolly, 2016).

Öğrenciler programlama öğrenirken, öğrenme süreci onların anlayabileceği şekilde düzenlenmelidir. Öğrencilerin programlama kavramlarını anlamasını ve öğrenmesini desteklemek amacıyla bireylerin aktif katılımını sağlayacak alternatif öğretim metodlarını kullanmak birçok araştırmacı tarafından desteklenmektedir (Altıok ve Yükseltürk, 2015; Calder, 2010; Kausic ve Asic, 2011; Kelleher, Pausch ve Kiesler,

2007). Örneğin Tilki ve Öztaş (2014), geleneksel öğretim metotlarıyla bilgisayar programlamayı öğrenmenin, öğrenciler açısından programlama yapısını algılamada zorluk yaratabileceğini ifade etmektedirler. Bu nedenle öğrencilerin programlamayı etkili bir şekilde öğrenebilmesinde farklı ve yeni öğretim yaklaşımlarının kullanılmasının gerektiği ortaya koyulmaktadır. Programlama öğretiminde eğitsel oyunların kullanımının, yaratıcı düşünme ve problem çözme becerisini geliştirerek aktif katılımı birlikte bilgiyi anlamlandırmada katkı sağladığı sonucu ortaya çıkmıştır (Tilki ve Öztaş, 2014). Yapılan başka bir çalışmada da yine programlama öğretiminde oyuna dayalı öğretim yaklaşımının kullanımının öğrenmeye katkı sağladığı öne sürülmüştür (Westcott, 2008). İstekli ve motivasyonu yüksek bireylerin programlamaya daha yatkın olduğu söylenebilir (Yükseltürk ve Altıok, 2016). Bu nedenle öğrencilerin ilgisini çekecek uygulamalara yer verilmesi öğretim sürecinde işe yarayabilir. Bu amaçla yapılandırıcı yaklaşıma dayalı oyun tabanlı öğrenme, hedefe dayalı senaryo tabanlı öğrenme gibi öğretim yaklaşımlarının öğretim sürecine entegrasyonu etkili olabilir.

Son yıllarda, programlamaya olan talep ve öğrencilerin programlamaya karşı ilgisinde artış gözlemlenmekte ve tüm dünyada öğrencilerin erken yaştan itibaren programlamayı öğrenmesi gerektiği üzerinde durulmaktadır (Baz, 2018; Sayın ve Seferoğlu, 2016; Yecan, Özçınar ve Tanyeri, 2016). Programlama öğretimine erken yaşlarda başlamanın öğrencilerin gelişimleri açısından oldukça faydalı olacağı söylenebilir (Yükseltürk ve Altıok, 2016). Literatürde 21. yüzyıl becerileri olarak tanımlanan yaratıcı düşünme becerisi, eleştirel düşünme becerisi, problem çözme becerisi, analiz ve sentez yapabilme, akıl yürütme, algoritmik düşünme, analitik ve sistematik düşünme, medya ve teknoloji okuryazarlığı gibi bir takım becerilerin kazandırılmasının bireyin yaşam boyu etkili olarak öğrenme sürecinde oldukça önemli olduğu vurgulanmaktadır (Günüç, Odabaşı ve Kuzu, 2013; Karabak ve Güneş, 2013). Bu becerilerin kazandırılmasında ve geliştirilmesinde programlama eğitimi önemli rol oynamaktadır (Karabak ve Güneş, 2013; Monroy-Hernández ve Resnick, 2008; Shin, Park ve Bae, 2013). Programlamanın her öğrencinin kazanması gereken bir 21.yy becerisi olduğu düşüncesi, programlamanın çeşitli ülkelerde öğretim müfredatlarına entegre edilmesinde etkili olduğu söylenebilir. Programlama eğitimi ile bireyler karşılaştıkları problem durumuna karşı kendine özgü farklı çözümler ortaya atabilir ve

farklı performanslar sergileyebilir. Burada bireyin hem algoritmik düşünme becerisi hem de yaratıcı düşünme becerisini harekete geçiren programlama becerisi, bir nevi problem çözme becerisi olarak da düşünülebilir (Kalelioğlu ve Gülbahar, 2014; Karriker ve Heersink, 2011; Yükseltürk ve Altıok, 2016). Böylece öğrenciler farklı projeler tasarlayabilir, farklı ürünler oluşturabilir (Shin, Park ve Bae, 2013). Ayrıca Papert (1980) tarafından ortaya atılan bilgi-işlemsel düşünmenin (computational thinking), 21. yüzyılda her öğrenciye öğretilmesi gerektiği ile ilgili farklı görüşler ortaya atılmıştır. Bilgi işlemsel düşünme, en genel anlamda bir problem durumunun çözümünde bir insan ya da bilgisayarın kullanabileceği düşünme biçimi olarak tanımlanabilir (Wing, 2006). Bilgi işlemsel düşünme kavramına bağlı olarak programlama da 21. yüzyıl becerisi olarak öğrencilere kazandırılması gereken beceri olarak üzerinde durulmaktadır.

6. sınıf öğrencileriyle yapılan bir araştırmada programlama öğretiminin öğrencilerin problem çözme becerilerini olumlu yönde etkilediği ortaya çıkmıştır (Lai ve Yang, 2012). Calder (2010) ise yaptığı çalışmada, görsel programlamanın öğrencilerin motivasyon düzeylerini arttırdığını ve problem çözme becerilerinin geliştirdiğini belirtmiştir. Fessakis, Gouli ve Mavroudi (2013) 5-6 yaş grubundaki öğrencilerle yaptıkları çalışmada da programlama öğretiminin öğrencilerin problem çözme becerilerini geliştirdiğini vurgulamışlardır. Bu bulgulardan farklı olarak, Kalelioğlu ve Gülbahar'ın (2014) yaptıkları çalışmada ise ilköğretim 5. sınıf öğrencilerinin problem çözme becerilerinin üzerinde Scratch programlama etkisini araştırmışlar ve Scratch'ın öğrencilerin problem çözme becerilerine anlamlı bir katkısının olmadığı sonucuna varılmıştır. Çatlak, Tekdal ve Baz (2015) tarafından yapılan bir çalışmada programlama ile ilgili yapılmış 32 çalışmanın sonuçları değerlendirilmiş ve bilgisayar programlamanın problem çözme becerisi, yaratıcı düşünme, mantıksal düşünme, farklı derslerde kullanımı ve algoritma ve programlama öğrenimi gibi farklı alt başlıklar altında incelenmiştir. Bu inceleme sonucunda, Scratch programlamanın, algoritma ve programlama temellerinin öğretiminde ve ayrıca problem çözme ve yaratıcı düşünme gibi becerilerin geliştirilmesinde etkili olduğu, motivasyon ve tutum değişkenleri üzerinde olumlu etkiye sahip olduğu ve son olarak da Scratch'ın farklı derslerde de kullanılabileceği ifade edilmiştir (Çatlak ve diğerleri, 2015). Ayrıca programlama öğretiminde bireyler kodlama yaparken süreç içerisinde örtük olarak karşılaştıkları

problem durumlarına çözüm üretmek (Calder, 2010; Fessakis, Gouli ve Mavroudi, 2012; Lai ve Yang, 2012), sonuçlarını görmek (Gülbahar ve Kalelioğlu, 2014), deneyimleyerek özgüven kazanmalarını sağlamak (Ersoy ve Aydın, 2015), yaptıkları uygulamalarda kendi kendilerine geri dönüt almalarını sağlamak, hataları olduğunda onları düzeltmek (Cooper ve diğerleri, 2006; Shin ve Park, 2014; Shin, Park ve Bae, 2013) başarılı olduklarında motivasyon duygularının artmasını sağlamak (Yükseltürk ve Altıok, 2015), öğrencileri daha iyisini yapmaya hedeflemek, kendilerine olan güven duygusunu geliştirmek, başardığında mutlu olmak (Genç ve Karakuş, 2011) ve kendi oluşturdukları ürünleri başkalarıyla paylaşmak (Monroy ve Resnick, 2008) gibi birçok kazanımları kazanabileceği söylenebilir.

Programlamanın öğrenciler üzerindeki bu geliştirici ve olumlu etkilerine bakıldığında her ne kadar kazandırılması gerekli görülse de programlamayı öğrenmek zor ve karmaşık bir süreçtir. Daha önceden hiç programlama eğitimi almamış ve hiç programlama deneyimi olmayan bireyler için programlama yapıları ve kod parçaları soyut gelebilir. Programlamada kodların yapısını ve kural yapılarını anlatmadan önce algoritmik düşünme becerisini ve programlama mantığını kazandırmak gerekebilir. Algoritmik düşünme becerisini kazandıktan sonra verilen problem durumuna kolaylıkla çözümler üretebilir ve daha sonra kod yapıları ve programlama yapılarını daha kolay öğrenebilir.

Programlama mantığı ise algoritma yapısı geliştirilerek öğrenilmektedir (Bennedsen ve Caspersen, 2008). Algoritma, herhangi bir problem durumuna karşı çözüm basamaklarının aşama aşama ifade edilmesidir (Vatansever, 2011). Bu nedenle herhangi bir problem durumuna ilişkin programlamaya başlamadan önce temel algoritma yapısının belirlenmesi ve aşama aşama çözüm basamaklarının oluşturulması gerekmektedir. Ancak herhangi bir programlama geçmişi olmayan öğrenciler tarafından algoritma yapısının anlaşılması ve problem durumunun algoritmaya dönüştürülmesinde sıkıntılar yaşanabilmektedir (Özmen ve Altun, 2014).

Bilgisayarların sıradan insanların kullanabileceği hale gelmesinden yani 1980'lerden günümüze kadar hem etkileşim hem de ara yüz olarak ciddi gelişmeler olmuştur. Grafik ara yüzlerin olmadığı dönemlerde etkileşim çok daha zorken, şimdi bakış, ses ve hareket gibi birçok alternatifle etkileşimi zenginleştirebiliyoruz. Programlama

dillerindeki gelişim de yine benzer şekilde kodlamayı kolaylaştıracak şekilde değişmiştir. Soyut kodların yerini okul öncesi öğrencilerinin bile mantığını kavrayabileceği ve hatta kendi kod parçalarını oluşturabilecekleri görsel kodlar yani bloklar almaya başlamıştır. Bu sayede, ileri düzey programlama dillerinin öncesinde algoritma yapısı gibi temel alt yapı bilgileri ve becerileri erken yaşlarda edinilebilmektedir. Görsel kod blokları sadece erken yaşlar için değil, programlamada acemi olan her yaş grubuna hitap edebilir.

Scratch, Codemonkey, Alice ve Star Logo kod bloklarıyla etkileşim kurma ve çıktılarını gözlemlene imkânı veren ortamlardan birkaçıdır. Bunlar oyun ve etkileşimli hikâyelerle algoritma yapısını somutlaştırmakta ve problem çözme sürecini daha kolay ve anlaşılır hale getirmektedir (Ramadhan, 2000; Utting, Cooper, Kölling, Maloney ve Resnick, 2010). Her öğrenci bu araçları kullanarak kendine özgü tasarımlar oluşturabilir ve kendi tasarladığı karakterlere hareket özelliği kazandırabilir. Ayrıca bireylerin tasarladıkları oyunları bitirdikten sonra etkileşimli olarak oynayabilmesi öğrencinin ilgisini çekebilecek noktalardandır. Scratch, basit ara yüzü ve sürükle bırak özelliğiyle anasınıfından üniversite öğrencilerine kadar farklı yaşlardaki öğrencilere algoritma ve programlama mantığını eğlenceli bir şekilde öğretmek amacıyla kullanılmaktadır.

Scratch programıyla oluşturulan projelerde resim, müzik, video ve simülasyonlar kullanılarak etkileşimli oyunlar ve hikâyeler oluşturulabilmektedir. Bu kadar çeşitli materyalin hazırlanmasına imkân sunan bu ara yüz, farklı derslere de adapte edilebilmektedir. Algoritma ve programlama öğretiminin yanında matematik, fen ve teknoloji, sosyal bilgiler ve yabancı dil gibi derslerde öğrenilmesi zor ve karışık olan konular, Scratch ile zevkli hale getirilebilir ve bu karmaşık konular öğrencilere etkileşimli oyunlar, çeşitli animasyonlar ya da simülasyonlar tasarlamaları sağlanarak öğretilir. Örneğin, Calder (2010), yaptığı çalışmada matematiksel kavramların öğretiminde Scratch görsel programlama aracının etkisini incelemiş ve öğrencilerle görüşmeler yapmıştır. Bu araştırmanın sonucunda, Scratch'ın matematiksel kavramların öğrenilmesinde etkili olduğunu, problem çözme sürecini geliştirmesi bakımından mantıksal ve matematiksel düşünme becerisinin geliştirilmesinde ve soyut olan matematik kavramlarının daha somut ve anlaşılır hale gelmesinde işe yarayacağını vurgulamıştır (Calder, 2010). Scratch ile oyunlar tasarlanmasının veya

etkileşimli hikâyeler oluşturmanın öğrencilerin yaratıcı problem çözme becerilerine olan olumlu etkisi de yapılan çalışmalarda mevcuttur (Kim, Chung ve Yu, 2013). Sanjanaashree, ve Soman (2014), lise düzeyinde, Scratch programının yabancı dil öğretimiyle ilgili çalışmalarında, Scratch'ın öğrenme sürecini zevkli ve eğlenceli hale getirdiği ve olumlu etkilerinin olduğu sonucuna varmışlardır. Öğrenciler, bu öğretim sürecinde hem eğlenerek hem de etkili bir şekilde öğrendikleri için öğrenme daha anlamlı ve kalıcı hale gelebilir. Scratch görsel programlama aracının farklı derslerde kullanımıyla ilgili literatüre bakıldığında en fazla algoritma ve programlama öğretiminde etkileri incelenmiş, daha sonra matematik dersinde matematiksel becerilerinin geliştirilmesi üzerine etkilerinin araştırıldığı pek çok çalışmaya rastlanmıştır (Calder, 2010; Özsoy, 2005; Taylor, Harlow ve Forret, 2010). Bu çalışmalarda, Scratch'ın genellikle matematiksel becerileri düşük olan öğrencilerin algoritmik ve analitik düşünme gibi karmaşık düşünme becerilerinin geliştirilmesinde etkili olduğu sonucuna varılmıştır (Taylor ve diğerleri, 2010). Ancak yabancı dil ve fen öğretimi alanında çok fazla çalışma bulunmamaktadır.

Programlama ve algoritma öğretiminin soyut olması, döngüler, değişkenler, koşul ifadeleri gibi temel programlama yapılarının ve kod parçalarının zor anlaşılması öğrencilerin programlamaya karşı motivasyonlarının düşük olmasına neden olabilir (Monroy- Hernandez ve Resnick, 2010). Programlama zor ve soyut bir süreçtir. Bu nedenle programlamayla ilk karşılaşan bireylerin programlama eğitiminin zor bir süreç olduğunu düşünerek programlamaya karşı olumsuz bir tutum geliştirmelerine neden olmaktadır (Anastasiadou ve Karakos, 2011). Geleneksel programlama dillerindeki soyut ifadeler, teknik kavramlar ve metin tabanlı ifadeler öğrencilerin programlama sürecinde ilgisini çekmemektedir (Mayer, 1981). Ayrıca anadili İngilizce olmayan bireyler için de programlama dilindeki komutları anlamak da sıkıntı yaratabilen başka bir durumdur (Sanjanaashree ve Soman, 2014). Dolayısıyla motivasyon düşüklüğü ve olumsuz tutumlar, öğrencilerin programlamaya ilişkin başarılarını olumsuz etkilemektedir (Lahtinen, Ala-Mutka ve Järvinen, 2005). Bu süreçte öğrencilerin daha çok ilgisini çekmeyi hedefleyen, kendi dillerinde kod blokları içeren ve eğlence öğeleri barındıran görsel programlama araçlarının kullanılması, programlama öğretimine farklı bir boyut kazandırmıştır. Çünkü bu programların genel amacı, programlama yapısını görselleştirerek somutlaştırmak ve daha anlaşılır hale gelmesini sağlamaktır

(Bergin ve Martinez, 1996; Yükseltürk ve Altıok, 2016). Ayrıca öğrencilerin kodlama süreçlerinde herhangi bir hata durumu olduğunda anında geri dönüt imkânı sağladığı için bireyler hatalarını görebilir, düzeltebilir ve hatalarını en aza indirebilir (Cooper ve diğerleri, 2006).

Her ne kadar kodları somut hale getiren ortamlar olsa da becerilerin gelişmesi ve anlamlandırılması geleneksel olarak tasarlanmış bir ders ortamında çok etkili olmayabilir. Bu çalışmada amaç, Scratch gibi görsel kodlama imkânı sunan bir araçtaki kodları kullanabilme becerilerini HDS ilkeleriyle birleştirerek sunmak ve bunun etkisini gözlemlemektir. Alan yazına bakıldığında HDS'nin uygulandığı birçok alan olmasına rağmen kodlamayla ilgili herhangi bir uygulamaya rastlanmamıştır. Bu nedenle, HDS'nin öğrencilerin programlama performanslarına etkisi ele alınarak öğrenci görüşlerine de yer verilmeye çalışılmıştır.

1.1 Araştırmanın Konusu, Amacı ve Önemi

Bu çalışmanın temel amacı, hedefe dayalı senaryo (HDS) yaklaşımının görsel kodlamayla bağdaştırıldığında öğrencilerin ortaya koydukları ürünlerdeki performanslarına nasıl bir etkisi olduğunu gözlemlemektir. Daha önce herhangi bir kodlama tecrübesi olmayan ilköğretim 5. sınıf öğrencilerinin temel programlama mantığını öğrenebilmeleri için Scratch görsel programlama aracı kullanılmıştır. Bu kodlama aracının seçilmesinin sebebi, basit ara yüzü ve Türkçe dil desteğinin olmasıdır. Alan yazın taramaları sonucunda incelenen çalışmalarda Scratch ile ilgili çok fazla sayıda çalışmaya rastlanmıştır (Calder, 2010; Çatlak ve diğerleri, 2015; Yükseltürk ve Altıok, 2015) fakat kodlama öğretiminde HDS kullanımıyla ilgili pek fazla çalışmaya rastlanmamıştır. Öğrenciler karmaşık problem durumuyla karşılaştıklarında verilen problem durumunu tasarlamada zorluk çekebilir. Bu aşamada Senaryo Tabanlı Öğrenme (STÖ) yaklaşımı, düşünceyi davranışa dönüştürmede ve öğrenmenin somut hale getirilmesinde kullanılan bir öğretim yaklaşımıdır. STÖ yaklaşımı ile öğrenci ve öğretmenlerin hayal güçlerinin ve yaratıcılığının artırılmasıyla daha eğlenceli ve daha etkili öğretim süreci sunar. Somut içerikli senaryolarla ve öğrencilerin yaptıkları eğlenceli ve kendine özgün senaryo etkinlikleriyle soyut olan programlama ve algoritma kavramları açık ve anlaşılır hale gelebilir. Yaratıcı ve sistematik düşünme becerisini geliştiren HDS problem durumunu somutlaştırarak öğrencilerin Scratch programlama performanslarının daha iyi olmasını

sağlayabilir. Bir hedefe dayalı senaryoda motivasyonu artırma ve başarı duygusunu geliştirme amaçlanarak öğretim sürecinin daha etkili bir şekilde sürmesi sağlanmaktadır.

Programlama öğretiminde Scratch ile öğrenciler kendi oyunlarını tasarlarken soyut olan kavramları zihninde canlandıramayabilir. Bu nedenle hedefe dayalı senaryolar daha somut ve görsel olarak onlara destek sağlayabilir. Hedefe dayalı senaryolar, genel geçer durumlardan daha çok becerilere odaklanacak şekilde tasarlandığında daha etkili olabilir (Schank, 1992). Bir işi ya da davranışı yapmadan önce durumların bilinmesinden çok, net bir hedefe sahip olmak ve yapılan davranışın gerçekten nedenini bilmek insanı motive eder. Hedefe dayalı senaryolar, daha eğlenceli bir şekilde yapma isteğini arttırırken aynı zamanda öğrencinin yapacağı işin yönergesini akılda canlandırmasına katkıda bulunabilir.

Öğrenciler Scratch programını kullanarak etkileşimli oyunlar tasarlamadan önce hedefe dayalı senaryoların verilmesi onların motivasyonunu arttırabilir. Böylece öğrenciler hangi amaçla neyi yapacaklarını daha iyi kurgulayabilir. Hedefe dayalı senaryoların tasarımında önemli faktörlerden biri de; öğrencinin HDS' de kazanmasını istediği hedef becerilerin tasarımcı tarafından belirlenmesidir (Schank ve Cleary, 1989). Böylece hedef beceriler belirlendikten sonra o senaryonun misyonu belirlenir. HDS'nin misyonu ile öğrencilere ne yapmaya, neye ulaşmaya çalıştıkları önceden öğrencilere belirtilir. Böylece öğrencilerin öğretim sürecine güdülenmiş bir şekilde daha aktif olarak katılmaları sağlanır. STÖ modeliyle, öğrenmeler anlamlı hale getirilir ve karmaşık olan problem durumlarına çözümler sunarak yaratıcı fikirlerle daha eğlenceli bir öğretim süreci sunar (Schank, 1999). Bu nedenle programlama öğretiminde öğrenciler için daha eğlenceli ve daha etkili öğrenme süreci sağlayan Scratch görsel programlama aracının kullanılması, öğrencilerin programlamaya karşı olumlu tutumlarının artması ve ayrıca Scratch ortamında yapacakları ürünlerin daha önceden bir hedef çerçevesinde ve senaryo dâhilinde planlanmasıyla motivasyonlarının artması beklenmektedir.

1.2 Araştırma Soruları

1. Hedefe dayalı senaryoların, 5. sınıf öğrencilerin Scratch programını kullanma performansı üzerine etkisi nasıldır?
 - 1.1. HDS doğrultusunda geliştirilen Scratch uygulamalarının,
 - 1.1.1. Görsel tasarımı,
 - 1.1.2. Kod bloklarının doğruluğu,
 - 1.1.3. Ürün tamamlama durumları üzerine etkisi nasıldır?
 - 1.2. HDS grubundaki 5. sınıf öğrencilerinin, blok temelli programlama son-test puanları kontrol grubundaki öğrencilere göre anlamlı şekilde farklılık göstermekte midir?
 - 1.3. HDS grubundaki 5.sınıf öğrencilerinin ortaya çıkardıkları performans ürünlerinin ortalama puanı, kontrol grubundaki öğrencilere göre anlamlı olarak farklılık göstermekte midir?
2. Katılımcıların, Scratch'ın ve senaryoların/çalışma kâğıtlarının derse dâhil edilmesiyle ilgili görüşleri nelerdir?

1.3 Terimler

Hedefe Dayalı Senaryo: Roger Schank tarafından öne sürülen hedefe dayalı senaryolar, yaparak yaşayarak öğrenme ile durum tabanlı öğrenmeyi birleştiren yapılandırmacı öğrenmenin bir modelidir (Schank, 1992).

Senaryo Tabanlı Öğrenme: Etkileşimli senaryolar kullanarak problem temelli öğrenme veya durum tabanlı öğrenme gibi aktif öğrenme stratejilerini destekleyen öğrenme ortamlarıdır (Clark, 2009).

Problem Temelli Öğrenme: Gerçek problem durumuna gereken çözümü oluşturarak öğrencilerde öğrenmeyi sağlayan öğretim metodudur. Problem çözme sürecinde öğrenciler problem çözmeye çalışırken içerik bilgilerini yapılandırır ve kendi kendine bağımsız öğrenme becerilerinin yanı sıra problem çözme becerilerini de geliştirir. 1950 yılında ilk defa tıp eğitiminde kullanılmak üzere geliştirilmiştir. Ayrıca öğrencilerin grupla etkileşim ve işbirliği çalışma becerilerini geliştiren öğrenci merkezli bir yaklaşımdır (Hung, Jonassen ve Liu, 2007; Polya, 1945).

Kodlama: Kodlama bir programı derleyebilmek amacıyla kullanılan komut sistemleri ya da programlama dili olarak tanımlanabilir. Kodlamanın eğitimsel olarak kullanımı 60'lı yıllarda Logo programlama dilinin kullanımı ile başlamıştır. Kodlama, çeşitli ülkelerde farklı biçimlerde tanımlanmıştır. Örneğin; Belçika'da “bilişimsel düşünce ve programlama”, Bulgaristan'da “algoritmik problem çözme ve programlama”, Estonya'da “programlama”, İspanya'da “programlama, algoritma ve robotik” ve İngiltere'de “computing” olarak yer almaktadır.

Kod blokları: Geleneksel programlama dillerinde yaygın olan metin tabanlı kodlar yerine, çeşitli renk ve şekillerde, yapboz benzeri ve tut-sürükle-bırak mantığıyla program parçaları oluşturmaya yarayan ortamlardır.

Scratch: Kelime anlamı çizmek, tırmalamak ya da iz anlamlarına gelmektedir. Mitchell Resnick ve ekibi tarafından 2003 yılında Massachusetts Teknoloji Enstitüsü (MIT) Medya laboratuvarında geliştirilen her yaşa hitap edebilen, zengin medya içeriğine sahip, detaylı programlama bilgisi olmaksızın etkileşimli hikâyeler, animasyonlar, oyunlar, müzik videoları gibi projeler geliştirebilmeyi ve çevrimiçi olarak başkalarıyla paylaşabilme imkânı sağlayan iki boyutlu görsel programlama aracı ya da programlama dilidir (Scratch about, 2017).

Programlama: Bilgisayar tarafından belirli bir görevi gerçekleştirmek ve insan bilgisayar etkileşimi sağlamak amacıyla bilgisayara ne yapacağını, belirli bir talimatla çeşitli komutlar ya da programlama dili kullanılarak adım adım ifade etme durumudur. Aynı zamanda bilgisayar programlama sistematik ve analitik düşünme süreçlerini içine alan, problem çözme becerisi olarak da ifade edilebilir. Bir problemin nasıl çözüleceğinin, çözüm aşamalarının adım adım belirlenmesiyle, algoritmik düşünme becerisini de kazandıran bir beceri olarak tanımlanabilir. Programlama, öğrencileri problem çözmeye, uygulamalar tasarlamaya ve çevrimiçi bağlantılar kurmaya yönlendirir (Kafai ve Burke, 2014).

Görsel Programlama: Görsel ve grafiksel öğelerle yapılan bilgisayar programlamasına denir. Programın belirli bir bölümünün ya da tamamının görsel araçlar kullanılarak kodun otomatik olarak yazılmasıyla oluşan programlamadır. Görsel programlama araçlarının en yaygın olarak kullanılanlarından bazıları şöyledir: Scratch, Small Basic, Alice, Code.Org ve App Inventor.

İKİNCİ BÖLÜM

II. KURAMSAL ÇERÇEVE

21.yüzyıl becerileri olarak tanımlanan problem çözme, eleştirel düşünme, yaratıcı düşünme, yenilikçi ve üretken olabilme, akıl yürütme, karar verebilme, analitik düşünebilme, analiz veya sentez yapabilme, medya okuryazarlığı gibi beceriler; günümüz öğrencilerine kazandırılması beklenen üst düzey düşünme becerileridir. Bir bireyin gerçek hayatta karşılaştığı problemi aşama aşama çözebilmesinde algoritmik düşünce ve programlama becerisini kullanabilmesi, o problemin çözümünde kolaylık sağlamaktadır. Çünkü birey karşılaştığı problemi nasıl çözeceğini kafasında adım adım tasarlarken nasıl plan yapıyorsa aslında bilgisayar sisteminde de problemi çözmek adına algoritmalara dayanarak kodlama yaptığında burada bir nevi programlama yapmış bulunmaktadır. Bu nedenle programlama aslında problem çözme becerisi olarak tanımlanabilir (Altıok ve Yükseltürk, 2015; Gezgin, Özcan, Ergün, Köse ve Emir, 2017). Programlama becerisi sadece bilgisayarda program yazmakla sınırlı olmamakla birlikte günlük hayatımızın hemen hemen her alanında algoritma aşamalarını kullanarak düşünce sistemi geliştirdiğimizde karşımıza çıkmaktadır. Örneğin bir matematik probleminin çözümünde; öncelikle çözüm basamaklarının belirlenmesi gerekir. Programlama becerisi ile bireylerin üst düzey becerilerini kullanmalarını gerektiren, analitik düşünme, problem durumunu farklı açılardan analiz edebilme, sistematik ya da eleştirel düşünme ve problem durumuna farklı çözümler üretmeyi amaçlayan yaratıcı düşünme gibi becerileri de beraberinde gerektirmektedir (Calder, 2010; Fessakis, Gouli ve Mavroudi, 2013; Papert, 1980; Park, 2013; Yükseltürk ve Altıok, 2016). Yapılan araştırmalarda programlama öğretiminin öğrencilerin problem çözme becerileri üzerinde olumlu yönde etkilerinin olduğu sonucuna varılmıştır (Calder, 2010; Fessakis, Gouli ve Mavroudi, 2013; Lai ve Yang, 2011). Bu nedenle dünyanın çeşitli yerlerinde, öğrencilerin erken yaşlardan itibaren programlama becerisinin geliştirilmesi gerektiği üzerinde durulmuştur. Bunun etkisi

eđitim m¼fredatlarına yansımıř ve ilgili eđitim m¼fredatlarında deęiřiklikler yapılarak b¼lgesel ve ulusal boyutta ¼nemli projeler geliřtirilmeye bařlanmıřtır (Balanskat ve Engelhardt, 2014).

D¼nya genelinde ¼ocuklara programlamayı daha etkili ¼ğretmek ve ¼ocukların programlama becerisini kazanmasını saęlamak amacıyla programlamayı somutlařtıran ve daha eęlenceli hale getiren farklı ara y¼zlere sahip ¼eřitli g¼rsel programlama ara ¼aları geliřtirilmiřtir. Bunlardan Scratch g¼rsel programlama aracı, ilköęretim ve orta¼ęretimin ilk basamaklarında kullanılabilecek g¼rsel ara y¼ze sahip ve kolay ulařılabilirlięi a ¼ısından kodlama ¼ęretiminde ve programlama becerisinin geliřtirilmesinde ¼nemli bir yere sahiptir. Ancak her ne kadar g¼rsel bloklar ¼zerinden programlama yapısı oluřturmaya imk¼n verse de Scratch gibi bir ortamda bile problem durumunun doęrudan verilmesi ¼ęrenciler i ¼in soyut ve anlamsız gelebilir. Bu nedenle problem durumunun bir senaryo ¼er ¼evesinde verilmesi ¼ęrencinin motivasyonunu istedięimiz hedefe y¼nlendirebilmemizde etkili olabilir. ¼ęrenciye senaryo durumu verildięinde, kendisini o senaryonun bir par ¼ası olarak d¼řünebilir ve problemin ¼öz¼m¼ne aktif olarak katılım saęlayabilir. Bu nedenle Scratch programlama ¼ęretiminde problem durumunun hedefe dayalı senaryolar ile verilmesi, ¼ęrencinin programlamada algoritma mantıęını daha ¼abuk kavraması ve problem durumunda daha istekli bir řekilde ¼öz¼me ulařmasını ve daha aktif bir řekilde s¼rece katılmasında etkili olabilir.

2.1 Programlama

Bilgisayar tarafından belirli bir g¼revi ger ¼ekleřtirmek ve insan bilgisayar etkileřimi saęlamak amacıyla bilgisayara ne yapacaęını, belirli bir talimatla ¼eřitli komutlar ya da programlama dili kullanılarak adım adım ifade etme durumudur. Programlama, bir problemin ¼öz¼m¼ i ¼in kod yazma olarak da tanımlanabilir. Kesici ve Kocabař'a (2007) g¼re programlama ger ¼ek hayattaki iřlemlerin, komutlar dizisinin, bilgisayar ortamında derlenmesi ve ¼alıřtırılması olarak tanımlanmıřtır. Bilgisayar programlama zor, karmařık ve uzmanlık gerektiren bir iř olarak g¼r¼lmektedir (Eckerdal, 2009; Gen ¼ ve Karakuř, 2012; Gomes ve Mendes, 2007; Robins, Rountree ve Rountree, 2003). Metin temelli programlama dillerinin soyut olması ¼ęrencilerin ilgisini ¼ekmemekte ve programlamaya kar řı olumsuz bakıř a ¼ısı ve ¼nyargıyla yaklařmalarına neden olmaktadır (Demirer, 2016).

Bilgisayar teknolojilerinde geçmişten günümüze kadar önemli derecede değişiklikler olmuştur. 1945'te Memex adlı tasarımın ortaya atılmasıyla birlikte bilgisayar alanının temel bileşenleri somut bir boyut kazanmış ve böylece bilgisayarın depolama birimi olmasının yanında programlanabilir bir makine olduğu da modellenmiştir (Çağiltay, 2018). Fakat bilgisayarın kullanıldığı ilk dönemlerde kullanıcı ile bilgisayar arasındaki etkileşimi sağlamak amacıyla makine dili kullanılmıştır. 1954 yılından itibaren Basic, Fortran, Cobol, Algol, Pascal ve C gibi yordayıcı diller ortaya çıkmış (Çamoğlu, 2009) ve bu diller genellikle mühendislik hesapları ve ticari uygulamalar için sadece mühendisler tarafından kullanılırken zamanla, özellikle 1980'li yıllardan itibaren günümüze kadar dünyada sıradan insanların bilgisayar kullanımına başlamasıyla birlikte İnsan- Bilgisayar etkileşimi farklı bir boyut kazanmıştır. Bilgisayar alanındaki gelişmelerle birlikte, programlama boyutunda da sadece mühendislerin hesaplama ve ticari uygulamalar için kullanılan bir araç olmaktan çıkmış ve programlama süreci önemli boyutta gelişme göstermiştir. Ayrıca programlama günümüzde görsel ve işitsel programlama araçlarının gelişmesiyle birlikte Anasınıfından itibaren hemen hemen her yaşta bireyin ilgisini çeker hale gelmiştir. Blok tabanlı ve görsel uygulamaların gelişmesiyle birlikte, ülkemizde ve dünyada hemen hemen bütün insanların programlamaya ilgisi artmış küçük yaşta bireyler bile kendi programlarını kendi kendilerine yazabilir hale gelmiştir.

2.2 Metin Temelli Programlama

Metin temelli programlama dilleri 1980'li yıllardan itibaren ortaya çıkmıştır. Söz dizim kurallarına göre belirli formatta ve mantık süzgecinden geçirilerek kod yapılarının bir araya getirilmesiyle oluşturulan programlama olarak ifade edilebilir. Programlamaya ilk başlayanlar için programlamadaki terimler yapısı itibariyle zor ve soyut gelebilir. Programlama öğretiminde öncelikle algoritmalar ile programlama mantığının kazandırılması gerekmektedir. Aksi takdirde programlama öğretiminde algoritmalar ve programlama mantığı kazandırılmadan direkt olarak herhangi bir programlama dilinin öğretimine geçildiğinde bireylerin öğrenme sürecinde ne yaptıklarının çok bilincinde olmadan öğrenmeye çalıştıkları ve hatta kodları anlamsızca ezberlemeye çalıştıkları gözlemlenebilir.

Programlamaya yeni başlayan bir birey programlama dillerinin karmaşık mantıksal yapısını anlamakta güçlük çekebilir ve algoritmik düşünme biçiminde zorlanabilir.

Metin temelli programlamada kodları öğrenmek, yazmak, anlamak ve hatırlamak zor olabilmektedir. Programlamada kod yapısı karışık gelebilir ve kodlama aşamasında hatalarla karşılaşılabilir. Metin temelli programlamada öğrenciler döngü, değişken, operatörler gibi temel programlama yapılarını kullanırken zorluk çekebilir.

Geçmişten günümüze programlamanın öğretimi ile ilgili zorluklar incelendiğinde çeşitli faktörlerin etkisinden bahsedilebilir. Programlama dilinin yapısı (Esteves ve Mendes, 2004; Gomes ve Mendes, 2007; Ozoran, Çağıltay ve Topalli, 2012), programlama mantığının oluşturulamaması (Duboulay, 1986; Lahtinen, Ala Mutka ve Jarvinen, 2005; Özmen ve Altun, 2014; Pea ve Kurland, 1983), programlama öğretim yöntemi (Esteves ve Mendes, 2004; Gomes ve Mendes, 2007), programlamada yabancı dil kullanımı (Arabacıoğlu, Bülbül ve Filiz, 2007); programlama öğrenmeye yönelik tutum (Hongwarittorn ve Krairit, 2010), öğrencilerin özyeterlik düzeyi, motivasyon ve deneyim eksikliği (Gomes ve Mendes, 2007; Özmen ve Altun, 2014) gibi faktörler bunlara örnek olarak verilebilir.

İleri seviyedeki programlama dilleri (C/C++ , Java gibi) programlamaya yeni başlayanlar ve daha önceden tecrübesi olmayanlar için ilgi çekici olmayabilir. Özellikle 7-12 yaş grubundaki çocuklara gösterilen programlama öğretiminde, ilköğretimin ilk ve son aşamalarında soyut düşünebilme ve anlama kabiliyetlerinin daha düşük olmasının yanı sıra görsel olmayan programlama öğretiminin ilgi çekme olasılığı oldukça düşüktür. Bu durum öğrencilerin motivasyonlarını azaltmakla birlikte programlamayı çabuk bırakma eğilimine neden olmaktadır. Bu nedenle programlamaya yeni başlayanlar için bu sıkıntıları ortadan kaldırmak ve programlamayı öğrenmelerini kolaylaştırmak, daha somut ve anlaşılır hale getirmek amacıyla görsel tabanlı programlama araçları geliştirilmiştir. Görsel programlama araçları, programlamayı somut hale getirerek öğrencilerdeki bilişsel engelleri ortadan kaldırabilir (Hill, 2015).

Görsel programlama araçlarından biri olan Scratch; öğrencilerin sürükle bırak yöntemiyle etkileşimli kod bloklarını kullanmalarına izin vererek metin temelli programlamanın neden olabileceği bilişsel engelleri azaltır. Scratch programlamada öğrenciler problem durumuna karşı odaklanır, hedefler belirler, çözüm yollarına dair planlar oluşturabilir, hatalarını gözlemleyebilir ve değerlendirebilir. Bireyler Scratch

platformu üzerinde paylaşılan diğer insanların farklı projelerini gözlemleyebilir ve oradaki projeleri gözlemleyerek motivasyonlarının artmasını ve daha iyilerini yapabileceğini düşünmelerini sağlayabilir (Coravu, 2015).

Scratch alanında yapılan akademik çalışmalarda Scratch görsel programlama aracının öğrenciler için soyut gelen döngü, dizi, değişken gibi programlama yapılarını öğrenmeyi kolaylaştırdığını, öğrencilerin anlama düzeylerini geliştirdiğini ve oyun tasarlama sürecinde öğrencilerde merak ve heyecan duygusu uyandırdığı sonuçları ortaya çıkmıştır (Calder, 2010; Malan ve Leitner, 2007). Ayrıca öğrencilerin programlamaya yönelik motivasyonlarını yükselterek öğrencilerin kendi başarılarıyla ilgili inancı ve özgüvenlerini artırdığı birçok araştırmacı tarafından ifade edilmektedir (Calder, 2010).

2.3 Blok Temelli Programlama

Günümüzde öğrencilerin programlamaya karşı ilgilerini çekebilmek ve soyut olan algoritma ve programlama mantığını daha somut hale getirmek amacıyla öğrencilerin mantıksal düşünme becerilerini destekleyecek öğrenme ortamlarının farklı tasarlanması işe yarayabilir. Programlama öğretiminde soyut olan kod yapısı, karmaşık algoritma yapısı ve programlama mantığının zor anlaşılması gibi sıkıntıları gidermek amacıyla blok temelli görsel programlama araçları geliştirilmiştir. Blok temelli görsel programlama ile öğrenciler kodları tek tek yazmak yerine kod bloklarından sürükleyip bırak yöntemiyle oluşturdukları programlama ile öğrenme süreci daha somut ve anlaşılır hale gelmiştir. Programlamayla ilk defa tanışan çocuklara kodlamayı sevdirmek amacıyla kolay ara yüze sahip görsel programlamanın kullanılması daha etkili olacaktır. Görsel programlama dilleri öğrencilerin hayal gücünün ve düşünme becerilerinin geliştirilmesine katkıda bulunur (Coravu, 2015). Öğrenciler kendi başlarına etkileşimli hikâyeler, oyunlar, animasyon videoları, simülasyonlar ya da oyunlar tasarlayabilir.

Programlama zor ve soyut bir süreç olmasına rağmen, günümüzde Scratch, Blockly, Alice vb. görsel programlar yardımıyla daha somut hale gelmiştir. Scratch görsel programlama ile bloklarla programlama öğretimi sayesinde programlama yaşı daha da düşmüştür. Bu da her yaşta ve ilgi alanından bireylerin kodlamayla tanışmasına ve

sıkılmadan daha aktif ve eğlenceli bir şekilde oyunlar, etkileşimli hikâyeler, animasyonlar, simülasyonlar, dijital öyküler vb. geliştirebilmesine imkân verir.

Çocuklar için kullanılan eğlenceli ve etkili görsel programlama araçlarından biride Scratch'dir. Çocuklar kendi oluşturdukları hikâyeleri ya da verilen başka bir hikâyeyi Scratch yardımıyla etkileşimli karakterler kullanarak hareket özellikleri ile animasyon biçiminde daha da geliştirebilirler. En eski görsel programlama aracı 60 lı yıllarda logo'dur (Calao, Moreno-Leon, Correa ve Robles, 2015; Papert, 1980). Daha sonra görsel programlama Alice, Code.org ve Scratch gibi programlama dilleri ile gelişmeye başlamıştır (Sayın ve Seferoğlu, 2016).

Blok tabanlı programlama ortamları çeşitli görsel programlama dillerinin kullanıldığı temel puzzle parçalarına benzetilebilir. Böyle ortamlarda öğrenenler işlevsel programlarda sadece mouse kullanarak görsel kod bloklarını açıklamalarını kendi zihninde anlamlandırarak bir araya getirir. Bazı bloklar kullanıcıya nasıl olduğu ve blok şeklinden ve renklerinden dolayı nerede kullanılacağı ile ilgili görsel ipucu sağlayıcı niteliktedir. Ayrıca kod açıklamaları blokların yapısına bitişik şekilde kullanılır. Görsel açıklama bazı bloklar tarafından ifade edilmesiyle birlikte bloklar arasındaki yapı, bloklar çeşitli formlarda yapı iskelesi kullanıldığında oluşur. Sonuç olarak, bu gibi özellikler blok tabanlı programlama ile kullanım kolaylığı olduğunu göstermektedir. Kodlama alanında bulunan bloklardaki kod parçacığını sürükleyip bırak yöntemiyle sahne alanına bırakarak yeni tasarımlar oluşturulabilmesine imkân sağlar. Böylece kod hataları olmadan kod dizimini doğru şekilde oluşturabilmeyi ve uyarlanan kodların aktif bir şekilde çalışmasına katkıda bulunur.

Scratch ve Alice gibi iyi bilinen blok tabanlı programlama ortamları, bireylere etkileşimli hikâyeler, animasyonlar ve dijital oyunlar gibi yaratıcı aktivitelerin geliştirilmesine katkı sağlamaktadır (Resnick ve diğerleri, 2009).

Scratch programlamanın genellikle ilk ve orta dereceli okullarda uygulanmasına yönelik pek çok çalışma mevcuttur. Malan ve Leitner (2007); ilk defa Scratch Programlamayı Harvard Üniversitesi'nde programlamaya giriş dersine entegre ederek kullanmışlar ve etkililiğini değerlendirmişlerdir. Scratch'ın öğrencilerin temel programlama konularını öğrenmelerinde başarılı olduğunu raporlamışlardır. Ozoran, Çağıltay ve Topalli (2012) da Scratch'ın öğrencilerin performansı üzerine yaptıkları

çalışmalarında, Scratch'ın öğrencilerin programlama performanslarını arttırdığını, programlamayı daha eğlenceli hale getirdiğini ve öğrencilerin yaratıcılığına da olumlu katkısı olduğunu raporlamışlardır. Sadece programlama alanında değil, diğer alanlarda da Scratch'ın etkisi çeşitli çalışmalarla gözlemlenmiştir. Örneğin, Rizvi ve diğerleri (2010), bir dönem boyunca Scratch kullanımına dayalı bir giriş dersi tasarlamışlar ve bu dersi düşük matematik becerisi olan öğrenciler üzerinde uygulamışlardır. Bu çalışmanın sonucunda, öğrencilerin performansını ve öz yeterliklerini arttırdığını raporlamışlardır (Rizvi ve diğerleri, 2010).

2.3.1 Programlama Öğretiminde Kullanılan Araçlar

Scratch, Alice, Blockly, Kodu Game Lab, MIT App Inventor, Tynker, Kodable, Fiziksel robot, Robomind, Etoys, Code.org, Stency gibi araçlar programlama öğretiminde kullanılmaktadır. Bu programlama araçları görsel grafik tabanlı programlama dilleri olup öğrencilerin daha somut ve eğlenceli bir şekilde kodlama yapabilmelerini kolaylaştırmaktadır. Bunlardan en çok kullanılanların Alice, App Inventor, Code. Org ve Scratch olduğu söylenebilir. 5-7 yaş grubundaki anasınıfı öğrencileri için de Scratch Jr. tercih edilmektedir.

2.3.1.1 Alice

Alice, 1998 yılında Carnegie Mellon Üniversitesi tarafından geliştirilen, kullanıcılara kodları sürükleyip bırakarak 3 boyutlu dijital animasyon geliştirmelerine imkân veren, programın çalışıp çalışmadığını test ederek kod yazımından kaynaklı hataları gösteren; ses, müzik, resim gibi interaktif medya araçlarını ekleyerek nesnelere hareket özelliği kazandırılmasını sağlayan blok tabanlı programlama dilidir. Genellikle 8 yaş ve üzeri bireylere hitap etmektedir (Alice, 2019). Alice programının ara yüzü Şekil-1'de gösterilmiştir.



Şekil 1: Alice Arayüzü

2.3.1.2 Blockly

Google tarafından geliştirilen, kullanıcılara sürükle bırak metodunu kullanarak dijital uygulamalar ve etkileşimli oyunlar geliştirmelerine fırsat sunan ve bunun yanında Blockly üzerinde yapılan uygulamaların Javascript, Python, PHP, Lua ve Dart programlama dillerindeki kodlarını da oluşturan blok tabanlı görsel programlamadır (Blockly, 2019). Blockly programının ara yüzü Şekil-2’de gösterilmiştir.



Şekil 2: Blockly Arayüzü

2.3.1.3 Kodu Game Lab

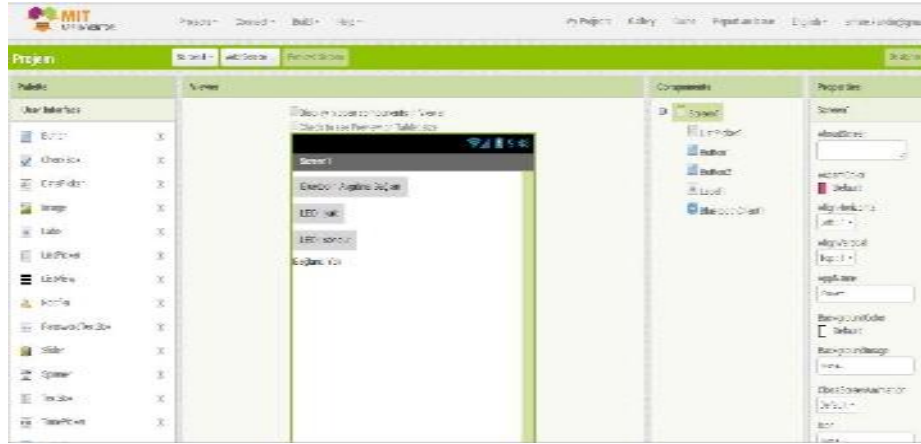
Microsoft Research tarafından ilk başlarda Xbox için geliştirilen ancak zamanla bilgisayarlarda kullanılabilen nesne tabanlı görsel programlama dilidir. 3B boyutlu ortamda 3B karakterler kullanılarak kendilerine ait kurguladıkları senaryolarla animasyonlar ve farklı oyunlar tasarlamalarını ve böylece yaratıcılık ve programlama becerilerinin geliştirilmesini sağlamaktadır. Kodu Game Lab ortamında bulunan hazır özellikler sayesinde herhangi bir programlama geçmişi olmayan küçük yaşta bireylerin bile kodlayarak kendi oyunlarını tasarlamalarına imkân vermektedir. Geliştirdikleri oyun projelerini web ortamında arkadaşlarıyla paylaşabilmektedirler (Kodu Game Lab, 2019). Kodu Game Lab programının ara yüzü Şekil-3’de gösterilmiştir.



Şekil 3: Kodu Game Lab Arayüzü

2.3.1.4 MIT App Inventor

MIT App Inventor; 2012 yılında Google ve Massachusetts Teknoloji Enstitüsü (MIT) iş birliğiyle geliştirilen ve Android İşletim sistemi uyumlu mobil cihazlara uygulamalar geliştirilmesini sağlayan blok tabanlı ve görsel programlama ortamıdır. Sürükle- bırak kodlama ara yüzüne ve online kurulum, kullanım yapısına sahip olan MIT App Inventor hemen hemen her yaştan kullanıcıya hitap etmektedir (MIT App Inventor, 2019). MIT App Inventor programının ara yüzü Şekil-4'te gösterilmiştir.

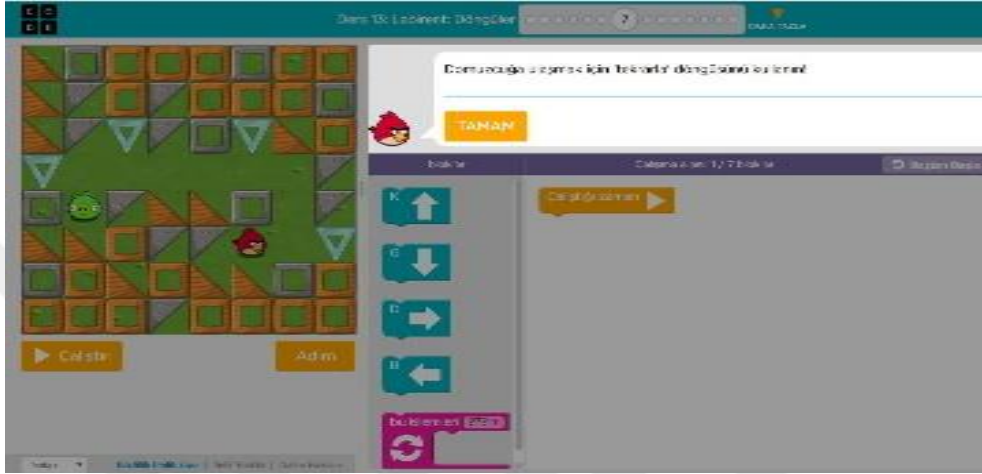


Şekil 4: MIT App Inventor Arayüzü

2.3.1.5 Code Org

Microsoft'un Hours of Code (Kod saatleri) projesi kapsamında geliştirilen, bireylerin oyunlar, etkileşimli animasyonlar oluştururken algoritma ve programlama mantığını kazandıran görsel web tabanlı blok programlamadır. Özellikle küçük yaştaki bireylerin

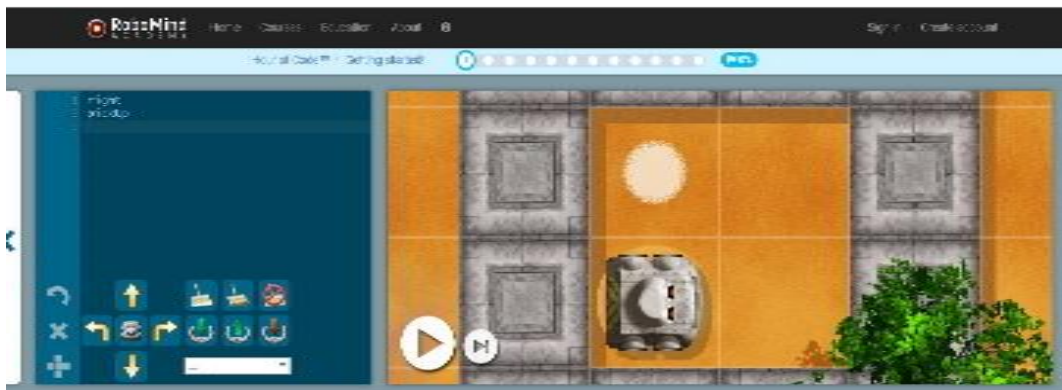
kodlamayı ve programlama mantığını aşama aşama öğrenmesinde oldukça etkilidir. Algoritmik düşünme becerilerinin geliştirilmesinde önemli rol oynayan Code.Org; aşama aşama basitten karmaşığa doğru görevlerin başarıyla tamamlanmasıyla ilerleme göstermektedir. Ayrıca görevlerin nasıl yapılacağıyla ilgili açıklayıcı video içerikleri de bulunmaktadır (Code Org, 2019). Code Org programının ara yüzü Şekil-5’de gösterilmiştir.



Şekil 5: Code Org Arayüzü

2.3.1.6 RoboMind

RoboMind Van Amsterdam üniversitesi tarafından geliştirilmiştir. Robo ile robotun hareketlerini kod bloklarıyla yönlendirmeyi sağlayan ve programlamanın temelini öğreten görsel ve eğlenceli programlama aracıdır. İlk 30 gün İnternet üzerinden ücretsiz kullanma imkânına sahip olan RoboMind’in kullanılabilmesi için kurulununun yapılması gerekmektedir. Türkçe dil desteği de bulunmaktadır (RoboMind, 2019). RoboMind programının ara yüzü Şekil-6’da gösterilmiştir.



Şekil 6: Robomind Arayüzü

2.3.1.7 Tynker

Tynker, 7 yaş ve üzeri kullanıcılara hitap eden, kullanıcıların kod yapılarını ve programlama mantığını görsel olarak ifade eden 2013 yılında Tynker Şirketi tarafından geliştirilmiş blok tabanlı görsel programlama aracıdır. İnternet tarayıcısı üzerinden online olarak çalışma imkanı sunmaktadır. Başlangıç, Orta ve İleri düzey gibi farklı düzeylere göre tasarlanmış uygulamalarda görevleri yerine getirirken kodların doğru biçimde sürüklenip bırakılmasıyla gerçekleştirilir. Tynker görsel programlama ortamında, Javascript ve Python dillerinde blok tabanlı olarak kod yazımı gerçekleştirilmektedir. Tynker üzerinde robotik, müzik, hikayeleştirme, animasyon, simülasyon, oyunlar ve tasarım temaları adı altında projeler yapılmasına fırsat sunar (Tynker, 2019). Tynker programının ara yüzü Şekil-7’de gösterilmiştir.



Şekil 7: Tynker Arayüzü

2.3.1.8 Kodable

Kodable, 2012 yılında Louisville Üniversitesinden Grechen Huebner ve John Mattingly tarafından geliştirilmiş görsel programlama aracıdır. Özellikle Okul öncesi dönemden başlayarak (4-7 yaş) ilkokul 5. Sınıf düzeyine kadar olan kullanıcıların problem çözme ve algoritmik düşünme becerilerini geliştirmek amacıyla tasarlanmıştır. ABD’de ilkokul düzeyinde çok fazla kullanılmaktadır (BilgeAdam, 2016). Toplam 90 seviyeden oluşan Kodable’ ın ilk 30 seviyesi kullanıcılar tarafından ücretsiz olarak kullanılabilir. Kodable üzerinde sınıf için uygulama yapmadan önce bir öğretmen veya ebeveyn üyeliği ile sınıf oluşturulur, daha sonra öğrencilere bu sınıfın kodu verilerek öğrencilerin giriş yapması sağlanır (Kodable, 2019). Kodable programının ara yüzü Şekil-8’de gösterilmiştir.



Şekil 8: Kodable Arayüzü

2.3.1.9 Scratch Jr.

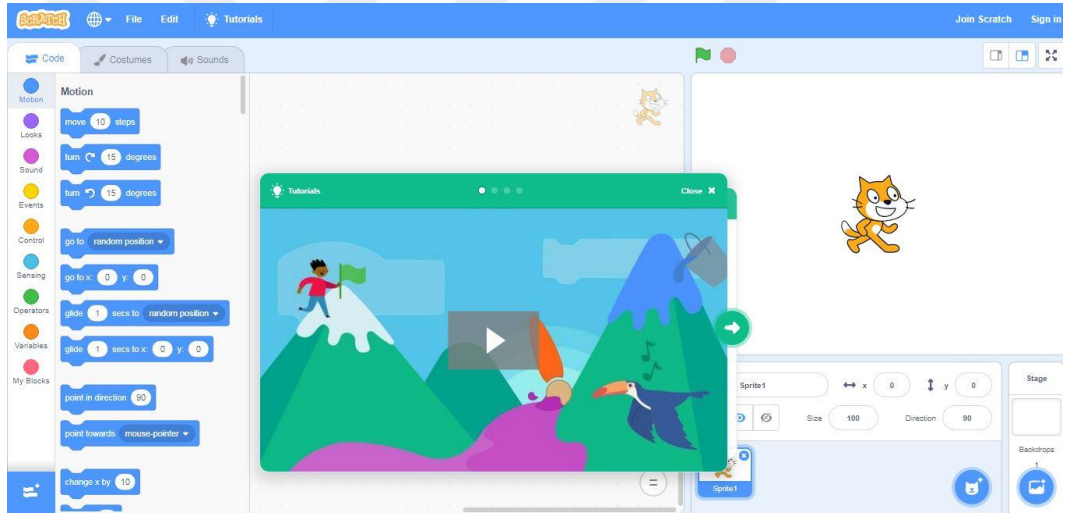
Scratch Jr., 2014 yılında MIT Media Laboratuvarında Tufts Üniversitesi ve Playful şirketinin ortak projesi olarak geliştirilmiştir. 5-7 yaş aralığındaki bireylerin kod bloklarını sürükleyip bırak yöntemiyle, karakterlere hareket ve etkileşim özelliği kazandırmalarına, kendi tasarımlarıyla özgün dijital oyun, etkileşimli hikâyeler ve animasyonlar oluşturmalarına ve bu Scratch Jr. üzerinde yaptıkları projeleri web ortamında birbirleriyle paylaşmalarına fırsat veren blok tabanlı görsel programlama aracıdır. Öğrencilerin yaratıcılık, algoritmik düşünme ve programlama becerilerinin gelişmesinde oldukça etkilidir (Scratch Jr., 2019). Scratch Jr. programının ara yüzü Şekil-9’da gösterilmiştir.



Şekil 9: Scratch Jr. Arayüzü

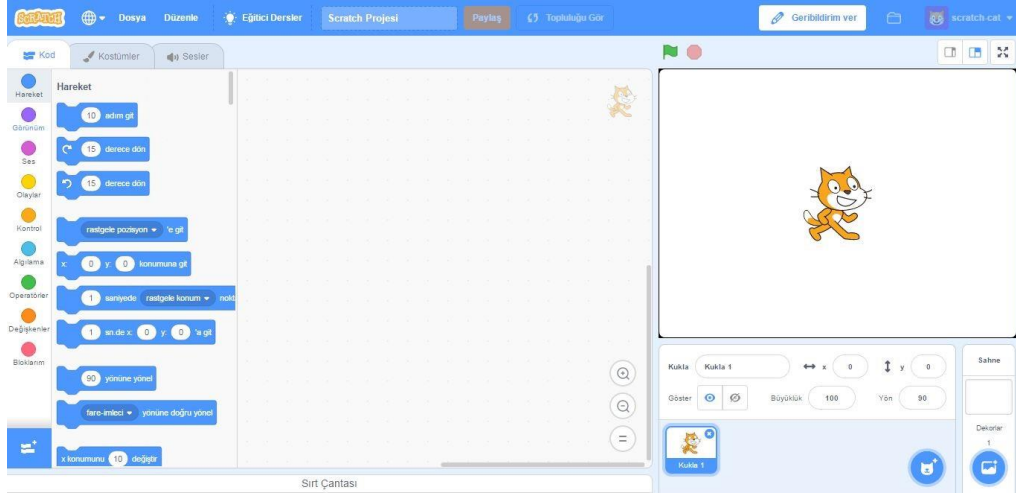
2.3.1.10 Scratch

Scratch, 2003 yılında Lifelong Kindergarten grubu Mitchell Resnick ve arkadaşları tarafından MIT (Massachusetts Teknoloji Enstitüsü) Medya laboratuvarında geliştirilen, üzerinde yapılan projeleri barındıran, 40 dan fazla dil desteği olan, çoklu ortam desteği sunabilen, web ortamında paylaşılabilir özelliği bulunan, sürükle-bırak etkileşimiyle kod yapılarını kolayca bir araya getirebilmeyi sağlayan, her yaş ve düzeye hitap edebilen ücretsiz, blok tabanlı ve görsel programlama dilidir (Scratch, 2019). Scratch, bireylerin etkileşimli hikâyeler, animasyonlar, dijital oyunlar, simülasyonlar ve eğlenceli oyunlar geliştirebilmesine fırsat veren görsel programlama ortamıdır. Scratch programının ara yüzü Şekil-10'da gösterilmiştir.



Şekil 10: Scratch Arayüzü-1

2.3.2 Scratch Programlama



Şekil 11: Scratch Arayüzü-2

Scratch, 2003 yılında Yaşam boyu Okul Öncesi Grubu (Lifelong Kindergarten Group), Mitchell Resnick ve arkadaşları tarafından Massachusetts Teknoloji Enstitüsü MIT Medya Laboratuvarında geliştirilmiştir. Scratch programı programlama dili olduğu gibi, üzerinde projeleri barındıran ve yapılan bu projelerin paylaşılabilirdiği tamamen ücretsiz görsel programlama aracıdır. Yaklaşık 1, 5 milyon üye kayıtlıdır. 9 milyonun üzerinde projeyi barındıran web portalı ve etkileşimli ara yüzüyle Scratch görsel programlama her yaş ve düzeyde öğrencinin kodlamayla tanışmasına imkân sağlamaktadır. 2007 yılında tüm dünyada yaklaşık 40 dil desteği ile yayınlanan görsel programlama aracı olan Scratch'ın her yaş aralığından kullanıcısı mevcut olmakla birlikte en aktif ve yoğun kullanma yaş grubu 9-18 olarak bilinmektedir; ancak her yaş ve düzeyde bireyler Scratch üzerinde projeler oluşturabilmekte ve paylaşımına sunabilmektedir (Scratch about, 2019).

Scratch programında karmaşık kodlar yerine kod bloklarında sürükleyip bırak özelliği bulunmaktadır. Program tasarlamak ve kodlama yapmak için bu blokları alt alta sürükleyerek bırakmak yeterlidir. Bu bloklar farklı renklerde tasarlanmıştır ve her kod bloğu aslında farklı renklerde. "Hareket" blok kodları mavi rengi temsil ederken, "Görünüm" blok kodları mor renkli, "Ses" blok kodları pembe renkli, "Kontrol" blok kodları sarı renkte, "Kalem" blok kodları koyu yeşil renklerinde iken, "Algılama" blok kodları açık mavi renginde, "Operatörler" blok kodları yeşil renkte ve "Değişkenler" blok kodları da turuncu renk ile tasarlanmıştır. Bu blok kodların üzerinde ne işe yaradığı yazılıdır. Örneğin "10 adım git", "kostüm 2 kostümüne geç", "miyav sesini

çal” ya da “ rengine deđiyor mu” gibi. Bu kodlar bir programlama dilindeki karmaşık kod (syntax) yapısından daha çok günlük konuşma diline uygun şekilde tasarlanmıştır. Ayrıca sadece birbiriyle uyumlu bloklar birbiriyle kenetlenmektedir ve böylece hata yapma olasılığı daha da azalmaktadır. Aynı zamanda Scratch görsel programlama aracının çoklu ortam desteđi, etkileşimli olması ve web ortamında paylaşılabilir olması gibi özelliklerle tasarlanmış olması öğrencilerin programlamayı öğrenmesinde ve algoritmik düşünme becerilerinin kazandırılmasında oldukça etkilidir. Farklı derslerde kullanılabilir yapıda olmasının da öğrencilerin motivasyonunu arttırmada etkili olduđu söylenebilir.

2.3.3 Scratch Programlama ile İlgili Yapılmış Çalışmalar

Alan yazın taraması yapıldığında Scratch programlama ile ilgili yapılmış birçok çalışma vardır. Bu çalışmaların sonuçlarına göre farklı etkiler araştırılmıştır. Çatlak ve diđerleri (2015), Scratch yazılımı ile programlama öğretiminin durumu üzerine yaptıkları araştırmada programlama öğretiminde Scratch görsel programlamanın kullanımının öğrencilerin ilgilerini çektiđi, motivasyonlarının olumlu yönde deđişiklik gösterdiđi sonucu ortaya çıkmıştır. Bu inceleme sonucunda, Scratch programlamanın, algoritma ve programlama temellerinin öğretiminde ve ayrıca problem çözme ve yaratıcı düşünme gibi becerilerin geliştirilmesinde etkili olduđu, motivasyon ve ilgileri üzerinde olumlu etkiye sahip olduđu ve son olarak da Scratch’ın farklı derslerde de kullanılabileređi ifade edilmiştir (Çatlak ve diđerleri, 2015).

Scratch yazılımını kullanarak öğrencilere temel programlamayı öğretmek ve öğrencilerin programlamaya karşı motivasyonlarını incelemek amacıyla Ouahbi ve diđerleri (2015) tarafından yapılan bir başka çalışmada Scratch programı kullanılarak yapılan programlama eğitiminde, öğrencilerin başarıları ve motivasyonları olumlu yönde gelişim göstermiştir. 6. sınıf öğrencileriyle yapılan bir araştırmada programlama öğretiminin öğrencilerin problem çözme becerilerini olumlu yönde etkilediđi ortaya çıkmıştır (Lai ve Yang, 2012). Calder (2010) ise yaptıđı çalışmada, görsel programlamanın öğrencilerin motivasyon düzeylerini arttırdığını ve problem çözme becerilerinin geliştirdiđini belirtmiştir. Fessakis, Gouli ve Mavroudi (2013) 5-6 yaş grubundaki öğrencilerle yaptıkları çalışmada da programlama öğretiminin öğrencilerin problem çözme becerilerini geliştirdiđini vurgulamışlardır.

Sanjanaashree ve Soman (2014) ise lise düzeyinde, Scratch programının yabancı dil öğretimiyle ilgili çalışmalarında, Scratch'ın öğrenme sürecini zevkli ve eğlenceli hale getirdiği, olumlu etkilerinin olduğu ve Scratch programının kullanışlı olabileceği sonucuna varmışlardır. Yükseltürk ve Altıok (2016), programlama eğitiminde Scratch aracı ile ilgili BT Öğretmen Adayların algılarına yönelik bir çalışmada, BT öğretmen adaylarının Scratch ile programlamaya ilişkin motivasyon ve kullanım kolaylığına ilişkin algılarının olumlu yönde olduğunu raporlamışlardır. Bu bulgulardan farklı olarak, Kalelioğlu ve Gülbahar'ın (2014) yaptıkları çalışmada ise ilköğretim 5. sınıf öğrencilerinin problem çözme becerilerinin üzerinde Scratch programlama etkisini araştırmışlar ve Scratch'ın öğrencilerin problem çözme becerilerine anlamlı bir katkısının olmadığı sonucuna varılmıştır. Sonuç olarak alan yazına bakıldığında Scratch görsel programlamanın öğrenme sürecinde olumlu ve olumsuz yönde etkileri olduğunu gösteren çalışmaların olduğu görülmüştür. Ancak Scratch programlamanın, 21. yy becerilerinden programlama, problem çözme, yaratıcılık gibi düşünme becerilerinin gelişiminde olumlu etkilerinin olduğu alanların olumsuzlara göre daha fazla olduğu söylenebilir.

2.4 Çocuklara Yönelik Programlama (4P)

Scratch, 2003 yılında Yaşam boyu Okul Öncesi Grubu (Lifelong Kindergarten Group), Mitchell Resnick ve arkadaşları tarafından Massachusetts Teknoloji Enstitüsü (MIT) Medya Laboratuvarı'nda geliştirilmiştir. Bu grup, Scratch programını tasarlariken yaratıcı öğrenmenin (Creative Learning) temel ilkelerini göz önünde bulundurmuşlardır. Bu temel İlkeler Resnick tarafından ortaya atılan Yaratıcı Öğrenme Modelinin 4P si; proje (project), akranlar (peers), ilgi (passion), oyun (play) olarak tanımlanabilir (Resnick, 2014). Yaratıcı Öğrenme-4P ilkesine dayalı bu aşamalar Şekil 12'de gösterilmiştir.



Şekil 12: Creative Learning (Yaratıcı Öğrenme-4P İlkesi)

2.4.1 Project (Proje)

İnsanlar yeni fikirler üretmek, prototip yada örneğini tasarlamak, inşa etmek, sürekli düzenlemelerde bulunmak gibi anlamlı projeler üzerinde aktif bir şekilde çalıştıklarında en iyi şekilde öğrenirler. Scratch üzerinde bireyler, sürece aktif olarak katılmakta ve sürecin ilk ögesi olarak animasyon, hikâye ya da oyun gibi yaratıcı projeler ortaya çıkarmaktadırlar. Böylece Scratch görsel programlama aracının bu özelliği sayesinde daha anlaşılır ve somut olarak programlamayı öğrenmeleri kolaylaşır (Scratch, 2019). Dale'in (1945) yaşantı konisine göre, tabakanın en altında en anlamlı öğrenmelerin yaparak yaşayarak öğrenme olduğu üzerinde durulmaktadır. Resnick'in (2014) proje modeliyle, öğrenciler kendi projelerini kendileri özgün ve farklı şekilde tasarladığında, yaparak yaşayarak öğrenerek öğrenme sürecinde aktif rol alır. Böylece bireylerin programlamayı öğrenmesi anlamlı hale gelir.

2.4.2 Peers (Akranlar)

Bireyin akranlarıyla aynı ortamda bulunması, öğrenme, insanlarla fikir alışverişinde bulunarak, projelerde işbirliği içinde olarak ve birbirlerinin çalışmalarının üzerine koyarak, yani katkı sağlayarak sosyal bir aktivite olarak açığa çıkar. Scratch programlama, öğrencilerin projelerini birbirleri ile paylaşabilmesinin yanında öğrencilerin birbirlerinin projelerine yorum yapmalarına imkân verir. Bunun yanında

öğrencilerin projelerini yeniden revize etmelerine olanak sağlayarak en iyi ürünün ortaya çıkmasını sağlayabilir. Aynı zamanda öğrencilerin kendi yaptıkları projeleri başkalarıyla paylaşmaları ve akranlarıyla iş birliği oluşturmaları onların özgüven duygularını da geliştirmelerini sağlayacaktır. Yapılandırmacı yaklaşımın önemli savunucularından olan Vygotsky'nin Sosyal yapılandırmacı anlayışına göre, bireyin anlamlı öğrenmesinde sosyal etkileşimin ve akranlarla olan iletişimin oldukça önemli olduğu vurgulanmaktadır. Vygotsky'e göre, bireyin öğrenme potansiyeli akranlarıyla iletişime ve etkileşim sürecine girdiğinde ortaya çıkmaktadır. Vygotsky öğrenmenin sosyal boyutunda Zone of Proximal (Yakınsak Gelişim Alanı) 'a göre bireylerin bir sonraki gelişim aşamasına geçebilmesi için akranlarıyla ya da yetişkinlerine ihtiyaç duyduğunu ifade etmektedir. Bruner de, eğitim öğretim sürecinde öğrencilerin aktif rol aldıklarında ve kendi öğrenmelerinin mimarı olarak kendi öğrenmelerinden sorumlu olduklarını ifade etmektedir (Wood, 1998). Ayrıca Vygotsky kadar olmasa da Bruner da bireylerin zihin gelişimlerinde dil ve sosyal etkileşimin önemli olduğuna vurgu yapmıştır (Wood, 1998).

2.4.3 Passion (İlgi ve Motivasyon)

Bireylerin ilgi alanlarıyla ilgili projelerde çalışma fırsatı olarak tanımlanabilir. İnsanlar önemsedikleri, ilgi duydukları konular üzerinde çalıştıklarında daha uzun süre ve daha sıkı çalışırlar, ilgi duydukları alanlarda çalıştıklarında ise özellikle süreç içinde daha iyi öğrenirler. Scratch üzerinde de değişik ilgi alanlarına sahip öğrenciler kendi ilgi alanlarına yönelik projeler oluşturabildiklerinde programlama ya karşı daha iyi bir tutum sergiler. Csikszentmihalyi'nin (1977) Akış Teoremi'ne göre (Flow Theory) bireylerin yetenekleri ve ilgi alanları doğrultusunda yapabilecekleri aktivitelere ilişkin motivasyonlarını toplayarak tamamen odaklanmaları sürecidir. Csikszentmihalyi'e (1977) göre birey tamamen ilgi odağı olarak yaptıkları eylemlere tamamen odaklanmakta, çevredeki uyarıcıları göz ardı ederek akış (flow) içine girmektedirler. Csikszentmihalyi'e (1977) göre bu akış; bireyin belirli koşul altında algıladığı ve bireyin yetenekleri arasında denge oluşması olarak ifade edilmektedir. Csikszentmihalyi'e (1988) göre akış içindeki birey, tamamıyla ilgi duyduğu aktiviteye odaklandığında zamanın nasıl geçtiğini anlamaz ve bu süreçten keyif alır. Böylece bireyin yapılan faaliyete odaklanarak, ilgi duyarak ve merak duygusunun ön plana çıkarılmasıyla faaliyete ilişkin beklenenden daha fazla verim alabileceği ve

performansının artacağı söylenebilir (Snyder ve Shane, 2009). 1987 yılında Keller tarafından geliştirilen ARCS (Attention, Relevance, Confidence, Satisfaction) Motivasyon Modelinde de, öğrenme sürecinde motivasyonun önemine değinmektedir. Keller (1987), ARCS Motivasyon tasarım modelinde öğrencilerin gelecekte gerçek yaşam koşullarında kullanabilecekleri bilgiyi öğrenmelerinin onları motive etmede önemli rol oynadığı vurgulanmaktadır. Keller (1987) ARCS Modeline göre, öğrenme sürecinde motivasyonu ve öğrenme güdüsünü arttıran 4 boyutu temel olarak ele almıştır. Bunlar; Dikkat, İlişki, Güven ve Memnuniyet olarak ifade edilmiştir (Driscoll, 1993). Keller 'in güdülenmeye dayalı ARCS Modeline göre, öğretim sürecinde bilginin öğrencilerin dikkatini çekecek şekilde sunulması, öğrencilere bilgiyi direkt vermek yerine bilgiyi özümseyecek şekilde ilişkilendirilerek verilmesi ve anlamlandırılması, öğrencilere güven duygusunun hissettirilmesi ve aktif katılımlarının sağlanması ve sürecin sonunda bireyin öğrenme çıktılarıyla memnun olması dikkate alındığında öğrenmenin kalıcı hale gelmesi beklenir. Ayrıca literatürde yapılan incelemeler sonucunda, Arcs Motivasyon Modelinin öğrencilerin akademik başarıları üzerinde de olumlu etkisi olduğu ve anlamlı öğrenmelerinin kalıcı hale geldiği sonuçlarının olduğu görülmüştür (Cengiz, 2009; Chyung, Winiecki ve Fenner, 1999; Feng ve Tuan, 2005).

2.4.4 Play (Oyun)

Öğrenme, yeni şeyleri denemek, farklı materyallerle haşır neşir olmak, yapabileceği faaliyetleri test etmek, kendine güvenerek risk almak ve yinelemek gibi oyun temelli deneyimleri içerir (Resnick, 2009). Scratch üzerinde de yapılan oyunu test etme, çalışıp çalışmadığını değerlendirme, kodlama hatalarını görme yeniden düzenleme ve çalışan oyunu oynama gibi deneyim sağlayarak öğrenciler programlamaya karşı daha olumlu bir tutum sergiler. Resnick ve diğerlerine (2014) göre oyun, Scratch üzerinde yapılan kodları bir araya getirme ve çalışıp çalışmadığını deneme ve yeni şeyler keşfetme olarak açıklanmaktadır (Knierzinger, Gradinarova, 2016; Resnick, 2014). Böylece bireyler yaptıkları hataları görebilecek, düzeltebilecek ve daha anlamlı öğrenmeler meydana gelecektir. Ayrıca bireyler kendileri deneyimlediklerinde daha eğlenceli öğrenmelerin gerçekleşmesi beklenir.

2.5 Programlama Öğretiminin Faydaları/Avantajları

21. yüzyıl becerileri olarak tanımlanan problem çözme becerisi, eleştirel düşünme becerisi, yaratıcı düşünme, yenilikçi ve üretken olabilme, analitik düşünebilme, analiz veya sentez yapabilme, medya okuryazarlığı gibi beceriler; günümüz öğrencilerine kazandırılması beklenen temel becerilerdir. Programlama da problem çözme becerisi olarak tanımlanabilir. Son yıllarda yapılan çalışmalarda programlama öğrenen bireylerin karşılaştıkları problem durumlarına karşı çözüm üretme becerilerinin geliştiği görülmüştür (Coravu, Marian ve Ganea, 2015; Resnick ve Silverman, 2005). Programlama becerisi, herhangi bir problem durumuyla karşılaştığımızda o duruma ilişkin çözüm basamaklarını belirleyebilmede etkilidir. Programlama sürecinde algoritmalar geliştirilerek öğrencinin bilişsel süreçleri aktif olarak kullanması beklenir. Algoritma, problem durumunun adım adım çözülme sürecidir. Bu süreçte problemin çözümü için ihtiyaç duyulan adımlar mantıksal olarak sıralı bir şekilde ifade edilir.

Algoritma, programlama geliştirmenin yanı sıra günlük hayatta karşılaşılan sorunları daha kolaylıkla çözme olanağı sağlar. Burada problemin çözümü için aşamaların zihinde tasarlanması, algoritmik ve sistematik düşünme becerisinin kullanılması gerekmektedir. Bu nedenle programlama problem çözme becerisi olarak da ifade edilebilir (Karabak ve Güneş, 2013; Munson, Moskal, Harriger, Karriker ve Heersink, 2011; Shin, Park ve Bae, 2013; Yükseltürk ve Altıok, 2015). Problem çözme modellerinin temel bileşenleri incelendiğinde, problem çözme süreci 4 aşamada ele alınmıştır: problemi anlama, plan hazırlama, planı uygulama ve kontrol etme (Polya, 1945). Polya'nın (1945) problem çözme aşamaları, programlamaya uyarlandığında ilk aşama problemin tanımlanması, açık ve net bir şekilde ifade edilmesidir. İkinci aşamada problemin çözüm basamaklarına ilişkin algoritmaların geliştirilmesi ifade edilebilir. Üçüncü aşamada problem çözme modelinde karşılık gelen planın uygulanmasına karşılık kodlamaların yapılması, programlama sürecinin uygulama aşaması olarak söylenebilir. Son aşama olan kontrol etme ve değerlendirme aşamasında ise; programın çalışıp çalışmadığının sınanması ve değerlendirilmesi olarak ifade edilebilir. Kontrol aşamasında farklı çözüm yolları belirlenir ve farklı problem durumlarına uyarlanabilir. Problem çözme becerisi olarak tanımlanan programlama becerisinde öncelikle problem durumunun ne olduğu tanımlanır, daha sonra problem durumuna ilişkin çözüm basamakları yani algoritması hazırlanır. Bir

sonraki aşamada hazırlanan algoritma uygun bir programlama diliyle kodlanır ve son basamakta çözüm yolları gözden geçirilir, programın çalışıp çalışmadığı ve problem durumunun çözüme ulaşip ulaşmadığı test edilerek süreç tamamlanır. Bu işleyiş göz önüne alındığında programlamanın öğrencilerin problem çözme becerilerini oldukça geliştirdiği ifade edilebilir.

Programlama, sadece bilgisayarda program yazımı ile sınırlı kalmamakla birlikte, öğrencilerin hayal gücünün geliştirilmesinde ve yaratıcı düşünme becerilerinin geliştirilmesinde etkilidir (Çatlak, Tekdal ve Baz, 2015; Shin, Park ve Bae, 2013). Programlama, aynı zamanda öğrencilerin mantıksal düşünme, yaratıcı düşünme, sistematik düşünme ve problem çözme becerisi gibi üst bilişsel düşünme becerilerinin geliştirilmesinde yardımcı olmaktadır (Fesakis ve Serafeim, 2009). Programlama ile birey uzamsal düşünme ve analitik düşünme becerisini de kazanır. Birey programlama yaparken ne yapacağını düşünme sürecinde uzun süreli hafıza kullanımı ile bilgiyi içselleştirir. Yapılan çalışmalara bakıldığında programlama sürecinde hem süreç hem de sonuç odaklı düşünme önemlidir (Akpınar ve Altun, 2014; Karabak ve Güneş, 2013).

Programlama becerisi ile bireyler gerçek hayatta karşılaştıkları problem durumlarına karşı çözüm yolları üretebilir. Bu çözüm yollarını önce zihinde tasarlar, aşama aşama planlama yapar. Burada birey programlama yaparken algoritmik ve sistematik becerisini ortaya koymaktadır. Aynı zamanda problemi çözme sürecinde analitik düşünme becerisiyle problem durumunu en küçük parçalara ayırarak, bu parçaları ayrı ayrı analiz etme durumunu kullanır. Böylece öğrencilerin analitik düşünme becerilerinin de gelişmesine katkı sağlayabilir. Scratch, Alice ve App Inventor gibi görsel programlama araçlarında öğrenciler etkileşimli hikâyeler veya animasyonlar tasarlama boyutunda öğrenciler hayal güçlerini kullanarak farklı tasarımlar yapabilir.

Programlama ile birlikte öğrencilerden uzamsal düşünme becerilerini ve hayal güçlerini kullanarak tasarımlar yapması beklenir. Bu anlamda yaratıcı düşünme becerilerinin gelişmesinde programlama önemli ölçüde fayda sağlayabilir. Aynı zamanda programlama eğitimi, ülkenin gelişiminde de katkı sağlamakta ve programlama eğitimi verilen ülkelerde üretken nesillerin yetiştirilmesinde önemli rol oynamaktadır.

Programlama ile birlikte bireyler yenilikçi öğrenme deneyiminde gelişen teknolojiyi daha etkin kullanarak 21. yy teknolojilerinin gerektirdiği becerileri kazanabilmektedir. Örneğin çocuklar erken yaşta algoritmik düşünme becerilerini kazanarak farklı alanlarda sorunlarla karşılaşma durumunda problem çözmeleri kolaylaşacaktır. Ezberleyen, öğrendiklerini sorgulamayan, herhangi bir eleştiride bulunmayan bireyler yerine 21. yy becerileri ile donatılmış yeni bireylerin yetiştirilmesi için erken yaşta kodlama eğitimi ile bu becerilerin geliştirilmesi hedeflenmektedir. Programlama, aynı zamanda öğrencilerin yaptıkları projelerle somut ürünler ortaya çıkarmalarını ve kendilerine olan öz güvenlerinin gelişmesini sağlayacaktır. Ayrıca programlama becerisi ile çocukların kendi tasarladığı projeleri başkalarıyla paylaşma ortamının olması ve geliştirdikleri projeleri çevresiyle paylaşması onların motivasyonunu arttıracaktır.

2.6 Eğitim Müfredatlarında Kodlamanın Yeri

Avrupa'daki ülkelerde programlama kavramı farklı şekillerde yer almaktadır. Örneğin Belçika'da "bilgi-işlemsel düşünme ve programlama" (computational thinking and programming); Polonya, İngiltere ve Norveç 'te Kodlama ve Bilgisayar Programlama (Coding and Computer programming), İrlanda ve Fransa'da Kodlama (coding); Bulgaristan'da "algoritmik problem çözme ve programlama" (algorithmic problem solving and programming), İspanya'da "programlama, algoritma ve robotik" (programming, algorithmic and robotics), Estonya'da "programlama, robotik" (programming, robotics) ve İngiltere'de "bilgi işlem" (computing) olarak eğitim müfredatlarında geçmektedir (Balanskat ve Engelhardt, 2014). İngiltere, Estonya ve Finlandiya gibi bazı ülkelerde kodlamanın okul müfredatına eklenmesi için ciddi girişimlerde bulunulmuştur. Amerika Birleşik Devletleri'nde ulusal bir müfredat bulunmamasıyla birlikte eğitim sistemlerinde kodlamayı ve bilgisayar becerilerini, New York ve Los Angeles gibi yerlerde müfredatlarına programlamayı entegre etmiştir. ABD'de yönetim programlama öğretiminin yaygınlaştırılması gerektiğini savunmuş ve buna dair çalışmalar yapmıştır.

European Schoolnet (2015) yayını, 16 Avrupa ülkesinin müfredatta Avusturya, Macaristan, Danimarka, Fransa, İspanya, Portekiz ve Bulgaristan da dahil olmak üzere ulusal, bölgesel veya yerel düzeyde kodlamayı entegre ettiğini bildirmiştir. Ülkelerin çoğunluğu programlama ile öğrencilerin mantıksal düşünme becerisi ve problem

çözme becerilerini, böylece 21.yy becerilerini geliştirmeyi hedeflemektedir. Estonya’da programlama eğitimi 2012 den bu yana “Proge Tiiger” adlı bir pilot uygulamayla ilköğretim programına dâhil edilmiştir (Demirer ve Sak, 2016).

2014 yılında İngiltere, bilgisayar programlamayı okulun her kademesinde zorunlu olarak tutan programlama odaklı yeni müfredat uygulamasına geçmiştir; İngiltere ortaöğretim müfredatın da yapacağı yenilikçi uygulamadan önce 2014 yılını "Kodlama yılı" olarak ilan etmiş çocuklarda kodlama bilincinin oluşturulması amacıyla harekete geçmiştir. Avrupa’da bu konuda öncelik göstermiş ve 5-16 yaş aralığındaki çocukların uzamsal ve analitik düşünme becerilerini geliştirmeyi hedeflemiştir (Balanskat ve Engelhardt, 2014).

Finlandiya ‘da ise 2016 dan itibaren 7-16 yaş aralığındaki bireylerin zorunlu olarak programlamanın temellerini öğrenmeleri amacıyla müfredatlarına eklenmiştir. Böylece öğrencilerin programlama becerilerinin geliştirilmesi desteklenmektedir. Danimarka’da ise programlama eğitimi 1-10. Sınıf sınıflarda matematik ve fen bilimleri derslerinde verilmesiyle ilgili bir uygulama yapılmıştır (Balanskat ve Engelhardt, 2014).

Ülkemizde de programlama eğitiminin küçük yaşlardaki öğrencilere kazandırılması amacıyla çeşitli yerlerde kodlama atölyeleri açılmaya ve çalışmalar yapılmaya başlanmıştır (Kodlama Atölyesi, 2017; MEB, 2017). Milli Eğitim Bakanlığının yaptığı çalışmayla Bilişim Teknolojileri ve Yazılım dersi 5. ve 6. Sınıf düzeylerinde zorunlu 7. ve 8. Sınıf düzeyinde seçmeli olarak güncellenmiş ve dersin eğitim müfredatının içerisine programlama eğitimi de dâhil edilmiştir. Böylece öğrencilere temel programlama becerilerinin kazandırılması hedeflenmektedir (MEB, 2017). Programlama Eğitimi sadece ilköğretim ve ortaöğretimle kalmamış, üniversite lisans eğitiminde öğretim müfredatı içerisinde de yer almaktadır. Ülkemizdeki çeşitli üniversitelerde lisans ve yüksek lisans eğitimlerinde programlama ağırlıklı müfredatla birlikte öğrencilerin problem çözme ve analitik düşünme becerilerinin geliştirilmesi sağlanabilmektedir.

2018’de güncellenen 5. ve 6. Sınıf Bilişim teknolojileri ve Yazılım dersi öğretim programı müfredatına ilişkin problem çözme ve programlama ünitesine ilişkin konular ve kazanımlardan bazıları Ekler (Ek H) kısmında gösterilmiştir.

2.7 Senaryo Tabanlı Öğrenme (STÖ)

Senaryo Tabanlı Öğrenme (STÖ), belirlenen hedef kazanımların bir senaryo çerçevesinde yapılandırılmasıyla öğrencinin kendini o senaryonun bir parçası olarak gördüğü ve karşılaştığı problemleri çözerek kazanımlara ulaşması esasına dayanan bir öğrenme yaklaşımıdır (Schank, 2000). STÖ ile gerçek hayattaki problem durumunun sınıf ortamına taşınarak öğrencilere problem üzerinde çözüme ulaşmaları ve düşünme becerilerinin geliştirilmesi hedeflenmektedir. STÖ ortamlarında probleme dayalı öğrenme stratejileri gibi aktif öğrenme stratejilerini desteklemek amacıyla etkileşimli senaryolar kullanılır. Senaryolar üzerinde yoğunlaşan öğrenciler, problem çözme sürecinde analitik düşünme, eleştirel düşünme ve karar verme gibi üst düzey becerilerini geliştirmektedir. STÖ genelde çözmeleri gereken karmaşık yapılandırılmış veya basit bir probleme dayanan bir tema üzerinde yavaş yavaş ilerleme kaydeden öğrencilerin öğrenme sürecini kapsar. Bu süreçte öğrenciler gerçek hayattaki problem çözme becerileri ve eleştirel düşünme becerileri ile konu ile ilgili bilgilerine ve deneyimlerine başvururlar. STÖ, öğrencilere süreçteki her bir aşamada geri bildirim fırsatı sağlar ve ayrıca öğrencilerin hayal güçlerini harekete geçirerek onların yaratıcı düşünme becerilerinin gelişmesini sağlar (Clark, 2012). Öğrenciler öncelikle gözlemlediği senaryoları kendilerince içselleştirir ve o senaryo kurgusuna kendilerini dâhil ederler, böylelikle öğrenciler öğrenme sürecine aktif olarak katıldıklarında anlamlı öğrenmeler gerçekleşmektedir (Rybarczyk ve diğerleri, 2007).

STÖ sürecinde öğrenci, kendisini o senaryonun içinde hissetmekte ve senaryonun bir karakteri olarak karşılaştığı problemi çözme sürecine girmektedir. Öğrenciler bu süreçte iş birliği içerisinde çalışarak birbirlerinin öğrenmelerini destekleyebilir. Ayrıca öğrencilerin araştırma yaparak sonuca ulaşmaları beklenmektedir. Gerçek hayatta karşılaştıkları problem durumlarında öğrencilerin bir problem üzerinde düşünme, sorgulayabilme, bilgilerini gerçeğe benzer durumlara uyarlayabilme, eksikliklerle karşılaştığında araştırma yapabilme ve problem durumuna çözüm üretebilmelerini sağlar (Schank, 2000; Veznedaroğlu, 2005). Öğrencilerin izlediği senaryolar sayesinde, okulda öğrendikleri ön bilgileri ile gerçek yaşam arasındaki bağ kurulduğunda, öğrendikleri bilgileri transfer edebilmeleri ve günlük hayatta karşılaştıkları problemleri çözebilmeleri daha kolay hale gelmektedir (Er-Nas, 2008; Süğümlü, 2009).

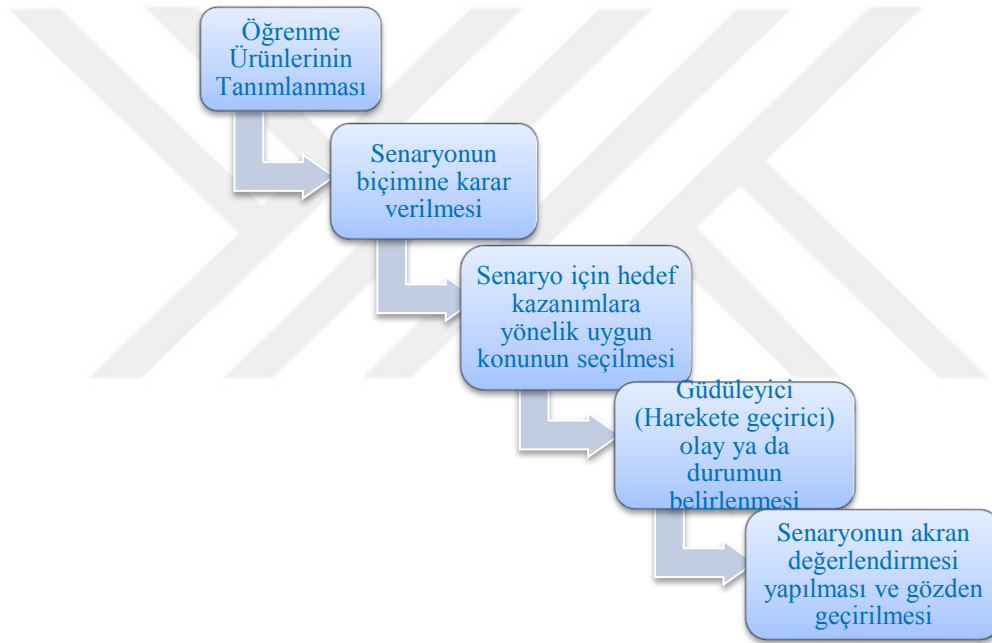
STÖ'nün en önemli noktası senaryodaki hedeflenen görevleri tamamlamaktır. Bunun yanında STÖ süreci, senaryonun yazılı, işitsel ya da görsel olarak sunulması, öğrencinin senaryoyu kendi zihninde yapılandırması ve ayrıca öğrencinin bu süreçte kendini değerlendirerek öz değerlendirme yapmasını içine alan geniş bir süreçtir (Schank, 2000). Errington'a (2003) göre STÖ; hedefe dayalı, oyun temelli, sorgulama temelli, beceri geliştirme temelli senaryolar olarak çeşitlilik gösterir. Öğrenme konusunun içeriğine ve hedef kazanıma ilişkin uygun senaryoların seçilmesi etkili öğrenmeyi sağlamak adına önemli bir husustur.

STÖ, duruma dayalı öğrenme teorisi ilkelerine dayanmaktadır. Bu teori ilkeleri, öğrenme sürecinde bilişsel süreçlerin aktif olarak kullanılacağını ve böylece öğrenmenin en iyi şekilde gerçekleşeceğini ifade etmektedir (Kindley, 2002). STÖ birçok alana adapte edilebilir ama özellikle geleneksel sınıf ortamlarında öğrencilerin bu senaryoları deneyimlemeleri ve öğretmenlerin bu tarz ortamlar yaratmaları her zaman mümkün olmayabilir. Ancak fırsatlar sağlayarak gerçek dünyaya ait uygulamalarla benzetim yapıldığında daha etkili bir şekilde çalışır ve senaryolar daha çok başarı gösterir.

Dil öğretiminde, ulaşım alanında, kalıtım ve genetik alanında, trafik eğitiminde, tarım faaliyetleri alanında, bitki yetiştirilmede, mühendislik alanında, hemşirelik alanında ve sağlık alanında hastalıkların teşhisi ve kalp cerrahi ameliyatlarına kadar geniş kapsamlı konular etrafında senaryolar geliştirilmektedir (Akbiyık ve Tokem, 2018; Carney ve diğerleri, 2011; Clark, 2012; Korhan, Yılmaz, Ceylan, Haynes, Spence ve Lenze, 2009; Mariappan, Shih ve Schrader, 2004; Süğümlü ve Yaman, 2009; Yaman, 2005). Bu alanlarda özellikle soyut konular için gösterilen animasyon ya da simülasyon biçiminde verilen senaryolar çerçevesinde öğrenmeler daha anlamlı hale gelmektedir. Örneğin öğretmenlik ve mesleki uygulamalarda, hukuk ve sağlık alanında yapılan uygulamalarda, vb. alanlarda senaryolar öğrenme boyutunda oluşan soyut boşlukları doldurabilir. STÖ ilk olarak 19. Yüzyıl sonlarında Harvard Üniversitesi Hukuk Fakültesinde hukuk dersinin öğretiminde uygulanmaya başlanmıştır (Kimball, 1995). STÖ süreç ya da sonuç odaklı değerlendirmenin bir parçası olarak kullanılabilir (Clark, 2012). Genellikle programlama gibi karmaşık durumlardaki karar verme ve eleştirel düşünme gerektiren görevlere uygulandığında STÖ oldukça işe yarayabilir. Yapılan araştırmaların sonucunda fen öğretiminde

kavram yanlışlarının giderilmesinde ve somutlaştırılmasında STÖ'nün etkili ve verimli bir öğretim aracı olduğu saptanmıştır (Kocadağ, 2010). Yabancı dil öğretiminde de yine STÖ'nün öğrencilerin yazma ve konuşma becerilerinin geliştirilmesinde hayal güçlerini ve yaratıcılıklarını arttırdığı sonucuna varılmıştır (Clark, 2012). Aynı zamanda öğrencilerin sosyal etkileşimlerine dayalı performanslarını arttırarak motivasyon düzeylerini yüksek seviyede tutmak için de STÖ'nün etkili olduğu ifade edilmektedir (Clark, 2012; Kindley, 2002).

Schank'a (2000) göre, STÖ belli aşamalarda gerçekleşir. Bu aşamalar Şekil 13'de gösterilmiştir (s. 191).



Şekil 13: Senaryo Tabanlı Öğrenme (STÖ) Aşamaları

2.8 Senaryo Tabanlı Öğrenme (STÖ) Aşamaları

2.8.1 Öğrenme Ürünlerinin Tanımlanması

Schank, Berman ve Macperson'a (1999) göre STÖ, hedef beceriler ve öğrenme kazanımları ile ilgilidir. STÖ öğrencilerin belirlenen bir hedef doğrultusunda istenilen davranışları gerçekleştirmek ve uygulama performanslarını geliştirmede etkili olmaktadır. Öğrencilerden senaryonun tamamlanmasında aktif rol almaları için senaryoda istenilen hedef kazanımları önceden belirlemek önemlidir. Bu nedenle onları öğrenmeye güdüleyecek durumu oluşturmaları için öğrenme ürünlerinin

tanımlanması ve geriye doğru aşama aşama açıklanması da anlamlı öğrenme açısından önemlidir. Öğrenme ürünlerinin geriye doğru aşama aşama sönümlenmesi, tamamlanılması istenilen görevlerin aşama aşama ilerleme kaydedilerek tamamı uygulamalı göreve dönüştürülmesidir (Clark, Nguyen ve Sweller, 2006). Clark ve diğerlerine (2006) göre problem durumuna ilişkin verilen görevlerin aşama aşama yönergelerle açıklanması, etkili öğretimin gerçekleşmesinde önemli rol oynamaktadır. Bu durum öğrencilerin öğretim sürecinde hem anlamlı öğrenmesinde hem de hiç bilmedikleri bir beceri durumuna ilişkin uzmanlık kazanmasında oldukça etkilidir (Sweller ve Cooper, 1985; Sweller, 1988). Örneğin programlama gibi soyut konularda öğrencilerin aktif katılımını sağlamak ve etkili öğrenmeyi gerçekleştirmek amacıyla istenilen hedef becerilerin aşama aşama açıklanması oldukça faydalıdır. Öğrencilere verilen senaryoda hedef kazanımların ve öğrenme ürünlerinin aşama aşama değil de doğrudan sunulduğunda bu durum onlar için karmaşık gelebilir. Ayrıca uygulamalı görevleri öğrencilerin bilişsel olarak anlamlandırmada zorluk yaşamalarına neden olabilir. Bu nedenle, hedef becerilerin ya da öğrenme ürünlerinin bireylerin bilişsel kapasitelerine uygun olarak geriye doğru aşama aşama verilmesi, etkili öğretimi gerçekleştirmek adına bilişsel zorlukları ve bilişsel yük durumunu ortadan kaldırabilir. Böylece öğrenmenin karmaşık halden çıkarak daha açık ve anlaşılır hale gelmesi beklenmektedir (Clark ve diğerleri, 2006).

Ayrıca Öğrencilerden yapılması beklenen hedef kazanımlarla ilgili açıklamalar senaryoda açıklayıcı ve anlaşılır biçimde verildiğinde senaryo öğrencilerin motivasyonunu arttıracaktır. Böylece öğrenciler o senaryodan hareketle o görevlerin tamamlanmasında daha istekli olacaktır. Senaryo, öğrenciler izlediklerinde onların hayal güçlerini harekete geçirecek şekilde tasarlandığında, öğrencilerin belirlenen hedef kazanımları gerçekleştirirken aynı zamanda yaratıcılık becerilerinin de artması beklenir (Schank, Fano, Bell ve Jona, 1993; Schank, 2000; Veznedaroğlu, 2005).

2.8.2 Biçimine Karar Verilmesi

Senaryonun öğrenme ortamına entegrasyonunda hangi biçimde sunulacağı önceden planlanmalıdır. Senaryonun yüz yüze mi yoksa sanal ortamlarda mı verileceği, senaryoda hangi ortam (görsel, işitsel ve çeşitleri) kullanılacağı ve başka farklı kaynaklara ihtiyaç duyulup duyulmayacağına karar verilmesi gerekir. Senaryolar öğrencilere birebir anlatım yöntemi kullanılarak yüz yüze olabileceği gibi fotoğraf,

video ya da etkileşimli animasyonlar şeklinde de sunulabilir. Senaryoların birden fazla duyuya hitap edebilmesi öğrenmede kalıcılığı artırmaktadır. Bu Dale'in (1946) yaşantılar konisinden de bilinen bir durumdur. Ayrıca sanal bir senaryo kullanılıyorsa vikiler ve tartışma forumları gibi başka yardımcı etkinlikler de kullanılabilir (Schank, 2000). Günümüzde Web 2.0 teknolojilerinin sunduğu yeniliklerle, senaryonun farklı sunum şekilleri de oluşturulabilir. Örneğin ActionBound gibi bir Web 2.0 uygulamasıyla hem senaryonun sunulduğu hem de misyonların tamamlanması bir arada aktif bir şekilde yürütülebilir.

2.8.3 Uygun Konunun Seçilmesi

STÖ içeriğinin belirlenmesinde, yaparak yaşayarak öğrenme deneyimi oluşturabilecek günlük hayatın içinden konuların belirlenmesi oldukça önemlidir. Ders alanında, günlük hayatta gerçekleşen olaylar ve ilgi çekici durumları kullanmak STÖ'nün etkililiğini arttırmasını sağlayabilir. Matematik, Fen Bilimleri, Yabancı dil ya da Programlama gibi daha soyut konuların öğretilmesinde senaryolar kullanıldığında daha etkili olması beklenebilir çünkü soyut konuların somutlaştırılmasına katkısı olabilir. Bilgisayar bilimi eğitimi kapsamında, özellikle algoritma ve programlama gibi soyut yapıların öğretilmesinde STÖ modelinin kullanılmasının potansiyel etkisi olabileceği varsayılabilir.

2.8.4 GÜdüleyici (Harekete Geçirici) Olay ya da Durumun Belirlenmesi

Bu aşama senaryonun etkili olması adına en önemli aşama olarak ifade edilebilir. Bu aşamada öğrencileri güdüleyici ve onlara geri bildirim için can alıcı noktalar belirlendiğinde etkinin artması beklenir. Bunu yapmanın en etkili yolu öncelikle senaryonun taslağını oluşturmaktır. Senaryo üzerindeki problem durumu senaryoda gömülü olarak verildiğinde, öğrencilere bu problem durumu üzerinde düşünceleri ve problem çözme becerilerini kullanmaları söylenir. Verilen senaryo kapsamındaki problemin günlük hayattaki durumlara denk gelmesi güdüleyici etkiyi artırabilir. Çünkü öğrenci duruma aşinadır ve olası çözümleri daha önce zihninde modellemiştir. Böylece, benzer bir durum karşısında yeni yaklaşımlar ve çözümler üretebilir. Bunun ortaya çıkabilmesi için de güdüleyici olayın seçilmesi ve sunulması oldukça önem taşımaktadır.

2.8.5 Senaryonun Akran Değerlendirilmesi ve Gözden Geçirilmesi

STÖ sürecinde önemli aşamalardan biri de değerlendirme sürecidir. Senaryonun hedeflenen doğrultuda gelişim göstermesi ve istenilen hedef kazanımların aktarılması amacıyla öğrencilerin akranlarıyla birlikte senaryoyu detaylı bir şekilde incelemeleri faydalı olabilir. Böylece öğrenciler, akranlarıyla birlikte değerlendirirken senaryoları tekrar gözden geçirerek içselleştirebilirler. Bununla birlikte öğretimin kalitesinin artırılması beklenir. Bu nedenle senaryoların gözden geçirilmesi oldukça önemli bir aşamadır.

2.9 Sanal Senaryo Tabanlı Öğrenme Araçları

STÖ, yapılandırılmış senaryoları öğrencilere sunma imkânı sağlayan öğrenme ortamıdır. PowerPoint uygulamaları, Adobe Captivate, Articulate Storyline, Stream ve Powtoon gibi araçlar etkileşimli senaryolar geliştirmek amacıyla kullanılabilir. STÖ ortamı geliştirmek, hedef kazanımların karmaşıklığına bağlı olarak zaman alıcı olabilir ama ortaya çıkan ve geliştirilen ürün, öğrencilere hem ilgi çekici hem de onları motive edici özgün bir öğrenme deneyimi sağlayabilir. Hedefe dayalı senaryolar, 1991 yılında Roger Schank tarafından geliştirilmiştir. Bu senaryolar metin, video, etkileşimli animasyon, simülasyon, haber yayını gibi çoklu öğrenme ortamları ile geliştirilmiş ana hikâyeler şeklinde olabilmektedir. Bu yaklaşımda öğrencilere direkt olarak bilgi ve beceri aktarımı yerine onların performanslarının gelişim süreci izlenir (Schradler, 2004). Schank'ın (2000) belirlediği aşamalara göre STÖ süreci şunları içerir:

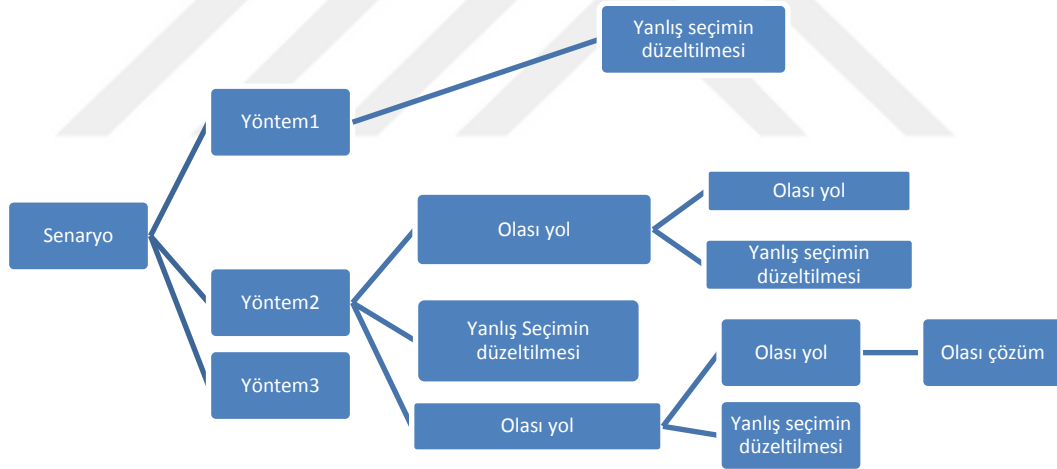
- i. **Öğrenme Hedefleri:** Akranlar arası işbirliğine dayalı grup çalışmaları dâhilinde bireysel sorumluluk alabilme, yapabileceği yetkinliklere karar verme, öğrenme sürecini organize etme yeteneği gibi öğrencilerden beklenen temel becerilerdir. Örneğin, Öğrencilerin Scratch görsel programlama aracını kullanarak piyano oyununu tasarlamaları ve etkileşimli piyanoda notaların ses ve kontrol kodlarının çalıştırılarak algoritmik, mantıksal, programlama becerilerinin geliştirilmesi ve yaratıcı düşünme becerilerinin geliştirilmesi Öğrenme hedefleri arasında gösterilebilir.
- ii. **Hedef Becerilerin Geliştirilmesi:** Senaryoda verilen durumdan yola çıkarak, öğrencilerin ne tür görevler alacağı, kazanımlara uygun olarak

hangi çalışmalarını oluşturacakları ve STÖ sürecinde nereden başlamaları gerektiğine karar verme aşamasıdır. Örneğin piyano oyununda hedeflenen becerilerin geliştirilmesine ilişkin öncelikle piyano tuşlarını tasarlamak, sonra dans eden karakterlerin oluşturulması gibi spesifik görevlerin ayrıntılı olarak açıklanması ve öğrencilerin her aşamada geri dönüt alabilmeleri amacıyla kazanımlara ulaşıp ulaşılmadığının tespit edilmesidir.

- iii. **İçerik Odağının Düzenlenmesi:** Hedef becerileri karşılayacak görevlerin belirtilerek senaryo içeriğinin oluşturulma aşamasıdır. Örneğin, “Öğrencilerden piyano oyununu tasarlama hedef becerisinin gerçekleşmesi konusunda, Erenin doğum günü partisinde arkadaşlarına çalacağı piyano çalışmamaktadır. Ancak Eren partiye katılan arkadaşlarıyla keyifli zaman geçirmek ve eğlenmek istemektedir. Bu konuda Eren ve arkadaşlarının eğlenmesi amacıyla onlara Piyano tasarlayabilir misin?” şeklinde senaryo içeriğinin oluşturulması beklenmektedir.
- iv. **Örtük (Gizil) Hedeflerin Belirlenmesi:** Öğrencilerden beklenen hedef becerilerin yanı sıra algoritmik düşünme, yaratıcı düşünme, problem çözme, mantıksal düşünme gibi 21. yy düşünme becerilerinin geliştirilmesi örtük hedefler olarak ifade edilebilir.
- v. **Doğrudan içerik:** Hedefe dayalı senaryolarda hedeflenen temel beceriler ve kazanımları da içinde barındıran senaryo içeriğinin yazılı, görsel ya da işitsel olarak doğrudan sunulması olarak ifade edilebilir.
- vi. **Etkinliklerin ve Öğrenme Çevresinin Düzenlenmesi:** Gerek asıl içeriğe gerekse örtük hedeflere dayalı içeriğe göre, etkinliklerin düzenlenmesi sağlanır. Öğrencilerin hangi rolleri ve sorumlulukları alacaklarına karar vermeleri için gereken düzenlemeler yapılır. Bütün adımları ile etkinlikler gerçekleştirilir.
- vii. **Değerlendirme:** Adımların uygulanmasından sonra, değerlendirme aşamasına geçilir. Nasıl başladı, nasıl gelişti, nereye ulaştık, ne kazandık, ne öğrendik, kimdik, rolümüz bize ne kattı, biz role ne kattık? gibi sorular

yöneltilerek, değerlendirme çalışması yapılır. Nasıl oldu, nasıl olmasını beklerdiniz? Siz olsaydınız ne yapardınız? gibi sorularla hedeflere ulaşıp ulaşılmadığı değerlendirilmiş olur (Yaman, 2005).

Senaryolardaki hedef kazanımlar öğrenciler tarafından anlamlandırılıp oluşturulduğundan, kendilerinin öğrenmede aktif oldukları ve öğrenmeyi anlamlı hale getirdikleri söylenebilir. Schank'a (2000) göre STÖ ortamının içeriğinin düzenlenmesinde gerçek hayattaki problem durumlarında problem çözme becerilerini kullanabilecekleri gerçeklik durumunun esas olduğu, öğrenmenin tıpkı gerçek hayatta sevdiğin bir sporu yapıyormuşsun gibi eğlenceli ve keyif verici olması gerektiği ileri sürülmektedir. Aynı zamanda öğrenmede hatalar için öğrencilere izin verilmeli, öğrencilerin hatalarından geri dönüt aldıklarında öğretimin daha etkili olduğu ve daha iyi öğrendikleri savunulmaktadır. Mariappan, Shih ve Schrader'den (2004) uyarlanan Senaryo Tabanlı Öğrenme Ortamı Örneği Şekil 14'te gösterilmiştir.



Şekil 14: Senaryo Tabanlı Öğrenme Ortamı Örneği (Mariappan, Shih ve Schrader'den (2004) uyarlanmıştır)

2.10 Hedefe Dayalı Senaryolar (Goal-Based Scenarios)

Roger Schank tarafından öne sürülen hedefe dayalı senaryolar (HDS), yaparak yaşayarak öğrenme ile durum tabanlı öğrenmeyi birleştiren yapılandırmacı öğrenmenin bir modelidir. HDS'nin amacı; kazanıma ulaşmak amacıyla güdülemek,

öğrenmeye destek oluşturmak ve yapılması beklenen temel becerilere odaklanmak amacıyla öğrenenlerin motivasyonunu sağlamaktır (Schank, 1992). Bu yaklaşımın temelinde, insan davranışlarının her boyutunda belirlenen hedeflere ulaşmayı kapsar. Eğer hedefler, bireyin düşünce sürecinin temeliyse öğrenmenin aynı zamanda hedefe dayalı bir alanda olması gerektiğini savunur (Schank, 1992).

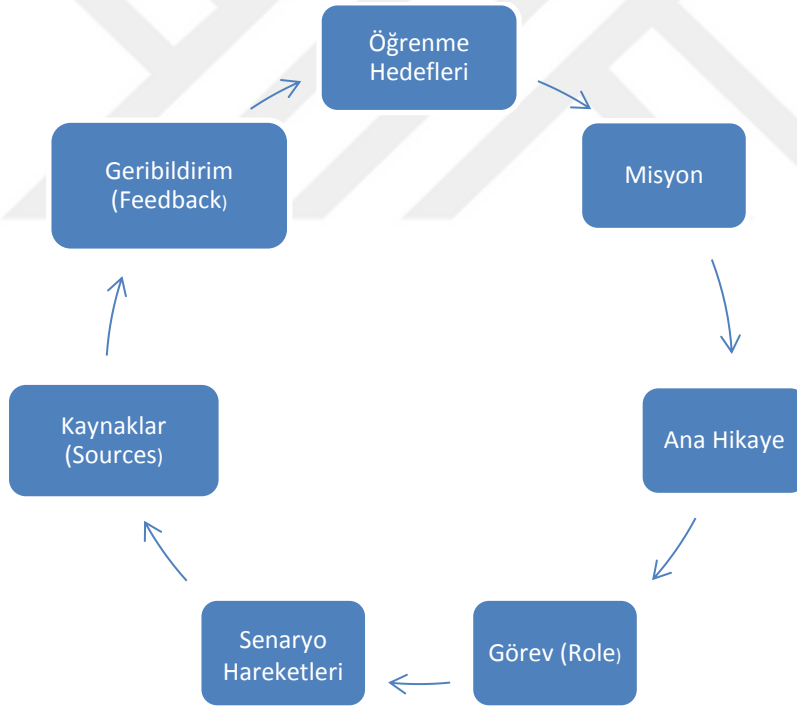
HDS, Öğrenme sürecinde verilen karmaşık öğrenme görevlerini gerçekleştirmek amacıyla öğrencilere gerçekçi ortamlar sunarak, onların öğrenme motivasyonlarını artırma potansiyeline sahiptir, ancak HDS'deki görevler bilişsel yük teorisi (BYT) ilkeleri göz önüne alınarak tasarlandığında daha etkili olması beklenir (Kılıç ve Yıldırım, 2012). Bilişsel yük teorisi (BTY), bir bireyin çalışma hafızasının sınırlı kapasitede olduğunu ifade etmektedir (Clark, Nguyen ve Sweller, 2006; Kirschner, 2002; Sweller ve Cooper, 1985; Sweller, 1988).

Kılıç ve Yıldırım'ın (2012), Bilişsel yük ve Hedefe dayalı senaryo merkezli geliştirilen 3D Multimedya Öğrenme ortamlarına ilişkin öğrencilerin motivasyonu, memnuniyeti ve zihinsel çabalarıyla ilgili yaptıkları çalışmada Multimedya öğrenme ortamı tasarlamada HDS'nin etkili bir yaklaşım olduğu ve öğrenciler üzerinde içsel ve dışsal yönden motivasyon sağladığı hatta senaryoların öğrenciler üzerinde sorumluluk bilinci kazandırdığı, görevleri bitirdiğinde başarıya duygusunu ortaya çıkardığı sonuçları ortaya çıkmıştır. Yapılan bu çalışmada bilişsel yük teorisi ilkeleri ile ve bilişsel yük teorisi ilkeleri olmadan tasarlanan iki farklı HDS merkezli multimedya tasarımına yönelik öğrencilerin görüşleri incelenmiş ve her iki grupta HDS'nin etkili ve eğlenceli olduğu, öğrenmeyi kolaylaştırdığı, multimedya ile görselleştirilen HDS'lerin daha kalıcı öğrenmeler sağladığı, öğrenmeye daha istekli oldukları ve motivasyonlarını arttırdığı gibi olumlu görüşler ortaya çıkmıştır. Ancak Bilişsel yük teorisi dikkate alındığında HDS'lerin bilişsel yükü azaltacak şekilde sunulduğunda öğrenme motivasyonlarını arttırdığı, bilişsel yükü arttıracak şekilde sunulduğunda ise aksine dikkatlerini dağıttığı, öğrenmelerini zorlaştırdığı ve motivasyonlarını ve memnuniyetlerini azalttığı görüşleri dile getirilmiştir. Okulda geleneksel biçimde sunulan öğrenme modeli öğrenciler için çoğunlukla sıkıcı olabileceği gibi öğretim süreci ve sonuçları da istenilen şekilde olmayabilir. Çünkü öğretim sürecinde doğal öğrenme hedeflerinin yerine genellikle öğrencilere başkaları tarafından belirlenmiş dışsal yapay hedefler yer almıştır. Örneğin birey öğretim sürecinde kendi

performansının üzerine geçmeyi hedeflemek yerine başkaları tarafından üzerinde etki bırakılarak bir başka arkadaşına göre kendini kıyaslamaktadır. Ya da birey kendi kişisel gelişimini arttırmayı hedeflemek yerine iş pozisyonunu yükseltmek ya da daha fazla maddi kazanç elde etmeyi hedeflemektedir. Öğrenciler ya da bireyler bir şeyi gerçekten öğrenmek istediği için değil de öğretmenlerinin ya da patronlarının gözüne girmek, onların gözünde yükselmek, statü ya da makam sahibi olmak gibi kaygılarla öğrenmeye başlarlar. Bu tür hedefler aslında etkili öğretim sürecinde pek de anlamlı değildir. Ancak HDS, bu tür yapay hedeflerin aksine eğitimsel amaçları elde etme aracı olarak çalışmaktadır. Bununla birlikte öğrenciler için daha anlamlı ve daha motive edici olan senaryo hedefleri dizisine ulaşmaya çalışır.

2.10.1 Hedefe Dayalı Senaryo (HDS) Öğeleri

Schank'den (1992) adapte edilen HDS Öğeleri Şekil 15'te gösterilmiştir.



Şekil 15: Hedefe dayalı Senaryo (HDS) Öğeleri (Schank'den (1992) adapte edilmiştir.)

2.10.1.1 Öğrenme Hedefleri (Goals)

Öğrencilerin öğrenmesi gereken temel becerilerdir. Öğrenme Hedefleri prosedürel ya da tanımlayıcı bilgi şeklinde ifade edilir. Schank, Berman ve Macpherson (1999)

öğrenim kazanımlarını içerik bilgisi ve bilginin uygulama sürecinde kullanımı olmak üzere iki ayrı boyutta ele almışlardır.

2.10.1.2 Misyon (Mission)

Öğrencilerden çözmesi hedeflenen problem durumunu açıklar. Motivasyonel (güdüleyici) veya gerçekçi objektif şekilde öğrencileri izler ve takip eder. Öğrencinin sürece aktif katılmasını sağlayacak şekilde oluşturulmalıdır.

2.10.1.3 Ana Hikaye (Cover story)

Misyonun ihtiyaçlarını açıklayan ve oluşturan motive edici hikâyedir.

2.10.1.4 Görev (Role)

Oynayacak ve görev alacak öğrenci karakteri, Temel becerileri gerektirir.

2.10.1.5 Senaryo Hareketleri (Scenario Operations)

Misyonu yerine getirmek amacıyla gerçekleştirilen bütün öğrenci aktiviteleridir. Misyon ve kazanımlar doğrultusunda öğrencinin ne yapması gerektiğini açıklar. Kararların verildiği ve sonuçları olan olaylardan oluşmalıdır.

2.10.1.6 Kaynaklar (Sources)

Öğrencilerin hedef becerileri kazanmalarını sağlayacak iyi organize edilmiş ve erişilebilir bilgi kaynaklarıdır.

2.10.1.7 Geribildirim (Feedback)

Benzer deneyimlerle ilgili hikâye ve olayların sonuçları özel ders şeklinde Uzman tarafından ve tam zamanında sağlanmalıdır. Zamanında verilmesi oldukça önemlidir.

HDS'nin motivasyon üzerindeki etkisiyle birlikte öğrenmeyi arttırdığı savunulmaktadır (Schank, Berman ve Macpherson, 1999). Bununla birlikte HDS ile ilgili yapılan alan yazın taramaları sonucunda farklı alanlara entegre edildiği ancak programlama öğretimi literatüründe çok fazla uygulamasının olmadığı gözlemlenmiştir. HDS'nin farklı disiplinlerde çeşitli örnekleri görülmektedir. HDS ile ilgili yapılan birçok çalışmada, HDS'nin öğrencilere bilgilerin amacını ve nasıl kullanacaklarını sunan etkili bir öğretim yaklaşımı olduğu vurgulanmaktadır (Bell, Bareiss ve Beckwith, 1993; Schoenfeld- Tacher, Persichitte ve Jones, 2001; Zumbach ve Reimann, 2002). Bu çalışmalara benzer bir çok çalışma örneği daha verilebilir. Örneğin Zumbach ve Reimann'ın (2002) Hiper medya karşılaştırması ile ilgili hedefe dayalı senaryo yaklaşımı değerlendirmesi üzerine yapılan çalışmada HDS'nin

öğrencilerin içsel motivasyonlarında artış gösterdiği sonucuna ulaşılmıştır. Bell, Bareiss ve Beckwith (1993) Bilim müzesinde kalıcı olarak sergilenen “Orak Hücre Danışmanı (Sickle Cell Counselor)” olarak adlandırılan bir programı geliştirmişlerdir. Araştırmalara göre bu programın bireyler için ilgi çekici olduğu ancak fazla zaman harcadıkları sonuçları ortaya çıkmıştır.

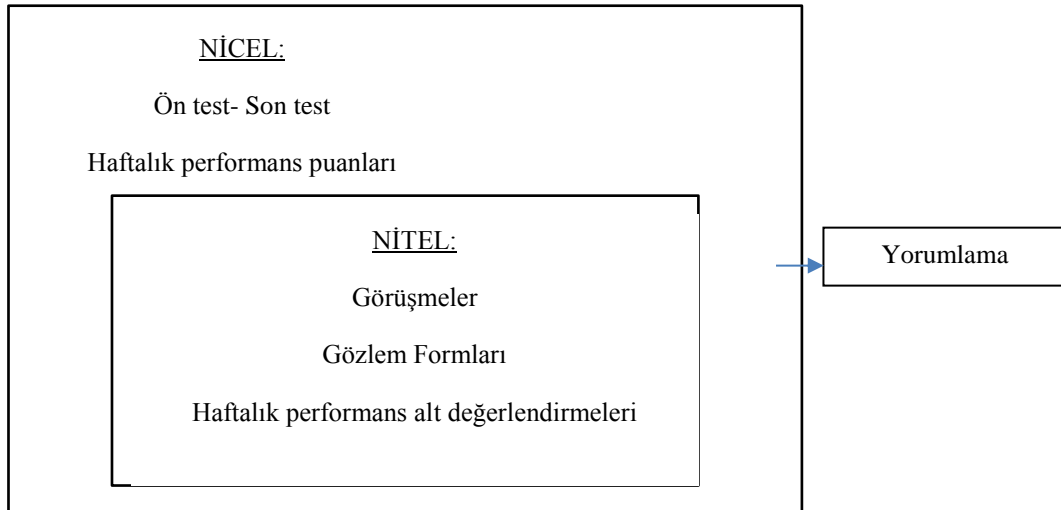
Foster (1994) tarafından yapılan çalışmada da yine HDS'nin geleneksel öğretim yöntemlerindeki bazı eksiklikleri ortadan kaldırdığı ve öğrencilerin motivasyonunu arttırdığı sonucuna ulaşılmıştır. Schaller, Bunnell ve Nagel'in (2001) yaptığı çalışmada HDS'nin etkileşimli animasyonlar, farklı simülasyonlar ve yaratıcı oyun etkinlikleriyle sunulduğunda ilgisi düşük fakat potansiyeli yüksek öğrencilerin dışsal motivasyonlarını arttırdığı sonucu ortaya çıkmıştır. Naidu, Ip ve Linser (2000), çalışmalarında HDS'nin etkililiğini araştırmaları sonucunda öğrencilerin HDS'ye karşı olumlu tutum sergiledikleri sonucuna ulaştıklarını belirtmişlerdir. Bilgisayar teknolojisini kullanarak HDS geliştirmek, HDS'nin önkoşul ilkelerinin sağlanmasını kolaylaştırmaktadır (Schank ve diğerleri, 1994). Literatür incelendiğinde, bilgisayar tabanlı HDS'lerin öğrenciler için görsel açıdan daha anlamlı ve kalıcı öğrenme ortamları sağladığı ancak bilgisayar bilimi ve programlama alanlarında literatürün yetersiz kaldığı görülmüştür. HDS'nin farklı alanlara entegre edilmesinin yanında hiç entegre edilmediği alanlar da vardır. Örneğin bilgisayar bilimi eğitimi. Son yıllarda, drama ve bilgisayarsız bilgisayar bilimi etkinlikleri yenilikçi programlama öğretimi yaklaşımları olarak alan yazında yerini almıştır. Bu tarz yaklaşımlar, bilgisayarın ve özellikle programlamanın sadece bilgisayar başında gerçekleşmeyeceğine vurgu yapmaktadır. Benzer bir yaklaşımı temel alarak, bu çalışmada HDS'nin entegre edilmesinin programlama öğretimine etkisi gözlemlenecektir. Henüz alanyazında benzer çalışmalar yer almadığı için bu çalışmanın diğer çalışmalara ışık tutması hedeflenmektedir.

ÜÇÜNCÜ BÖLÜM

III. YÖNTEM

3.1 Araştırma Yöntemi

Bu araştırmada, karma yöntemlerden olan gömülü desen (embedded design) temel araştırma deseni olarak belirlenmiştir (Creswell, 2012). Çalışmanın yarı deneysel olarak yürütülen kısmında, deney ve kontrol grubu olarak iki grup kullanılmıştır. Bu gruplardan birine HDS ile hazırlanan uygulamalar yaptırılırken diğerine çalışma kâğıtlarıyla hazırlanan uygulamalar yaptırılmıştır. Toplam 10 haftalık uygulama sürecine ek olarak hazırlık, ön-test, son-test ve görüşmeler için de ayrı haftalar söz konusu olmuştur. Nicel olarak kullanılan veriler, ön-test, son-test ve her hafta tamamlanan performans puanlarıdır. Bunun yanında nitel olarak görüşme, gözlem ve haftalık alt-performans göstergeleri kullanılmıştır. Böylece her iki veri türü birbirini tamamlayıcı niteliktedir ve yorumları bir arada yapılmıştır. Şekil 16’da araştırma deseni özetlenmiştir.



Şekil 16: Gömülü Karma Araştırma Deseni (Cresswell’den (2012) uyarlanmıştır.)

3.2 Katılımcılar

Katılımcılar, Karadeniz bölgesindeki bir devlet okulundaki 5. sınıf öğrencilerinden oluşmaktadır. Hedef popülasyon, bilgisayar dersi kapsamında Scratch arayüzünü kullanan 5. Sınıf öğrencilerinden oluşmaktadır, fakat ulaşılabilir popülasyon, Samsun'un merkez ilçelerindeki 5. Sınıf öğrencilerinden oluşmaktadır. Bu bağlamda, katılımcılar ilköğretim 5. sınıf öğretim müfredatında yer alan problem çözme ve programlama ile ilgili blok tabanlı programlama araçlarını tanıma ve kullanabilme konu ve kazanımlarına uygun olarak daha önce hiç programlama deneyimi olmayan öğrencilerden oluşturulmuştur. Katılımcıların uygun örneklem (convenience sampling) yöntemiyle bir merkez ilçedeki devlet okulundaki 2 şubede yer alan 5. Sınıf öğrencilerinden oluşmaktadır. 20'şer kişilik 2 şubeden oluşan 5. sınıfların bir şubesi deney grubu ($N=20$), diğer şubesi kontrol grubu ($N=20$) toplam 40 öğrenciden oluşmaktadır. Çalışma öncesinde öğrencilerin velileriyle görüşülerek veli onam formları alınarak gönüllük esas çerçevesinde kalınmıştır.

Katılımcıların ortak özellikleri şu şekilde özetlenebilir:

1. İlköğretim devlet okulunda okuyor olması.
2. 5. sınıf öğrencisi olmaları.
3. Daha önceden hiç programlamayla karşılaşmamış olması ve programlama deneyimi olmaması
4. Öğrencilerin ve velilerin araştırmaya katılmaya gönüllü olması
5. Temel bilgisayar kullanım becerilerine sahip olmaları.

Katılımcılarla ilgili demografik bilgiler, Tablo 1'de özetlenmiştir.

Tablo 1: Katılımcılar

	Deney Grubu (N)	Kontrol Grubu (N)
Cinsiyet		
Kız	9	7
Erkek	11	13
Yaş Ortalaması	11	11

3.3 Prosedürler

3.3.1 Hazırlık Aşaması

Çalışmaya hazırlık aşamasında senaryolar, müfredattaki kazanımlar doğrultusunda geliştirilmiş ve uzman görüşü alınarak son haline getirilmiştir. Gerekli Etik Kurul izinleri alındıktan sonra ise velilerle iletişim kurularak veli onam formları toplanmıştır. Çalışmadaki verilerin sadece bilimsel amaçlı kullanılacağı belirtilerek çalışmanın içeriği hakkında kısa bilgi verilmiştir. Geliştirilen hedefe dayalı senaryolar Powtoon programı aracılığıyla tasarlanmıştır. Hedefe dayalı senaryolar bir karakter üzerinden birbiriyle bağlantılı şekilde ilerlemektedir. Hazırlık aşamasında Excel konusu üzerinden Pilot çalışma gerçekleştirilmiştir.

3.3.1.1 Pilot Çalışma

Araştırmacı tarafından deney grubu öğrencilerine senaryo yazmayla ilgili hedef senaryo yönergesi verilmiş olup öğrencilerden belirlenen misyona göre senaryo hareketleri yapmaları istenmiştir. Pilot çalışma Excel programındaki bir uygulama şeklinde yapılmıştır. Kontrol grubu öğrencilerinden ise direkt Excel de kişilerin sınav notlarını gösteren bir tablo verilmiş olup ortalamayı ve sonuç kısmını hesaplayın şeklinde yönerge verilerek uygulama yapmaları istenmiştir. Bu süreçte deney grubu öğrencilerinden oluşan sınıf ortamının daha eğlenceli ve araştırmacı şekliyle birbirlerinden yardımlaşarak iş birliği içerisinde motive edici çalışmalar yaptıkları gözlenmiştir. Kontrol grubunun ise öğretilen formüller ve işlevlerle ilgili formül yazarken sıkıldıkları, birebir formülize ettikleri ve çoğu kişinin ders sonuna kadar istenilen görevi tamamlayamadığı görülmüştür.

Öğrencilerin konuyu algılama biçimlerine ve motivasyon düzeylerine bakıldığında yapacağı uygulamanın daha sonra hayatında işe yarayacağı düşüncesini oluşturmak senaryo tabanlı öğrenme yaklaşımının ve hedefe dayalı senaryoların konuyu kavratmada öğrencinin daha çok güdülenmesini sağladığı görülmüştür. Deney grubuna sunulan Hedefe dayalı senaryo Powtoon programı aracılığıyla tasarlanmış ve uzman görüşleri alınarak gerekli revizeler yapılmıştır.

Excel programını kullanarak ortalama ve sonuç kısmını (geçti/kaldı) şeklinde hesaplamasına yardımcı olmak ile ilgili görevi gerçekleştirmek üzere Schank in hedefe dayalı senaryolar doğrultusunda oluşturulan örnek Pilot uygulamada kullanılan hedefe dayalı senaryoda Öğrenme Hedefleri Tablo 2’de gösterilmiş olup aşağıdaki gibidir (Schank’den (1992) uyarlanmıştır).

Öğrenme Hedefleri:

Tablo 2: Pilot Uygulama – HDS

Dersin Konusu	: Hesaplama Yapıyorum
Amaçlar ve Kazanımlar	Amaç: Belirli problemlerin çözümüne yönelik formülleri kullanarak hesaplamalar yapabilme. Kazanım: Excel programında kullanılacak olan hazır elektronik çizelgeyi ekleyebilme. Excel programında bulunan formüllerden uygun olanı seçebilme. Karar verilen formül için istenilen hücreyi seçebilme. Kullanılacak formülün doğru hücrede uygulayabilme. Uygulanan formülün doğruluğunu değerlendirme

Misyon:

Ders ortalamaları alındığında bütün derslerin ayrı ayrı kendi içinde 45 in üzerinde olup “geçti” bareminde olması ve bütün derslerin ayrı ayrı ortalamaları hesaplandıktan sonra genel ortalama değerinin [85-100]Aralığında Takdir Belgesi , [70-85] Aralığında Teşekkür Belgesi almaya hak kazanması, [0-69] aralığında ise belge alamaması durumu.

Ana Hikâye:

Zeynep sınıfın başarılı öğrencisidir. Zeynep dönem sonunda takdir yada teşekkür alıp alamayacağını bilmemektedir. Bütün sınav notlarını öğrenmiş ancak ortalama puanlarını hesaplayamamıştır. Hesap makinesi alacak durumu yoktur. Bir gün bilgisayar öğretmeni onun bu durumunu görmüş ve üzüntüsünün sebebini sormuştur. Zeynep de ağlamaya başlamış “Öğretmenim takdir yada teşekkür alıp alamadığımı bilmiyorum. Eğer takdir alırsam babaannem bana telefon alacaktı. Bu telefon benim için çok önemli, çünkü onunla daha çok konuşacağım. Babamı çok özledim.” diyerek durumu anlatmıştır. Bilgisayar öğretmeni onu laboratuvara götürmüş ve bilgisayarda Excel programında hesaplaması gerektiğini söylemiştir. Ancak Zeynep bu konuları öğrenmiş olmasına rağmen o telaşla formülü yazarken sürekli bir hatayla karşılaşmaktadır. Peki, siz şimdi Zeynep’in Sınav notlarının ortalamasını hesaplayıp Karnesinin yanında hangi belgeyi almaya hak kazandığını hesaplamaya yardım eder misiniz?

Rol (Görev):

Zeynep’in Excel programını kullanarak ortalama ve sonuç kısmını (geçti/kaldı) şeklinde hesaplamasına yardımcı olmak.

Senaryo Hareketleri:

- Öncelikle çalışma yaprağında bulunan Zeynep’in derslerine ilişkin sınav notlarını gösteren elektronik çizelgeyi çiz ve not değerlerini gir
- Ortalama sütunu oluştur ve hücre değerine uygun formülü ve uygun hücreleri seçerek formül satırına formülü yaz.
- Aynı işlemi bütün dersler için tekrarla.
- Sonuç (Geçti/Kaldı) sütunu oluştur ve Koşullu biçimlendirmeye uygun formülü ve uygun hücreleri seç ve Formül satırına uygun formülü yaz.
- En son kısımda ise bütün derslerin ortalamasını hesapladıktan sonra genel ortalama hesaplaması yap ve belirlenen kritere göre değerlendirmeni yap.

Kaynaklar (Sources):

Formül satırları ve işlev kodlarının bulunduğu çalışma yaprakları.

Geribildirim (Feedback):

-Gayet güzel ilerliyorsun, - Dikkat et , - Formül satırında noktalı virgülü unutmuşsun. gibi..

Deney ve Kontrol grubuyla yapılan pilot çalışmanın sonuçları incelenerek öğrenme ortamı ile ilgili bilgisayarların hızı, öğrencilerin bilgisayarları kullanma becerileri dikkate alınarak oluşan donanımsal aksaklıklar giderilmiş ve her iki grubun da aynı şartlarda uygulama yapabilmesi için ortam sağlanmıştır. Bununla birlikte deney grubundaki öğrencilerin hedefe dayalı senaryoyu kavrama durumları, senaryo sahnelerinin geçiş hızı ve sahneler arası geçiş süreleri pilot uygulama sonunda kontrol edilerek yapılan gözlemlerden edinilen bilgiye göre öğrencilere akışın hızlı geldiği ve öğrencilerin takip etmede zorlandıkları bu nedenle senaryoları tam olarak kavrayamadıkları görülmüştür. Bu nedenle senaryo sahneleri arasındaki geçiş sürelerinin biraz uzatılmasına karar verilmiştir. Ayrıca senaryolardaki karakterlerin konuşma baloncuklarındaki konuşma metinlerinin uzun oldukları, onların akışa göre bölünmeleri gerektiği kararına varılmış, uzman görüşleri de alınarak gerekli revizeler yapılmıştır. Bilgisayar derslerinde Excel ünitesi kazanımları ile ilgili Pilot çalışma olarak gerçekleştirilen senaryoda senaryo öğelerinin tamamına yer verilmiş ve istenilen görevler açıkça belirtilmiştir. Kontrol grubuna verilen çalışma kâğıtları ile ilgili ise açıklayıcı yönergelerin yetersiz kaldığı, görsel olmadığında öğrencilere akıcı gelmediği ile ilgili gözlemler yapılmış ve bununla ilgili de revizelerin yapılmasına karar verilmiştir.

3.3.2 Uygulama Aşaması

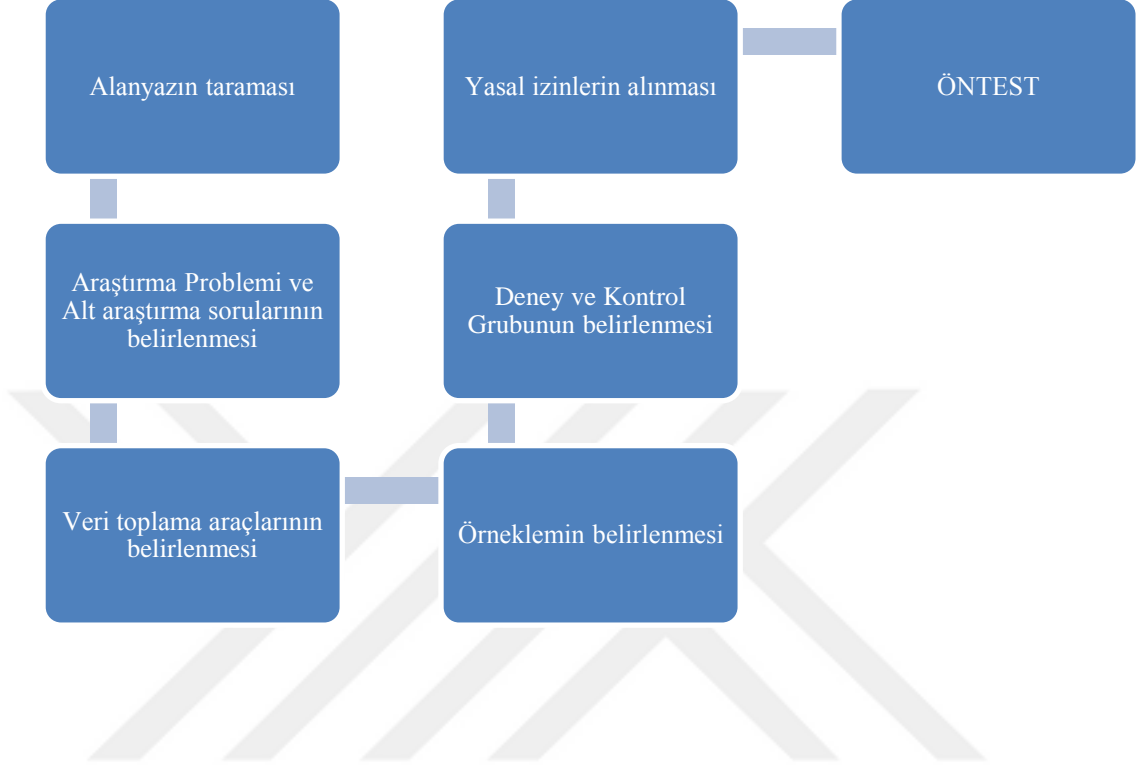
Araştırma 14 hafta sürecek şekilde planlanmıştır. Hiç programlama becerisi olmayan öğrencilerin gelişim süreçlerini izlemek amacıyla müfredatın izin verdiği süreç içerisinde nispeten uzun bir süreç olarak 14 haftalık uygulama süreci tamamlanmıştır. Her hafta deney grubu öğrencilerine birer hedef senaryo, kontrol grubu öğrencilerine ise çalışma yaprağı verilerek öğrencilerin düşünme becerilerini kullanarak görevleri yerine getirmeleri istenmiştir. İlk 2 hafta her iki gruba da algoritma, Scratch programı,

genel bilgiler, programın ara yüz tanıtımı, değişkenler, blokların tanıtılması ile ilgili temel kavramlar anlatılmıştır. Daha sonraki haftalarda ise uygulama aşaması da dâhil edilerek haftalık 4 saatin 2 saatlik bölümünde Scratch ile ilgili genel bilgiler verilmiş sonraki 2 saatlik bölümünde uygulama yapılmıştır. Kontrol grubunda bilgisayar dersi kapsamında Scratch görsel programlama ile ilgili öğrencilere programlama becerilerini kazandırma adına önce temel bilgiler sunuş yoluyla verilip daha sonra verilen bir problemin çözümüne ilişkin uygulama yapmaları istenmiştir. Deney grubunda da yine bilgisayar dersi kapsamında Scratch görsel programlama ile ilgili ise temel bilgiler verildikten sonra kazanımlara uygun hedefe dayalı senaryolar verilerek öğrencilerin düşünme becerilerini kullanarak uygulama yapmaları sağlanmıştır. Her hafta öğrenci ürünleri kaydedilmiş ve daha sonra analiz edilmiştir. Uygulama sürecinde 10 hafta boyunca izlenen içerikler Tablo 3’te özetlenmiştir. Uygulama sırası her iki grupta da aynı şekilde gerçekleşmiştir. Hazırlık ve uygulama aşamasına ait işlemler Şekil 17 ve Şekil 18’de özetlenmiştir.

Tablo 3: Uygulama Süreci

Hafta	Hedef Senaryo-Uygulama
Hafta1	Animasyonlu Davetiye Tasarlama
Hafta 2	Etkileşimli Piyano Oyunu
Hafta 3	Labirent Oyunu
Hafta 4	Hediye Paketi Toplama Oyunu
Hafta 5	Bilgi Yarışması Oyunu
Hafta 6	Balon Patlatma Oyunu
Hafta 7	Matematik 4 işlem Resimli Bulmaca Oyunu
Hafta 8	Tuğla Kırma Oyunu
Hafta 9	Trafik Kuralları Oyunu
Hafta 10	Mevsimler Oyunu

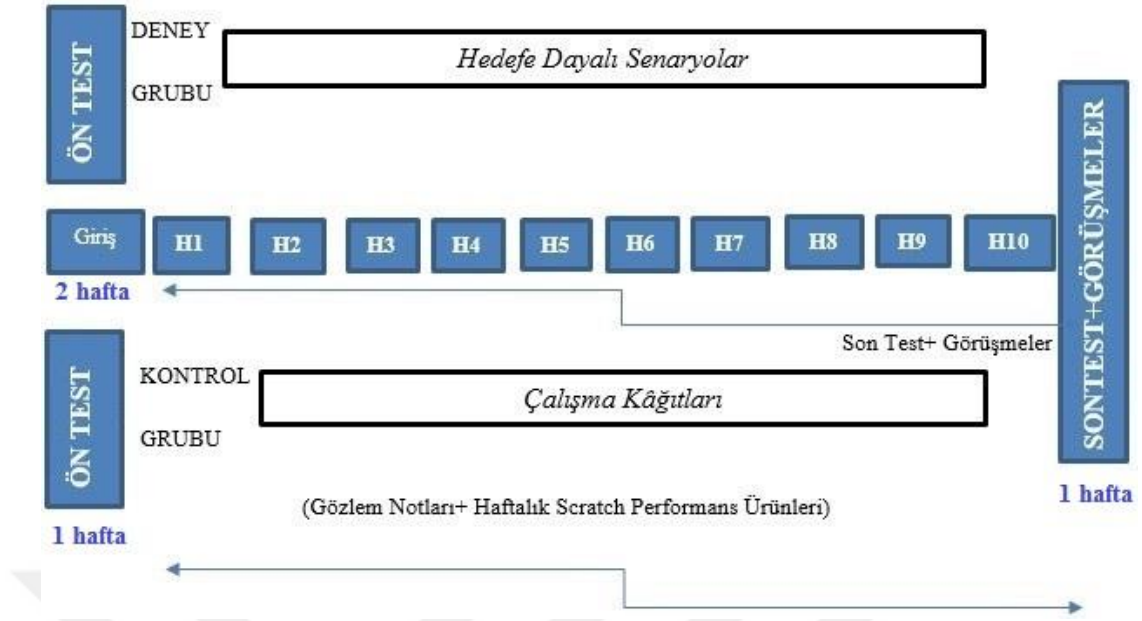
Uygulama Öncesi ile ilgili aşamalar Şekil 17’de özetlenmiştir.



Şekil 17: Hazırlık aşaması

Şekilde gösterildiği gibi, uygulama öncesinde bütün aşamalar tamamlandıktan sonra öğrencilere Scratch programlama ile ilgili temel bilgi ve becerilerini yoklayan öntest uygulanmıştır. Her iki gruba da ön test uygulandıktan sonra, uygulama sürecine geçilmiştir. Uygulama süreci tamamlandıktan sonra değerlendirme aşamalarında son test ve görüşmeler yapılmıştır. Ayrıca süreç içerisinde her iki gruptaki bireylerin haftalık Scratch performans ürünleri hafta hafta gözlemlenmiş ve gözlem notları alınmıştır.

Uygulama süreci ve sonrası Şekil 18’de özetlenmiştir.



Şekil 18: Uygulama aşaması

Bu çalışmada, öğrencilere kodlama öğretiminde HDS kullanımının etkisini araştırmak amacıyla öncelikle araştırmacı tarafından STÖ ve programlama konularında ayrıntılı biçimde alan yazın taraması yapılmıştır. Ortaokul öğrencileri için soyut olan programlama öğretiminde HDS'lerin etkisini belirlemek amaçlanmıştır. Araştırmacı, bu bağlamda veri toplamak amacıyla öncelikle uzman görüşleri alarak Powtoon aracıyla senaryolar geliştirmiş, uzman değerlendirmeleri yapıldıktan sonra öğrencilere sunulmuş ve öğrencilerden verilen HDS'lerden hareketle Scratch Görsel programlama aracını kullanarak tasarımları istenmiştir. Öğrencilerin Scratch programında hazırladıkları tasarımları HDS'ler daha sonrasında yine uzman görüşü alınarak hazırlanan performans değerlendirme aracı ile değerlendirilmiş, analizleri yapılmıştır. Ayrıca araştırma sürecinde öğrencilere görüşme soruları hazırlanmış, uzman değerlendirmeleri alınarak bu sorular geliştirilmiştir. Toplam 14 (7DG-7KG) öğrenciyle yüz yüze yarı yapılandırılmış görüşmeler yapılmıştır. Araştırmacı, topladığı bu verileri analiz ederek elde ettiği bulguları bu tezde ayrıntılı olarak açıklamıştır.

3.4 Hedefe Dayalı Senaryolar

Geliştirilen HDS'ler, Powtoon programı aracılığıyla tasarlanmıştır. Etkili öğrenmenin sağlanması amacıyla bilişsel yük teorisi ilkeleri göz önünde bulundurulmuştur. HDS'lerin tasarım aşamasında öğrencilerin seviyeleri de dikkate alınmıştır. Bununla birlikte HDS'lerin hazırlık ve tasarlama aşamasında hafta hafta hedef kazanımlara uygun olarak uzman görüşleri alınmıştır.

HDS'lerin geliştirilme aşamasında Hedefe dayalı senaryonun bütün öğelerine yer verilmiştir. HDS'nin başlıca öğeleri: öğrenme hedefleri, misyon, ana hikâye, görev (rol), senaryo hareketleri, kaynaklar ve geribildirimdir (Schank, 1992). Her haftanın senaryo akışı var olan bir karakter üzerinden gitmektedir. Ayrıca her haftanın senaryosu bir öncekiyle bağlantılı olarak ilerlemektedir. 10 haftalık uygulamada "animasyonlu davetiye, piyano oyunu, labirent oyunu, hediye toplama oyunu, bilgi yarışması oyunu, balon patlatma oyunu, matematik 4 işlem resim bulmaca oyunu, tuğla kırma oyunu, trafik kuralları oyunu, mevsimler oyunu" HDS leri geliştirilmiş ve öğrencilerden bu senaryoları tasarlama istenmiştir.

Hafta 2 Etkileşimli piyano oyununa ait örnek HDS Şekil 19'da ve Tablo 4'te verilmiş olup, diğer tüm HDS'ler ekler kısmında yer almaktadır (Ek D).



Şekil 19: Hafta 2- Etkileşimli Piyano Oyunu HDS Örneği

Tablo 4: Hafta2- Etkileşimli Piyano Oyunu Uygulaması

Hafta2	
Ünite: Scratch ile Programlamaya Giriş	
Etkinlik: Piyano Oyunu Uygulaması	
1)Öğrenme Hedefleri:	Kazanımlar: 1.Sahne , Karakter ve Kostüm kullanabilme 2.Hareket bloğunu kullanabilme 3.Görünüm bloğunu kullanabilme 4.Ses bloğunu kullanabilme 5.Kontrol bloğunu kullanabilme
2)Misyon:	1)Genel Tasarım Öğelerini Düzenleme Renk efektini değiştirme FarklıKarakterlere ekleyebilme (Balon, Nota, Şapka , İnsan, Mikrofon,) Piyano notalarını ekleyebilme ve yerleştirebilme Karakterlere farklı kostümler ekleyebilme Sahne arkaplanını değiştirme 2)Hareket Bloğu komutlarını kullanabilme Karakterleri hareket ettirebilme Döndürme kodunu kullanabilme 3)Görünüm Bloğu komutlarını kullanabilme Kostümler arası geçiş kodunu verebilme 4) Ses Bloğu komutlarını kullanabilme Tasarıma uygun her notaya vuruş ve nota belirtilerek farklı ses ekleyebilme 5) Kontrol Bloğu komutlarını kullanabilme Yeşil Bayrak Tıklandığında kodunu ekleyebilme Karakter Tıklandığında kodunu ekleyebilme Bekleme kodunu kullanabilme Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme Görevi Tamamlayabilme

Tablo 4: Hafta2- Etkileşimli Piyano Oyunu Uygulaması (devamı)

3)Ana Hikaye:

Erenlerin evinde doğum günü partisi yapılacaktır. O yüzden Erenin annesi partiye hazırlık yapmayı planlamaktadır. Öncelikle evde temizlik yapmayı düşünmektedir. Daha sonra Erenin arkadaşları parti için Rüya Mahallesindeki Erenlerin evine gelmek için yola çıkmışlardır. Erenin arkadaşlarından birisi evlerine geldiklerinde “Haydi kapıyı çalalım” diye seslenir.Erenin annesi Ezgi, Eren’in arkadaşlarını Erenin doğum günü partisine hoş geldiniz diye karşılar. Arkadaşları eve gelir gelmez Erenin nerede olduğunu sorarlar. Annesi Eren’in parti için hazırlandığını ve hemen çağıracağını söyler. Eren’e seslenir: “Eren bak oğlum , kimler gelmiş , arkadaşların seni bekliyor.”Eren bir anda annesi ve arkadaşlarını görünce çok şaşırarak mutlulukla havalara uçar. Eren annesine ve arkadaşlarına çok mutlu olduğunu ve bugün onun için anlamlı bir doğum günü olduğunu söyler.Erenin arkadaşları Eren ‘e çok güzel hediyeler almış ve bütün evi annesi balon ve konfetilerle süslemiştir.Eren hediyeleri görünce mutlu olur ve arkadaşlarına teşekkür eder. Daha sonra partinin devamında Erenin arkadaşları “Haydi dans edelim, Müzik zamanı diye düşünürler. Eren bunun için annesinden izin alıp kendi piyanosunu çalacağını söyler arkadaşlarına. Daha sonra Erenin annesi Erene piyano çalması için izin verir.Eren arkadaşlarının karşısında piyano çalmaya hazırlanmaktadır. Ancak piyanonun tuşlarına basıldığında ses bir türlü çıkmadığını ve piyanonun çalışmadığını fark eder. Erenin arkadaşı Mert neden çalışmadığını sorar ve piyanonun çalışmadığına ilişkin akıl yürütür. Eren ise arkadaşlarına piyanoya sürekli bastığını ancak herhangi bir sesin çıkmadığını söyleyerek piyanonun çalışmadığını görünce çok üzümüştür. O yüzden arkadaşlarından doğum günü partisinde eğlenmek için yardım istemektedir. Arkadaşlarına doğum günü partisinde piyano çalmak ve eğlenmek istediğini ancak ne yazık ki evlerinde bulunan piyanonun çalışmadığını açıklar. Daha sonra aklın a bilgisayarda piyano oyunu tasarlamak gelir. Bilgisayarda hazırlayacağı piyanoda her tuştan farklı notalardan farklı sesler çıkarmasını ve her tuşa bastıkça tuşa basma görünümü kazandırarak daha gerçekçi ve animasyonlu bir piyano tasarlamak istediğini söyleyerek yardım istemektedir.

Tablo 4: Hafta2- Etkileşimli Piyano Oyunu Uygulaması (devamı)

4) Görev (Role):	<ul style="list-style-type: none">*Piyano çalmaya uygun sahne tasarımı yapılmalı*13 Notadan oluşmalı*Her nota farklı ses çıkarmalı*Animasyonlu dönen, rengi yanıp sönen nota işaretleri olsun.*Piyano çalan bir karakter oluşturun.*Piyanonun yanında dans eden karakterler veya mikrofon ile şarkı söyleyen karakterler oluşturun.
5)Senaryo Hareketleri:	<p>Öncelikle Scratch senaryosunda bulunan Eren'in doğum günü partisinde çalacağı piyano ve sahne ortamı için sahne arka plan tasarımını oluştur. Farklı renkler ve şekillerle tasarlayabilirsin. İlk önce piyano tuşları için 8 beyaz 5 siyah olmak üzere toplam 13 tuş tasarla ve her tuşa basma görünümü kazandırmak amacıyla farklı kostümler ekle oluştur. Piyano tuşlarına sırasıyla Karakter tıkladığında kodunu ekle .Farklı kostümler arası geçiş kodları için kostüme geç kodunu ekle. Ve her tuş için farklı nota seslerini vermek amacıyla... notasını ... vuruş çal kodunu ekle. Mikrofon karakteri ekle ve animasyon oluşturmak amacıyla renk efektini değiştir kodunu ekle. Nota işaretleri tasarla ve bunlara dönme efekti için 15 derece dön kodunu, renk efekti vermek için renk efektini ... değiştir kodunu ekle. Kendiniz tasarlayarak ya da hazır karakterler kullanarak piyanoyu çalan bir karakter oluşturun. Piyanonun yanında dans eden karakterleri oluştur. Ve dans etme görünümü sağlamak amacıyla kostüm değiştirme kodunu ekle. Ve mikrofon ile şarkı söyleyen karakter oluşturabilirsin.</p>
6) Kaynaklar (Sources):	Powtoon programıyla hazırlanmış Piyano Oyunu senaryosu, Çalışma Kağıdı
7)Geribildirim (Feedback):	-Gayet güzel ilerliyorsun, Bu kodu burada eksik kullanmışsın, Tasarımında renk efekti uygulayabilirsin, Piyano tuşlarına bastığında ses çıkmıyor, nota seslerini ekleyebilirsin.

3.5 Veri Toplama Araçları

Bu çalışmada veri toplama kısmında, başlangıçta nicel verileri elde etmede öğrencilerin temel programlama performanslarını ölçen öntest-sontest şeklinde uygulama yapılmıştır. Nitel veriler ise öğrenciler tarafından Scratch üzerinde yapılan hedef senaryo uygulamalarının ekran kayıtları alınarak gözlem formları ve yarı yapılandırılmış mülakat kullanılarak toplanmıştır. Yaklaşık olarak 14 hafta süren bu araştırma sürecinde temel programlama bilgilerinin verilmesi ve kazanımlara uygun senaryoların oluşturularak öğrencilere algoritmik düşünme becerisi ve programlama becerisini kazandırılması hedeflenmiştir. Öğrencilerin hedef senaryoyu tasarlama ve geliştirme sürecinde uygun geri dönütler öğretmen tarafından verilmiş olup gelişim düzeyleri her hafta aşama aşama incelenmiş ve gözlemlerle kayıt altına alınmıştır. Öğrencilerin haftalık Scratch performans değerlendirmeleri Scratch programlama alan uzmanı, içerik uzmanı ve ölçek geliştirme uzmanı görüşleri alınarak her hafta için ayrı ayrı hazırlanan Rubrik (Scratch Performans değerlendirme aracı) ile yapılmıştır.

3.5.1 Ön Test- Son Test

Bu çalışmada veri toplama kısmında, başlangıçta nicel verileri elde etmede öğrencilerin temel programlama performanslarını ölçen 16 sorudan (Ek B) oluşan bir ön test-son test şeklinde uygulama yapılmıştır. Araştırmacı tarafından hazırlanan bu ölçme araçlarının geliştirilmesinde 1 konu alanı uzmanı, 1 içerik uzmanı ve 1 ölçek geliştirme uzmanı görüşleri alınarak üzerinde değişiklikler yapılmış ve yeniden revize edilerek geçerliği ve güvenilirliği kontrol altına alınmaya çalışılmıştır. Öğrencilerin temel programlama bilgilerini ve performanslarını ölçmek amacıyla geliştirilen ön test ve son test ölçme araçları öğrencilerin seviyelerine uygun olarak hazırlanmıştır. Geliştirilen ön test ve son test ölçme araçlarının tüm soruları ekler (Ek B) kısmında yer almaktadır. Ön test ve son test ölçme araçlarının sorularının hangi kazanımlara karşılık geldiğine dair hedef kazanım-soru belirtke tablosu Tablo 5’de gösterilmiştir.

Tablo 5: Ön Test-Son Test Hedef Kazanımlar Belirtke Tablosu

Kazanımlar	Soru Sayısı
Blok tabanlı programlama aracının arayüzünü ve özelliklerini tanıır.	5
Blok tabanlı programlama ortamında sunulan hedeflere ulaşmak için doğru algoritmayı oluşturur.	2
Doğrusal mantık yapısını kullanan algoritmalar geliştirir.	2
Karar yapıları içeren algoritmalar geliştirir.	2
Döngü yapısını ve işlevlerini açıklar. Döngü yapısı içeren algoritmalar oluşturur.	2
Farklı yapılar için oluşturduğu algoritmaların sonucunu yordayarak hatalarını ayıklar.	3

3.5.2 Mülakat (Görüşmeler)

Araştırmada yarı yapılandırılmış mülakat tekniği kullanılmıştır. Öğrencilerle son test sonrasında, STÖ yaklaşımına dayalı öğretim süreci sonunda öğrencilerin Scratch programlamaya yönelik algıları, hedefe dayalı senaryolar ve çalışma kâğıtları ile ilgili düşüncelerini öğrenmek amacıyla görüşmeler yapılmıştır. Görüşmeler yaklaşık olarak 15-20 dakikada gönüllü öğrencilerle ayrı ayrı yapılmıştır. 7 kişi deney grubu 7 kişi kontrol grubu olmak üzere toplam 14 gönüllü öğrenciyle görüşme yapılmıştır. Her iki grupta da görüşme yapılan öğrenciler süreç içerisinde gözlemlenen performanslarına göre hem gönüllü olmaları hem de düzeylerine göre belirlenerek heterojen gruplar oluşturulmuştur. Araştırmacı tarafından hazırlanan yarı yapılandırılmış görüşme soruları alan yazın desteğiyle yapılandırılmış ve bu görüşme soruları alan uzmanı, içerik uzmanı ve ölçek geliştirme uzmanı tarafından kontrol edilmiştir. Yapılan değerlendirmelerin sonucunda üzerinde değişiklikler yapılarak geçerlik ve güvenilirliği sağlanmaya çalışılmıştır. Mülakat (Görüşmeler) deney grubunda Scratch Programlamaya yönelik algılar (5 Soru), HDS'ye yönelik algılar (4 Soru) ve Öz değerlendirmeye yönelik algılar (3 Soru) olmak üzere toplam 12 soru içerirken kontrol grubunda ise Scratch programlamaya yönelik algılar (5 Soru), Çalışma kâğıtlarına yönelik algılar (3 Soru) ve Öz değerlendirmeye yönelik algılar (3 Soru) olmak üzere toplam 11 soru içermektedir. Deney ve kontrol grubu için geliştirilen yarı yapılandırılmış görüşmelerin tüm soruları ekler (Ek A) kısmında yer almaktadır.

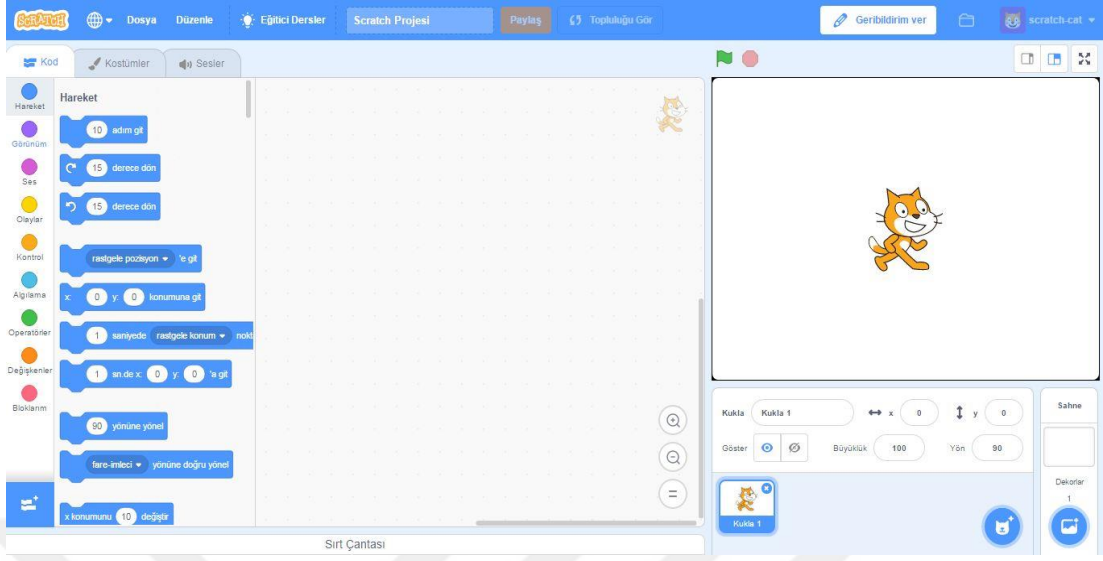
3.5.3 Gözlem Formları

Öğretim sürecinde her hafta öğrencilerin Scratch performansları gözlemlenmiştir. Bununla birlikte gözlem formları uygulanmış ve ara değerlendirmeler yapılmıştır. Böylece her öğrencinin gelişim aşamaları hafta hafta izlenmiştir. Her haftanın gözlem formları, ayrı ayrı ve farklı olarak hazırlanmıştır. Güvenirlik ve geçerlik analizlerini sağlamak amacıyla blok temelli Scratch programlama ile ilgili konu alanında uzman, ölçek geliştirme alanında uzman ve içerik uzmanı tarafından değerlendirmeler yapılmıştır. Bu değerlendirmelerin sonucunda üzerinde değişiklikler yapılarak geçerlik ve güvenilirliği sağlanmaya çalışılmıştır. Geliştirilen gözlem formlarının tümü ekler (Ek E) kısmında yer almaktadır.

3.5.4 Rubrik (Scratch Performans Değerlendirme Aracı)

Araştırmacı tarafından hazırlanan; Scratch Performans Değerlendirme Dereceli Puanlama Anahtarı (Rubrik) ile öğrencilerin performansları değerlendirilmiştir. Uzman değerlendirmesi yapıldıktan sonra üzerinde gerekli değişiklikler yapılmıştır. Her hafta uygulanacak olan senaryonun kazanımları belirlenmiş, senaryo içeriği ve kazanımlara bağlı olarak rubrikler hafta hafta hazırlanmıştır. Geliştirilmeli (1), Orta düzeyde (2) ve Mükemmel (3) dereceli puanlama anahtarlarına göre öğrencilerin performansları görsel tasarım, kod bloklarının doğruluğu ve ürün tamamlama durumları açısından gözlemlenerek değerlendirilmiştir. Her hafta için ayrı ayrı ve farklı rubrik hazırlanmıştır. Ayrıca güvenirlik ve geçerlik analizleri blok temelli Scratch programlama ile ilgili alan uzmanı, içerik uzmanı ve ölçek geliştirme uzmanı görüşleri alınarak araştırmacı tarafından gerçekleştirilmiştir. Bu görüşmelerin sonucunda üzerinde değişiklikler yapılarak geçerlik ve güvenilirliği sağlanmaya çalışılmıştır. Geliştirilen bu ölçme araçlarının tümü ekler (Ek C) kısmında yer almaktadır.

3.5.5 Scratch Görsel Programlama Aracı (Scratch 3.0 Beta Sürümü)

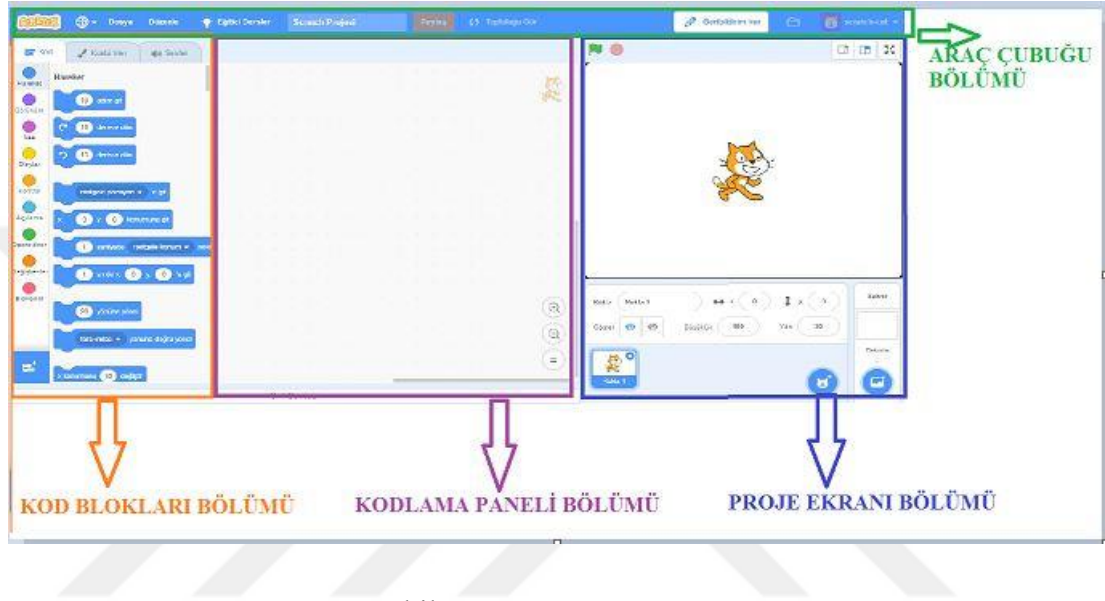


Şekil 20: Scratch Ara yüzü-3

Scratch, 2003 yılında Yaşam boyu Okul Öncesi Grubu (Lifelong Kindergarten Group), Mitchell Resnick ve arkadaşları tarafından Massachusetts Teknoloji Enstitüsü MIT Medya Laboratuvarında geliştirilmiştir. Scratch programı programlama dili olduğu gibi, üzerinde projeleri barındıran ve yapılan bu projelerin paylaşılabilirdiği tamamen ücretsiz görsel programlama aracıdır. Yaklaşık 1, 5 milyon üye kayıtlıdır. 9 milyonun üzerinde projeyi barındıran web portalı ve etkileşimli ara yüzüyle Scratch görsel programlama her yaş ve düzeyde öğrencinin kodlamayla tanışmasına imkân sağlamaktadır. 2007 yılında tüm dünyada 150'den fazla ülkede yaklaşık 40 tan fazla dil desteği ile yayınlanan görsel programlama aracı olan Scratch'ın anaokulundan lise kademesine her yaş aralığından kullanıcısı mevcut olmakla birlikte en aktif ve yoğun kullanma yaş grubu 8-16 olarak bilinmektedir (Scratch about, 2019). Ancak her yaş ve düzeyde bireyler Scratch üzerinde projeler oluşturabilmekte ve paylaşımına sunabilmektedir.

Scratch programının ara yüzü dört bölümden oluşmaktadır. Şekil 20 ve Şekil 21'de gösterilen bu bölümler şu şekilde ifade edilebilir.

- 1.Kod Blokları bölümü
- 2.Kodlama Paneli bölümü
- 3.Proje Ekranı bölümü
- 4.Araç Çubuğu bölümüdür.

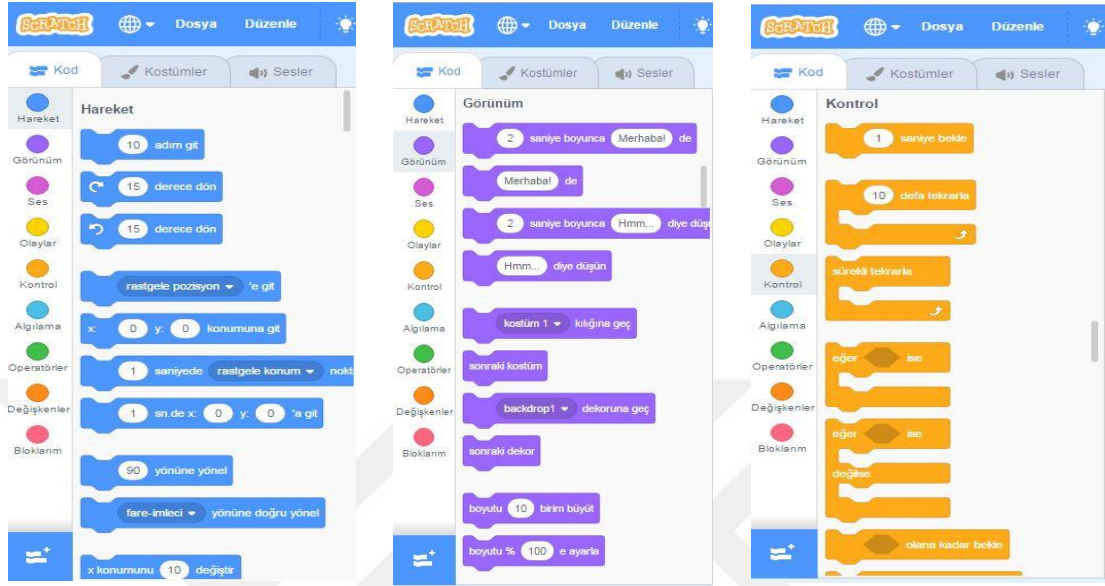


Şekil 21: Scratch Ara yüzü-4

3.5.5.1 Kod Blokları Bölümü

Sol bölüm kod blokları bölümü olarak adlandırılır. Bu bölümde karaktere yapmasını istediğimiz komutlara ait çeşitli kod blokları ve kodlar hazır olarak bulunur ve bu kodlar sürükle- bırak yöntemiyle çeşitli kod bloklarının olduğu bölümden alınarak kodlama paneline bırakılır. Bu bölümde farklı renklerde 9 farklı kod bloğu bulunur. Hareket, Görünüm, Algılama, Ses, Kontrol, Olaylar, Değişkenler, Operatörler ve Bloklarım bloklarından oluşan bu bölümde alt başlıklara ayrılacak şekilde farklı kodlar içermektedir. Her bloğun altında o bloğa ait kodlar yer almaktadır. Bu bloklar farklı renklerde tasarlanmıştır. Hareket blok kodları mavi renkte, Görünüm kod blokları mor renkte, Ses kod blokları lila renkte, Bloklarım kod blokları pembe renkte, Kontrol kod blokları turuncu renkte, Olaylar kod blokları sarı renkte, Algılama kod blokları mavi renkte, Operatörler kod blokları yeşil renkte ve Değişkenler kod blokları turuncu renkte gösterilmektedir. Bu blokların üzerinde işlevleriyle ilgili açıklamalar yer almaktadır. Bu kodlar metin temelli programlamadaki kompleks syntax kod yapısından daha çok günlük konuşma dilinde ve daha basit düzeydedir. Scratch

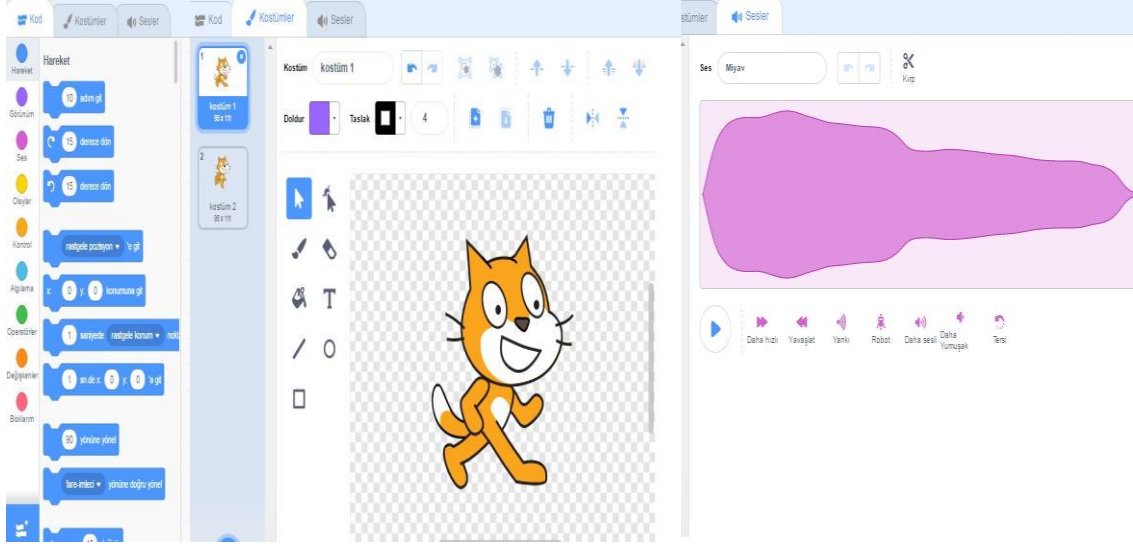
programlama ile karmaşık kodlar yerine bu kod bloklarının sürükle bırak yöntemiyle arka arkaya yerleştirilerek program tasarlanabilir. Scratch programının kod blokları bölümü Şekil 22’de gösterilmiştir.



Şekil 22: Scratch programı kod blokları bölümü

3.5.5.2 Kodlama Paneli Bölümü

Bu bölümde karaktere ait kodlamaların yapıldığı bölümdür. Kod bloklarındaki komutlar sürükle bırak yöntemiyle bu bölüme bırakılarak karaktere hareket özelliği kazandırılır. Yazılar, kostümler ve sesler bölümlerinden oluşmaktadır. Yazılar bölümünde kod bloklarından alınan komutlar bir araya getirilerek proje ekranındaki karakter hareket ettirilir. Kostümler bölümünde karaktere farklı görünümler kazandırılır. Sesler bölümünde ise karakterlere ses özelliği kazandırılabilir. Ayrıca bu bölümün üst orta kısmında karakterin adı ve karakterin yönünü belirleyen oklar gibi bilgiler yer almaktadır. Bu oklar aracılığıyla karakterin yönü belirlenmektedir. Scratch programının kodlama paneli bölümü Şekil 23’te gösterilmiştir.

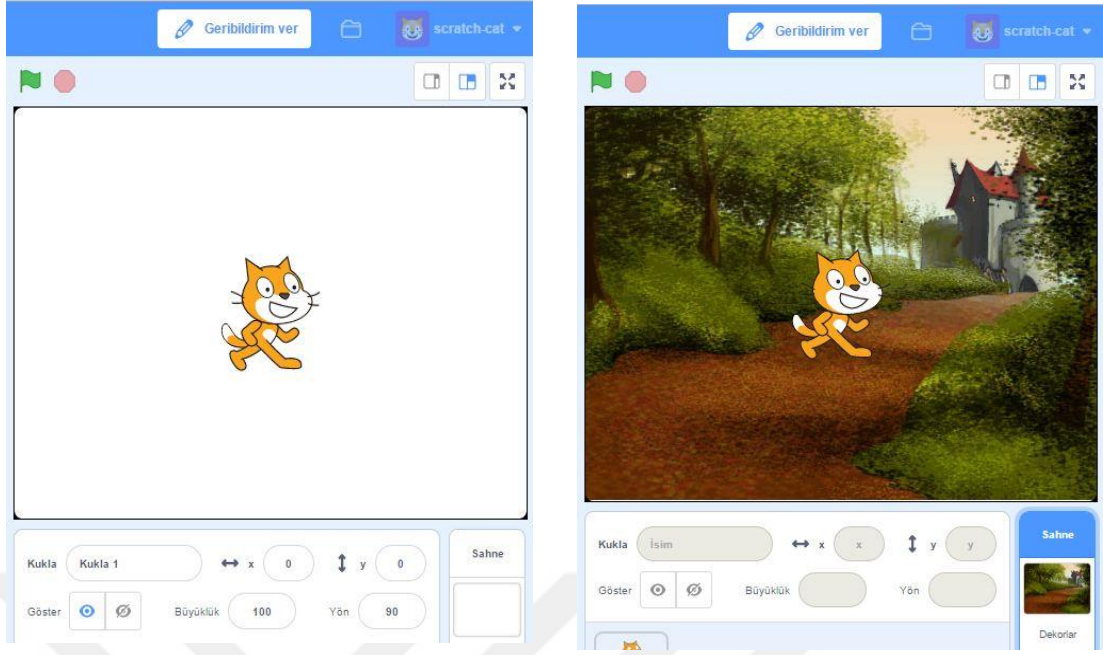


Şekil 23: Scratch programı kodlama paneli bölümü

3.5.5.3 Proje Ekranı Bölümü

Bu bölüm projemizin uygulamaya geçtiği, karaktere dair yazılan kodların karakterlere nasıl görünüm kazandıracağını ve tasarladığımız her şeyi görebileceğimiz sahnemizdir. Oluşturduğumuz interaktif yani etkileşimli animasyonların, oyunlarımızın, projelerimizin canlandığı yerdir. Projede yazdığımız kodlarla bu bölümde sahne üzerinde uygulanmakta ve gösterilmektedir. Ayrıca bu bölümde karakterin bulunduğu yerin hemen altında hangi sahnede proje yapıldığı ve proje ekranındaki hangi karaktere komut verildiği gösterilmektedir.

Bu bölümde yeşil bir bayrak ve kırmızı bir düğme vardır. Yeşil bayrak projenin başlamasını sağlayan komuttur. Yeşil bayrağa tıklandığında program başlayacak ve kodlar çalışır hale gelecektir. Kırmızı düğme komutu ise tıklandığında programın durdurulmasını sağlayan komuttur. Proje ekranındaki karakterler dosyadan hazır olarak ekle seçeneğiyle hazır olarak kullanılabilceği gibi ayrıca projeyi tasarlayan kişinin kendisi tarafından da isteğe göre tasarlanabilmektedir. Scratch programının proje ekranı bölümü Şekil 24'te gösterilmiştir:

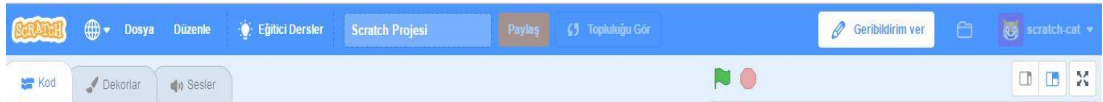


Şekil 24: Scratch programı proje ekranı bölümü

3.5.5.4 Araç Çubuğu Bölümü

Bu bölüm proje ekranının en üst kısmında bulunmaktadır. Araç çubuğunda dil seçeneği, projeyi kaydetme ve projeyi paylaşma simgeleri ile dosya, düzenle, internette paylaşım, yardım seçeneği menüleri bulunmaktadır. Bu kısımların hemen yanında sahnedeki karakterleri çoğaltma, silme, büyültme-küçültme ikonları bulunmaktadır. Bu ikonlar ile sahnedeki karakterler üzerinde seçme, çoğaltma, silme, karakteri büyütme ve karakteri küçültme işlemleri yapılabiliriz. Son sırada ise ekranı küçültme, ekranı büyültme ve proje sunma ekranı kısımları yer almaktadır. Bu seçeneklerle ekranı istediğimiz boyutta kullanabilme imkânı sunmaktadır.

Scratch programının araç çubuğu bölümü Şekil 25'te gösterilmiştir.

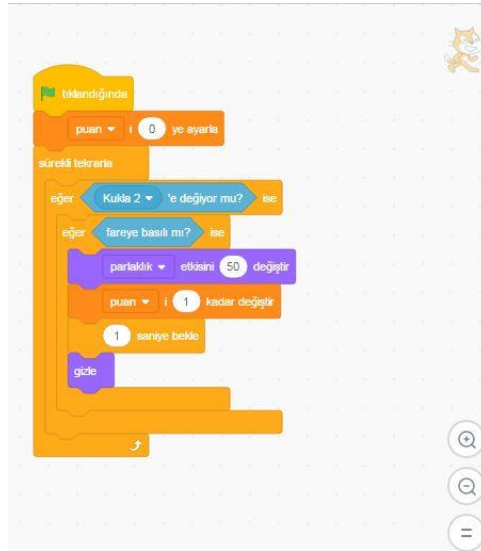


Şekil 25: Scratch programı araç çubuğu bölümü

(Scratch About, 2019).

3.5.5.5 Blok Kod Yapısı

Scratch programında karmaşık kodlar yerine kod bloklarında sürükle bırak özelliği bulunmaktadır. Bu özelliği ile programlamaya yeni başlayan ve acemi olan bireylere oldukça kolaylık sağlamaktadır. Program tasarlamak ve kodlama yapmak için bu blokları alt alta sürükleyerek bırakmak yeterlidir. Bu bloklar farklı renklerde tasarlanmıştır ve her kod bloğu aslında farklı renklerde. Görsel olarak farklı renklerde olması bireylere motivasyon ve ilgi çekicilik kazandırması bakımından oldukça faydalıdır. "Hareket" blok kodları mavi rengi temsil ederken , "Görünüm" blok kodları mor renkli, "Ses" blok kodları lila renkli, "Kontrol" blok kodları turuncu renkte, "Olaylar" blok kodları sarı renkte, "Bloklarım" blok kodları pembe renkte iken , "Algılama" blok kodları açık mavi renğinde, "Operatörler" blok kodları yeşil renkte ve "Değişkenler" blok kodları da turuncu renk ile tasarlanmıştır. Bu blok kodların üzerinde ne işe yaradığı yazılıdır. Örneğin "10 adım git", "kostüm 2 kostümüne geç", "miyav sesini çal" ya da " rengine değişiyor mu" gibi. Bu kodlar bir programlama dilindeki karmaşık kod (syntax) yapısından daha çok günlük konuşma diline uygun şekilde tasarlanmıştır. Ayrıca sadece birbiriyle uyumlu bloklar birbiriyle kenetlenmektedir ve böylelikle hata yapma olasılığı daha da azalmaktadır. Scratch programı örnek kod blokları ile kod yazma örneği aşağıdaki Şekil 26'da gösterilmiştir.



Şekil 26: Scratch programı örnek kod bloğu

3.6 Veri Analizi Süreci

Bu araştırma sürecinde elde edilen nicel verilerin analizi bilgisayar ortamında Spss 25.0 istatistik programı ile değerlendirilmiştir. Gerçekleştirilen yarı deneysel araştırma deseni sonrasında HDS uygulamalarının gruplar üzerindeki etkisini belirlemek amacıyla öğrencilerin temel programlama performanslarını ölçen son test başarı puanları karşılaştırılmıştır. Deney ve kontrol grubunun ön test sonuçlarına bakıldığında iki grup arasında anlamlı farkın olmadığı görülmüştür. Bu nedenle iki grubun son test sonuçları arasında anlamlı farkın olup olmadığına dair bağımsız örneklem t testi ile analiz yapılmıştır. Araştırmayla ilgili bağımsız örneklem t testinin bütün varsayımları incelenmiş ve iki grup arasında anlamlı farkın olup olmadığına dair $p=0.05$ değeri ile yorumlanmıştır. Bununla birlikte haftalık gözlemler ve yarı yapılandırılmış görüşmeler gibi nitel veriler tematik içerik analizi yöntemiyle analiz edilmiştir.

Bu analizler sonucunda görüşmeler “Scratch programlamaya yönelik algılar”, “Hedefe dayalı senaryoya yönelik algılar”, “Çalışma kâğıtlarına yönelik algılar” ve “Öz değerlendirmeye yönelik algılar” temaları altında kategorize edilmiştir. Scratch programlamaya yönelik öğrencilerin görüşleri analiz edildiğinde kodlama aşamasında ve tasarım aşamasında zorluk/kolaylık algıları; Scratch programında hoşça giden durumlar (Çizilen yada hazır karakterlerin canlılık özelliği kazanması, dinamik (hareketli öğeler) , kodların görsel ve sürükle bırak yöntemiyle oluşturulması, kendi istedikleri şekilde özgün biçimde tasarlamak, hazır karakter kullanmak vb.); hoşça gitmeyen durumlar (kodlamada zorluk yaşama, kodların yerini karıştırma ve bulamama, kodlar arasında mantık kuramama, bazı senaryo oyunlarında kodların fazla ve uzun olması, tasarımda kendi çizimde zorluk yaşama, bazı hazır karakterlerin hoşça gitmemesi, kodlarda hata oluşma durumu, kodların hatalı şekilde yerleştirilmesi vb.); program çalıştığında mutluluk, ilgi çekme, hoşça gitme, beğenme, hayal gücünü geliştirme, özgüven, motivasyon, pekiştirici (ödül) etkisi, hüsrana, kuşku, endişe/kaygılanma, başarı duygusu, hırs duygusu, hemen bitirme isteği, eğlenceli ve zevkli bulma, sinirlenme/öfke/ kızgınlık/program çalışmadığında üzülmeye gibi duyuşsal faktörler; Hedefe dayalı senaryoya yönelik öğrencilerin görüşleri analiz edildiğinde karikatürize şemalar, devam niteliğinde senaryoların hikaye bitişikliği, konuşma baloncuklarındaki metinler gibi görsel boyutuyla ilgili faktörler, senaryoların

kolaylaştırma etkisi, gösterilen senaryolara bağlılık, senaryoların tasarımı sınırlandırma durumu, senaryoların öğretici olması, anlatımının açık ve anlaşılır olması, öğrencilerin seviyelerine uygunluk, iyi yada kötü hazırlanmış olması, hikayeye aitlik duygusu (gerçek yaşam etkisi), senaryoları kurgulayabilme ve fikir oluşturabilme, senaryodaki karakterler, senaryoda açıklayıcı yönergelerin bulunması gibi yapısal boyutuyla ilgili faktörler, esneklik boyutunda öğrencilerin kendi düşüncelerine göre tasarım yapabilmesi, sürenin yeterliliği ve öz değerlendirme ile ilgili faktörler (kendi yaptığı ürünü ve kendi Scratch performans puanlarını değerlendirebilme) ele alınarak nitel temalar oluşturulmuş ve bu temalar kod analizleri ile belirtilerek yorumlanmıştır. Görüşmelerin analizi ile ilgili olarak görsel tasarım ve kod blokları ile ilgili örnek kod şemaları ve bir öğrenciye ait yapılan görüşmenin analizi Tablo 6, Tablo 7 ve Tablo 8’de gösterilmiştir.

Tablo 6: Öğrencilerin HDS/ Çalışma kağıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Görsel Tasarıma ilişkin Görüşleri

Scratch Görsel Programlamanın Tanımlanması
Tasarım aşamasında kolaylık
Tasarım aşamasında zorluk
Hoşuna giden yanları
❖ Görsellik
❖ Tasarım (Özgün Tasarım)
❖ Öğrencilerin Kendi çizimleri
❖ Hazır karakterleri kullanmak
❖ Çizilen ya da hazır karakterlerin canlılık (hareket özelliği kazanması)
Hoşa gitmeyen yanları
❖ Tasarım
❖ Kendi çizimde zorluk yaşama
❖ Hazır karakterler kullanmak
Esneklik (Farklı karakterler kullanma)
Tasarıma yoğunluk verme

Tablo 7: Öğrencilerin HDS/ Çalışma kağıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Kodlama ve Kod Bloklarına ilişkin Görüşleri

Scratch Görsel Programlamanın Tanımlanması

Kodlama aşamasında kolaylık

Kodlama aşamasında zorluk

Hoşuna giden yanları

- ❖ Kodlama
- ❖ Kodlarda Sürükle Bırak Yöntemi
- ❖ Kodlarını çalıştırabilme

Hoşa gitmeyen yanları

- ❖ Kodlama
- ❖ Kodlamada zorluk yaşama
- ❖ Kodlarda mantık kuramama
- ❖ Kodlamaların fazla ve uzun olması
- ❖ Kodlarda sürükle bırak yöntemi- Yanlış yere koyma
- ❖ Kodlarda hata oluşma durumu
- ❖ Program çalışmadığında öfkelenme
- ❖ Sahneler arası geçişlerde zorlanma
- ❖ Kostümler arası geçiş kodlarında zorlanma
- ❖ Kodlama aşamasında yeterli zaman kalmama ve zorlanma

Hata oluşma durumu

- ❖ Kodlarda renkleri karıştırma
- ❖ Yanlış kod parçacığı ekleme
- ❖ Kodları şekil olarak karıştırma
- ❖ Hareket bloğunda hata
- ❖ x, y koordinat kodlarında hata
- ❖ Doğru kodun yerini bulamama
- ❖ Kodlamada sürükle bırak yöntemiyle yanlış yere koyma
- ❖ Sesle ilgili kodlarda hata
- ❖ Görünüm bloğunda hata
- ❖ Kontrol bloğunda hata
- ❖ Değişkenler bloğunda hata
- ❖ Algılama yada Operatörler bloğunda hata
- ❖ Kodlamada bazı şeyleri eklemeyi unutma

Hata oluşmama durumu

Tablo 8: Öğrenci (5) ile gerçekleştirilen Görüşme Kod Analizi Örneği

DG-Öğrenci (5)	Yarı Yapılandırılmış Görüşme Kod Analizi Örneği- Öğrenci5
Zorluk Algısı	“Hem zor hem kolaydı. Zor olmasının nedeni kodlamaların bazen <i>uzun</i> olması, kolay olmasının nedeni ise <i>tasarımların kendimiz yapıyor olmamız</i> ”
Kodlarda hata oluşma durumu	“Doğru kod olmadığına hatalarla karşılaşabiliyoruz. Ve bazen hangi kodları ekleyeceğimiz aklımıza gelmiyor bazen ya da <i>kodların yerini bulmada zorlanabiliyoruz.</i> ”
Hoşa giden yanlar	“ <i>Görsellik ve Kodlama.</i> Çünkü zaten görselleri de eminim pek çok kişi sever. Ama bazıları kodlamayı pek sevmiyor. Benim kodlamayı sevme değişik ve zor olmasına rağmen kodların sürekli bırak şeklinde olması ve <i>renkli renkli olması biraz da güzel ve eğlenceli bence.</i> Hazır olan tasarımlar hoşuma gitti. Bazı görsel karakterler çok komikti. Bana eğlenceli gelmişti.”
Hoşa gitmeyen yanlar	“Tasarımı Biraz hoşuma gitmedi. Tasarımı hani bilgisayarda yapıyoruz ya bana göre de, bilgisayarda karakterleri oluştururken <i>acaba yapabilecek miyim diye düşünürken elim biraz titriyordu.</i> Hani Kendimiz çiziyoruz ya o yüzden biraz zorlandım. Zorlandığım için de <i>tasarımını kendimiz yapmamız konusunda zorluklar yaşadığımız için tasarımını yapmak pek hoşuma gitmiyordu.</i> (heyecan, kaygı, endişelenme, bilgisayar ortamında tasarımda zorluk)”
Hata Oluşma Durumu	“Birkaç kere olmuştu. Yani karıştırdım. <i>Kodların yerlerini karıştırdığım için de bazen hatalar oluştu.</i> ”
Senaryoya Yönelik Algılar	“Tabiki de beğendim ama birşeyi beğenmedim labirent oyununu yaparken çok zorlanmıştım. <i>Zorlandığım için de beğenmedim.</i> İzlediğim senaryolardaki <i>animasyonlar güzeldi, anlaşılırdı .</i> Senaryolar çok <i>farklıydı</i> böyle çok <i>beğenerek izledim.</i> Eksik yanları yoktu bence. <i>Çok iyi</i> hazırlanmıştı senaryolar. Güzeldi yani. Benim <i>hoşuma gitti</i> senaryolar. (Zorluk yaşadığını beğenmeme, senaryoları beğenme, hoşa gitme, ilgi çekici ve farklı bulma, anlaşılabilirlik ve açıklayıcılık)”
Senaryoların kolaylaştırma etkisi	“Evet çünkü o senaryoyu izliyorsun ve izlerken ne yapacağımızı önceden <i>kafamızda canlandırıyorduk</i> zaten kendiliğinden geliyor diğerleri de (Akılda canlandırma)”

Tablo 8: Öğrenci (5) ile gerçekleştirilen Görüşme Kod Analizi Örneği-devamı

Gösterilen senaryoya bağlılık/ Sınırlandırma etkisi	“Hayır. Senaryolar beni hiç sınırlandırmadı. <i>Hayal Gücümü geliştirdi.</i> Faydası da nasıl desem daha çok hayal gücümü geliştirdi. Aklımda yani o senaryoyu izlemeden önce aklımda parti oyunu yapmak falan vardı. Ama o senaryoları izlediğim zaman daha farklı oyunlar da tasarlayabileceğimizi gördüm. Bu da beni <i>çok mutlu etti.</i> Hayal dünyam geliyor ve aklıma daha farklı şeyler geliyordu. Bide Oyunları yaptıkça <i>kendime olan güvenim</i> de arttı. Kendim artık <i>güzel oyunlar yapabileceğime inanıyorum.</i> (Hayal gücünü geliştirme, Mutluluk, özgüven, motivasyon)”
Senaryodaki karakterler	“Erenin annesi çok kibar . Zaten Erenin arkadaşları da böyle. Senaryoyu izlediğimde zaten her yere saçılmış hediyeler. Eren de aynı nazik annesi gibi. Güzel tasarlanmış. Şimdi onlar gerçek hayatta olsaydı gerçekten de güzel olurdu. Mesela ben Eren’in arkadaşı olsaydım. O da beni partiye çağırırdı. Gerçekten de çok mutlu olurum. Onlar benim hayal gücümü genişletti. (<i>içselleştirme, gerçek yaşam etkisi, beğenme, hayal gücünü genişletme</i>)”
Senaryoların yapısal boyutu	“ <i>Senaryoda izlediğimiz açıklamalar</i> ve bizden ne yapmamızı istediği açıklamalar oyun tasarlamamızı <i>kolaylaştırdı.</i> ”
Verilen sürenin yeterliliği	“Bazılarını bitirebildim bazılarını bitiremedim. Yaptığım oyunların bazıları yarım bıraktım çünkü tasarımlarına çok zaman ayırdığımda kodları yetiştiremedim ama bazıları da güzel oldu. Süre yetmediği için yarım bıraktım, sürem yetseydi hepsini tamamlardım.
Senaryo olmadan oyun tasarlama	“Kendimi biraz tuhaf hissedirdim çünkü senaryodaki animasyonu izlemediğimde ne yapacağımı hemen aklıma gelmez. Hayal edemedim. Yada basit şeyler düşünürüm, ve bişeyler düşünmek için fazladan zaman harcamak zorunda kalırım. Ama Senaryodaki Animasyonlar hayallerimi geliştiriyordu. Aslında aklımda bir şeyler geliştiriyordu. Şimdi kendim oturduğumda düşünmeye başlayınca ne yapacağımı bilemediğim için başım ağrıyor. Ama Direk izlediğimde nasıl yapacağım aklıma geliyor.”

Ayrıca veri analizi sürecinde Öğrencilerin haftalık Scratch performans uygulamaları Dereceli puanlama anahtarı (Rubrik) ile değerlendirilmiştir. Rubriklerin kodlama aşamasında Hedefe dayalı senaryonun öğeleri göz önünde bulundurularak öğrencilerin performansları; Geliştirilmeli (1), Orta Düzeyde (2) ve Mükemmel (3) dereceli kriterlerine göre değerlendirilmiştir. Her hafta için dereceli puanlama anahtarı kriterleri ayrı ayrı belirlenmiş ve Öğrencilerin Scratch performansları bu kriterlere göre karşılaştırılmıştır. Örneğin Hafta 2- Piyano oyununa ilişkin dereceli puanlama

anahtarı (Rubrik) in kod analizi Tablo 9’da gösterilmiş olup diğer tüm haftaların Rubrik kod analizleri de Ekler (Ek C) kısmında gösterilmiştir.

Tablo 9: Hafta 2- Etkileşimli Piyano Oyunu Rubrik Kod Analizi

Hafta2

Ünite: Scratch ile Programlamaya Giriş

Etkinlik: Etkileşimli Piyano Oyunu Uygulaması

Kazanımlar: 1.Sahne , Karakter ve Kostüm kullanabilme

2.Hareket bloğunu kullanabilme

3.Görünüm bloğunu kullanabilme

4.Ses bloğunu kullanabilme

5.Kontrol bloğunu kullanabilme

1)Genel Tasarım Öğelerini Düzenleme

Nitelikler	Geliştirilmeli	Orta Düzeyde	Mükemmel
	(1)	(2)	(3)
Renk efektini değiştirme	Hiçbir karakter üzerinde renk efektini değiştirilmemiştir veya Sadece 1 veya 2 Karaktere Renk efektini değiştirme uygulanmıştır.	2 den fazla karaktere Renk efektini değiştirme etkisi uygulanmıştır.	Sahnedeki bütün karakterlere Renk efektini değiştirme doğru bir şekilde uygulanmıştır. Böylece estetik olarak güzel bir görünüm yakalanmıştır.

Tablo 9: Hafta 2- Etkileşimli Piyano Oyunu Rubrik Kod Analizi devamı

Farklı Karakterler ekleyebilme (Balon, Nota, Şapka, İnsan, mikrofon,)	Kazanıma uygun hiç karakter eklenmemiştir veya Sadece 1 karakter eklenmiş yada tasarıma uygun olmayan karakterler eklenmiştir.	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	Tasarıma ait bütün istenilen karakterler doğru bir şekilde eklenmiştir.	
Piyano notalarını ekleyebilme ve yerleştirebilme	Piyano notaları hiç eklenmemiştir veya Piyano notalarından sadece bir tanesi eklenmiştir.	Piyano notalarından sadece 2 3 tanesi eklenmiştir.	Piyano notalarının hepsi doğru bir şekilde eklenmiştir.	
Karakterlere farklı kostümler ekleyebilme	Karakterlere hiç farklı kostümler eklenmemiştir veya Karakterlerden sadece 1 tanesine farklı kostümler eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere farklı kostümler eklenmiştir.	Tasarıma ait bütün istenilen karakterlere farklı kostümler eklenmiştir.	
Sahne arkaplanını değiştirme	Sahne arkaplanı değiştirilmemiştir veya Sahneye arkaplan eklenmiş ancak hedefe uygun bir arkaplan seçilmemiştir.	Sahneye arkaplan eklenmiş ve hedefe uygunluğu kabul edilebilir niteliktedir.	Sahneye hedefe uygun olarak arkaplan eklenmiş ve aynı zamanda sahenin arkaplanına renk efekti uyarlanmıştır.	
2)Hareket Bloğu komutlarını kullanabilme	Karakterleri hareket ettirebilme	Karakterlere hiçbir şekilde hareket etme özelliği kazandırılmamıştır veya Karakterlerden sadece 1 tanesine hareket eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere farklı hareketler eklenmiştir.	Tasarıma ait bütün istenilen karakterlere farklı hareketler eklenmiştir.

Tablo 9: Hafta 2- Etkileşimli Piyano Oyunu Rubrik Kod Analizi devamı

Döndürme kodunu kullanabilme	Karakterlere döndürme kodu eklenmemiş yada doğru şekilde eklenmemiştir veya sadece 1 tanesine eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere farklı döndürme kodları eklenmiştir.	Tasarıma ait bütün istenilen karakterlere farklı döndürme kodları eklenmiştir.
3)Görünüm Bloğu komutlarını kullanabilme			
Kostümler arası geçiş kodunu verebilme	Kostümler arası geçiş kodları doğru şekilde eklenmemiştir veya Karakterlerden sadece 1 tanesine kostümler arası geçiş kodları eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere kostümler arası geçiş kodları eklenmiştir.	Tasarıma ait bütün istenilen karakterlere kostümler arası geçiş kodları eklenmiştir.
4) Ses Bloğu komutlarını kullanabilme			
Tasarıma uygun her notaya vuruş ve nota belirtilerek farklı ses ekleyebilme	Tasarıma Uygun her notaya vuruş ve nota ses kodları hiç kullanılmamıştır veya Karakterlerden sadece 1 , 2 tanesine ses kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere vuruş ve nota ses kodları birbirine karışmayacak şekilde eklenmiştir.	Sahnedeki bütün piyano notalarına tasarıma uygun her notaya vuruş ve nota belirtilerek farklı ses kodları birbirine karışmayacak şekilde eklenmiştir.
5) Kontrol ve Olaylar Bloğu komutlarını kullanabilme			
Yeşil Bayrak Tıklandığında kodunu ekleyebilme	Yeşil Bayrak tıklandığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.

Tablo 9: Hafta 2- Etkileşimli Piyano Oyunu Rubrik Kod Analizi devamı

Karakter Tıklandığında kodunu ekleyebilme	Karakter tıklanmışında bloğu kullanılmamıştır veya Karakterlerden sadece 1, 2 tanesine karakter tıklanmışında kodu eklenmiştir.	Karakterlerden (Piyano notalarından) 2 ve 2 den fazla karaktere karakter tıklanmışında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere (piyano notalarına) karakter tıklanmışında kontrol kodu eklenmiştir.
Bekleme kodunu kullanabilme	Bekleme kodu hiç kullanılmamıştır veya Karakterlerden sadece 1, 2 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	Karakterlerden (piyano notalarından) 2 ve 2 den fazla karaktere bekleme kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere (piyano notalarına) kostümler arası geçişin başarılı şekilde çalışması amacıyla bekleme kodu doğru bir şekilde eklenmiştir.
Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme	Sürekli döngü kodu hiç kullanılmamıştır veya Karakterlerden sadece 1, 2 tanesine sürekli döngü kodu doğru eklenmiştir.	Karakterlerden (piyano notalarından) 2 ve 2 den fazla karaktere sürekli döngü kodu doğru eklenmiştir.	Tasarıma ait bütün istenilen karakterlere (piyano notalarına) sürekli döngü kodu doğru eklenmiştir.
Görevi Tamamlayabilme	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlanmış ve kodları doğru bir şekilde yerleştirmiştir. Yeşil bayrağa tıklanmışında görsel olarak animasyonlu davetiye uygulaması başarılı bir şekilde çalıştırılmıştır.	Görevi kazanımlara uygun olarak yerine getirmemiş, tasarımda ve kodlamada birtakım hatalar yapmış ve yeşil bayrak tıklanmışında istenilen bir biçimde animasyonlu davetiye çalıştıramamıştır.	Puan

Öğrencilerin haftalık Scratch performans uygulamaları sürecinde hafta hafta yapım aşamaları gözlemlenerek gözlem formu oluşturulmuştur. Örneğin Hafta 2- Piyano oyununa ilişkin gözlem formunun kod analizi Tablo 10’da gösterilmiş olup diğer tüm haftaların Gözlem formu kod analizleri de Ekler (Ek E) kısmında gösterilmiştir.

Tablo 10: Hafta 2- Etkileşimli Piyano Oyunu Gözlem Formu Kod Analizi

Öğrenci Adı-Soyadı:	Süre	Başarı	Notlar				
Gözlem Tarihi:							
Saat:							
Değerlendirme	Kısa	Normal	Uzun	Kötü	Orta	İyi	
Kriterleri							
Scratch’te sahneyi değiştirebilir.		*			*		
Scratch’te yeni karakter ekleyebilir.	*					*	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlamış ve kodları doğru bir şekilde yerleştirmiştir. Yeşil bayrağa tıkladığında görsel olarak etkileşimli piyano oyunu uygulaması başarılı bir şekilde çalıştırılmıştır.
Scratch’te bir karaktere farklı kostümler ekleyebilir		*			*		
Hareket kod bloğunu kullanabilir.		*				*	
Kontrol ve Olaylar kod bloklarını kullanabilir.			*		*		
Görünüm kod bloğunu kullanarak, eklenen karakterlere kostüm değişikliği yaptırabilir.			*			*	
Ses kod bloğunu kullanabilir.			*		*		

3.7 Geçerlik ve Güvenirlik

Uygulama aşamasında deney ve kontrol grubunda hem teorik hem de uygulama süreci esnasında,

- Eğiticinin aynı olması,
- Öğretim ortamının aynı olması,
- Öğrenme içeriğinin aynı olması,
- Öğrenme içeriğinin ve hedef senaryoların aynı işleyiş sırasını takip etmesi,
- Öğrencilerin aynı okulda aynı bilgisayar laboratuvarında aynı bilgisayarları kullanması,
- Öğrencilerin uygulama saat aralıklarının ve sürelerinin aynı şekilde planlanması gibi hususlar göz önünde bulundurularak gerçekleştirilmiştir.

Bununla birlikte uygulama sürecinde geçerlik ve güvenilirlik sağlanmaya çalışılmıştır.

- Öğrencilerle yapılan yarı yapılandırılmış görüşmelerin kod analizleri iki ayrı araştırmacı tarafından değerlendirilerek, nitel verilerin analizinde gözlemciler arası güvenilirlik sağlanmaya çalışılmıştır.
- Ayrıca Öğrencilerin haftalık Scratch performanslarını değerlendirmek amacıyla her hafta için ayrı ayrı tasarlanan rubriklerin oluşturulma sürecinde programlama öğretiminde 1 alan uzmanı, 1 içerik uzmanı ve 1 ölçek geliştirme uzmanlarının görüşleri alınarak geçerlik ve güvenilirlik sağlanmaya çalışılmıştır.

DÖRDÜNCÜ BÖLÜM

IV. BULGULAR VE YORUM

Bu bölümde yarı deneysel araştırma deseniyle yapılan araştırma sürecinde elde edilen bulgular ve bu bulguların yorumlanması ifade edilmiştir. Araştırmacı elde ettiği bulgularla Araştırmanın amacı ve alt hedefleri doğrultusunda araştırma sorularına yanıt aramaya çalışmış ve araştırma soruları temel alınarak bu bulgular ortaya koyulmuştur. Araştırmada nicel verilere yönelik yapılan istatistik analizlerinden elde edilen veriler tablolar aracılığıyla gösterilmiş ve yorumlanmıştır. Bunun yanında gözlemler ve yarı yapılandırılmış görüşmeler gibi nitel verilere yönelik tematik içerik analizi yöntemiyle elde edilen bulgular ise temalarla ve kod analizleri ile belirtilmiş ve yorumlanmıştır.

4.1 HDS'lerin Scratch Kullanma Sürecine Etkisi

Yapılan analizler sonucunda elde edilen nicel ve nitel bulguların sunulması ve bu bulgulara ilişkin yorumların yapılması amacıyla gerçekleştirilen analiz sonuçları tablolara aktarılarak gerçekleştirilmiştir. Araştırma sürecinin objektif şekilde aktarılması amacıyla veri toplama araçları olan, ön test- son test ölçme aracı, öğrencilerin Scratch uygulama projeleri ekran kayıtları, öğrencilerle yapılan yarı yapılandırılmış görüşmeler ve araştırmacı gözlem formlarından alıntılar birlikte yorumlanmıştır.

Öğrencilerin Scratch programını kullanma performanslarına dair birçok şekilde veri toplanmıştır. Bunlardan ilki haftalık olarak uygulama esnasında oluşturdukları ürünlerdir. Tasarlanan ürünler Scratch dereceli puanlama anahtarı (Rubrik) ile değerlendirilmiş ve öğrencilerin *görsel tasarım*, *kod bloklarının doğruluğu* ve *ürün tamamlama durumlarından* aldıkları puan ortalamaları tablolarla ve grafiklerle gösterilmiştir.

4.1.1 Görsel Tasarım

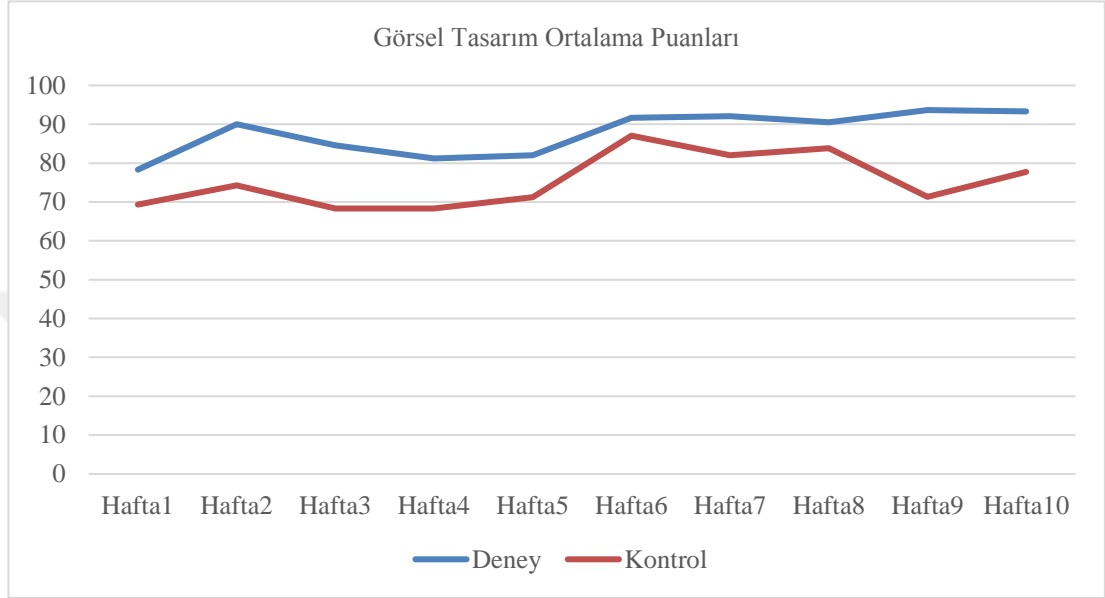
Öğrencilerin Görsel tasarım ortalama puanları, deney ve kontrol grupları için karşılaştırılmalı olarak Tablo 11 ve Şekil 27’de özetlenmiştir.

Tablo 11: Görsel Tasarım Ortalama Puanları

Grup	Hafta 1	Hafta 2	Hafta 3	Hafta 4	Hafta 5	Hafta 6	Hafta 7	Hafta 8	Hafta 9	Hafta 10
Deney	78.33	90	84.58	81.25	82.08	91.67	92.08	90.56	93.67	93.33
Kontrol	69.33	74.33	68.33	68.33	71.25	87.08	82.08	83.89	71.33	77.78

Tablodaki Öğrencilerin HDS/ Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Görsel Tasarım Ortalama Puanlarına bakıldığında Deney grubu öğrencilerinin Kontrol grubu öğrencilerine kıyasla daha yüksek skorlar elde ettiği ve ortalama puanlarının her hafta daha yüksek olduğu görülmüştür. HDS’lerle çalışan deney grubu öğrencilerinin bu doğrultuda geliştirdikleri Scratch uygulama performansları ve öğretim süreçleri gözlemlendiğinde öğrencilerin, kontrol grubuna kıyasla tasarımlarının daha yaratıcı ve senaryolara benzetim yapılarak ve içselleştirilerek daha özgün yapıldığı ancak kontrol grubundaki öğrencilerin tasarım sürecinde zorlandıkları, ne çizeceklerini hemen düşünemedikleri ve tasarım için daha fazla zaman ayırdıkları gözlemlenmiştir. Burada HDS’lerin öğrencilerin Scratch uygulama performanslarına bakıldığında yaratıcı düşünme becerilerini desteklediği ve hayal gücünü geliştirdiği göz önünde bulundurularak öğrencilerin tasarım sürecinde daha keyifli, daha eğlenceli bir süreç izledikleri ve daha güzel, farklı ve yaratıcı tasarımlar yaptıkları, istenilen görevleri yerine getirdikleri ve bunun yanında farklı farklı karakterler de oluşturularak esneklik kazandırdığı gözlemlenmiştir. Grafik incelendiğinde Deney Grubu ve Kontrol grubunun Hafta6 balon patlatma ve Hafta8 tuğla kırma oyununda görsel tasarım puanlarının oldukça birbirine yakın olduğu görülmüştür. Bu senaryolarda istenilen görevler daha çok kod bloklarıyla ilgili olup tasarımsal olarak anlamlı farklılık görülmemiştir. Bununla birlikte Hafta9 trafik

kuralları oyunu senaryosunda deney grubu ve kontrol grubu arasında oldukça fazla fark olduğu gözlemlenmiştir. Bu haftanın özelliği diğer haftalara göre nispeten daha kompleks bir yapıda olmasıdır.



Şekil 27: Görsel Tasarım Ortalama Puanları

Ayrıca öğrencilerle yapılan görüşmeler sonucunda da deney grubundaki öğrencilerin tasarım aşamasında çok fazla zorlanmadıkları aksine eğlenceli ve keyifli bir şekilde tasarım görevlerini tamamladıkları ifade edilmiştir. Öğrencilerin kendi tasarımlarını beğendikleri, senaryolardaki hazır görsellerden yararlandıkları ve hatta kendi çizimleriyle oluşturdukları kısımlarda motivasyonlarının ve özgüvenlerinin arttığına dair bildirimlerde bulunmuşlardır. Örneğin deney grubundan D3 (deneygrupöğrenci3) ve D7 (deneygrupöğrenci7) bu durumu şöyle dile getirmiştir.

“Tasarım aşaması daha çok hoşuma gitti. Çünkü insan tasarımı daha kolay yapabiliyor. Tasarım aşamasında hem karakterleri bazen kendim çizdiğim hem de bazen hazır karakterleri kullanmak hoşuma gidiyordu. Ve daha kolay geliyordu bana.” (D3)

“Görsel tasarımını kendim yaptığım için, kendim tasarladığım için güzel görünüyor. Kendi çizdiğimde daha çok hoşuma gidiyor hazır karakter eklemektense. Nasıl olsa kendi emeğinle yapıyorsun yani.” (D7)

Bununla birlikte deney grubundaki öğrencilerden bazılarının tasarım aşamasında zorlandıkları ancak kontrol grubundaki öğrencilerden tasarım aşamasında zorluk çekenlerin deney grubuna kıyasla daha fazla olduğu gözlemlenmiştir. Örneğin Deney grubundan D5 ve K4 (kontrolgrupöğrenci4) bu durumu şöyle ifade etmiştir.

“Tasarımda, karakter falan bulmakta daha çok zorlandım. Bilgisayarda olduğu için çizemedim.” (D5)

“Tasarımını oluştururken zorlanıyordum ve geç kalıyordum yapamıyordum bazı tasarlayacağım şeyleri.” (K4)

Kontrol grubu öğrencileriyle yapılan görüşmeler incelendiğinde tasarlarken akılda canlandırmada zorlandıkları, neyi tasarlayacaklarını düşünmek için ekstra süre harcadıkları ancak yine de tasarlamaya çalıştıkları ifade edilmiştir. Öğrencilerin tasarımda daha çok hazır karakterler kullandıkları, bazı senaryolarda tasarım aşamasını beğendikleri, kendi çizimiyle tasarladıkları ya da hazır kullandıkları nesnelerin hareket özelliği kazanmasının kontrol grubu öğrencilerini de motive ettiği belirtilmiştir. Ayrıca bazı öğrencilerin tasarım aşamasının onların daha çok hoşuna gittiği sonuçları ortaya çıkmıştır. Örneğin kontrol grubundan K2 bu durumu şöyle ifade etmiştir.

“Kendim tasarlayınca daha çok beğeniyorum. Kendim çizince daha kendime yakın hissediyorum, mutlu oluyorum.” (K2)

Hem deney grubunda hem de kontrol grubunda tasarım kısmında kendi çizim aşamasında ya da hazır görsel karakterler bulmada zorluk yaşayan öğrenciler olmuştur. Bu durumu bazı öğrenciler şu şekilde dile getirmiştir.

“Görsellikte kendim çizmek biraz daha zor olduğu için daha çok hazır karakterlerle tasarım daha iyi oluyordu benim için. Çünkü çizmede daha çok zorlandığımız için biraz vakit kaybı oluyordu.” (D1)

“Oradaki hazır karakterleri bazen bulmakta zorlanıyorum bazen de tam istediğim şekilde güzel bulamıyorum. O yüzden kendim çizerek karakterleri

kendim tasarlıyorum. Oradaki karakterleri bazen güzel bulmuyorum, beğenmiyorum.” (D4)

“Bazen tasarlarken zorlandım. Kendim çizemedim bazı şeyleri.” (K4)

Deney grubundaki öğrenciler tarafından tasarladıkları nesnelere ya da hazır kullandıkları görsel nesnelere hareket ve animasyon özelliği kazandırmalarının onları motive ettiği ifade edilmiştir. Örneğin deney grubundan D2 bu durumu şöyle ifade etmiştir.

“Kendim hareket ettirebiliyorum sonuçta kendim çizebiliyorum. Orada kendi çizdiğim karakterler hareket edince daha da motive ediyor beni.” (D2)

Aynı şekilde Kontrol grubu öğrencilerinden bazıları da tasarladıkları nesnelere hareket ve animasyon özelliği kazanmasının onlar için anlam ifade ettiğini belirtmiştir. Örneğin kontrol grubundan K6 bu durumu şöyle ifade etmiştir.

“Adamın yürüme özelliği ya da dalgıcın yüzme özelliği kazanması gibi yaptığımız karakterler hareket edince hoşumuza gidiyordu.” (K6)

Deney grubundaki öğrencilerin genellikle senaryolardaki karakterleri birebir içselleştirdikleri, bu nedenle tasarıma çok fazla ağırlık verdiklerinde bazı öğrencilerin süre olarak görevi yetiştirmede zorlandıkları görüşleri ifade edilmiştir. Kontrol grubundaki öğrencilerinde ise neyi tasarlayacaklarını düşünmek için ekstra süre harcadıkları ve bu nedenle görevi yetiştirmede zorlandıkları gözlemlenmiştir. Bu durumla ilgili Örneğin deney grubundan D1 şu şekilde görüş belirtmiştir.

“Tasarımına çok ağırlık verdim sanırım. Çünkü yaptığım oyunun güzel olmasını isterim. Her ne kadar kodların çalışması önemli ise tasarım da önemli bence. Bu yüzden tasarıma da çok zaman ayırdım.” (D1)

Öğrencilerle yapılan yarı yapılandırılmış görüşmeler sonucunda öğrencilerin görsel tasarıma ilişkin görüş frekansları Tablo 12’de özetlenmiştir.

Tablo 12: Öğrencilerin HDS/ Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Görsel Tasarıma ilişkin Görüş Frekansları (DG-KG)

Scratch	Görsel	Deney Grubu (f)	Kontrol Grubu (f)
Programlamanın Tanımlanması			
Tasarım aşamasında kolaylık		7	4
Tasarım aşamasında zorluk		4	5
Hoşuna Giden yanları			
Görsellik		6	4
Tasarım (Özgün tasarım)		7	4
Öğrencilerin Kendi Çizimleri		3	3
Hazır karakterleri kullanmak		4	7
Çizilen ya da hazır karakterlerin canlılık (hareket) özelliği kazanması		5	4
Hoşa gitmeyen yanlar			
Tasarım		3	1
Kendi çizimde zorluk yaşama		5	6
Hazır karakterler kullanmak		1	-
Esneklik			
Kendi düşüncelerimize göre tasarım		5	2
Farklı karakterler kullanma		6	-
Tasarıma yoğunluk verme		4	1
Tasarımı beğenme		6	4

4.1.2 Kod Bloklarının Doğruluğu

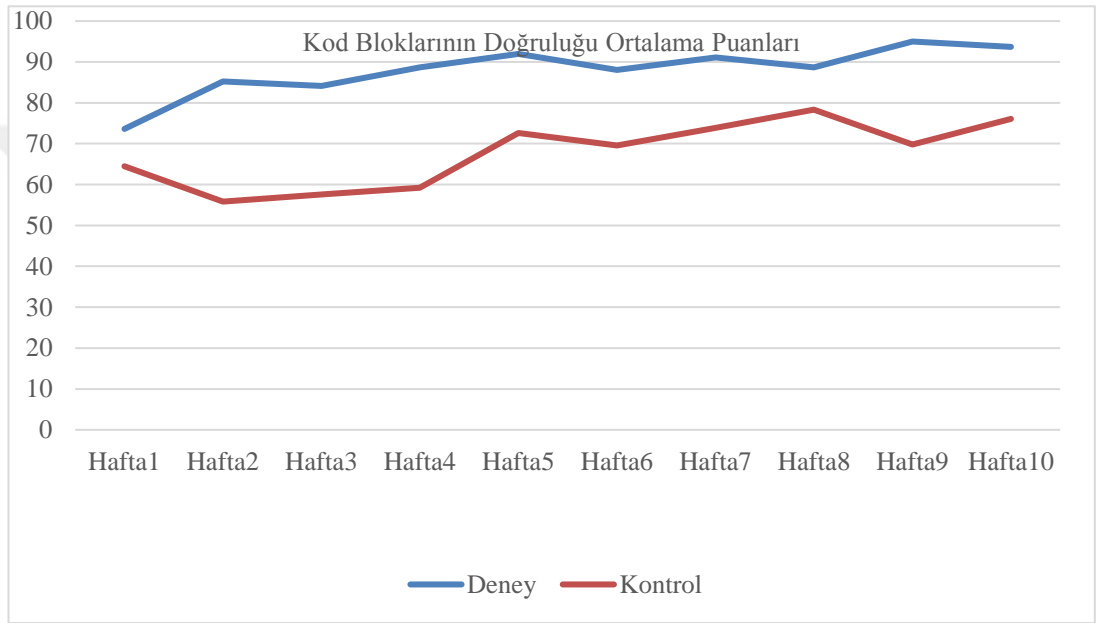
Öğrencilerin HDS/ Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Kod Bloklarının Doğruluğu Ortalama Puanları Tablo 13 ve Şekil 28’de gösterilmiştir.

Tablo 13: Kod Bloklarının Doğruluğu Ortalama Puanları

Grup	Hafta 1	Hafta 2	Hafta 3	Hafta 4	Hafta 5	Hafta 6	Hafta 7	Hafta 8	Hafta 9	Hafta 10
Deney	73.61	85.20	84.16	88.67	91.94	88.06	91.11	88.64	95.00	93.72
Kontrol	64.44	55.83	57.60	59.25	72.64	69.58	73.89	78.33	69.79	76.03

Tablodaki Öğrencilerin HDS/ Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch kod bloklarının doğruluğu ortalama puanlarına bakıldığında deney grubu öğrencilerinin kontrol grubu öğrencilerine kıyasla daha yüksek skorlar elde ettiği ve ortalama puanlarının her hafta daha yüksek olduğu görülmüştür. Deney grubu öğrencilerinin Scratch kullanım performansları ve öğretim süreçleri gözlemlendiğinde öğrencilerin kod bloklarıyla ilgili görevleri gerçekleştirmede kontrol grubuna kıyasla daha başarılı oldukları, daha az hata yaptıkları ve daha az zorlandıkları gözlemlenmiştir. Kontrol grubu öğrencilerinin kodlamada zorluk yaşama, kodlarda mantık kuramama, kodlamaların fazla ve uzun olması, kodlarda sürükle bırak yöntemi ile yanlış yere koyma, kodlarda hata oluşma durumu, program çalışmadığında öfkelenme, kodlarda renkleri karıştırma, yanlış kod parçacığı ekleme, kodları şekil olarak karıştırma, doğru kodun yerini bulamama, kostümler arası geçiş kodlarında zorlanma, kodlamaya yeterli zaman kalmama/zorlanma ve kodlamayı istenilen sürede yetiştiremememe, kod bloklarını yerleştirmede yanlış yere koyma ya da eksik kod yazımı şeklinde sorunlarla daha çok karşılaştıkları ve kod yazımında daha çok kaygı yaşadıkları bu nedenle performanslarının daha düşük olduğu gözlemlenmiştir.

Deney Grubu ve Kontrol grubunun Hafta1 animasyonlu davetiye ve Hafta8 tuğla kırma oyununda Kod bloklarının doğruluğu ortalama puanlarının oldukça birbirine yakın olduğu görülmüştür. Bu senaryolarda istenilen görevler daha çok görsel tasarımıyla ilgili olup kod bloklarının doğruluğu bakımından anlamlı farklılık görülmemiştir. Bununla birlikte hafta2 etkileşimli piyano oyunu ve hafta 9 trafik kuralları oyunu senaryosunda deney grubu ve kontrol grubu arasında oldukça fazla fark olduğu gözlemlenmiştir.



Şekil 28: Kod Bloklarının Doğruluğu Ortalama Puanları

Ayrıca öğrencilerle yapılan görüşmeler sonucunda her iki gruptaki öğrenciler tarafından kodlamanın sürükleyerek bırakma yöntemiyle yapılmasının onlar için kolaylaştırıcı ve eğlenceli şekilde olduğu ifade edilmiştir. Örneğin her iki gruptaki öğrencilerden bazıları bununla ilgili görüş belirtmiştir.

“Kodların sürükleyerek bırakma şeklinde eklenmesi bize kolaylık sağlıyordu. Öyle olmasa belki 2 saatte bitirebileceğimiz oyunu 4 saatte bitiremeyebilirdik belki.” (D3)

“Kodlarını sürükleyerek bırakıp hazır aldığımızda bize kolaylık sağlıyordu. Daha kolay yazabilmemizi sağlıyordu. Kodların sürükleyerek bırakma şeklinde olması ve renkli yapıda olması biraz da güzel ve eğlenceli bence.” (D1)

“Kodları sürükleyerek bıraktığımızda kolay bir şekilde yapabiliyoruz, bu hoşuma gidiyor.” (K4)

Bununla birlikte kodların sürükleyip bırak yöntemiyle kodların yanlış yere koyulmasından kaynaklı bazı öğrencilerin hata yapmalarına neden olduğu da görüşlerde ifade edilmiştir. Örneğin deney grubu öğrencilerinden D3 şu şekilde görüş belirtmiştir.

“Bazı kodlarda döngüleri iç içe koyarken üst üste koyuyordum. Koyamıyordum yerine sonra geri çekerken bütün kod bloğu geliyordu. Kodlar siliniyordu, karışıklık oluyordu.” (D3)

Kodlama aşamasının deney grubu ve kontrol grubu öğrencilerinin bazılarının hoşuna gitmediği, kodların yerini bulmada zorluk yaşadıkları ve kodlamayla ilgili kaygı yaşadıkları görüşleri belirtilmiştir. Örneğin deney grubu öğrencilerinden D4 ve kontrol grubu öğrencilerinden K7 şu görüşleri ifade etmiştir.

“Bazı kodlar blokların en altında olduğunda kodları bulamıyordum. O yüzden kodlamada zorlanıyorum. Mesela kontrol bloğunda en alta on kez tekrarlı kodu var mesela, onu bulamayınca zorlandım. Bazen bloklarda en alta bakmayı unutuyorum. Bu yüzden döngü kurarken kodlarda yanlışlarım oluyor.” (D4)

“Kodları bloklarda bulurken zorlanıyor, yerlerini karıştırıyordum. Kodları arayıp bulmaya çalışırken vakit kaybediyorduk.” (K7)

Bazı haftalarda kodlamaların uzun ve fazla olması, süre kaygısından dolayı her iki gruptaki öğrenciler için de olumsuz durum oluşturmuştur. Bununla ilgili bazı öğrencilerin görüşleri şu şekildedir:

“Kodlamalarda bazen şöyle çok uzun oluyor. Uzun olunca insan yapamayacağı hissine kapılıyor. Kısa süreli yapabilseydik hemen, hem de onu oynamaya da daha çok zamanımız olsaydı daha güzel olurdu.” (D4)

“Kodları uzun olduğunda hoşuma gitmiyordu. Bazı oyunlarda kodlar kısa olduğunda onlar hoşuma gidiyordu.” (K2)

Ayrıca her iki gruptaki öğrencilerin kodlama aşamasında dikkatsizlik ya da kaygıdan dolayı kod bloklarında doğru kodu bulamama ya da yanlış yere yerleştirme gibi nedenlerden dolayı hata yaptıkları gözlemlenmiştir. Deney grubundaki öğrencilerin kontrol grubuna kıyasla daha az hata yaptığı, kontrol grubundaki öğrencilerin hangi kodları kullanacağını bilememesinden kaynaklı yanlış kodları eklemesine ve bu nedenle fazla süre kaybetmesine neden olduğu görülmüştür. Örneğin bazı öğrenciler bu durumla ilgili görüşünü şu şekilde dile getirmiştir.

“Hata oluştu mesela -90 yapmayı unuttum sadece 90 yapmışım, sağ yön yerine sol yönle karıştırdım. Başka hatam olmadı.” (D5)

“Biraz hatalar oldu ben yanlış yaptım yanlış yerlere koymuşum kodları birkaç kez.” (K3)

“Kodlar zor geldiğinde yetiştiremedim. Kodları bulurken biraz zaman geçiyordu. Bide kodları ekledik diyelim yanlış kod ekleyince oradan çıkartırken onu alıyorduk onu koyuyorduk fln kafam karışıyordu. Kodları yapıştırdıktan sonra ayırması zor oluyordu yanlışları düzeltirken.” (K7)

Ancak kodlama ile ilgili her iki gruptaki öğrencilerin de kodlarını doğru yazıp çalıştırabildiklerinde onları motive ettiği ve tasarladıkları oyunları çalıştırıp ve hatta kalan sürede kendi oyunlarını oynamaları onları oldukça mutlu ettiği, başarabildiklerinde kodlamanın hoşlarına gittiği gözlemlenmiştir. Deney grubundaki öğrencilerin kontrol grubuna kıyasla oyunları bitirip çalıştırmada daha başarılı oldukları ve tasarladıkları oyunları oynadıkları, motivasyonlarının bu şekilde arttığı gözlemlenmiştir. Ayrıca öğrencilerle yapılan görüşmelerde de bazı öğrenciler tarafından şu şekilde dile getirilmiştir.

“Kodlarda hemen yapayım da hemen çalıştırayım diye acele ederken bazı kodlarda karıştırmıştım. Hemen bitirmeye çalışıyordum herkesten önce bitirip oyunu çalıştırayım ve kalan sürede de biraz oyun oynamak için hızlı yapmaya çalışıyordum.” (D6)

“Yaptıktan sonra oynamak benim hoşuma gidiyordu.” (K1)

Bunun yanında deney grubu öğrencilerinin hedefe dayalı senaryoları izleme sürecinde hangi kodları kullanacaklarını düşünmeleri için ekstra zaman olması onlar için avantajlı durum oluşturmuştur. Bununla birlikte çalışma kâğıtlarında kod bloklarıyla ilgili yönergelerin bulunması, kodların öğreticiliği açısından kontrol grubu öğrencilerine kolaylık sağladığı gözlemlenmiştir. Örneğin kontrol grubu öğrencilerinden K6 bu durumla ilgili şöyle bir görüş belirtmiştir.

“Çalışma kâğıtları sayesinde kodların yapısını ve mantığını anlayabildim. Çok öğreticiydi. Scratch’i iyi anlamama yardımcı oldu.” (K6)

Öğrencilerle yapılan yarı yapılandırılmış görüşmeler sonucunda öğrencilerin kod bloklarının doğruluğuna ilişkin görüşleri Tablo 14’de özetlenmiştir.

Tablo 14: Öğrencilerin HDS/ Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Kod bloklarının doğruluğu üzerine Görüşleri (DG-KG)

Scratch	Görsel	Deney Grup	Kontrol Grup
Programlamanın Tanımlanması			
Kodlama aşamasında kolaylık		5	1
Kodlama aşamasında zorluk		4	6
Hoşuna giden yanları			
Kodlama		5	2
Kodlarda sürükle bırak yöntemi		3	2
Hoşa gitmeyen yanları			
Kodlama		2	4
Kodlamada zorluk yaşama		4	4
Kodlarda mantık kuramama		1	2
Kodlamaların fazla ve uzun olması		1	3

Tablo 14: Öğrencilerin HDS/ Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Kod bloklarının doğruluğu üzerine Görüşleri (DG-KG)-devamı

Scratch	Görsel	Deney Grup	Kontrol Grup
Programlamanın Tanımlanması			
Kodlarda sürükle bırak yöntemi- Yanlış yere koyma		2	3
Kodlarda hata oluşma durumu		4	5
Program çalışmadığında öfkelenme		1	4
Hata Oluşma Durumu			
Kodlarda renkleri karıştırma		1	-
Yanlış kod parçacığı ekleme		3	5
Kodları şekil olarak karıştırma		1	1
Hareket bloğunda hata		1	4
x, y koordinat kodlarında hata		-	4
Doğru kodun yerini bulamama		2	4
Kodlamada sürükle bırak yöntemiyle yanlış yere koyma		-	3
Hata olmama durumu		3	-
Sesle ilgili kodları aktarma/hata		-	1
Sahneler arası geçişlerde zorlanma		-	2
Görünüm bloğunda hata		2	4

Tablo 14: Öğrencilerin HDS/ Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Kod bloklarının doğruluğu üzerine Görüşleri (DG-KG)-devamı

Scratch	Görsel	Deney Grup	Kontrol Grup
Programlamanın Tanımlanması			
Kontrol bloğunda hata		2	3
Değişkenler bloğunda hata		1	4
Algılama ya da Operatörler bloğunda hata		2	4
Kostümler arası geçiş kodlarında zorlanma		1	3
Kodlamada bazı şeyleri eklemeyi unutma		3	5
Kodlamaya yeterli zaman kalmama/zorlanma		2	6
Kodlarını çalıştırabilme		5	1

4.1.3 Ürün Tamamlama Durumu

Öğrencilerin HDS/Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch uygulamalarının ürün tamamlama durumları ortalama puanları Tablo 15 ve Şekil 29'da gösterilmiştir.

Tablo 15: Ürün Tamamlama Durumları Ortalama Puanları

Grup	Hafta 1	Hafta 2	Hafta 3	Hafta 4	Hafta 5	Hafta 6	Hafta 7	Hafta 8	Hafta 9	Hafta 10
Deney	75.75	87.05	84.25	87.43	89.48	88.96	91.41	89.05	94.68	93.65
Kontrol	66.66	62.95	59.75	60.76	72.29	73.96	76.41	79.52	70.16	76.35

Tablodaki Öğrencilerin HDS/ Çalışma kâğıtları doğrultusunda geliştirdikleri Scratch Ürün Tamamlama Durumları Ortalama Puanlarına bakıldığında Deney grubu

öğrencilerinin Kontrol grubu öğrencilerine kıyasla daha yüksek skorlar elde ettiği ve ortalama puanlarının her hafta daha yüksek olduğu görülmüştür. HDS lerle çalışan deney grubu öğrencilerinin Hedefe dayalı Senaryolar doğrultusunda geliştirdikleri Scratch uygulama performansları ve öğretim süreçleri gözlemlendiğinde öğrencilerin görevi tamamlama durumlarının daha yüksek olduğu ve çoğunluğun senaryolardaki görevleri istenilen sürede bitirdikleri gözlemlenmiştir. Örneğin deney grubu öğrencilerinden D2 şu şekilde görüş belirtmiştir.

“Bitirebildim bence güzel oldu ve ben kendi yaptığım oyunları beğeniyordum. Hem tasarım hem kodlama yönünden güzel oldu.” (D2)

Bunun yanında kod bloklarında hata oluşma durumu, tasarıma yoğunluk verme ve bu nedenle kodlamaya yeterli zaman ayıramama gibi sebeplerden dolayı deney grubu öğrencilerinden bazılarının istenilen sürede yetiştiremediği gözlemlenmiştir. Bu durumla ilgili deney grubundan bazı öğrenciler görüşünü şu şekilde dile getirmiştir.

“Çünkü tasarıma çok ağırlık verdim. Bazı uygulamalarda kendi yaptığım tasarımı beğenmeyip birkaç kere yapıp tekrar sildiğim zaman kodlamaya yeterli zaman kalmadı.” (D3)

“Tasarımına çok ağırlık verdim sanırım. Çünkü yaptığım oyunun güzel olmasını isterim. Her ne kadar kodların çalışması önemli ise tasarım da önemli bence. Bu yüzden tasarımına da çok zaman ayırdım. Bitiremememin sebebi bazı oyunlarda tasarımla uğraşırken böyle çok dikkat ediyoruz. Bir şeyin yanlış gitmesini istemeyiz. O yüzden fazla önem veriyoruz yapmamıza.” (D4)

Ayrıca deney grubu öğrencilerin senaryoları gerçekleştirme amacıyla daha heyecanlı oldukları, tasarım aşamasında daha yaratıcı ve esnek oldukları, oyunları çalıştırıp oynadıkları, kendi ürünlerini yüksek skorlarla değerlendirdikleri ve birbirleriyle paylaştıkları gözlemlenmiştir. Örneğin deney grubu öğrencilerinden D6 görüşünü şu şekilde ifade etmiştir.

“Bitirebildim bence güzel oldu ve ben kendi yaptığım oyunları beğeniyordum. Hem tasarım hem kodlama yönünden güzel oldu.” (D6)

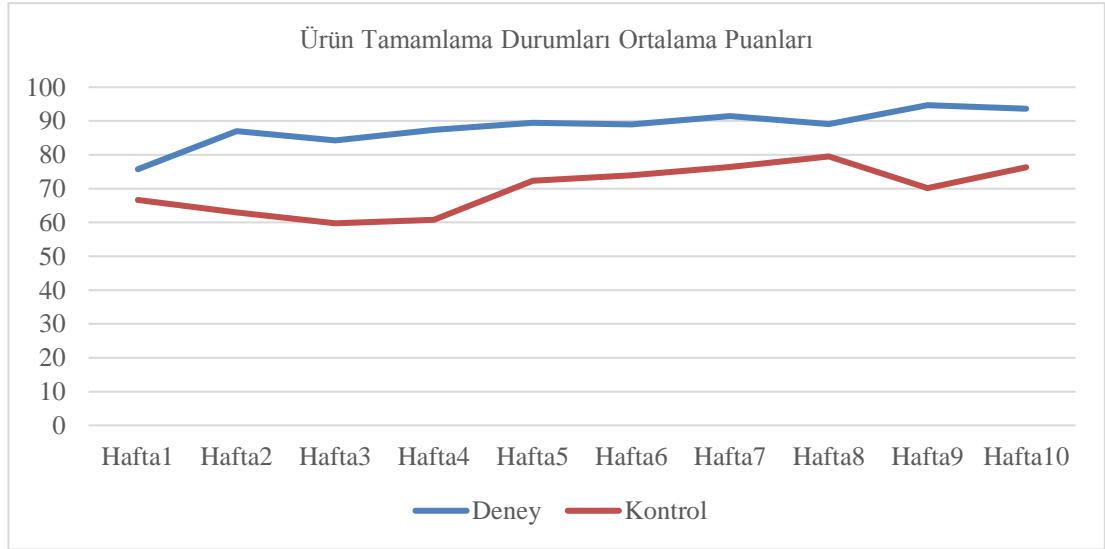
Kontrol grubunda ise genellikle kodlama aşamasında hata yaşadıkları, tasarımla ilgili ne yapacaklarını düşünmek için fazladan zaman harcadıkları, kodlamaya yeterli zaman ayıramadıkları, kaygı yaşadıkları ve bu nedenle oyunu istenilen sürede yetiştiremedikleri gözlemlenmiştir. Bu durumla ilgili kontrol grubu öğrencilerinden bazıları görüşlerini şu şekilde ifade etmiştir.

“Sahneleri ve karakterleri kendim çizdiğimde yetiştiremedim. Kodlamaya yeterli sürem kalmadı.” (K4)

“Bitiremedim çünkü tasarımda çok vakit harcadım. Tasarlama aşamasında ne yapacağımı düşünmek için biraz vakit harcadım” (K2)

Ancak kontrol grubunda yine de bazı oyunları tamamen bitirip başarılı bir şekilde çalıştıran öğrencilerin olduğu ancak deney grubunda tamamlayanların sayısının daha fazla olduğu gözlemlenmiştir. Örneğin kontrol grubu öğrencilerinden K5 bu durumla ilgili şu şekilde görüş belirtmiştir.

“Bitirdiğim oyunlar gayet iyi oldu ama bitiremediğim oyunlar eksik olduğu için tam güzel olmadı.” (K5)



Şekil 29: Ürün Tamamlama Durumları Ortalama Puanları

Deney Grubu ve Kontrol grubunun hafta1 animasyonlu davetiye ve hafta8 tuğla kırma oyununda Ürün tamamlama ortalama puanlarının oldukça birbirine yakın olduğu görülmüştür. Bununla birlikte hafta3 labirent oyunu, hafta4 hediye toplama oyunu ve

hafta 9 trafik kuralları oyunu senaryosunda deney grubu ve kontrol grubu arasında oldukça fazla fark olduğu gözlemlenmiştir.

Ayrıca öğrencilerle yapılan yarı yapılandırılmış görüşmeler incelendiğinde, deney grubundaki öğrencilerin istenilen sürede bitirebilen öğrencilerin kontrol grubuna kıyasla daha fazla olduğu gözlemlenmiştir. Öğrencilerden kendilerini değerlendirmeleri istendiğinde deney grubundaki öğrenciler kontrol grubundakilere kıyasla daha yüksek puanlarla değerlendirmişlerdir. Deney grubundaki öğrencilerin hepsi 91-100 aralığında not baremi ile değerlendirirken kontrol grubundaki öğrencilerden bir kısmı ($f=4$) 91-100 aralığında, bir kısmı ise ($f=3$) 81-90 aralığında not baremi ile değerlendirmiştir. Deney grubundaki öğrencilerin yaptıkları ürünlerde tasarımı beğendikleri ancak kontrol grubundaki öğrencilerin tasarımdan çok keyif almadıkları da gözlemlenmiştir. Her iki grupta da öğrencilerin Animasyonlu davetiye, hediye toplama oyunu, tuğla kırma oyunu ve mevsimler oyununu kolaylıkla tamamladıkları ancak bilgi yarışması, trafik işaretleri oyunlarını tamamlamada zorluk yaşadıkları gözlemlenmiştir. Öğrencilerle yapılan yarı yapılandırılmış görüşmeler sonucunda öğrencilerin ürün tamamlama durumlarına ilişkin görüşleri Tablo 16'da özetlenmiştir.

Tablo 16: Öğrencilerin HDS/ Çalışma kağıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Ürün Tamamlama durumları üzerine ilişkin Görüşleri (DG-KG)

Scratch	Görsel	Deney Grup (f)	Kontrol Grup (f)
Programlamanın Tanımlanması			
Verilen Sürenin Yeterliliği			
İstenilen Sürede bitirebilme		5	1
İstenilen Sürede yetiştirememe		1	4
Hata/ Kod bloklarını karıştırma		2	5
Tasarıma ağırlık verme		2	1

Tablo 16: Öğrencilerin HDS/ Çalışma kağıtları doğrultusunda geliştirdikleri Scratch uygulamalarının Ürün Tamamlama durumları üzerine ilişkin Görüşleri -devamı

Scratch	Görsel	Deney Grup (f)	Kontrol Grup (f)
Programlamanın Tanımlanması			
Kodlamaya yeterli zaman kalmama/ Zorlanma		2	4
Bazen bitirebilme/ yetiştirememe	Bazen	2	6
Kendi yaptığı ürünü iyi olarak değerlendirme		6	5
Tasarımı beğenme		4	3
Kodlarını çalıştırabilme		5	2
Not-Puan Aralığı			
100-91		7	4
90-81			3
80-71			
70-61			

4.2 HDS'lerin Bilgi Düzeyindeki Etkisi

HDS grubundaki öğrencilerin, ön-test puanları kontrol edildiğinde diğer grup öğrencilerine göre son-test puanlarının anlamlı şekilde farklılık gösterip göstermediğini tespit etmek amacıyla analizler yapılmış ve bu analizler yorumlanmıştır. Bu çalışmada, daha önce hiç programlama becerisi olmayan ve motivasyonu düşük öğrencilerin programlama performanslarının, hedefe dayalı

senaryo uygulamaları sonrasında anlamlı şekilde deęişiklik göstermesi beklenmektedir.

Arařtırmacı tarafından geliştirilen öğrencilerin Scratch performanslarını yoklayan ön-test ve son-test ölçme aracı, HDS'lere ve Çalışma kâğıtlarına dayalı 14 haftalık uygulama öncesinde ve sonrasında Deney Grubu ve Kontrol Grubu öğrencilerine uygulanmış ve sonuçları bağımsız örneklem t testi ile karşılaştırılmıştır. Öncelikle her iki grubun ön-test ve son-test genel ortalamalarına ve ön test sonuçları arasında fark olup olmadığına bağımsız örneklem t testi (independent sample t-test) ile bakılmış, ön test sonuçları arasında anlamlı fark olmadığı görülmüştür. Bu durumda öğrencilerin başlangıç bilgi düzeylerinin benzer seviyede olduğu varsayılabilir. Böylece her iki gruptaki öğrencilerin ön-test düzeyleri kontrol altına alınarak performans gelişim düzeyleri incelenmiş ve gruplar arasında farklılık olup olmadığına son test değerleri üzerinden bakılmıştır.

Testin sonucunda her iki grubun da başlangıç becerileri arasında yüksek iliş ancak bitiş becerileri arasında anlamlı fark olduğu görülmüştür. Bu testin sonucunda her iki grubun betimsel istatistikler tablosu aşağıdaki Tablo 17'de gösterilmiştir.

Tablo 17: Betimsel İstatistikler (DG-KG Ön test-Son test)

	Grup	N	Mean		Std. Deviation			
			Statistic	Statistic	Statistic	Statistic	Std Error	Statistic
ÖN TEST	DG	20	54.45	15.656	-.627	.512	.565	.512
	KG	20	53.20	17.410	-.513	.992	-.519	.992
SON TEST	DG	20	93.40	3.966	.058	.512	-.154	.512
	KG	20	89.60	4.684	.114	.992	.062	.992

Betimsel sonuçlar incelendiğinde, genel ortalamaların her iki grupta da ön-testten son-teste geçişte artış gösterdiği gözlemlenmektedir. Fakat deney grubunun kontrol grubuna kıyasla yüksek skorlu ürünler yaptığı ve son test sonuç değerlerinin daha

yüksek olduğu gözlemlenmiştir. Parametrik testlerden olan bağımsız örneklem t testi uygulamak için iki örneklem grubunun birbirinden bağımsız olmasını ve bağımsız gözlemlerin yapılabilmesini gerektirir ki araştırmadaki deney ve kontrol grubunun birbirinden bağımsız ve farklı olması bu varsayımı karşılamaktadır. Ayrıca en önemli varsayımlardan bir diğeri de normal dağılım varsayımdır. Scratch performanslarını ölçen ön-test ve son-test puanları incelendiğinde çarpıklık (Skewness) ve basıklık (Kurtosis) değerlerinin -1.00 ile +1.00 aralığında olduğu gözlemlenmiştir. Bununla birlikte histogramlar incelendiğinde de bağımlı değişkene ilişkin ölçümlerin dağılımının her iki grupta da normal dağılımlı olduğu görülmüştür. Ayrıca puanların normalliğe uygunluğunu incelemek amacıyla Kolmogorov-Smirnov (K-S) ve Shapiro-Wilks testleri yapılmış ve herhangi bir aykırılık tespit edilmemiştir ($p > .05$). Bu testin başka bir varsayımı da kategorik değişkenlerin sürekli (continious) veya oranlı ölçümlerinden oluşmasıdır ve karşılaştırma yapılan iki grubun ortalaması aynı değişkene aittir. Bu testlerin sonucunda her iki grubun Normallik Testleri aşağıdaki Tablo 18’de gösterilmiştir.

Tablo 18: Normallik Testleri

	Grup	Kolmogorov-Smirnov			Shapiro-Wilk		
		Statistic	df	Sig.	Statistic	Df	Sig.
ÖN	Deney	.117	20	.200	.957	20	.491
TEST	Kontrol	.154	20	.200	.933	20	.174
SON	Deney	.157	20	.200	.916	20	.082
TEST	Kontrol	.184	20	.074	.966	20	.664

H_0 : Deney ve kontrol grubu öğrencilerinin ön-test/son-test puanları arasında anlamlı bir farklılık yoktur. ($\mu_1 = \mu_2$)

H_a : Deney ve kontrol grubu öğrencilerinin ön-test/son-test puanları arasında anlamlı bir farklılık vardır. ($\mu_1 \neq \mu_2$)

Deney Grubunda Scratch bilgisini ölçen ön-test ($M=54.45$, $SD=15.656$) ve son-test ($M=93.40$, $SD=3.966$) ortalamaları göz önüne alındığında ön testten son teste bir artış gözlemlenmektedir. Kontrol grubunda da ön-test ($M=53.20$, $SD=17.410$) ve son-test

($M=89.60$, $SD=4.684$) değerlerine bakıldığında yine bir artış gözlemlenmektedir. Grupların ön-test puanları dikkate alınarak çalıştırılan testin sonucunda, gruplar arasında anlamlı bir puan farkı olmadığı raporlanmıştır, $t(38) = .24$, $p > .05$; yani H_0 reddedilememiştir. Deney ve kontrol grubundaki öğrencilerin son-test puanları karşılaştırıldığında, bağımsız örneklem t-test sonucunda deney grubundakilerin puanlarının anlamlı şekilde diğer gruptan daha yüksek olduğu ortaya çıkmıştır, $t(38) = 2.77$, $p < .05$, $r = .41$. Bu durumda H_0 reddedilmiştir. Etki büyüklüğü orta olarak değerlendirilebilir. Analize ait değerler, Tablo 19’da özetlenmiştir.

Tablo 19: Bağımsız Örneklem T-testi Sonuçları (Ön-Test ve Son-Test)

	Deney Grubu			Kontrol Grubu			t	p	Std. Error
	M	SD	df	M	SD	df			
Öntest	54.45	15.656	38	53.20	17.410	37.579	.239	.813	5.236
Sontest	93.40	3.966	38	89.60	4.684	36.994	2.769*	.009	1.372

* $p < .05$.

4.3 HDS’lerin Performans Puanlarına Etkisi

H_0 : Deney ve kontrol grubu öğrencilerinin Scratch programını kullanma performansları arasında anlamlı bir farklılık yoktur. ($\mu_1 = \mu_2$)

H_a : Deney ve kontrol grubu öğrencilerinin Scratch programını kullanma performansları arasında anlamlı bir farklılık vardır. ($\mu_1 \neq \mu_2$)

Deney ve Kontrol grubunun performans değerleri Tablo 20’de özetlenmiştir. Tablo 20’de ki değerler incelendiğinde deney grubu öğrencilerin ortaya çıkardıkları performans ürünlerinin ortalama puanının ($M=88.17$, $SD=3.44$), kontrol grubundaki öğrencilere göre ($M=69.88$, $SD=6.65$) daha yüksek olduğu gözlemlenmiştir. Bağımsız örneklem t-testi sonucunda, deney grubunun anlamlı şekilde daha yüksek Scratch kullanma performansı ortaya çıkardığı, yani ürünler bazında HDS uygulamalarının etkisinin anlamlı olduğu sonucu gözlemlenmiştir, $t(38) = 10.92$, $p < .001$, $r = .87$. Etki değeri oldukça yüksek olan bu sonuca göre H_0 reddedilmiştir.

Tablo 20: Performans Değerleri

	Grup	N	Mean	Std.Deviation	Std.Error Mean
Performans puanı	Deney	20	88.17	3.44	.77
	Kontrol	20	69.88	6.65	1.49

Bağımsız Örneklem T-Testi ilgili performans puanları analizine ait değerler Tablo 21’de özetlenmiştir. Tablolardaki verilere göre deney ve kontrol grubu öğrencilerinin Scratch programını kullanma performansları arasında anlamlı bir farklılığın olduğu söylenebilir.

Tablo 21: Bağımsız Örneklem T-Testi Sonuçları (Performans Puanları)

	Deney Grubu		Kontrol Grubu				t	p
	M	SD	df	M	SD	Df		
Performans puanı	88.17	3.44	38	69.88	6.65	28.47	10.92	.000

4.4 Katılımcı Görüşleri

Katılımcıların, Scratch’ın ve senaryoların/çalışma kâğıtlarının derse dâhil edilmesiyle ilgili görüşleri incelendiğinde HDS grubunun senaryolar için öğretici, bilgi verici olduğunu, senaryoların tasarımda fikir verme ve kurgulayabilme açısından ne yapacaklarını zihinde canlandırmalarına yardımcı olduğunu ifade etmişlerdir. Bununla ilgili deney grubundan bazı öğrencilerin ifade ettikleri görüşler şu şekildedir.

“Senaryolar oyunu yapabilmek için bize fikir veriyordu. Nasıl canlandırabileceğimizi nasıl yapabileceğimizi gösteriyordu sanki beynimizde canlandırıyordu her şeyi.” (D4)

“Senaryolar olduğunda muhtemelen hemen aklıma bir şeyler geliyordu ve kolaylıkla oyun tasarlayabiliyordum.” (D5)

HDS lerin öğretici olduğu ve gerçek hayata transfer edilebilmesi ile ilgili anlamlı öğrenmelerin gerçekleştiği söylenebilir. Bu durumla ilgili bazı öğrenciler farklı görüşlerde bulunmuşlardır. Örneğin trafik işaretleri oyunu ile gerçek hayatta trafik işaretlerine uyarak kontrollü bir şekilde geçilebilmesi ile ilgili sunulan HDS de öğrencilere bu bilgileri anlamlı şekilde öğrenmesi ve gerçek hayatta kullanabilmesi sağlanmaktadır.

“Mesela trafik işaretleri oyununda hocam. Trafik işaretleri olmasa arabalar hareket ettiğinde ölebiliriz! uyarısı veriyor. Buda bizim için öğretici oluyor aslında. Ve gerçek hayatta trafik esnasında ışıkları kontrol ederek daha dikkatli davranmamızı gösteriyor.” (D5)

“Daha kolay yapabilmemizi ve daha iyi öğrenmemizi sağlıyordu.” (D2)

Bununla birlikte Deney grubu öğrencilerinin dersin başında izledikleri senaryoların hayal güçlerini geliştirdiğine, tasarlarken yaratıcı ve esnek tasarımlar yapabildiklerine ve özgür düşünme becerilerinin desteklediğine dair düşünceleri ortaya çıkmıştır. Buradan sonuçla HDS’lerin öğrencilerin yaratıcı düşünme becerilerinin gelişimine katkı sağladığı söylenebilir. Örneğin deney grubundan bazı öğrenciler görüşlerini şu şekilde ifade etmişlerdir.

“Senaryoyu izlediğimde mutlu hissediyorum çünkü aklıma daha değişik fikirler geliyor. Senaryolar hayal gücümü genişletti.” (D3)

“Senaryolar hayal gücümü geliştirdi. Faydası, daha çok hayal gücümü geliştirdi. Aklımda yani o senaryoyu izlemeden önce aklıma bir şeyler geliyordu tabi ama o senaryoları izledikten sonra daha farklı oyunlar tasarlayabileceğimi gördüm buda beni çok mutlu etti. Hayal dünyam geliyor ve aklıma daha farklı şeyler geliyordu.” (D4)

Bununla birlikte deney grubu öğrencileri, senaryoları beğendiklerini, senaryolardaki karakterlerin ve görsel karikatürize şemaların ilgi çekici olduğunu, her hafta gelecek olan senaryoyu merak ettiklerini, senaryoyu izlerken heyecanlandıklarını belirterek

senaryoların Scratch de oyun tasarlama isteklerini arttırdığı ile ilgili görüşlerde bulunmuşlardır. Örneğin deney grubundan D2, D4 ve D7 görüşlerini şu şekilde dile getirmiştir.

“İnsan senaryoları izleyince böyle onu hemen yapası geliyordu.” (D2)

“Senaryoların görsel olması daha anlaşılır olmasını sağlıyordu. Oradaki karakterleri izleyince bizim de yapma isteğimiz geliyordu. Daha çok onlarında hareket edip konuşuyor olmaları daha açıklayıcı oluyordu.” (D4)

“Senaryolar bence güzel hazırlanmıştı. Oldukça gerçekçiydi. İzlerken çok eğlendim. Senaryolar hoşuma gitti hatta evde de abimle birlikte de tasarlamaya çalıştım.” (D7)

Ayrıca deney grubu öğrencilerinden bazıları senaryolar ile ilgili devam niteliğinde senaryoların hikaye bitişikliğinin motivasyon düzeylerini arttırdığını bir sonraki senaryoyu ilgiyle merak ettiklerini dile getirmişlerdir. Ancak bazı öğrenciler ise senaryoların hikâye bitişikliğinde hep aynı karakterleri görmenin onları motive etmediği, farklı ve yeni karakterler görmek istediklerini ifade etmişlerdir. Örneğin deney grubundan D6 görüşünü şu şekilde belirtmiştir.

“Bence her senaryoda her hafta farklı karakterler olsaydı daha güzel olurdu. Anlatım şekilleri görünüm şekilleri belki farklı olurdu. Yeni yeni karakterler görmek isterdim her hafta, o zaman daha çok ilgi çekici ve ilginç olurdu. Sürekli ereni görmek kötü olmadı ama hafta hafta başka karakterler de olsaydı daha güzel olabilirdi. Ne bileyim yani Eren de sürekli hep farklı oyunları da onun yapabileceğini düşünmüyorum yani. Her oyunu yapamaz diye düşünüyorum. Farklı oyunlarda sürekli eren yapamaz diye düşünüyorum.” (D6)

Deney grubu öğrencileri kendilerini senaryolardaki hikâyeye aidiyet duygusu hissettikleri ve senaryoları içselleştirdiklerini belirterek senaryolara odaklandıklarında güzel tasarımlar yapabildikleri ve senaryoların yaratıcı ve esnek düşüncelerini sağladıkları ortaya çıkmıştır. Örneğin deney grubundan D1 senaryolarla ilgili düşüncelerini şu şekilde ifade etmiştir.

“Hem de sanki o hikâyeye aitmişiz gibi hissediyoruz. Çünkü oradaki karakter yardım edebilir misiniz bana diyor ve bizde bütün sınıf olarak Eren ve Ezgi’ye yardım ediyoruz.” (D1)

HDS’lerin ve çalışma kâğıtlarının öğrencilerin Scratch uygulama performanslarında tasarım ve kodlama aşamasında kolaylık sağladığı gözlemlenmiş ve bu durumla ilgili bazı öğrenciler tarafından görüşler ortaya çıkmıştır.

“Evet, kolaylaştırdı çünkü o senaryoları dinledikten sonra eksiksiz yapmaya çalıştım. Git gide daha iyi yönde ilerlememi sağladı. Tasarımlarda ve kuklalarda bana faydalı oldu.” (D3)

“Evet kolaylaştırdı. Çalışma kâğıtları daha kolay oyunlar tasarlayabilmemizi sağladı. Çünkü zihinden bulamadığımızda çalışma kâğıdındaki yönergeleri takip ediyoruz.” (K1)

“Evet, işimizi kolaylaştırdı. Çünkü kâğıtta kodlarda yanlarında ipuçları vardı hangi blokta olduğunu gösteriyordu çoğu zaman bize. Hem resimler le gösterilmiş olması güzeldi.” (K2)

Bununla birlikte HDS’lerin bazı öğrencileri sınırlandırmadığı, onların kendi düşüncelerine göre özgür tasarımlar yapabildiklerini ancak bazı öğrencileri bazen sınırlandırdığı, senaryolara bağlı yapmaya çalıştıkları gözlemlenmiştir. Öğrencilerle yapılan görüşmelerde de örneğin deney grubundan D3 sınırlandırmadığını, deney grubundan D5 ise bazı senaryolarda sınırlandırdığını ifade etmiştir.

“Sonuçta kendi özgür düşüncelerimize göre tasarım yapıyoruz. İstedığımız sahneyi seçebiliyoruz, istediğimiz şeyi yapabiliyoruz. Herkes farklı renklendirme, farklı şekilde yapabiliyor. Yani senaryo o olabilir ama herkes farklı düşüncelerini ortaya koyarsa herkesin ki farklı olur yani. Bu yüzden sınırlandırmaz diye düşünüyorum.” (D3)

“Mesela eren yerine bir kız değil de yine erene benzer erkek karakteri koymaya çalışıyordum. Çünkü senaryoda anlatılanı canlandırmaya çalışıyordum Ancak renk değişimi ve sahne arka plan değişikliğini kendimizce yapabiliyorduk. Buda hem sınırlandırır hem sınırlı olmadığımızı gösterir.” (D5)

Deney grubu öğrencilerinden D5'in de ifade ettiği gibi HDS'lerde sunulan örnek uygulamaların da kolaylaştırıcı olduğu ve senaryoların sonunda koyulan ödül gibi pekiştiricilerin öğrencilerin motivasyon düzeylerini arttırdığı gözlemlenmiştir. Örneğin deney grubu öğrencilerinden D6 görüşünü şu şekilde dile getirmiştir.

“Senaryolarda örnek uygulamalarda nasıl olduğunu görebiliyorduk, yapınca nasıl şekle geleceğini görebiliyorduk ona göre de hazırlıyorduk.” (D5)

“Hem trafik işaretlerine uyarsa ödül kazanıyor orada. Bizi de mutlu ediyor o yoldan geçirdiğimizde ve lunaparka ulaştırdığımız için. Her oyunun sonunda ödül olması, tebrikler mesajları bile mutlu ediyor insanı. Ödüle ulaşıldığında bitirdik diyoruz ve sonunda başardık duygusuna ulaşıyoruz.” (D6)

Deney grubu öğrencilerinin HDS'lere bakış açıları incelendiğinde genel olarak hoşlarına gittiği gözlemlenmiştir. Ancak bazı öğrenciler eleştirel düşünme becerilerine dayalı olarak farklı yorumlarda bulunmuşlardır. Örneğin deney grubu öğrencilerinden D2 ve D4 şu şekilde görüş belirtmişlerdir. Buradan sonuçla HDS'lerin öğrencilerin eleştirel düşünme becerilerinin gelişimine katkı sağladığı söylenebilir.

“Senaryolar sıkıcı değildi ancak siz anlatsaydınız daha iyi olurdu. Siz daha yakınsınız bize senaryolardan. Siz anlatsaydınız da ben o şekilde kafamda canlandırabilirdim. Ama sizde anlatırken böyle hikâyelerle anlatırsanız iyi olur. Yoksa görevi verip direkt yapın dersiniz o zaman canlandıramam.” (D2)

“Ben olsaydım belki o senaryoları daha farklı yapabilirdim. Senaryolarda kötü bir yer yoktu ama benim hayal gücüm olduğu için mesela farklı bir şey koyabilirim mesela sahne farklı renk değişimi yapabilirdim.” (D4)

Kontrol grubunun çalışma kâğıtları için bilgi verici olduğu ancak daha fazla görsel içerikler olduğunda daha ilgi çekici olabileceği, ne yapacakları ile ilgili görevleri okuduklarında hemen anlamadıkları ancak kod bloklarında yardım edici ve açıklayıcı olduğu ve bu sayede daha erken sürede bitirdikleri görüşleri ortaya çıkmıştır. Örneğin kontrol grubundan K4 ve K5 şu şekilde görüş belirtmiştir.

“Çalışma kâğıtları sayesinde kodların mantığını anlayabildim. Çok öğreticiydi. Scratch'i iyi anlamama yardımcı oldu.” (K4)

“Açıklayıcıydı. Bir çalışma kâğıdı olmasaydı birkaç saatte felan bitirirdim. Yani çalışma kâğıtları olmasaydı birkaç saatte biterdi ama olduğunda yarım saat gibi bitirdim. Yani daha erken sürede bitirmemizi sağladı verdiğiniz çalışma kâğıtları.” (K5)

Deney grubu öğrencilerinin senaryo olmadan tasarım yapabilme ile ilgili zorlanacakları, hemen akıllarına bir şey gelemeyeceği görüşleri de ifade edilmiştir. Bazı öğrenciler için senaryoların bitişikliği anlamlı ve motive edici olduğu ifade edilirken bazı öğrenciler için de her hafta aynı karakterler yerine farklı farklı yeni karakterler görme isteğinin onları motive edeceği ifade edilmiştir. Deney grubu öğrencileri senaryoların oyunu tasarlama ve kodları yazmada işlerini kolaylaştırdığını ve onları sınırlandırmadığı çoğunluk olarak ifade edilmiştir. Aynı şekilde kontrol grubu öğrencilerinde de çalışma kâğıtlarının onların oyun tasarlama ve kodları yazmada işlerini kolaylaştırdığı, çalışma kâğıtlarındaki yönergeleri takip ettikleri ve böylece daha kısa sürede yapabildikleri görüşleri ifade edilmiştir. Kontrol grubundaki bazı öğrenciler tarafından çalışma kâğıtlarında yazılan görevleri takip ettikleri ve bunun onları sınırlandırdığı düşüncesi de ortaya çıkmıştır. Ancak deney grubundaki öğrencilerin senaryoyu sınırlandırmadıkları, aksine kendi özgür düşüncelerine göre tasarım yaptıkları, yaratıcı düşünebildiklerini ifade etmişlerdir. Ayrıca deney grubundaki öğrenciler tarafından senaryoların sonlarında pekiştireç (ödül) sunulmasının onları motive ettiği ve o ödüle ulaştıklarında başarıma duygusunun onları mutlu ettiği görüşleri ifade edilmiştir.

Ayrıca Deney Grubu ve Kontrol grubu öğrencilerinin Scratch programlamaya ilişkin algılarına bakıldığında genel olarak Scratch üzerinde oyun tasarlama isteklerinin fazla olduğu, esnek ve yaratıcı düşünme becerilerini geliştirdiği, tasarım ve kodlama aşamalarında keyif aldıkları, öğrencilerin ürünleri tamamladıklarında başarı duygusunun ön plana çıktığı, katılımlarının yüksek olduğu ve ürün ortaya koyma performanslarının yüksek olduğu, hafta hafta uzmanlık kazandıkları, sürecin sonunda kendi tasarladıkları oyunları bitirdikten sonra oynamalarının ve akranlarıyla paylaşımlarının motivasyon düzeylerini arttırdığı ve hatta bunun için etkili ve hızlı çalışma performansı gösterdikleri gözlemlenmiştir.

BEŞİNCİ BÖLÜM

V. SONUÇ, TARTIŞMA VE ÖNERİLER

Günümüzde öğrencilerin 21.yy becerilerinin kazandırılması üzerinde durulmaktadır. Bireylerin gerçek hayatta karşılaştıkları problem durumlarına çözüm geliştirebilmeleri amacıyla erken yaştan itibaren programlama ve problem çözme becerilerinin kazandırılması öngörülmektedir. Programlamanın çeşitli ülkelerde öğretim müfredatlarında yer alması, her yaştan öğrencinin programlama ile birlikte mantıksal düşünme ve problem çözme becerisi gibi 21. yy becerilerinin kazandırılmasına önem verildiğini göstermektedir. Ancak programlamanın genel olarak karmaşık ve soyut yapısı itibariyle öğrencilerin olumsuz bir tutum geliştirmelerine ve ön yargılı bir şekilde yaklaşmalarına neden olduğu belirtilmektedir (Kinnunen ve Malmi, 2008; Winslow, 1996). Bu nedenle Scratch gibi görsel blok tabanlı programlama araçlarının geliştirilmesinin öğrencilerin daha çok ilgilerini çektiği ve motivasyon düzeylerini arttırmada etkili olduğu savunulmaktadır (Cooper ve diğerleri, 2006). Ayrıca Öğrencilerin Scratch programı ile tasarım aşamasında hayal güçlerini geliştirerek yaratıcı ve esnek düşünme becerilerinin gelişiminde oldukça etkili olduğu ifade edilmektedir (Çatlak, Tekdal ve Baz, 2015; Shin, Park ve Bae, 2013). Yapılan bu çalışmada da benzer sonuçlar ortaya çıkmıştır. Scratch gibi görsel ve blok tabanlı öğrenme aracıyla oyun tasarlama sayesinde öğrenciler, tasarım aşamasında daha aktif ve eğlenceli öğrenme deneyimi yaşamaktadırlar. Böylece öğrencilerin programlamayı öğrenmesi daha keyifli ve anlamlı hale gelmektedir. Scratch görsel programlama aracı, tasarımsal arayüzü, kodların sürükle bırak yöntemiyle kolay kullanılabilir olması, bireylerin ilgisini çekebilecek oyun tasarımına uygun olması ve her yaştan bireye hitap etmesi bakımından programlama öğretiminde etkili bir şekilde kullanılabilir. Yapılan çalışmalar incelendiğinde programlama öğretiminde Scratch'in öğrencilerin başarısını ve motivasyon düzeylerini arttırdığına dair yapılmış pek çok çalışma mevcuttur (Calder, 2010; Çatlak ve diğerleri, 2015; Quahbi ve diğerleri, 2015; Yükseltürk ve Altıok, 2015).

Ancak programlamanın soyut ve karmaşık yapısını somutlaştıran ve görsel olarak eğlenceli hale getiren görsel programlama araçları geleneksel olarak tasarlanmış bir öğretim sürecinde çok etkili olmayabilir. Bu nedenle Scratch programı ile birlikte öğrenci merkezli ve etkileşimli farklı öğretim tasarımları kullanılabilir.

Öğrencilerin Scratch ile etkileşimli oyunlar ya da animasyonlar tasarlamadan önce problem durumunun karmaşık şekilde verilmesi onlar için bir anlam ifade etmeyebilir. Bu nedenle problem durumunun hedefe dayalı senaryolarla verilmesi öğrencilerin motivasyonlarını arttırmada etkilidir (Schank, 1999). Böylece öğrenciler hangi amaçla neyi yapacaklarını daha iyi kurgulayabilir. HDS'ler ile öğrencilerin öğretim sürecine güdülenmiş bir şekilde daha aktif olarak katılmaları sağlanır. (Schank, 1999). Bu nedenle programlama öğretiminde öğrenciler için daha eğlenceli ve daha etkili öğrenme süreci sağlayan Scratch görsel programlama aracının kullanılması, öğrencilerin programlamaya karşı olumlu tutumlarının artması ve ayrıca Scratch ortamında yapacakları ürünlerin daha önceden bir hedef çerçevesinde ve senaryo dâhilinde planlanmasıyla motivasyonlarının artması beklenmektedir. Yaptığımız bu çalışmada HDS'lerin öğrencilerin yaratıcı ve esnek düşünme becerilerini geliştirdiği, motivasyon düzeylerini arttırdığı ve eğlenceli bir şekilde görevi tamamlamaya hevesli oldukları gözlemlenmiştir. Bu amaçla Scratch ile ilgili yapılan çalışmalar incelenmiş ve HDS ile entegre edilmiş programlama çalışmasına rastlanmamıştır.

Bu çalışmada Scratch gibi görsel kodlama imkânı sunan araç HDS ilkeleriyle birleştirilerek öğrencilere sunulmuş, öğrencilerin programlama performanslarında, tasarım ve kodlama becerilerinde HDS'nin etkisi gözlemlenmiştir. Gerçekleştirilen bu çalışmada yapılan analizler sonucunda Deney ve kontrol grubundaki öğrencilerin Scratch programlama başarılarına bakıldığında haftalık gelişimleri baz alındığında her iki grubunda gelişim gösterdiği görülmüştür. Ancak deney grubunun kontrol grubuna kıyasla daha fazla gelişim gösterdiği sonucuna ulaşılmıştır. Buradan hareketle HDS'lerin Scratch programlama performanslarının gelişiminde olumlu katkısı olduğu çıkarılabilir. HDS'lerin bilgi düzeyinde ve performans düzeyinde etkileri ortaya koyulmuştur. Bununla birlikte HDS'nin öğrencilerin programlama performanslarına etkisi gözlemlenmiş ve öğrencilerin farklı bakış açılarıyla ifade ettikleri görüşleri ele alınmıştır.

HDS'nin bilgi düzeyindeki etkisine bakıldığında iki grup arasında ön test değerleri arasında anlamlı bir fark görülmediği ortaya çıkmıştır. Buradan her iki grubunda birbirine yakın seviyede olduğu ve başlangıç performanslarının birbirine yakın derecelerde olduğu söylenebilir. Bunun nedeni her iki grubun da aynı düzeyde ve daha önceden hiç programlamayla karşılaşmamış acemi bireylerden oluşması ile açıklanabilir. Yaklaşık olarak 14 haftalık uygulama sürecinin sonunda yapılan istatistiksel analizler ortaya koyulmuş ve ayrıca araştırmanın gözlemlerine de yer verilmiştir.

HDS'lerin öğrencilerin performanslarını ve başarılarını arttırdığına dair çalışmalar ortaya çıkmıştır. Örneğin Kılıç ve Yıldırım (2012) in yaptığı bir çalışmada öğrencilerin içsel ve dışsal motivasyonlarını arttırdığı sonucu ortaya çıkmıştır. Bu çalışmamızda da öğrencilerle yapılan görüşmelere göre HDS'lerin öğrencilerin ilgiyle takip ettikleri ve motivasyon düzeylerini arttırdığı sonuçları paralellik göstermektedir. Yaptığımız bu çalışma da da HDS'lerin Öğrencilerin Scratch Programlama performanslarını arttırdığına dayalı sonuçlar ortaya çıkmıştır (Kılıç ve Yıldırım, 2012)

Yapılan analizler sonucunda her iki gruptaki öğrenciler arasında bilgi düzeyinde son test puanları karşılaştırıldığında ortalama etki büyüklüğüne sahip bir sonuç bulunmuştur. Buradan sonuçla deney grubu ve kontrol grubu öğrencileri arasında anlamlı fark olduğu ifade edilebilir. Bununla ilgili Schaller, Bunnell ve Nagel'in (2001) ve Foster (1994) tarafından yapılan çalışmalarda da benzer sonuçların bulunduğu gözlemlenmiştir.

HDS'nin performans düzeyindeki etkisine bakıldığında yapılan analizler sonucunda her iki gruptaki öğrencilerin performans puanları karşılaştırıldığında yüksek etki büyüklüğüne sahip bir sonuç bulunmuştur. Buradan hareketle deney grubunun anlamlı şekilde daha yüksek Scratch kullanma performansı gösterdiği yani tasarladıkları ürünler bazında HDS uygulamalarının etkisinin anlamlı olduğu sonucu gözlemlenmiştir. HDS'nin performans düzeyindeki etki değeri, HDS'nin bilgi düzeyindeki etki değerinden daha yüksek olduğu gözlemlenmiştir. Bu durumda HDS nin süreç içerisindeki etkisinin ürünler bazında ortaya çıktığı söylenebilir.

Yapılan bu çalışmada, Scratch uygulama performansları incelendiğinde öğrencilerin, kontrol grubuna kıyasla tasarımlarının daha yaratıcı ve senaryolarla içselleştirilerek

daha özgün yapıldığı ancak kontrol grubundaki öğrencilerin tasarım sürecinde zorlandıkları, ne çizeceklerini hemen düşünemedikleri ve tasarım için daha fazla zaman ayırdıkları gözlemlenmiştir. Buradan sonuçla HDS'lerin öğrenciyi öğrenmeye olan hazır bulunuşluğu arttırdığı söylenebilir. Ausubel in de ifade ettiği gibi birtakım görseller ön hazırlayıcı niteliğindedir. Bu çalışmadaki powtoon aracılığıyla hazırlanan senaryolar ön hazırlayıcı niteliğinde öğrencilere sunulmuştur. Öğrencilere öğretim sürecinde sunulan ön hazırlayıcılar öğrenmenin anlamlı hale gelmesinde oldukça etkili olduğunu ifade etmiştir. Yapılan bu çalışmadaki sonuçlar da bu sonuçlarla paralellik göstermektedir. Öğrenciler karmaşık problem durumunu tasarlamak yerine verilen HDS'ler ile ne yapacaklarını kurgulayabilir ve fikir oluşturabilirler. Öğrenciler için senaryoların bitişikliği bazı öğrencilerin her hafta ilgi düzeylerini arttırdığı ve merakla birlikte isteklerinin fazla olduğuna dair sonuçlar ortaya çıkmıştır.

Csikszentmihalyi'nin (1977) akış teoremi'ne göre bireylerin yetenekleri ve ilgi alanları doğrultusunda yapabilecekleri aktivitelere ilişkin motivasyonlarını toplayarak tamamen odaklandıklarında verimli öğrenmeler gerçekleşir. Yaptığımız çalışma da öğrenciler her hafta gelecek olan senaryoya ilişkin merak ve ilgileri üst düzeyde olmuştur. Csikszentmihalyi (1977), bireylerin ilgi duyarak ve merak duygusunun ön plana çıkarılmasıyla bir faaliyete odaklandıklarında maksimum verimi alabileceğini ve performansının artacağını savunur. Yapılan bu çalışmada da buna benzer durumlar ortaya çıkmıştır.

Ayrıca öğrencilerle yapılan görüşmeler ve süreç içerisindeki gözlemlerden hareketle HDS'lerin öğrencilerin hayal güçlerini geliştirerek öğrencilerin esnek ve yaratıcı düşünme becerilerine katkı sağlayabileceği şeklinde yorumlanabilir.

Araştırmacı tarafından yapılan gözlemler ve Öğrencilerle yapılan görüşmeler sonucunda HDS'lerin öğrenciler için gerçekçi ortamlar sağladığı, senaryolardaki hikâyelere aidiyet hissettikleri ve böylece öğrencilerin öğrenme motivasyonlarını arttırdığı sonuçları ortaya çıkmıştır.

Bazı haftalarda performanslarının çok yakın olduğu, bazılarının çok uzak olduğu gözlemlenmiştir. Bunun nedeni öğelerin etkileşiminden kaynaklanabilir. Bilişsel yük kuramına göre nesnelere etkileşimi kompleks senaryolarda daha fazladır bu da öğrenme performansını etkileyebilir.

Süre kısıtlılığı öğrenciler için bir zorluk olarak görülebilir. Çünkü öğrenci mevcut sürede oyunu hemen yapıp kendi tasarladığı oyunu oynamak için hızlı ve acele tasarlama sürecine gidebilir. Ancak tasarladıkları oyunu oynama isteyeceklerinden oyunu tasarlamada daha kısa süreden dolayı performansları olumsuz etkileyebilir.

HDS'lerin Öğrencilerin Scratch Performanslarına olan etkisi analizler sonucunda incelendiğinde Görsel Tasarım boyutunda iki grubun Hafta6 balon patlatma ve Hafta8 tuğla kırma senaryolarında görsel tasarım puanlarının oldukça birbirine yakın olduğu görülmüştür. Bu senaryolarda istenilen görevler daha çok kod bloklarıyla ilgili olup tasarımsal olarak anlamlı farklılık görülmemiştir. Bununla birlikte Hafta9 trafik kuralları oyunu senaryosunda iki grup arasında oldukça fazla fark ortaya çıkmıştır.

Deney Grubu ve Kontrol grubunun Hafta1 animasyonlu davetiye ve Hafta8 tuğla kırma oyununda Kod bloklarının doğruluğu ortalama puanlarının oldukça birbirine yakın olduğu görülmüştür. Bu senaryolarda istenilen görevler daha çok görsel tasarımla ilgili ve daha basit düzeyde olup kod bloklarının doğruluğu bakımından anlamlı farklılık görülmemiştir. Bununla birlikte hafta2 etkileşimli piyano oyunu ve hafta 9 trafik kuralları oyunu senaryosunda hem tasarım hem de kodlama bakımından kompleks görevler içerdiğinden dolayı deney grubu ve kontrol grubu arasında oldukça fazla fark ortaya çıkmıştır. Buradan hareketle hedefe dayalı senaryoların öğrencilerin problem durumuyla karşılaştıklarında problem çözme becerilerini kullanarak, kod bloklarını tamamlama performanslarını arttırdığı ortaya çıkmıştır. Problem çözme modeline ilişkin ilk aşama olarak hafta hafta verilen HDS lerde belirlenen problem durumu, öğrenciler tarafından anlamlandırılır. İkinci aşamada öğrenciler senaryoları kurgulayarak zihinlerinde algoritmaları tasarladıkları gözlemlenmiştir. Bu aşamada öğrencilerin senaryoları izlerken akılda canlandırarak hayal gücünü geliştirdikleri söylenebilir. Böylece daha esnek ve yaratıcı düşünme becerilerini sergileyerek farklı oyunlar tasarladıkları gözlemlenmiştir. Clark (2012)' a göre hedefe dayalı senaryoların bu yönde katkı sağlaması beklenen bir durumdur. Üçüncü aşamada öğrenciler istenilen görevleri gerçekleştirmek amacıyla tasarım ve kodlamalar yapmışlardır. Bu aşamada öğrenciler sürece aktif olarak katıldıkları ve oyunları birebir tasarlamaya çalıştıklarında daha etkili ve anlamlı öğrenmelerin gerçekleştiği söylenebilir (Resnick, 2009). Son aşamada ise öğrenciler tasarladıkları oyunların çalışıp çalışmadığını denetleyerek aslında problem çözme modelinin aşamalarını gerçekleştirerek problem

çözme sürecini tamamladıkları görülmüştür (Polya, 1945). Tasarladıkları HDS'lerde hata oluşma durumlarında hatalarını gözlemedikleri farklı çözüm yolları, farklı kod dizilimleri test edilerek istenilen görevleri gerçekleştirdikleri gözlemlenmiştir (Knierzinger, Gradinarova, 2016; Resnick, 2014). Buradan hareketle HDS'lerin öğrencilerin motivasyonlarının arttığı ve problem çözme becerilerini geliştirdiği ifade edilebilir. Ayrıca programlama öğretiminde bireyler kodlama yaparken süreç içerisinde karşılaştıkları problem durumlarına yaratıcı çözümler ürettikleri (Calder, 2010; Fessakis, Gouli ve Mavroudi, 2012), tasarladıkları senaryoların sonuçlarını gözlemedikleri (Gülbahar ve Kalelioğlu, 2014), Scratch üzerinde deneyimleyerek özgüven ve motivasyon kazandıkları (Ersoy ve Aydın, 2015), yaptıkları uygulamalarda kendi kendilerine geri dönüt almalarının onları motive ettiği, hataları olduğunda onları düzelterek alternatif farklı kod bloklarını denedikleri (Cooper ve diğerleri, 2006; Shin ve Park, 2014; Shin, Park ve Bae, 2013) başarılı olduklarında motivasyon duygularının arttığı (Yükseltürk ve Altıok, 2015), ve daha iyisini yapmaya güdülendikleri, başardıklarında mutlu oldukları (Genç ve Karakuş, 2011) ve kendi oluşturdukları ürünleri başkalarıyla paylaştıkları (Monroy ve Resnick, 2008) gibi birçok kazanımların ortaya çıktığı söylenebilir.

Deney Grubu ve Kontrol grubunun Hafta1 animasyonlu davetiye ve Hafta8 tuğla kırma oyununda Ürün tamamlama ortalama puanlarının oldukça birbirine yakın olduğu görülmüştür. Bununla birlikte hafta3 labirent oyunu, hafta4 hediye toplama oyunu, ve hafta 9 trafik kuralları oyunu senaryosunda deney grubu ve kontrol grubu arasında oldukça fazla fark ortaya çıkmıştır.

Bununla birlikte deney grubu öğrencilerinin genel olarak verilen senaryoları istenilen sürede görevleri tamamlayıp oyunları çalıştırabildikleri ancak kontrol grubunun tasarımda ve kodlamada deney grubuna oranla daha fazla zorlandıkları, kodlamaya yeterli zaman ayıramadıkları gözlemlenmiştir. Ancak her iki grubun da Scratch ortamındaki uygulama performansları sırasında keyif aldıkları gözlemlenmiştir.

Öneriler

Tasarımcılara Öneriler

- ✓ Programlama öğretim sürecine görsel kodlama araçları ve farklı öğretim tasarımları entegre edilerek öğrencilerin birebir etkileşim kurabileceği şekilde düzenlenebilir.
- ✓ Scratch ortamının dâhil edilmesinin yarattığı pozitif etkilerin duygusal boyuttaki katkıları programlama öğretimi açısından destekleyici şekilde kullanılabilir. Özellikle hayali karakterlerin dahil edilerek maceralar kurgulanması öğrenmenin sürekliliği açısından katkı sağlayabilir.
- ✓ Hikayeleştirilmiş içeriklerde yapılması istenen hedef kazanımlar görsel araçlarla açıklanabilir ve örnek uygulamalar gösterilebilir.
- ✓ Öğrenme sürecinde programlamaya ayrılan süre arttırılabilir.
- ✓ Öğrencilere gerçek hayatta karşılaşılabilecekleri problem durumunu deneyimleyebilecekleri özgür düşünme ortamları ve öğrenme içerikleri sunulabilir.
- ✓ Tasarlanan uygulamalarda öğrencinin yaratıcılığını destekleyecek tasarımlara yer verilmesi motivasyonu arttırabilir.

Eğitim Müfredatına İlişkin Öneriler

- ✓ HDS ile entegre edilmiş programlama ile birlikte disiplinler arası yaklaşımla farklı derslerle de kullanabilecek hedef kazanımlara yer verilebilir. Örneğin Matematik dersiyle, Hafta 7: Matematik 4 İşlem Resim Bulmaca Oyunu senaryosu ya da Bilgisayar dersiyle ilgili Bilgi Yarışması Oyunu senaryosu birleştirilebilir.
- ✓ STÖ, Problem Temelli Öğrenme, Yaparak Yaşayarak Öğrenme gibi farklı öğretim metodlarıyla ilgili Öğretmenlere hizmetiçi eğitimler verilebilir.

- ✓ Programlama öğretiminin daha erken seviyelerdeki müfredatlara dahil edilmesi programlama performansı açısından olumlu katkılar sağlayabilir.

Uygulayıcılara Öneriler

- ✓ Özellikle küçük yaştaki bireyler olmak üzere tüm eğitim kademelerinde Scratch blok temelli programlama sadece algoritma öğretimi amacıyla değil 21. Yy becerilerinin gelişimi amacıyla kullanılabilir.
- ✓ Senaryolar dönem başında ön hazırlayıcı olarak öğrencilere sunulabilir. Örneğin tasarımla ilgili öğrencilerin becerisini geliştirecek hedef kazanımlarla ilgili Animasyonlu davetiye tasarlama veya Mevsimler Oyunu senaryoları kullanılabilir. Scratch becerisi gelişmiş öğrenciler içinde süreç içerisinde Trafik Kuralları Oyunu senaryosu, Matematik 4 İşlem Resim Bulmaca Oyunu senaryoları şeklinde izlenilebilir.
- ✓ HDS ve Programlama Öğretimi bir araya getirilerek ders tasarımında öğrencilere bilişsel hedeflerin yanında psikososyal ve duyuşsal becerilerin de kazandırılması sağlanabilir.

KAYNAKÇA

- Akınoğlu, O. ve Tandoğan, R. Ö. (2007). The effects of problem-based active learning in science education on students' academic achievement, attitude and concept learning. *Eurasia Journal of Mathematics, Science & Technology Education*, 3 (1), 71-81.
- Alice, (2019, 15 Mayıs). Alice. Erişim adresi: <http://www.alice.org/index.php>
- Allsopp, B. B., ve Ejsing-Duun, S. (2016). Programming concepts in playful programming products. in *proceedings of the 10th European Conference on games based learning: ECGBL 2016 proceeding* (1-10). Paisley, Scotland: University of West Scotland. Academic Bookshop Proceedings Series
- Anastasiadou, S.D. ve Karakos, A.S. (2011). The beliefs of electrical and computer engineering students' regarding computer programming. *The International Journal of Technology, Knowledge and Society*, 7 (1), 37-51.
- Baz, F. Ç. (2018). Çocuklar için kodlama yazılımları üzerine karşılaştırmalı bir inceleme. *Current Research in Education*, 4(1), 36-47.
- Batista, A. L. F., Connolly, T., ve Angotti, J. A. P. (2016, Ekim). A Framework for games-based construction learning: A text-based programming languages approach. in *European Conference on Games Based Learning* (815). Academic conferences international limited.
- Bennedsen, J., ve Caspersen, M. E. (2008). Exposing the programming process. In *Reflections on the Teaching of Programming* (pp. 6-16). Springer, Berlin, Heidelberg.
- Bennedsen, J., Caspersen, M. E., ve Kölling, M. (Eds.). (2008). *Reflections on the teaching of programming: methods and implementations* (4821). Springer. Heidelberg New York.
- Blockly, (2019, 15 Mayıs). Blockly. Erişim adresi: <https://developers.google.com/blockly/>
- Calder, N. (2010). Using Scratch: An Integrated problem-solving approach to mathematical thinking. *Australian Primary Mathematics Classroom*, 15 (4), 9-14.
- Clark, R. C., Nguyen, F., ve Sweller, J. (2011). *Efficiency in learning: Evidence-based guidelines to manage cognitive load*. John Wiley & Sons. San Francisco
- Clark, R. C., ve Mayer, R. E. (2012). *Scenario-based e-learning: Evidence-based guidelines for online workforce learning*. John Wiley & Sons. San Francisco
- Code.org, (2019, 15 Mayıs). Code.org. Erişim adresi: <https://code.org>
- Cooper, S., Powers, K., Gross, P., McNally, M., Goldman, K. J., Proulx, V., ve Carlisle, M. (2006, March). Tools for teaching introductory programming: what works?. In *ACM SIGCSE Bulletin* (Vol. 38, No. 1, pp. 560-561). *ACM Bulletin*, 38, 560-561.
- Coravu, L., Marian, M., ve Ganea, E. (2015, September). Scratch and recreational coding for kids. In *2015 14th RoEduNet International Conference-Networking in Education and Research (RoEduNet NER)* (pp. 85-89). IEEE.

- Dale, E. (1946). Audio-visual methods in teaching. New York: The Dryden Press.
Dale, Edgar. Audio-visual methods in teaching, 3rd ed., Holt, Rinehart ve Winston, New York, 1969, 108.
- Demirer, V. (2016). Programming education and new approaches around the world and in Turkey. *Journal of Theory and Practice in Education*, 12 (3), 521-546.
- Eckerdal, A. (2009). *Novice programming students' learning of concepts and practise* (Doctoral dissertation, Acta Universitatis Upsaliensis).
- European Schoolnet (2015) European Schoolnet report. http://fcl.eun.org/documents/10180/14689/Computing+our+future_final.pdf/746e36b1-e1a6-4bf1-8105-ea27c0d2bbe0
- Fessakis, G., Gouli, E., ve Mavroudi, E. (2013). Problem solving by 5–6 years old kindergarten children in a computer programming environment: A case study. *Computers & Education*, 63, 87-97.
- Genç, Z., ve Karakuş, S. (2012). Tasarımla öğrenme: Eğitsel bilgisayar oyunları Tasarımında Scratch Kullanımı. *5th International Computer ve Instructional Technologies Symposium*, 22-24 September 2011 Fırat University, Elazığ-Turkey.
- Gomes, A., ve Mendes, A. J. (2007, September). Learning to program-difficulties and solutions. In *International Conference on Engineering Education–ICEE* (Vol. 2007).
- Günüç, S., Odabaşı, H. F., ve Kuzu, A. (2013). 21.yüzyıl öğrenci özelliklerinin öğretmen adayları tarafından tanımlanması: Bir twitter uygulaması. *Journal of Theory and Practice in Education*, 9 (4), 436-455.
- Hung, W., Jonassen, D. H., & Liu, R. (2008). Problem-based learning. *Handbook of research on educational communications and technology*, 3(1), 485-506.
- Kafai, Y., ve Burke, Q. (2014). *Connected code: why children need to learn programming*. MIT Press, Cambridge, Massachusetts.
- Kafai, Y. B., ve Resnick, M. (Eds.). (2012). *Constructionism in practice: Designing, thinking, and learning in a digital world*. Routledge, New York.
- Kalelioglu, F., & Gülbahar, Y. (2014). The Effects of Teaching Programming via Scratch on Problem Solving Skills: A Discussion from Learners' Perspective. *Informatics in Education*, 13(1), 33-50.
- Knierzinger, A. J., & Gradinarova, B. (2016). Learning to Code, Coding to Learn the World of Algorithms in Higher Education. *The Online Journal of Quality in Higher Education*, 3(3), 25.
- Karabak, D., & Güneş, A. (2013). Ortaokul birinci sınıf öğrencileri için yazılım geliştirme alanında müfredat önerisi. *Eğitim ve Öğretim Araştırmaları Dergisi*, 21(2-3), 163-169.
- Kilroy, D. A. (2004). Problem based learning. *Emergency medicine journal*, 21(4), 411-413.
- Kodu Game Lab, (2019, 15 Mayıs). Kodu Game Lab. Erişim Adresi: <http://www.kodugamelab.com>

- Korhan, E. A., Yılmaz, D. U., Ceylan, B., Akbıyık, A., ve Tokem, Y. (2018). Hemşirelikte Psikomotor Becerilerin Öğretiminde Senaryo Temelli Öğrenme: Bir Deneyim Paylaşımı. *İzmir Katip Çelebi Üniversitesi Sağlık Bilimleri Fakültesi Dergisi*, 3(3), 11-16.
- Kreijns, K., Kirschner, P. A., & Jochems, W. (2003). Identifying the pitfalls for social interaction in computer-supported collaborative learning environments: a review of the research. *Computers in human behavior*, 19(3), 335-353.
- Mariappan, J., Shih, A., & Schrader, P. G. (2004). Use of scenario-based learning approach in teaching statics. In *Proceedings of the 2004 American Society for Engineering Education Annual Conference and Exposition, American Society for Engineering Education*.
- MEB (2018). Bilişim Teknolojileri ve Yazılım Dersi (5, 6, 7 ve 8. Sınıflar) Öğretim Programı. Erişim Adresi
<http://mufredat.meb.gov.tr/Dosyalar/2018124103559587-Bili%C5%9Fim%20Teknolojileri%20ve%20Yaz%C4%B1%C4%B1m%205-6.%20S%C4%B1n%C4%B1flar.pdf>
- MIT App Inventor 2, (2019, 15 Mayıs). App Inventor for Android. Erişim Adresi: <http://www.appinventor.org>, <http://appinventor.mit.edu/explore/>
- Naidu, S., Ip, A., & Linser, R. (2000). Dynamic goal-based role-play simulation on the web: A case study. *Journal of Educational Technology & Society*, 3(3), 190-202.
- Ouahbi, I., Kaddari, F., Darhmaoui, H., Elachqar, A., ve Lahmine, S. (2015). Learning basic programming concepts by creating games with Scratch programming environment. *Procedia-Social and Behavioral Sciences*, 191, 1479-1482.
- Özmen, B., & Altun, A. (2014). Undergraduate students' experiences in programming: difficulties and obstacles. *Turkish Online Journal of Qualitative Inquiry*, 5(3), 1-27.
- Özsevgeç, L. C., ve Kocadağ, Y. (2013). Senaryo tabanlı öğrenme yaklaşımının öğrencilerin kalıtım konusundaki yanlışlarının giderilmesi üzerindeki etkileri. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 28(28-3), 83-96.
- Özsoy, G. (2005). Problem çözme becerisi ile matematik başarısı arasındaki ilişki. *Gazi Eğitim Fakültesi Dergisi*, 25 (3), 179-190.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York Basic Books, Inc..
- Ramadhan, H. A. (2000). Programming by discovery. *Journal of Computer Assisted Learning*, 16(1), 83-93.
- Resnick, M., ve Silverman, B. (2005, June). Some reflections on designing construction kits for kids. In *Proceedings of the 2005 conference on Interaction design and children* (pp. 117-122). ACM.
- Resnick, M. (2014, August). Give P's a chance: Projects, peers, passion, play. In *Constructionism and creativity: Proceedings of the Third International Constructionism Conference*. Austrian Computer Society, Vienna (pp. 13-20).

- Resnick, M. ve Siegel, D. (2015). A different approach to coding. *Bright/Medium*.10(4) 1-4
- Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer Science Education*, 13(2), 137-172.
- Kindley, R. W. (2002). Scenario-based e-learning: A step beyond traditional e-learning. *Learning Circuits*, 3(5).
- Sayın, Z., ve Seferoğlu, S. S. (2016). Yeni bir 21. yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi. *Akademik Bilişim Konferansı*, 3-5.
- Scratch, (2019, 15 Mayıs). Scratch. Erişim Adresi: <https://Scratch.mit.edu>
- ScratchJr, (2019, 15 Mayıs). ScratchJr. Erişim Adresi: <https://www.Scratchjr.org>
- Schank, R.C. ve Cleary, C. (1989) Goal Based Scenarios on the net. Erişim Adresi: <http://www.engines4ed.org/hyperbook/nodes/NODE-227-pg.html>
- Schank, R. C. (1992). Goal-based Scenarios. In R. C. Schank, *Beliefs, reasoning and decision making: psycho-logic in honour of Bob Abelson* (1-32).
- Schank, R. C. (1994). Goal-based scenarios: A radical look at education. *The Journal of the Learning Sciences*, 3(4), 429-453.
- Schank, R. C., Fano, A., Bell, B., & Jona, M. (1994). The design of goal-based scenarios. *The journal of the learning sciences*, 3(4), 305-345.
- Schank, R. C., Berman, T. R., ve Macpherson, K. A. (1999). Learning by doing. *Instructional-design theories and models: A new paradigm of instructional theory*, 2(2), 161-181.
- Schank, R. C. (2000). Designing a goal based scenario. Erişim Adresi: <http://www.engines4ed.org/hyperbook/misc/rcs.html> 06.09.2017 tarihinde erişilmiştir.
- Semerci, Ç. (2010). Başarı odaklı motivasyon (BOM) ölçeğinin geliştirilmesi. *E-Journal of New World Sciences Academy*, 5(4), 2123-2133.
- Sweller, J., & Cooper, G. A. (1985). The use of worked examples as a substitute for problem solving in learning algebra. *Cognition and Instruction*, 2(1), 59-89.
- Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science*, 12(2), 257-285.
- Snyder, C. R., ve Lopez, S. J. (Eds.). (2009). *Oxford handbook of positive psychology*. Oxford library of psychology. New York
- Taylor, M., Harlow, A., ve Forret, M. (2010). Using a computer programming environment and an interactive whiteboard to investigate some mathematical thinking. *Procedia-Social and Behavioral Sciences*, 8, 561-570.
- Utting, I., Cooper, S., Kölling, M., Maloney, J., ve Resnick, M. (2010). Alice, greenfoot, and Scratch--a discussion. *ACM Transactions on Computing Education (TOCE)*, 10(4), 17.

- Van den Bossche, P., Gijsselaers, W. H., Segers, M., ve Kirschner, P. A. (2006). Social and cognitive factors driving teamwork in collaborative learning environments: Team learning beliefs and behaviors. *Small group research*, 37(5), 490-521.
- Walker, J. T., ve Lofton, S. P. (2003). Effect of a problem based learning curriculum on students' perceptions of self directed learning. *Issues in Educational Research*, 13(2), 71-100.
- Wood, D. F. (2003). Problem based learning. *BMJ: British Medical Journal*, , 326(7384), 328-330.
- Yaman, B. (2005). Senaryo Tabanlı Öğrenme yaklaşımına (STÖY) dayalı eğitimde drama yönteminin, ilköğretim beşinci sınıf öğrencilerinin okuduğunu anlama başarılarına etkisi. *Çukurova Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 14(2), 465-482.
- Yaman, H., ve Süğümlü, Ü. (2009). Dilbilgisi öğretiminde Senaryo tabanlı öğrenme yaklaşımının etkililiği, *Kelime türleri örneği. Dil Dergisi*,(144) 56-71.
- Yecan, E., Özçınar, H. ve Tanyeri, T. (2017). Bilişim teknolojileri öğretmenlerinin görsel programlama öğretimi deneyimleri. *Elementary Education Online*, 16 (1), 377-393.
- Yıldırım, A., ve Şimşek, H. (2000). *Sosyal Bilimlerde Nitel Araştırma Yöntemleri*. Ankara: Seçkin Yayınevi.
- Yükseltürk, E. ve Altıok, S. (2015). Bilişim teknolojileri öğretmen adaylarının Bilgisayar programlama öğretimine yönelik görüşleri. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 4 (1), 50-65.
- Yükseltürk, E. ve Altıok, S. (2016). Bilişim teknolojileri öğretmen adaylarının programlama öğretiminde Scratch aracının kullanımına ilişkin algıları. *Mersin Üniversitesi Eğitim Fakültesi Dergisi*, 12(1), 39-52.

EKLER

EK A: Görüşme Soruları

“5. Sınıf Öğrencilerine Programlama Öğretiminde Hedefe Dayalı Senaryo Kullanımının Etkisi ve Öğrenci Görüşleri ”

Çalışmanın Amacı: Bu çalışmanın amacı, senaryo tabanlı öğrenme yaklaşımına bağlı olarak gösterilen hedefe dayalı senaryoların ilköğretim 6. Sınıf öğrencilerinin programlama performanslarının gelişimi üzerindeki etkililiğini belirlemektir.

A.1 Deney Grubu

Temel Bilgiler

- **Demografik Sorular**

1. Adınız
2. Soyadınız
3. Sınıf
4. Şube

- **Öğrenci Algılarına (Genel duygu ve düşünceleri) yönelik sorular**

- **Scratch Programlamaya yönelik algılar**

1. Bilişim Teknolojileri ders sürecinde gördüğünüz Scratch Görsel Programlama sence nasıl bir programdır? Zor mu kolay mı? Neden?
2. Scratch üzerinde oyun tasarlama sürecinde hangi aşamada zorlandın?
A.Kodlama
B.Tasarım
3. Scratch Programının hoşuna giden yanları nelerdir?
A.Görsellik
B.Kodlama
C.Tasarım
4. Scratch Programının hoşuna gitmeyen yanları nelerdir?
A.Görsellik
B.Kodlama
C.Tasarım
5. Scratch programında oyun tasarlarken Hareket, Görünüm, Ses gibi çeşitli blokları kullanırken oluşturduğunuz kodlarda hiç hata oluştu mu?

○ **Hedefe dayalı Senaryoya yönelik algılar**

1. Oyun tasarlama sürecinde dersin başında gösterilen senaryolar hakkında ne düşünüyorsun?
2. Bu senaryolar oyun tasarlamada senin işini kolaylaştırdı mı? (evetse neden? hayırsa neden?)
3. Senaryoların senin tasarımını sınırlandırdığını düşünüyor musun? (evetse neden? /hayırsa neden?)
4. Senaryolarda gördüğün Eren , arkadaşları ve Annesi Ezgi hakkında ne düşünüyorsun?

○ **Öz Değerlendirme**

1. Verilen görevleri istenilen sürede bitirebildin mi ? Bu süreçte sence yaptığın ürünler nasıl oldu? Neden? (İyi/ Kötü)
2. Şuanda sana senaryo vermeden oyun yapman istendiğinde kendini oyun tasarlamada nasıl hissedersin?
3. Scratch programında kendine 100 üzerinden bir puan verecek olsan kaç puan ile değerlendirirsin?

A.2 Kontrol Grubu

Temel Bilgiler

• **Demografik Sorular**

1. Adınız
2. Soyadınız
3. Sınıf
4. Şube

• **Öğrenci Algılarına (Genel duygu ve düşünceleri) yönelik sorular**

○ **Scratch Programlamaya Yönelik Algılar**

1. Bilişim Teknolojileri ders sürecinde gördüğünüz Scratch Görsel Programlama sence nasıl bir programdır? Zor mu kolay mı? Neden?
2. Scratch üzerinde oyun tasarlama sürecinde hangi aşamada zorlandın?
A.Kodlama
B.Tasarım

3. Scratch Programının hoşuna giden yanları nelerdir?
 - A.Görsellik
 - B.Kodlama
 - C.Tasarım
4. Scratch Programının hoşuna gitmeyen yanları nelerdir?
 - A.Görsellik
 - B.Kodlama
 - C.Tasarım
5. Scratch programında oyun tasarlarken Hareket, Görünüm , Ses gibi çeşitli blokları kullanırken oluşturduğun kodlarda hiç hata oluştu mu?

○ **Laboratuvar Çalışma Kağıtlarına yönelik Algılar**

1. Oyun tasarlama sürecinde verilen çalışma kağıtları hakkında ne düşünüyorsun? Neden?
2. Bu çalışma kağıtları oyun tasarlamada senin işini kolaylaştırdı mı? (evetse neden? hayırsa neden?)
3. Bu çalışma kâğıtlarının seni sınırlandırdığını düşünüyor musun? Neden?

○ **Öz Değerlendirme**

1. Verilen görevleri istenilen sürede bitirebildin mi ? Bu süreçte yaptığın ürünler nasıl oldu? Neden? (iyi/ kötü)
2. Şuanda sana çalışma kağıdı verilmeden oyun yapman istendiğinde kendini oyun tasarlamada nasıl hissedersin?
3. Scratch programında kendine 100 üzerinden bir puan verecek olsan kaç puan ile değerlendirirsin

EK B: Öntest – Son test Soruları

2018-2019 EĞİTİM ÖĞRETİM YILI ██████████ ORTAOKULU

BİLİŞİM TEKNOLOJİLERİ VE YAZILIM DERSİ 5.SINIFLAR ÖN TEST-SON TEST SORULARI

Ad Soyad:

Sınıf:

No:

Puan:

1) Aşağıdaki a, b, c, ve d soruları yandaki tabloya göre cevaplandırınız. Size göre doğru olan kutucukların numaralarını soruların altındaki kutucuğa yazınız. (18p.)

 1	 2	 3
 4	 5	 6
 7	 8	 9

a) Yandaki tabloya göre kaç numaralı kutucuktakiler ses kodlarındandır?

--	--

b) Yeşil bayrağa tıklandığında sürekli 10 adım gitmesini sağlayan kodlar hangi kutucuktaki kodlardır?

--	--	--

c) Yukarıdaki tabloya göre kaç numaralı kutucuklar programı başlatan ve bitiren kodlardır?

--	--

d) Yeşil bayrak tıklandığında karakteri gizleyen kodlar kaç numaralı kutucuktaki kodlardır?

--	--

2)Scratch programının kullanım amacını hangi öğrenci doğru şekilde ifade etmiştir? (5p.)

A)



Ses, resim, müzik gibi araçları kullanarak çizgifilm ve animasyon oluşturabiliriz

B)



Sadece yazı yazmakta kullanabiliriz

C)



Scratch bir bilgisayarın çalışması için gereken kablolardur

D)



Scratchı internete girerken kullanırız.

3)Aşağıdaki kodların çalışması sonucu nasıl bir görünüm olur açıklayınız? (12p.)



Cevap:

4)



Komutunun görevi aşağıdakilerden hangisidir? (5p.)

A) Komut bloklarını tekrarlatır.

B) Komutları istenilen süre kadar bekletir.

C) Yeşil bayrağa tıkladığında bu blok altında sıralanmış kod kümesini çalıştırır.

D)Bütün kod kümelerini durdurur.

5)



Komutunun görevi aşağıdakilerden hangisidir? (5p.)

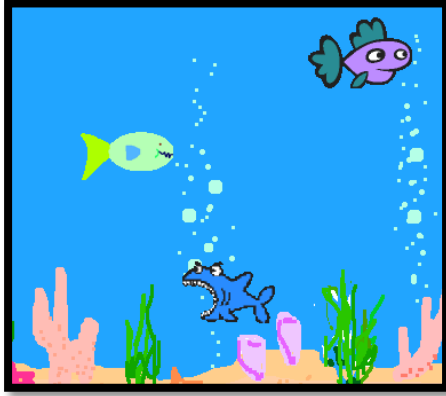
A) Karakterin sahnede resminin çıkmasını sağlar.

B) Bu komut içerisine yerleştirilen komutları çalışmayı durdurana kadar tekrarlar.

C) Karakteri 180 derece döndürür.

D) Karakteri kenarda ise zıplattır.

(6. ve 7. Soruları şekillere göre cevaplayınız)



6) Balıkların sürekli akvaryumda yüzebilmeleri için hangi kod bloğu eklenmelidir? (5p.)

A)



B)



C)



D)



7) Balıkların hızını artırmak için ne yapılmalıdır? (5p.)

- A) 1 adım git bloğundaki 1 yerine daha büyük bir sayı yazılmalıdır.
- B) 1 adım git bloğundaki 1 yerine daha küçük bir sayı yazılmalıdır.
- C) Kenara geldiğinde geri dön bloğuyla 1 adım git bloğunun yeri değiştirilmelidir.
- D) Tıklandığında bloğu kaldırılmalıdır.

8) Eğer kedi ağaca yaklaşıyor ise miyav desin, yaklaşmıyorsa beni yürüt desin.” Bu ifadeyi anlatan komut bloğunda aşağıdaki komutlardan hangisi kesin vardır? (5p.)



9)



Scratch karakteri olan kedinin sol yandaki gibi kostüm1 ve kostüm2 olmak üzere iki kostümü vardır. Kedinin devamlı hareket edip yürüyormuş gibi gözükmesi için yukarıdaki kodların ? işareti olan bölüme hangi kod gelmelidir? (5p.)

A)  B) 

C)  D) 

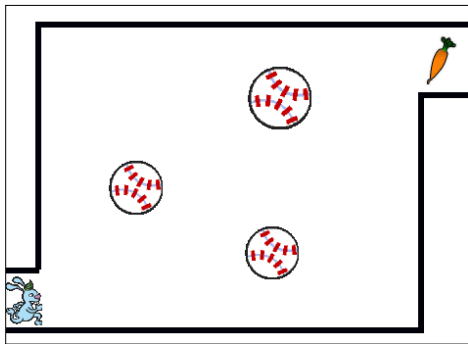
10)Yandaki kod bloğunun açıklaması aşağıdakilerden hangisidir? (5p.)

- A) Klavyede sağ yön tuşuna basıldığında sağa doğru dön ve 10 adım git.
B) Klavyede aşağı yön tuşuna basıldığında aşağı doğru dön ve 1 adım git.
C) Klavyede yukarı yön tuşuna basıldığında sağa dön ve 10 adım git.
D) Klavyede sol yön tuşuna basıldığında sola dön ve 5 adı git.



11) Scratch programında hazırlanmış bir oyunda, sahnedeki karakter bir şarta bağlı olarak hareket ediyorsa aşağıdaki kodların hangisi kesinlikle kullanılmıştır? (5p.)

- A)  B)  C)  D) 



“Yanda Scratch programında yapılan bir oyunun ekran görüntüsü verilmiştir. Oyunda sağ altta bulunan tavşan (-210, -145) noktasındadır. Tavşanın önünde hareket eden toplara ve kenar çizgilerine dokunmadan sağ üstte bulunan havuca, ulaşması gerekmektedir. Eğer tavşan toplara veya kenar çizgilerine dokunursa oyun baştan yani (-210, -145) noktasından başlamaktadır. Eğer tavşan havuca ulaşabilirse tavşanın üstünde tebrikler

balonu çıkmakta ve havuç kaybolmaktadır.” Yukarıda anlatılan oyun kurallarına göre aşağıdaki soruları cevaplayınız.

12) Tavşan toplara veya kenar çizgilerine dokunduğunda hangi kod bloğu çalışmaktadır? (5p.)

- A)  B)  C)  D) 

13) Tavşan havuca ulaştığında havuç kayboluyorsa havuç için hangi kod bloğu çalışıyordu? (5p.)

- A)  B)  C)  D) 

14) Tavşan havuca ulaştığında tavşanın üstünde tebrikler yazması hangi kod bloğu sayesinde olmuştur? (5p.)

- A)  B)  C)  D) 

15) Aşağıdakilerden hangi buton karakterimizi çoğaltmayı sağlar? (5p.)

- A) B) C) D)



16)



Yukarıda verilen boş kutulara 1.Rastgele (sürpriz) karakter ekleme, 2.Karakter çizme, 3.Dosyadan karakter ekleme işlevlerini doğru sırada yerleştiriniz. (5p.)

Başarılar Dilerim☺

Bilişim Teknolojileri Öğretmeni- Emine KANDİN

Ek C: Scratch Performans Değerlendirme Dereceli Puanlama Anahtarı-Rubrik
C.1: Hafta 1: Animasyonlu Davetiye

Tablo 22: Hafta 1: Animasyonlu Davetiye

Hafta1						
Ünite: Scratch ile Programlamaya Giriş						
Etkinlik: Animasyonlu Davetiye Hazırlama						
Kazanımlar: 1.Sahne, karakter ve kostüm kavramlarını ayırt eder.						
2. Yeni bir uygulama oluşturarak uygulama sahne ve karakter değiştirmeyi bilir.						
3. Kodlama alanında komutlar, kostümler ve sesler üzerinde çalışmayı bilir ve düzenler.						
4. Hareket ve görünüm komutlarını kullanarak uygulamalar geliştirir.						
1)Genel Tasarım Öğelerini Düzenleme						
Nitelikler		Geliştirilmeli (1)		Orta Düzeyde (2)		Mükemmel (3)
Renk değiştirme	efektini	Hiçbir üzerinde değiştirilmemiştir veya Sadece 1 Renk değiştirme uygulanmıştır.	karakter renk efektini veya Karaktere efektini	2 den fazla karaktere Renk değiştirme uygulanmıştır.	efektini etkisi	Sahnedeki karakterlere efektini değiştirme doğru bir şekilde uygulanmıştır. Böylece estetik olarak güzel bir görünüm yakalanmıştır.
Farklı ekleyebilme (Balon, Çiçek, Zarf, İnsan)	Karakterler	Kazanıma uygun hiç karakter eklenmemiştir veya Sadece 1 karakter eklenmiş yada tasarıma uygun olmayan karakterler eklenmiştir.	uygun karakter	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	olarak karakter	Tasarıma ait bütün istenilen karakterler doğru bir şekilde eklenmiştir.
Yer, Tarih ve Saat bilgilerini ekleyebilme	Yer, Tarih ve Saat bilgileri	hiç eklenmemiştir veya Bu bilgilerden sadece bir tanesi eklenmiştir.	hiç	Bu bilgilerden sadece 2 tanesi eklenmiştir.		Yer tarih ve saat bilgilerinin hepsi doğru bir şekilde eklenmiştir.
Karakterlere kostümler ekleyebilme	farklı	Karakterlere hiç kostümler eklenmemiştir veya	hiç farklı	Karakterlerden 2 ve 2 den fazla karaktere		Tasarıma ait bütün istenilen karakterlere

	Karakterlerden sadece 1 tanesine farklı kostümler eklenmiştir.	farklı kostümler eklenmiştir.	farklı kostümler eklenmiştir.
Sahne arkaplanını değiştirme	Sahne arkaplanı değiştirilmemiştir veya Sahneye arkaplan eklenmiş ancak hedefe uygun bir arkaplan seçilmemiştir.	Sahneye arkaplan eklenmiş ve hedefe uygunluğu kabul edilebilir niteliktedir.	Sahneye hedefe uygun olarak arkaplan eklenmiş ve aynı zamanda sahnenin arkaplanına renk efekti uyarlanmıştır.
2)Hareket Bloğu komutlarını kullanabilme			
Karakterleri hareket ettirebilme (yürüme, 10 adım git gibi)	Karakterlere hiçbir şekilde hareket etme özelliği kazandırılmamıştır veya Karakterlerden sadece 1 tanesine hareket eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere farklı hareketler eklenmiştir.	Tasarıma ait bütün istenilen karakterlere farklı hareketler eklenmiştir.
Döndürme kodunu kullanabilme	Karakterlere döndürme kodu eklenmemiş yada doğru şekilde eklenememiştir veya Karakterlerden sadece 1 tanesine döndürme kodları eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere farklı döndürme kodları eklenmiştir.	Tasarıma ait bütün istenilen karakterlere farklı döndürme kodları eklenmiştir.
3)Görünüm Bloğu komutlarını kullanabilme			
Kostümler arası geçiş kodunu verebilme	Kostümler arası geçiş kodları doğru şekilde eklenmemiştir veya Karakterlerden sadece 1 tanesine kostümler arası geçiş kodları eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere kostümler arası geçiş kodları eklenmiştir.	Tasarıma ait bütün istenilen karakterlere kostümler arası geçiş kodları eklenmiştir.
4) Ses Bloğu komutlarını kullanabilme			
Tasarıma uygun ses ekleyebilme	Tasarıma Uygun Ses Kodları hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine ses kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere ses kodları birbirine karışmayacak şekilde eklenmiştir.	Sahnenin kodlarına ses kodları eklenmiştir.

5) Kontrol ve Olaylar Bloğu komutlarını kullanabilme

Yeşil Bayrak	Yeşil Bayrak	Karakterlerden 2 ve 2	Tasarıma ait bütün
Tıklandığında kodunu ekleyebilme	tıklandığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıkladığında kontrol kodu eklenmiştir.	den fazla karaktere yeşil bayrak tıkladığında kontrol kodu eklenmiştir.	istenilen karakterlere yeşil bayrak tıkladığında kontrol kodu eklenmiştir.
Tekrar Döngüsü	Sürekli döngü kod	Karakterlerden 2 ve 2	Tasarıma ait bütün
olarak Sürekli	bloğu hiç veya	den fazla karaktere	istenilen karakterlere
kodunu ekleyebilme	kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	sürekli döngü kodu doğru bir şekilde eklenmiştir.	sürekli döngü kodu doğru bir şekilde eklenmiştir.
Görevi	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlamış ve kodları doğru bir şekilde yerleştirmiştir. Yeşil bayrağa tıkladığında görsel olarak animasyonlu davetiye uygulaması başarılı bir şekilde çalıştırılmıştır.	olarak animasyonlu başarılı bir şekilde çalıştırılmıştır.	Puan
Tamamlayabilme	Görevi kazanımlara uygun olarak yerine getirmemiş, tasarımda ve kodlamada birtakım hatalar yapmış ve yeşil bayrak tıkladığında istenilen bir biçimde animasyonlu davetiye çalıştıramamıştır.		Puan

C.2 Hafta 2: Etkileşimli Piyano Oyunu

Tablo 23: Hafta 2: Etkileşimli Piyano Oyunu

Hafta2			
Ünite: Scratch ile Programlamaya Giriş			
Etkinlik: Etkileşimli Piyano Oyunu Uygulaması			
Kazanımlar: 1.Sahne , Karakter ve Kostüm kullanabilme			
2.Hareket bloğunu kullanabilme			
3.Görünüm bloğunu kullanabilme			
4.Ses bloğunu kullanabilme			
5.Kontrol bloğunu kullanabilme			
1)Genel Tasarım Öğelerini Düzenleme			
Nitelikler	Geliştirilmeli (1)	Orta Düzeyde (2)	Mükemmel (3)
Renk efektini değiştirme	Hiçbir karakter üzerinde efektini değiştirilmemiştir veya Sadece 1 Karaktere efektini değiştirme uygulanmıştır.	2 den fazla karaktere Renk değiştirme uygulanmıştır.	Sahnedeki bütün karakterlere Renk efektini değiştirme doğru bir şekilde uygulanmıştır. Böylece estetik olarak güzel bir görünüm yakalanmıştır.
Farklı Karakterler ekleyebilme (Balon, Nota, Şapka, İnsan, Mikrofon)	Kazanıma uygun hiç karakter eklenmemiştir veya Sadece 1 karakter eklenmiş yada tasarıma uygun olmayan karakterler eklenmiştir.	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	Tasarıma ait bütün karakterler doğru bir şekilde eklenmiştir.
Piyano notalarını ekleyebilme ve yerleştirebilme	Piyano notaları hiç eklenmemiştir veya Piyano notalarından	Piyano notalarından sadece 2 3 tanesi eklenmiştir.	Piyano notalarının hepsi doğru bir şekilde eklenmiştir.

		sadece bir tanesi eklenmiştir.		
Karakterlere farklı kostümler ekleyebilme		Karakterlere hiç farklı kostümler eklenmemiştir veya Karakterlerden sadece 1 tanesine farklı kostümler eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere farklı kostümler eklenmiştir.	Tasarıma ait bütün istenilen karakterlere farklı kostümler eklenmiştir.
Sahne arkaplanını değiştirme		Sahne arkaplanı değiştirilmemiştir veya Sahneye arkaplan eklenmiş ancak hedefe uygun bir arkaplan seçilmemiştir.	Sahneye arkaplan eklenmiş ve hedefe uygunluğu kabul edilebilir niteliktedir.	Sahneye hedefe uygun olarak arkaplan eklenmiş ve aynı zamanda sahnenin arkaplanına renk efekti uyarlanmıştır.
2)Hareket Bloğu komutlarını kullanabilme				
Karakterleri hareket ettirebilme (yürüme, 10 adım git gibi)		Karakterlere hiçbir şekilde hareket etme özelliği kazandırılmamıştır veya Karakterlerden sadece 1 tanesine hareket eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere farklı hareketler eklenmiştir.	Tasarıma ait bütün istenilen karakterlere farklı hareketler eklenmiştir.
Döndürme kullanabilme	kodunu	Karakterlere döndürme kodu eklenmemiş yada doğru şekilde eklenmemiştir veya Karakterlerden sadece 1 tanesine döndürme kodları eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere farklı döndürme kodları eklenmiştir.	Tasarıma ait bütün istenilen karakterlere farklı döndürme kodları eklenmiştir.
3)Görünüm Bloğu komutlarını kullanabilme				
Kostümler arası geçiş kodunu verebilme		Kostümler arası geçiş kodları doğru şekilde eklenmemiştir veya	Karakterlerden 2 ve 2 den fazla karaktere kostümler arası geçiş kodları eklenmiştir.	Tasarıma ait bütün istenilen karakterlere kostümler arası

		Karakterlerden sadece 1 tanesine kostümler arası geçiş kodları eklenmiştir.	geçiş kodları eklenmiştir.
4) Ses Bloğu komutlarını kullanabilme			
Tasarıma uygun her notaya vuruş ve nota belirtilerek farklı ses ekleyebilme	Tasarıma Uygun her notaya vuruş ve nota ses kodları hiç kullanılmamıştır veya Karakterlerden sadece 1 , 2 tanesine ses kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere vuruş ve nota ses kodları birbirine karışmayacak şekilde eklenmiştir.	Sahnedeki bütün piyano notalarına tasarımı uygun her notaya vuruş ve nota belirtilerek farklı ses kodları birbirine karışmayacak şekilde eklenmiştir.
5) Kontrol ve Olaylar Bloğu komutlarını kullanabilme			
Yeşil Bayrak Tıklandığında kodunu ekleyebilme	Yeşil Bayrak tıklandığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.
Karakter Tıklandığında kodunu ekleyebilme	Karakter tıklandığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1, 2 tanesine karakter tıklandığında kontrol kodu eklenmiştir.	Karakterlerden (Piyano notalarından) 2 ve 2 den fazla karaktere karakter tıklandığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere (piyano notalarına) karakter tıklandığında kontrol kodu eklenmiştir.
Bekleme kodunu kullanabilme	Bekleme kodu hiç kullanılmamıştır veya Karakterlerden sadece 1, 2 tanesine sürekli döngü kodu	Karakterlerden (piyano notalarından) 2 ve 2 den fazla karaktere bekleme kodu	Tasarıma ait bütün istenilen karakterlere (piyano notalarına) kostümler arası

	dođru bir şekilde eklenmiřtir.	kodu dođru bir şekilde eklenmiřtir.	geçiřin başarılı şekilde çalışması amacıyla bekleme kodu dođru bir şekilde eklenmiřtir.
Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme	Sürekli döngü kod blođu hiç kullanılmamıřtır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu dođru bir şekilde eklenmiřtir.	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu dođru bir şekilde eklenmiřtir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu dođru bir şekilde eklenmiřtir.
Görevi Tamamlayabilme	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlamıř ve kodları dođru bir şekilde yerleřtirmiřtir. Yeřil bayrađa tıklandıđında görsel olarak etkileřimli piyano oyunu uygulaması başarılı bir şekilde çalıştırılmıřtır.	Görevi kazanımlara uygun olarak yerine getirmemiř, tasarımı ve kodlamada birtakım hatalar yapmıř ve yeřil bayrak tıklandıđında istenilen bir biçimde etkileřimli piyano oyunu uygulaması çalıştırılmamıřtır.	Puan
			Puan

C.3 Hafta 3: Labirent Oyunu

Tablo 24: Hafta 3: Labirent Oyunu

Hafta3						
Ünite: Scratch ile Programlamaya Giriş						
Etkinlik: Labirent Oyunu Uygulaması						
Kazanımlar: 1.Sahne, karakter ve kostüm kavramlarını ayırt eder.						
2. Yeni bir uygulama oluşturarak uygulama sahne ve karakter değiştirmeyi bilir.						
3. Kodlama alanında komutlar, kostümler ve sesler üzerinde çalışmayı bilir ve düzenler.						
4. Hareket ve görünüm komutlarını kullanarak uygulamalar geliştirir.						
5. Tekrar yapılarını kavrar ve örnek uygulamalar geliştirir.						
6. Uygulama yönetimini belirli şartlara göre yönlendirmeyi bilir ve düzenler.						
1)Genel Tasarım Öğelerini Düzenleme						
Nitelikler	Geliştirilmeli		Orta Düzeyde		Mükemmel	
	(1)	(2)	(3)	(4)	(5)	(6)
Renk efektini değiştirme	Hiçbir karakter üzerinde efektini değiştirilmemiştir veya Sadece 1 veya 2 Karaktere Renk efektini değiştirme uygulanmıştır.	2 den fazla karaktere Renk değiştirme uygulanmıştır.	2 den fazla karaktere efektini etkisi	Sahnedeki bütün karakterlere Renk efektini değiştirme doğru bir şekilde uygulanmıştır.	Böylece estetik olarak güzel bir görünüm yakalanmıştır.	
Farklı Karakterler ekleyebilme (Farklı renklerde hediyeler, İnsan paketleri, İnsan)	Kazanıma uygun hiç karakter eklenmemiştir veya Sadece 1 karakter eklenmiş yada tasarıma uygun olmayan karakterler eklenmiştir.	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	2	Tasarıma ait bütün istenilen karakterler doğru bir şekilde eklenmiştir.		
Labirent (İççe 3 Halka Duvar)düzenini	Labirent (İççe 3 Halka Duvar) düzeni hiç eklenmemiştir	Labirent (İççe 3 Halka düzeni) ve insan	Labirent (İççe 3 Halka düzeni) ve insan karakteri hepsi			

ekleyebilme yerleştirebilme	ve	veya	Labirent	karakteri	kısmen	doğru bir şekilde	eklenmiştir.
		duvarlarından	sadece bir tanesi	eklenmiştir.			
Karakterlere kostümler ekleyebilme	farklı	Karakterlere	hiç	Karakterlerden 2 ve 2	den fazla karaktere	Tasarıma ait bütün	istenilen karakterlere
		farklı	kostümler	den fazla karaktere	farklı kostümler	farklı kostümler	eklenmiştir.
		eklenmemiştir	veya	farklı kostümler	eklenmiştir.	eklenmiştir.	
		Karakterlerden	sadece 1 tanesine	eklenmiştir.			
		sadece 1 tanesine	farklı kostümler	eklenmiştir.			
Sahne değiştirme	arkaplanını	Sahne	arkaplanı	Sahneye	arkaplan	Sahneye	hedefe
		değiştirilmemiştir	veya	eklenmiş ve	hedefe	uygun	olarak
		veya	Sahneye	uygunluğu	kabul	arkaplan eklenmiş ve	
		arkaplan eklenmiş	ancak hedefe uygun	edilebilir niteliktedir.		aynı zamanda	
		ancak hedefe uygun	bir			sahnenin arkaplanına	
		bir	arkaplan			renk efekti	
		seçilmemiştir.				uyarlanmıştır.	
2)Hareket Bloğu komutlarını kullanabilme							
Karakterleri ettirebilme (yürüme, 10 adım git gibi)	hareket	Karakterlere	hiçbir	Karakterlerden 2 ve 2	den fazla karaktere	Tasarıma ait bütün	istenilen karakterlere
		şekilde hareket etme	özelliği	farklı hareketler	eklenmiştir.	farklı hareketler	eklenmiştir.
		kazandırılmamıştır	veya Karakterlerden	eklenmiştir.		eklenmiştir.	
		veya Karakterlerden	sadece 1 tanesine	eklenmiştir.			
		sadece 1 tanesine	hareket eklenmiştir.				
Belirli yönüne dön (Derece olarak doğru belirleyebilme)		Karakterlere belirli	yönüne dön kodu	Karakterlere belirli	yönüne dön kodu	Karakterlere belirli	yönüne dön kodu
		eklenmemiştir.		kısmen eklenmiştir.		doğru bir şekilde	eklenmiştir.
x , y konumuna git kodunu ekleyebilme		X y konumuna git	kodu eklenmemiştir.	X y konumuna git kodu	kısmen eklenmiştir.	X y konumuna git	kodu doğru bir
						şekilde eklenmiştir.	
Döndürme kullanabilme	kodunu	Karakterlere	döndürme kodu	Karakterlerden 2 ve 2	den fazla karaktere	Tasarıma ait bütün	istenilen karakterlere
		eklenmemiş	yada	farklı döndürme kodları	eklenmiştir.	farklı döndürme	kodları eklenmiştir.
		doğru şekilde	eklenmemiştir veya	eklenmiştir.		eklenmiştir.	
		eklenmemiştir veya	Karakterlerden				
		Karakterlerden					

	sadece 1 tanesine döndürme kodları eklenmiştir.			
3)Görünüm Bloğu komutlarını kullanabilme				
Kostümler arası geçiş kodunu verebilme	Kostümler arası geçiş kodları doğru şekilde eklenmemiştir veya Karakterlerden sadece 1 tanesine kostümler arası geçiş kodları eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere kostümler arası geçiş kodları eklenmiştir.	Tasarıma ait bütün istenilen karakterlere kostümler arası geçiş kodları eklenmiştir.	
Yanlış Balona geldiğinde “Bu Doğru Balon Değil” uyarı mesajını ekleyebilme	Kodlar hiç eklenmemiş yada eksik olarak eklenmiştir.	Balonlardan bazılarında istenilen kod blokları eklenmiş bazılarında eklenmemiştir.	Yanlış Balona geldiğinde “Bu Doğru Balon Değil” uyarı mesajı doğru bir şekilde eklenmiştir.	
Doğru Balona geldiğinde “Tebrikler ” uyarı mesajını ekleyebilme	Kodlar hiç eklenmemiş ya da eksik olarak eklenmiştir.	Balonlardan bazılarında istenilen kod blokları eklenmiş bazılarında eklenmemiştir.	Doğru Balona geldiğinde “Tebrikler ” uyarı mesajını ekleyebilme	
4) Ses Bloğu komutlarını kullanabilme				
Tasarıma uygun karakterin duvarlara çarptığında uyarı sesini ekleyebilme	Tasarıma uygun karakterin duvarlara çarptığında uyarı sesi hiç kullanılmamıştır veya Karakterlerden sadece 1 , 2 tanesine eklenmiştir.	Tasarıma uygun karakterin duvarlara çarptığında uyarı kodları kısmen eklenmiştir.	Tasarıma uygun karakterin duvarlara çarptığında uyarı ses kodları birbirine karışmayacak şekilde eklenmiştir.	
Tasarıma uygun karakterin doğru hediye paketine ulaştığında alkış sesini ekleyebilme	Tasarıma uygun karakterin doğru hediye paketine ulaştığında alkış sesi hiç kullanılmamıştır veya Karakterlerden	Tasarıma uygun karakterin doğru hediye paketine ulaştığında alkış kodları kısmen eklenmiştir.	Tasarıma uygun karakterin doğru hediye paketine ulaştığında alkış kodları birbirine karışmayacak şekilde eklenmiştir.	

				sadece 1 , 2 tanesine eklenmiştir.	
5) Kontrol ve Olaylar Bloğu komutlarını kullanabilme					
Yeşil Tıklandığında ekleyebilme	Bayrak kodunu	Yeşil bloğu kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Bayrak kodu hiç kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.
Yön basıldığında ekleyebilme	tuşlarına kodunu	Yön bloğu kullanılmamıştır yada sadece tek yön için eklenmiştir.	tuşlarına kodu hiç kullanılmamıştır	Yön bazı yerlerde eklenmiş bazı yerlerde eklenmemiştir.	tuşlarına kodu doğru ve eksiksiz şekilde eklenmiştir.
Bekleme kullanabilme	kodunu	Bekleme kodu hiç kullanılmamıştır veya Karakterlerden sadece 1, 2 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	kodu hiç kullanılmamıştır	Karakterlerden (piyano notalarından) 2 ve 2 den fazla karaktere bekleme kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere (piyano notalarına) kostümler arası geçişin başarılı şekilde çalışması amacıyla bekleme kodu doğru bir şekilde eklenmiştir.
Tekrar Sürekli ekleyebilme	Döngüsü olarak kodunu	Sürekli döngü bloğu kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	hiç kullanılmamıştır	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru bir şekilde eklenmiştir.
Koşul Eğer ise ekleyebilme	Döngüsü olarak kodunu	Koşul döngü bloğu kullanılmamıştır	hiç kullanılmamıştır	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu eklenmiştir.

	veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	doğru bir şekilde eklenmiştir.	doğru bir şekilde eklenmiştir.
6) Algılama Bloğu komutlarını kullanabilme			
Karaktere değişiyor mu kodunu kullanabilme	Karaktere değişiyor mu kodu hiç kullanılmamıştır.	Karaktere değişiyor mu kodu bazı yerlerde kullanılmış bazı yerlerde kullanılmamıştır.	Karaktere değişiyor mu kodu doğru bir şekilde eklenmiştir.
Rengine değişiyor mu kodunu kullanabilme	Rengine değişiyor mu kodu hiç kullanılmamıştır.	Rengine değişiyor mu kodu bazı yerlerde kullanılmış bazı yerlerde kullanılmamıştır.	Rengine değişiyor mu kodu doğru bir şekilde eklenmiştir.
Görevi Tamamlayabilme	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlamış ve kodları doğru bir şekilde yerleştirmiştir. Yeşil bayrağa tıkladığında görsel olarak labirent oyunu uygulaması başarılı bir şekilde çalıştırılmıştır.	Görevi kazanımlara uygun olarak yerine getirmemiş, tasarımda ve kodlamada birtakım hatalar yapmış ve yeşil bayrak tıkladığında istenilen bir biçimde labirent oyunu uygulaması çalıştıramamıştır.	Puan

C.4 Hafta 4: Hediye Toplama Oyunu

Tablo 25: Hafta 4: Hediye Toplama Oyunu

Hafta4			
Ünite: Scratch ile Programlamaya Giriş			
Etkinlik: Hediye Toplama Oyunu Uygulaması			
Kazanımlar: 1.Sahne, karakter ve kostüm kavramlarını ayırt eder. 2. Yeni bir uygulama oluşturarak uygulama sahne ve karakter değiştirmeyi bilir. 3. Kodlama alanında komutlar, kostümler ve sesler üzerinde çalışmayı bilir ve düzenler. 4. Hareket ve görünüm komutlarını kullanarak uygulamalar geliştirir. 5. Tekrar yapılarını kavrar ve örnek uygulamalar geliştirir. 6. Uygulama yönetimini belirli şartlara göre yönlendirmeyi bilir ve düzenler.			
1)Genel Tasarım Öğelerini Düzenleme			
Nitelikler	Geliştirilmeli (1)	Orta Düzeyde (2)	Mükemmel (3)
Farklı Karakterler ekleyebilme (Farklı renklerde hediye paketleri, farklı nesnelere, sepet, İnsan)	Kazanıma uygun hiçbir karakter eklenmemiştir veya Sadece 1 karakter eklenmiş ya da tasarıma uygun olmayan karakterler eklenmiştir.	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	Tasarıma ait bütün istenilen karakterler doğru bir şekilde eklenmiştir.
Puan ve Zaman değişkenlerini ekleyebilme	Puan ve zaman değişkenleri hiçbir şekilde eklenmemiştir	Puan ve zaman değişkenleri kısmen eklenmiştir.	Puan ve zaman değişkenleri hepsi doğru bir şekilde eklenmiştir.
Karakterlere farklı kostümler ekleyebilme	Karakterlere hiçbir farklı kostüm eklenmemiştir veya Karakterlerden sadece 1 tanesine farklı kostümler eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere farklı kostümler eklenmiştir.	Tasarıma ait bütün istenilen karakterlere farklı kostümler eklenmiştir.

Sahne arkaplanını deęiřtirme	Sahne arkaplanı deęiřtirilmemiřtir veya Sahneye arkaplan eklenmiř ancak hedefe uygun bir arkaplan seęilmemiřtir.	Sahneye arkaplan eklenmiř ve hedefe uygunluęu kabul edilebilir niteliktedir.	Sahneye hedefe uygun olarak arkaplan eklenmiř ve aynı zamanda sahnenin arkaplanına renk efekti uyarlanmıřtır.
2)Hareket Bloęu komutlarını kullanabilme			
Karakterleri hareket ettirebilme (yürüme, 10 adım git gibi)	Karakterlere hiębir şekilde hareket etme özellięi kazandırılmamıřtır veya Karakterlerden sadece 1 tanesine hareket eklenmiřtir.	Karakterlerden 2 ve 2 den fazla karaktere farklı hareketler eklenmiřtir.	Tasarıma ait bütün istenilen karakterlere farklı hareketler eklenmiřtir.
Belirli yönüne dön (Derece olarak doęru belirleyebilme)	Karakterlere belirli yönüne dön kodu eklenmemiřtir.	Karakterlere belirli yönüne dön kodu kısmen eklenmiřtir.	Karakterlere belirli yönüne dön kodu doęru bir şekilde eklenmiřtir.
x , y konumuna git kodunu ekleyebilme	X y konumuna git kodu eklenmemiřtir.	X y konumuna git kodu kısmen eklenmiřtir.	X y konumuna git kodu doęru bir şekilde eklenmiřtir.
Döndürme kodunu kullanabilme	Karakterlere döndürme kodu eklenmemiř yada doęru şekilde eklenmemiřtir veya Karakterlerden sadece 1 tanesine döndürme kodları eklenmiřtir.	Karakterlerden 2 ve 2 den fazla karaktere farklı döndürme kodları eklenmiřtir.	Tasarıma ait bütün istenilen karakterlere farklı döndürme kodları eklenmiřtir.
3)Görünüm Bloęu komutlarını kullanabilme			
Göster/Gizle kodlarını kullanabilme	Kodlar hię eklenmemiř yada eksik olarak eklenmiřtir.	Hediye paketlerinden bazılarında istenilen kod blokları eklenmiř bazılarında eklenmemiřtir.	Hediye paketleri sepete deędięinde gizlenme, haricinde görünme kodları doęru bir şekilde eklenmiřtir.
Sahne Arkaplanı geçiř kodlarını kullanabilme	Kodlar hię eklenmemiř ya da eksik olarak eklenmiřtir.	Balonlardan bazılarında istenilen kod blokları	Doęru Balona geldięinde

				eklenmiş bazılarında	“Tebrikler ” uyarı mesajını ekleyebilme
4) Ses Bloğu komutlarını kullanabilme				eklenmemiştir.	
Tasarıma uygun ses ekleyebilme	Tasarıma uygun ses kullanılmamıştır	hiç veya	Tasarıma uygun ses kodları	hiç veya kısmen eklenmiştir.	Tasarıma uygun ses kodları birbirine karışmayacak şekilde eklenmiştir.
5) Kontrol ve Olaylar Bloğu komutlarını kullanabilme					
Yeşil Tıklandığında ekleyebilme	Yeşil Bayrak kodunu	Yeşil Bayrak kodunu	Yeşil Bayrak kodunu	Karakterlerden 2 ve 2 den fazla karaktere yeşil bayrak tıkladığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere yeşil bayrak tıkladığında kontrol kodu eklenmiştir.
Boşluk tuşuna basıldığında ekleyebilme	Boşluk tuşuna kodunu	Boşluk tuşuna kodunu	Boşluk tuşuna kodunu	Boşluk tuşuna bazı yerlerde eklenmiş bazı yerlerde eklenmemiştir.	Boşluk tuşuna kodunu doğru ve eksiksiz şekilde eklenmiştir.
Bekleme kodunu kullanabilme	Bekleme kodunu	hiç veya	Karakterlerden (piyano notalarından) 2 ve 2 den fazla karaktere	hiç veya sürekli bekleme kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere (piyano notalarına) kostümler arası geçişin başarılı şekilde çalışması amacıyla bekleme kodu doğru bir şekilde eklenmiştir.
Tekrar Döngüsü olarak Sürekli ekleyebilme	Sürekli döngü kodunu	hiç veya	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	hiç veya sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru bir şekilde eklenmiştir.

Koşul Döngüsü olarak Eğer ise Sürekli kodunu ekleyebilme	Koşul döngü kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru bir şekilde eklenmiştir.
Bütün blokları durdur komutunu ekleyebilme	Bütün blokları durdur kodu eklenmemiştir.	Bütün blokları durdurma kodu doğru yerde ve doğru şekilde eklenmemiştir.	Sahnenin kodlarında zaman bittikten sonra oyun bitti ekranıyla karşılaşma kodlarından sonra bütün blokları durdur kodu doğru şekilde eklenmiş ve başarılı şekilde çalışması sağlanmıştır.
6) Algılama Bloğu komutlarını kullanabilme			
Sağ ok tuşuna basılı ise kodunu kullanabilme	Sağ ok tuşuna basılı ise kodu hiç eklenmemiştir.	Sağ ok tuşuna basılı ise kodu kısmen eklenmiştir.	Sepet karakterinin hediyeleri toplamak için sağ ve sol yönlere gitmesini sağlayacak olan sağ ok tuşuna basılı ise koşulu doğru şekilde eklenmiştir.
Sol ok tuşuna basılı ise kodunu kullanabilme	Sol ok tuşuna basılı ise kodu hiç eklenmemiştir.	Sol ok tuşuna basılı ise kodu kısmen eklenmiştir.	Sepet karakterinin hediyeleri toplamak için sağ ve sol yönlere gitmesini sağlayacak olan sol ok tuşuna basılı ise koşulu doğru şekilde eklenmiştir.
Karaktere değişiyor mu kodunu kullanabilme	Karaktere değişiyor mu kodu hiç kullanılmamıştır.	Karaktere değişiyor mu kodu bazı yerlerde kullanılmış bazı yerlerde kullanılmamıştır.	Karaktere değişiyor mu kodu doğru bir şekilde eklenmiştir.

7) Değişkenler Bloğu komutlarını kullanabilme

Puan ve Zaman değişkenini ekleyebilme	Puan ve zaman değişkenleri eklenmemiştir.	Puan ve zaman değişkenleri hiç eklenmiştir.	Puan ve zaman değişkenleri kısmen eklenmiştir.	Puan ve zaman değişkenleri doğru bir şekilde eklenmiştir.
Puan değişkenini ve Zaman değişkenini düzenleyebilme	Puan ve zaman değişkeni kodları eklenmemiştir.	Puan ve zaman değişkeni kodları hiç eklenmiştir.	Puan ve zaman değişkeni kodları kısmen eklenmiştir.	Puan ve zaman değişkeni kodları doğru bir şekilde eklenmiştir.

8) Operatörler Bloğu komutlarını kullanabilme

Belirli iki sayı arasında rastgele sayı tut tahmin kodlarını kullanabilme	Belirli iki sayı arasında rastgele sayı tut tahmin kodları eklenmemiştir.	Belirli iki sayı arasında rastgele sayı tut tahmin kodları hiç eklenmiştir.	Belirli iki sayı arasında rastgele sayı tut tahmin kodları kısmen eklenmiştir.	Belirli iki sayı arasında rastgele sayı tut tahmin kodları doğru bir şekilde eklenmiştir.
Büyük, Küçük, Eşittir operatörlerini kullanabilme	Büyük, Küçük, Eşittir operatörleri eklenmemiştir.	Büyük, Küçük, Eşittir operatörleri hiç eklenmiştir.	Büyük, Küçük, Eşittir operatörleri kısmen eklenmiştir.	Büyük, Küçük, Eşittir operatörleri doğru ve eksiksiz bir şekilde eklenmiştir.

Görevi Tamamlayabilme	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlamış ve kodları doğru bir şekilde yerleştirmiştir. Yeşil bayrağa tıkladığında görsel olarak hediye toplama oyunu uygulaması başarılı bir şekilde çalıştırılmıştır.	Puan
	Görevi kazanımlara uygun olarak yerine getirmemiş, tasarımda ve kodlamada birtakım hatalar yapmış ve yeşil bayrak tıkladığında istenilen bir biçimde hediye toplama oyunu uygulaması çalıştırılmamıştır.	Puan

C.5 Hafta 5: Bilgi Yarışması Oyunu

Tablo 26: Hafta 5: Bilgi Yarışması Oyunu

Hafta5			
Ünite: Scratch ile Programlamaya Giriş			
Etkinlik: Bilgi Yarışması Oyunu Uygulaması			
Kazanımlar: 1.Sahne , Karakter ve Kostüm kullanabilme			
2.Görünüm bloğunu kullanabilme			
3.Kontrol bloğunu kullanabilme			
4.Algılama bloğunu kullanabilme			
5. Değişkenler Bloğu komutlarını kullanabilme			
6. Operatörler Bloğu komutlarını kullanabilme			
1)Genel Tasarım Öğelerini Düzenleme			
Nitelikler	Geliştirilmeli (1)	Orta Düzeyde (2)	Mükemmel (3)
Farklı Karakterler ekleyebilme (Farklı sorulara görsel farklı seçenekler, Doğru yada Yanlış işareti gösteren True False Simgeleri İnsan)	Kazanıma uygun hiçbir karakter eklenmemiştir veya Sadece 1 karakter eklenmiş ya da tasarıma uygun olmayan karakterler eklenmiştir.	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	Tasarıma ait bütün istenilen karakterler doğru bir şekilde eklenmiştir.
Görsel seçenekleri ekleyebilme ve yerleştirebilme	Görsel seçenekler hiç eklenmemiştir	Sorulara cevap oluşturabilecek sorulara cevap görsel seçenekler kısmen eklenmiştir.	Görsel seçeneklerin hepsi doğru bir şekilde eklenmiştir.
True ve False simgelerinden her soru için eşit sayıda ekleyebilme ve yerleştirebilme	True ve False simgelerinden her soru için eklenmemiştir.	Sorulardan bir kaçına True ve False simgelerinden her soru için eşit sayıda eklenmiştir.	Bütün sorular için True ve False simgelerinden her soru için eşit sayıda eklenmiş ve yerleştirilmiştir.
Sahne arkaplanını değiştirme	Sahne arkaplanı değiştirilmemiştir veya	Sahneye arkaplan eklenmiş ve hedefe	Sahneye hedefe uygun olarak

	Sahneye arkaplan eklenmiş ancak hedefe uygun bir arkaplan seçilmemiştir.	uygunluğu kabul edilebilir niteliktedir.	arkaplan eklenmiş ve aynı zamanda sahnenin arkaplanına renk efekti uyarlanmıştır.
2)Görünüm Bloğu komutlarını kullanabilme			
Göster/Gizle kodlarını kullanabilme	Kodlar hiç eklenmemiş yada eksik olarak eklenmiştir.	Farklı soru seçeneklerinde cevap seçenekleri ve true false işaretleri yapılan duyurulara göre koşullara göre görünmesini ve gizlenmesini sağlayan kodlar sadece bazı seçeneklerde kullanılmıştır.	Farklı soru seçeneklerinde cevap seçenekleri ve true false işaretleri yapılan duyurulara göre koşullara göre görünmesini ve gizlenmesini sağlayan kodlar sadece bütün seçeneklerde kullanılmıştır.
Tebrikler , bildiniz; yada Bilemediniz, Yanlış uyarı mesajlarını doğru biçimde ekleyebilme	Kodlar hiç eklenmemiş yada eksik olarak eklenmiştir.	Tebrikler , bildiniz; yada Bilemediniz, Yanlış uyarı mesajları kısmen eklenmiştir.	4 farklı sorudada belirlenen yanıtlara doğru şekilde ulaşırsa tebrikler bildiniz yada eşit değilse yanlış bilemediniz uyarı mesajları 4 farklı soruda da doğru şekilde eklenmiştir.
3) Kontrol ve Olaylar Bloğu komutlarını kullanabilme			
Yeşil Bayrak Tıklandığında kodunu ekleyebilme	Yeşil Bayrak tıkladığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıkladığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere yeşil bayrak tıkladığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere yeşil bayrak tıkladığında kontrol kodu eklenmiştir.

Karakter Tıklandığında kodunu ekleyebilme	Karakter tıklandığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere karakter tıklandığında kodu kontrol eklenmiştir.	Tasarıma ait bütün istenilen karakterlere karakter tıklandığında kontrol kodu eklenmiştir.
Duyurusunu yap kodunu ekleyebilme	Duyurusunu yap kod bloğu hiç kullanılmamıştır yada sadece birkez eklenmiştir.	Duyurusunu yap kodu bazı yerlerde eklenmiş bazı yerlerde eklenmemiştir.	Duyurusunu yap kodu doğru ve eksiksiz şekilde eklenmiştir.
Duyurusunu yapıldığında kodunu ekleyebilme	Duyurusunu yapıldığında kod bloğu hiç kullanılmamıştır yada sadece birkez eklenmiştir.	Duyurusunu yapıldığında kodu bazı yerlerde eklenmiş bazı yerlerde eklenmemiştir.	Duyurusunu yapıldığında kodu doğru ve eksiksiz şekilde eklenmiştir.
Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme	Sürekli döngü kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru bir şekilde eklenmiştir.
Koşul Döngüsü olarak Eğer ise Sürekli kodunu ekleyebilme	Koşul döngü kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru bir şekilde eklenmiştir.
Bütün blokları durdur komutunu ekleyebilme	Bütün blokları durdur kodu eklenmemiştir.	Bütün blokları durdurma kodu doğru yerde ve doğru şekilde eklenmemiştir.	Sahnenin kodlarında zaman bittikten sonra oyun bitti ekranıyla karşılaşma kodlarından sonra bütün blokları durdur kodu doğru şekilde eklenmiş ve başarılı

					şekilde çalışması sağlanmıştır.
4) Algılama Bloğu komutlarını kullanabilme					
Soruyu sormak amacıyla -diye sor ve bekle kodunu kullanabilme	Soruyu sormak amacıyla -diye sor ve bekle kodu kullanılmamıştır.	Soruyu sormak amacıyla -diye sor ve bekle kodu hiç kullanılmamıştır.	Soruyu sormak amacıyla -diye sor ve bekle kodu bazı yerlerde kullanılmış bazı yerlerde kullanılmamıştır.	Soruyu sormak amacıyla -diye sor ve bekle kodu doğru bir şekilde eklenmiştir.	
Yanıt değişkenini ekleyebilme ve doğru şekilde kullanabilme	Yanıt değişkeni eklenmemiştir.	hiç	Algılama bloğundan yanıtın karşılığını kontrol etmek amacıyla birkaç soruda yanıt değişkeni kısmen eklenmiştir.	Algılama bloğundan yanıtın karşılığını kontrol etmek amacıyla 4 farklı yanıt değişkeni doğru şekilde eklenmiştir.	
5) Operatörler Bloğu komutlarını kullanabilme					
Eşittir operatörünü kullanabilme	Eşittir operatörü eklenmemiştir.	Eşittir operatörü kısmen eklenmiştir.		Eğer koşul döngüsü içinde doğru yanıtı belirlemek amacıyla eşittir operatörleri 4 farklı soru da doğru biçimde kullanılmıştır.	
Görevi Tamamlayabilme	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlamış ve kodları doğru bir şekilde yerleştirmiştir. Yeşil bayrağa tıkladığında görsel olarak bilgi yarışması oyunu uygulaması başarılı bir şekilde çalıştırılmıştır.		Görevi kazanımlara uygun olarak yerine getirmemiş, tasarımda ve kodlamada birtakım hatalar yapmış ve yeşil bayrak tıkladığında istenilen bir biçimde bilgi yarışması oyunu uygulaması çalıştırılmamıştır.	Puan	Puan

C.6 Hafta 6: Balon Patlatma Oyunu

Tablo 27: Hafta 6: Balon Patlatma Oyunu

Hafta6				
Ünite: Scratch ile Programlamaya Giriş				
Etkinlik: Balon Patlatma Oyunu Uygulaması				
Kazanımlar: 1.Sahne , Karakter ve Kostüm kullanabilme				
2.Hareket bloğunu kullanabilme				
3.Görünüm bloğunu kullanabilme				
4.Ses bloğunu kullanabilme				
5.Kontrol bloğunu kullanabilme				
6.Algılama bloğunu kullanabilme				
7. Değişkenler Bloğu komutlarını kullanabilme				
8. Operatörler Bloğu komutlarını kullanabilme				
1)Genel Tasarım Öğelerini Düzenleme				
Nitelikler		Geliştirilmeli (1)	Orta Düzeyde (2)	Mükemmel (3)
Farklı ekleyebilme (Farklı renklerde balonlar, balonları patlatacak nesne)	Karakterler	Kazanıma uygun hiç karakter eklenmemiştir veya Sadece 1 karakter eklenmiş ya da tasarıma uygun olmayan karakterler eklenmiştir.	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	Tasarıma ait bütün istenilen karakterler doğru bir şekilde eklenmiştir.
Oyunun kolaylık derecesini belirleyecek olan kolay orta ve zor gibi buton seçeneklerini ekleyebilme ve yerleştirebilme	Oyunun kolaylık derecesini belirleyecek olan kolay orta ve zor gibi buton seçeneklerini eklenmemiştir	Oyunun kolaylık derecesini belirleyecek olan kolay orta ve zor gibi buton seçenekleri hiç eklenmemiştir	Oyunun kolaylık derecesini belirleyecek olan kolay orta ve zor gibi buton seçenekleri kısmen eklenmiştir.	Oyunun kolaylık derecesini belirleyecek olan kolay orta ve zor gibi buton seçenekleri hepsi doğru bir şekilde eklenmiştir.
Oyuna başla butonunu tasarlayıp ekleyebilme ve yerleştirebilme	Oyuna başlamak amacıyla oyuna başla butonu eklenmemiştir.	Oyuna başlamak amacıyla oyuna başla butonu eklenmiş ancak tasarıma uygun değildir.	Oyuna başlamak amacıyla oyuna başla butonu eklenmiş ancak tasarıma uygun değildir.	Oyuna başlamak amacıyla oyuna başla butonu eklenmiş ve düzgün

					biçimde yerleştirilmiştir.
Sahne arkaplanını değiştirme	Sahne arkaplanı değiştirilmemiştir veya Sahneye arkaplan eklenmiş ancak hedefe uygun bir arkaplan seçilmemiştir.	Sahneye arkaplan eklenmiş ve hedefe uygunluğu kabul edilebilir niteliktedir.	Sahneye arkaplan eklenmiş ve hedefe uygunluğu kabul edilebilir niteliktedir.	Sahneye hedefe uygun olarak arkaplan eklenmiş ve aynı zamanda sahenin arkaplanına renk efekti uyarlanmıştır.	
2)Görünüm Bloğu komutlarını kullanabilme					
Göster/Gizle kodlarını kullanabilme	Kodlar hiç eklenmemiş yada eksik olarak eklenmiştir.	Göster/Gizle kodları bazı balonlar için doğru kullanılmış ancak bazılarında kullanılmamıştır.	Göster/Gizle kodları bazı balonlar için doğru kullanılmış ancak bazılarında kullanılmamıştır.	Göster/Gizle kodları bütün balonlar için doğru biçimde kullanılmıştır.	
Kolay, Orta , Zor geç kodlarını doğru biçimde ekleyebilme	Kodlar hiç eklenmemiş yada eksik olarak eklenmiştir.	Kolay orta ve zor butonları için bazı butonlarda kostümüne geç kodları kullanılmış bazılarında kullanılmamıştır.	Kolay orta ve zor butonları için bazı butonlarda kostümüne geç kodları kullanılmış bazılarında kullanılmamıştır.	Kolay orta ve zor butonları için bütün butonlarda kostümüne geç kodları doğru biçimde kullanılmıştır.	
3) Kontrol ve Olaylar Bloğu komutlarını kullanabilme					
Yeşil Bayrak Tıklandığında kodunu ekleyebilme	Yeşil Bayrak tıklandığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	
Karakter Tıklandığında kodunu ekleyebilme	Karakter tıklandığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere karakter tıklandığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere karakter tıklandığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere karakter tıklandığında kontrol kodu eklenmiştir.	
Duyurusunu yap kodunu ekleyebilme	Duyurusunu yap bloğu	hiç	hiç	Duyurusunu yap kodu bazı yerlerde eklenmiş	Duyurusunu yap kodu doğru ve

		kullanılmamıştır yada sadece birkez eklenmiştir.	bazı yerlerde eklenmemiştir.	eksiksiz şekilde eklenmiştir.
Duyurusunu yapıldığında ekleyebilme	Duyurusunu yapıldığında kodunu kod bloğu hiç kullanılmamıştır yada sadece birkez eklenmiştir.	Duyurusunu yapıldığında kodu bazı yerlerde eklenmiş bazı yerlerde eklenmemiştir.	Duyurusunu yapıldığında kodu doğru ve eksiksiz şekilde eklenmiştir.	
Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme	Sürekli döngü kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru bir şekilde eklenmiştir.	
Koşul Döngüsü olarak Eğer ise Sürekli kodunu ekleyebilme	Koşul döngü kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru bir şekilde eklenmiştir.	
Bütün blokları durdur komutunu ekleyebilme	Bütün blokları durdur kodu eklenmemiştir.	Bütün blokları durdurma kodu doğru yerde ve doğru şekilde eklenmemiştir.	Sahnenin kodlarında zaman bittikten sonra oyun bitti ekranıyla karşılaşma kodlarından sonra bütün blokları durdur kodu doğru şekilde eklenmiş ve başarılı şekilde çalışması sağlanmıştır.	
4) Algılama Bloğu komutlarını kullanabilme				
Balonu patlatacak nesne balonlara değişiyor mu kodunu ekleyebilme	Balonu patlatacak nesne balonlara değişiyor mu kodu kullanılmamıştır.	Balonu patlatacak nesne balonlara değişiyor mu kodu bazı yerlerde kullanılmış bazı yerlerde kullanılmamıştır.	Balonu patlatacak nesne balonlara değişiyor mu kodu doğru bir şekilde eklenmiştir.	

5) Değişkenler Bloğu komutlarını kullanabilme

Skor değişkenini ekleyebilme ve doğru şekilde kullanabilme Balonları patlattıkça skor değişkeninin artması amacıyla skor değişkeni eklenmemiştir. Balonları patlattıkça skor değişkeninin artması amacıyla skor değişkeni eklenmiş ancak doğru şekilde kullanılamamıştır. Balonları patlattıkça skor değişkeninin artması amacıyla skor değişkeni eklenmiş ve doğru biçimde kullanılmıştır.

Skor değişkenini 0 yap, Puan değişkenini değiştir kodlarını doğru biçimde ekleyebilme Skor değişkenini 0 yap, Puan değişkenini değiştir kodları eklenmemiştir. Skor değişkenini 0 yap veya skor değişkenini değiştir kodları bazı balonlarda eklenmiş bazılarında eklenmemiştir. Skor değişkenini 0 yap veya skor değişkenini değiştir kodları tüm balonlarda doğru biçimde eklenmiştir.

Görevi Tamamlayabilme Görevi bütünü kazanımlara uygun olarak istenilen bir şekilde tasarlamış ve kodları doğru bir şekilde yerleştirmiştir. Yeşil bayrağa tıkladığında görsel olarak balon patlatma oyunu uygulaması başarılı bir şekilde çalıştırılmıştır. Puan

Görevi kazanımlara uygun olarak yerine getirmemiş, tasarımda ve kodlamada birtakım hatalar yapmış ve yeşil bayrak tıkladığında istenilen bir biçimde balon patlatma oyunu uygulaması çalıştırılmamıştır. Puan

C.7 Hafta 7: Matematik 4 İşlem Resimli Bulmaca

Tablo 28: Hafta 7: Matematik 4 İşlem Resimli Bulmaca

Hafta7

Ünite: Scratch ile Programlamaya Giriş

Etkinlik: Matematik 4 İşlem Resimli Bulmaca Oyunu Uygulaması

Kazanımlar: 1.Sahne , Karakter ve Kostüm kullanabilme

2.Hareket bloğunu kullanabilme

3.Görünüm bloğunu kullanabilme

4.Ses bloğunu kullanabilme

5.Kontrol ve Olaylar bloğunu kullanabilme

6.Algılama bloğunu kullanabilme

7. Değişkenler Bloğu komutlarını kullanabilme

8. Operatörler Bloğu komutlarını kullanabilme

1)Genel Tasarım Öğelerini Düzenleme

Nitelikler	Geliştirilmeli (1)	Orta Düzeyde (2)	Mükemmel (3)
Farklı Karakterler ekleyebilme (Farklı renklerde soruların cevabı olacak sayılar, İnsan karakteri) ve yerleştirebilme	Kazanıma uygun hiç karakter eklenmemiştir veya Sadece 1 karakter eklenmiş ya da tasarıma uygun olmayan karakterler eklenmiştir.	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	Tasarıma ait bütün istenilen karakterler doğru bir şekilde eklenmiştir.
Görsel oyundaki soruların cevaplarını belirleyecek olan sayı butonu seçeneklerini ve arka plan bulmaca görselini ekleyebilme ve yerleştirebilme	Görsel oyundaki soruların cevaplarını belirleyecek olan sayı butonu seçenekleri ve arka plan bulmaca görseli eklenmemiş yada birkaçı eklenmiştir.	Görsel oyundaki soruların cevaplarını belirleyecek olan sayı butonu seçenekleri ve arka plan bulmaca görseli kısmen eklenmiştir.	Görsel oyundaki soruların cevaplarını belirleyecek olan sayı butonu seçenekleri ve arka plan bulmaca görseli doğru biçimde eklenmiş ve yerleştirilmiştir.
Oyuna başla butonunu tasarlayıp ekleyebilme ve yerleştirebilme	Oyuna başlamak amacıyla oyuna başla butonu eklenmemiştir.	Oyuna başlamak amacıyla oyuna başla butonu eklenmiş ancak tasarıma uygun değildir.	Oyuna başlamak amacıyla oyuna başla butonu eklenmiş ve düzgün biçimde yerleştirilmiştir.
Sahne arkaplanını değiştirme (Level 1, Level2, Resim1, Resim2, Game Over, Düşük bitiş, Orta bitiş, İyi bitiş)	Sahne arkaplanı değiştirilmemiştir veya Sahneye arkaplan eklenmiş ancak hedefe uygun bir arkaplan seçilmemiştir.	Resimli bulmaca oyununun zorluk seviyesine göre ve oyun bittiğinde geri dönüt vermesini sağlayacak arka planlardan bazıları eklenmiştir.	Sahneye hedefe uygun olarak arkaplan eklenmiş ve aynı zamanda sahnenin arkaplanına renk efekti uyarlanmıştır.

2)Görünüm Bloğu komutlarını kullanabilme

Göster/Gizle kodlarını kullanabilme	Kodlar hiç eklenmemiş yada eksik olarak eklenmiştir.	Göster/Gizle kodları bazı balonlar için doğru kullanılmış ancak bazılarında kullanılmamıştır.	Göster/Gizle kodları bütün balonlar için doğru biçimde kullanılmıştır.
Sahneler arası geçiş kodlarını doğru biçimde ekleyebilme	Sahneler arası geçiş kodları hiç eklenmemiş yada eksik olarak eklenmiştir.	Sahneler arası geçiş kodları bazı geçişlerde kullanılmış bazılarında kullanılmamıştır.	Sahneler arası geçiş kodları doğru biçimde kullanılmıştır.
3) Kontrol ve Olaylar Bloğu komutlarını kullanabilme			
Yeşil Bayrak Tıklandığında kodunu ekleyebilme	Yeşil Bayrak tıkladığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıkladığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere yeşil bayrak tıkladığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere yeşil bayrak tıkladığında kontrol kodu eklenmiştir.
Karakter Tıklandığında kodunu ekleyebilme	Karakter tıkladığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıkladığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere karakter tıkladığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere karakter tıkladığında kontrol kodu eklenmiştir.
Duyurusunu yap kodunu ekleyebilme	Duyurusunu yap bloğu hiç kullanılmamıştır yada sadece birkez eklenmiştir.	Duyurusunu yap kodu bazı yerlerde eklenmiş bazı yerlerde eklenmemiştir.	Duyurusunu yap kodu doğru ve eksiksiz şekilde eklenmiştir.
Duyurusunu yapıldığında kodunu ekleyebilme	Duyurusunu yapıldığında kod bloğu hiç kullanılmamıştır yada sadece birkez eklenmiştir.	Duyurusunu yapıldığında kodu bazı yerlerde eklenmiş bazı yerlerde eklenmemiştir.	Duyurusunu yapıldığında kodu doğru ve eksiksiz şekilde eklenmiştir.
Koşul Döngüsü olarak Eğer ise Sürekli kodunu ekleyebilme	Koşul döngü kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru

	kodu doğru bir şekilde eklenmiştir.	bir şekilde eklenmiştir.
4) Algılama Bloğu komutlarını kullanabilme	Soruları oluşturmak amacıyla -diye sor ve bekle kodunu ekleyebilme ve soruları oluşturabilme	Soruları oluşturmak amacıyla diye sor kodu bazı sorularda eklenmiş bazı sorularda eklenmemiştir.
5) Operatörler Bloğu komutlarını kullanabilme	Eşittir ve veya operatörünü kullanabilme	Bazı Soruların cevaplarının kontrolünü sağlamak amacıyla kullanılmış bazı sorularda kullanılmamıştır.
Görevi Tamamlayabilme	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlamış ve kodları doğru bir şekilde yerleştirmiştir. Yeşil bayrağa tıkladığında görsel olarak Matematik 4 İşlem Resim Bulmaca oyunu uygulaması başarılı bir şekilde çalıştırılmıştır.	Puan
	Görevi kazanımlara uygun olarak yerine getirmemiş, tasarımda ve kodlamada birtakım hatalar yapmış ve yeşil bayrak tıkladığında istenilen bir biçimde Matematik 4 İşlem Resim Bulmaca oyunu uygulaması çalıştırılmamıştır.	Puan

C.8 Hafta 8: Tuğla Kırma Oyunu

Tablo 29: Hafta 8: Tuğla Kırma Oyunu

Hafta8
Ünite: Scratch ile Programlamaya Giriş
Etkinlik: Tuğla Kırma Oyunu Uygulaması
Kazanımlar: 1.Sahne, Karakter ve Kostüm kullanabilme
2.Hareket bloğunu kullanabilme
3.Görünüm bloğunu kullanabilme
4.Ses bloğunu kullanabilme

5.Kontrol ve olay bloğunu kullanabilme			
6.Algılama bloğunu kullanabilme			
7. Değişkenler Bloğu komutlarını kullanabilme			
6. Operatörler Bloğu komutlarını kullanabilme			
1)Genel Tasarım Öğelerini Düzenleme			
Nitelikler	Geliştirilmeli (1)	Orta Düzeyde (2)	Mükemmel (3)
Farklı karakterler ekleyebilme (Farklı renklerde nesneleri, tuğlaları top karakteri yada herhangi bir nesne, kırmızı çubuğu, çubuğu)	Karakterler (Farklı görsel tuğla kıracak herhangi bir nesne, kırmızı çubuğu, çubuğu) kazanıma uygun hiç karakter eklenmemiştir veya Sadece 1 karakter eklenmiş ya da tasarıma uygun olmayan karakterler eklenmiştir.	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	Tasarıma ait bütün istenilen karakterler (Tasarladığı oyunda farklı renklerde görsel tuğla nesnelerini, tuğlaları kıracak bir karakter, kırmızı çubuğu, çubuğu hazır yada kendisi tasarlayarak.) doğru bir şekilde eklenmiştir.
Puan değişkenini ekleyebilme ve yerleştirebilme	Puan değişkeni eklenmemiştir.	Puan değişkeni eklenmiş ancak doğru biçimde kullanılmamıştır.	Puan değişkeni doğru biçimde eklenmiş ve yerleştirilmiştir.
Sahne arkaplanını değiştirme	Sahne arkaplanı değiştirilmemiştir veya Sahneye arkaplan eklenmiş ancak hedefe uygun bir arkaplan seçilmemiştir.	Sahneye arkaplan eklenmiş ve hedefe uygunluğu kabul edilebilir niteliktedir.	Sahneye hedefe uygun olarak arkaplan eklenmiş ve aynı zamanda sahenin arkaplanına renk efekti uyarlanmıştır.
2)Hareket Bloğu komutlarını kullanabilme			
Topun hareket etmesi amacıyla yönüne dön ve 10 adım git kodlarını ekleyebilme	Topun yukarı ve yanlara doğru hareket etmesi amacıyla yönüne dön ve 10 adım git kodları eklenmemiştir.	Topun yukarı ve yanlara doğru hareket etmesi amacıyla yönüne dön ve 10 adım git kodları kısmen eklenmiştir.	Topun yukarı ve yanlara doğru hareket etmesi amacıyla yönüne dön ve 10 adım git kodları doğru biçimde eklenmiştir.
3)Görünüm Bloğu komutlarını kullanabilme			

Göster/Gizle kodlarını kullanabilme	Kodlar hiç eklenmemiş yada eksik olarak eklenmiştir.	Göster/Gizle kodları bazı balonlar için doğru kullanılmış ancak bazılarında kullanılmamıştır.	Göster/Gizle kodları bütün balonlar için doğru biçimde kullanılmıştır.
-------------------------------------	--	---	--

4) Kontrol ve Olaylar Bloğu komutlarını kullanabilme

Yeşil Bayrak Tıklandığında kodunu ekleyebilme	Yeşil Bayrak Tıklandığında hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıkladığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere yeşil bayrak tıkladığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere yeşil bayrak tıkladığında kontrol kodu eklenmiştir.
---	---	---	---

Tekrar olarak Döngüsü Sürekli kodunu ekleyebilme	Tekrar döngüsü olarak Sürekli kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru bir şekilde eklenmiştir.
--	--	---	---

Koşul olarak Eğer ise değilse Döngüsü kodunu ekleyebilme	Koşul döngü kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine eğer ise öyle değilse kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere eğer ise öyle değilse döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere eğer ise öyle değilse döngü kodu doğru bir şekilde eklenmiştir.
--	--	---	---

5) Algılama Bloğu komutlarını kullanabilme

Rengine değişiyor mu kodunu ekleyebilme	Rengine değişiyor mu kodu hiç eklenmemiştir.	Rengine değişiyor mu kodu bazı tuğlalar için eklenmiş ve yerleştirilmiştir.	Rengine değişiyor mu kodu bütün tuğlalar için doğru biçimde eklenmiş ve yerleştirilmiştir.
---	--	---	--

Tuğlaların topa değiştiğinde kaybolması amacıyla top karakterine değişiyor mu kodunu ekleyebilme	Tuğlaların topa değiştiğinde kaybolması amacıyla top karakterine değişiyor mu kodu eklenmemiştir.	Tuğlaların topa değiştiğinde kaybolması amacıyla top karakterine değişiyor mu kodu bazı tuğlalarda eklenmiş	Tuğlaların topa değiştiğinde kaybolması amacıyla top karakterine değişiyor mu kodu bütün tuğlalarda eklenmiş
--	---	---	--

			bazılarında eklenmemiştir.	doğru biçimde eklenmiştir.
Topu yakalayacak olan çubuğu fare ile birlikte hareket etmesi amacıyla farenin x koordinatı kodunu ekleyebilme ve yerleştirebilme	Topu yakalayacak olan çubuğu fare ile birlikte hareket etmesi amacıyla farenin x koordinatı kodu hiç eklenmemiştir.	Topu yakalayacak olan çubuğu fare ile birlikte hareket etmesi amacıyla farenin x koordinatı kodu bazı yerlerde eklenmiş bazı yerlerde eklenmemiştir.	Topu yakalayacak olan çubuğu fare ile birlikte hareket etmesi amacıyla farenin x koordinatı kodu doğru biçimde eklenmiş ve yerleştirilmiştir.	
5) Operatörler Bloğu komutlarını kullanabilme				
180-yön operatörünü kullanabilme	Topun yukarı, aşağı hareketi için 180-yön kodu hiç eklenmemiştir.	Topun yukarı, aşağı hareketi için 180-yön kodu eklenmiş ancak doğru yerde kullanılmamıştır.	Topun yukarı, aşağı hareketi için 180-yön kodu eklenmiş ve doğru biçimde kullanılmıştır.	
6) Değişkenler Bloğu komutlarını kullanabilme				
Puan değişkenini ekleyebilme ve doğru şekilde kullanabilme	Puan değişkeni hiç eklenmemiştir.	Puan değişkeni eklenmiş ancak doğru biçimde kullanılmamıştır.	Puan değişkeni doğru biçimde eklenmiş ve yerleştirilmiştir.	
Puan değişkenini arttırma ve azaltma kodlarını doğru biçimde kullanabilme	Puan değişkenini arttırma ve azaltma kodları hiç eklenmemiştir.	Puan değişkenini arttırma ve azaltma kodları eklenmiş ancak doğru biçimde kullanılmamıştır.	Puan değişkenini arttırma ve azaltma kodları doğru biçimde eklenmiş ve yerleştirilmiştir.	
Görevi Tamamlayabilme	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlamış ve kodları doğru bir şekilde yerleştirmiştir. Yeşil bayrağa tıklanıldığında görsel olarak Tuğla Kırmaca oyunu uygulaması başarılı bir şekilde çalıştırılmıştır.	Görevi kazanımlara uygun olarak yerine getirmemiş, tasarımda ve kodlamada birtakım hatalar yapmış ve yeşil bayrak tıklanıldığında istenilen bir biçimde Tuğla Kırmaca oyunu uygulaması çalıştırılmamıştır.	Puan	Puan

C.9 Hafta 9:Trafik Kuralları Oyunu

Tablo 30: Hafta 9:Trafik Kuralları Oyunu

Hafta9			
Ünite: Scratch ile Programlamaya Giriş			
Etkinlik: Trafik Kuralları Oyunu Uygulaması			
Kazanımlar: 1.Sahne , Karakter ve Kostüm kullanabilme			
2.Hareket bloğunu kullanabilme			
3.Görünüm bloğunu kullanabilme			
4.Ses bloğunu kullanabilme			
5.Kontrol ve Olaylar bloğunu kullanabilme			
6.Algılama bloğunu kullanabilme			
7. Değişkenler Bloğu komutlarını kullanabilme			
8. Operatörler Bloğu komutlarını kullanabilme			
1)Genel Tasarım Öğelerini Düzenleme			
Nitelikler	Geliştirilmeli (1)	Orta Düzeyde (2)	Mükemmel (3)
Farklı karakterler ekleyebilme (Farklı renklerde görsel arabalar, İnsan karakterleri (Ezgi ve Eren), Trafik Polisi, Başla butonu, Kurallar butonu, Lunapark, Trafik lambası, Anayol)	Karakterler (Farklı hiç eklenmemiştir veya Sadece 1 karakter eklenmiş ya da tasarımı uygun olmayan karakterler eklenmiştir.	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	Tasarıma ait bütün istenilen karakterler (Tasarladığı oyunda farklı renklerde görsel arabalar, insan karakterleri (Ezgi ve Eren), Trafik polisi, başla butonu, kurallar butonu, lunapark, trafik lambası ve anayol karakterlerinin hepsini) doğru bir şekilde eklenmiştir.
Oyunun kurallarını açıklayacak olan sayfaya yönlendirmek amacıyla kurallar butonunu ve Oyuna başla butonunu tasarlayıp	Oyunun kurallarını açıklayacak olan sayfaya yönlendirmek amacıyla kurallar butonunu ve Oyuna başla butonunu eklenmemiştir.	Oyunun kurallarını açıklayacak olan sayfaya yönlendirmek amacıyla kurallar butonunu ve Oyuna başla butonu kısmen eklenmiştir.	Oyunun kurallarını açıklayacak olan sayfaya yönlendirmek amacıyla kurallar butonunu ve Oyuna başla butonunu düzgün biçimde eklenmiş ve başarılı şekilde çalışması sağlanmıştır.

ekleyebilme ve yerleştirebilme	Sahne arka planı için Oyuna Giriş sahnesi, Kuralların açıklandığı kurallar sahnesi, kazanma ve kaybetme sahnesini tasarlayıp ekleyebilme ve yerleştirebilme	Sahne arka planı için Oyuna Giriş sahnesi, Kuralların açıklandığı kurallar sahnesi, kazanma ve kaybetme sahnelerinden eklenmemiş yada birkaçı eklenmiştir.	Sahne arka planı için Oyuna Giriş sahnesi, Kuralların açıklandığı kurallar sahnesi, kazanma ve kaybetme sahnelerinden bazıları eklenmiş bazıları eklenmemiştir.	Sahne arka planı için Oyuna Giriş sahnesi, Kuralların açıklandığı kurallar sahnesi, kazanma ve kaybetme sahnelerinin hepsi eklenmiştir.
Arabaların trafik hareketinin olduğu anayol oyun sahnesi	Arabaların trafik hareketinin olduğu anayol oyun sahnesi eklenmemiştir.	Arabaların trafik hareketinin olduğu anayol oyun sahnesi kısmen eklenmiştir.	Arabaların trafik hareketinin olduğu anayol oyun sahnesi düzgün biçimde eklenmiştir.	
Sahne arkaplanını değiştirme	Sahne arkaplanı değiştirilmemiştir veya Sahneye arkaplan eklenmiş ancak hedefe uygun bir arkaplan seçilmemiştir.	Sahneye arkaplan eklenmiş ve hedefe uygunluğu kabul edilebilir niteliktedir.	Sahneye hedefe uygun olarak arkaplan eklenmiş ve aynı zamanda sahnenin arkaplanına renk efekti uyarlanmıştır.	
2)Hareket Bloğu komutlarını kullanabilme				
Arabaların trafikte hareketini sağlamak amacıyla 10 adım git kodunu ekleyebilme	Arabaların trafikte hareketini sağlamak amacıyla 10 adım git kodları eklenmemiştir.	Arabaların trafikte hareketini sağlamak amacıyla 10 adım git kodları kısmen eklenmiştir.	Arabaların trafikte hareketini sağlamak amacıyla 10 adım git kodları doğru biçimde eklenmiştir.	
Belirli saniyede x y konumuna git kodunu ekleyebilme ve doğru biçimde kullanabilme	Belirli saniyede x y konumuna git kodu eklenmemiştir.	Belirli saniyede x y konumuna git kodu kısmen eklenmiştir.	Belirli saniyede x y konumuna git kodu eklenmiş ve doğru biçimde kullanılmıştır.	
İnsan karakterinin her yöne gitmesini sağlayan yönüne dön kodunu doğru biçimde kullanabilme	İnsan karakterinin her yöne gitmesini sağlayan yönüne dön kodu eklenmemiştir.	İnsan karakterinin her yöne gitmesini sağlayan yönüne dön kodu kısmen eklenmiştir.	İnsan karakterinin her yöne gitmesini sağlayan yönüne dön kodu doğru biçimde eklenmiş ve hareketi sağlanmıştır.	

3)Görünüm Bloğu komutlarını kullanabilme

Göster/Gizle kodlarını kullanabilme	Kodlar hiç eklenmemiş yada eksik olarak eklenmiştir.	Göster/Gizle kodları bazı yerlerde doğru kullanılmış ancak bazılarında kullanılmamıştır.	Göster/Gizle kodları doğru biçimde eklenmiş ve başarılı bir şekilde çalışması sağlanmıştır.
Kırmızı, Sarı ve Yeşil kostümüne geç kodlarını doğru biçimde ekleyebilme	Kırmızı, Sarı ve Yeşil kostümüne geç kodları hiç eklenmemiştir.	Kırmızı, Sarı ve Yeşil kostümüne geç kodları bazı yerlerde doğru kullanılmış ancak bazılarında kullanılmamıştır.	Kırmızı, Sarı ve Yeşil kostümüne geç kodları doğru biçimde eklenmiş ve başarılı bir şekilde çalışması sağlanmıştır.
İnsan karakterinin yürümesi ve araba çarptığında kırmızı çarpı işareti gelecek şekilde kostümlerine geç kodlarını ekleyebilme	İnsan karakterinin yürümesi ve araba çarptığında kırmızı çarpı işareti gelecek şekilde kostümlerine geç kodları hiç eklenmemiştir.	İnsan karakterinin yürümesi ve araba çarptığında kırmızı çarpı işareti gelecek şekilde kostümlerine geç kodları kısmen eklenmiştir.	İnsan karakterinin yürümesi ve araba çarptığında kırmızı çarpı işareti gelecek şekilde kostümlerine geç kodları doğru biçimde eklenmiştir.
Arkaplanı değiştir kodunu ekleyebilme	Arkaplanı değiştir kodu hiç eklenmemiştir.	Arkaplanı değiştir kodu kısmen eklenmiştir.	Arka planı değiştir kodu doğru biçimde eklenmiş ve başarılı şekilde çalışması sağlanmıştır.

4) Ses Bloğu komutlarını kullanabilme

İnsanların arabaya çarptığında uyarı sesini ekleyebilme	İnsanların arabaya çarptığında uyarı sesi eklenmemiştir.	İnsanların arabaya çarptığında uyarı sesi eklenmiş ama çalışır konumda değildir.	İnsanların arabaya çarptığında uyarı sesi başarılı biçimde eklenmiştir.
---	--	--	---

5) Kontrol ve Olaylar Bloğu komutlarını kullanabilme

Yeşil Bayrak Tıklandığında kodunu ekleyebilme	Yeşil Bayrak Tıklandığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere yeşil bayrak tıklandığında kontrol kodu eklenmiştir.
---	--	--	--

Tekrar olarak kodunu ekleyebilme	Döngüsü Sürekli	Tekrar döngüsü olarak Sürekli kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru bir şekilde eklenmiştir.
Koşul olarak Eğer ise değilse ekleyebilme	Döngüsü Öyle	Koşul döngü kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine eğer ise öyle değilse kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere eğer ise öyle değilse döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere eğer ise öyle değilse döngü kodu doğru bir şekilde eklenmiştir.
Süre olarak bekleme kodunu ekleyebilme		Süre olarak bekleme kodu hiç eklenmemiştir.	Süre olarak bekleme kodu kısmen eklenmiştir.	Trafik işaretinin animasyonu amaçıyla süre olarak bekleme kodu doğru biçimde eklenmiştir.
6) Algılama Bloğu komutlarını kullanabilme				
Araba değişiyor mu ya da lunapark a değişiyor mu kodunu doğru biçimde ekleyebilme ve yerleştirebilme		Araba değişiyor mu ya da lunapark a değişiyor mu kodu hiç eklenmemiştir.	Araba değişiyor mu ya da lunapark a değişiyor mu kodu bazı yerlerde eklenmiş ve yerleştirilmiştir.	Araba değişiyor mu ya da lunapark a değişiyor mu kodunu doğru biçimde eklenmiş ve başarılı şekilde çalışması sağlanmıştır.
Kırmızı, Sarı yada Yeşil yandığında araba hareket ediyor mu kontrolünü sağlamak amacıyla duyurusunu yap ve duyurusu yapıldığında komutlarını ekleyebilme		Kırmızı, Sarı yada Yeşil yandığında araba hareket ediyor mu kontrolünü sağlamak amacıyla duyurusunu yap ve duyurusu yapıldığında komutları eklenmemiştir.	Kırmızı, Sarı yada Yeşil yandığında araba hareket ediyor mu kontrolünü sağlamak amacıyla duyurusunu yap ve duyurusu yapıldığında komutları bazılarında eklenmiş bazılarında eklenmemiştir.	Kırmızı, Sarı yada Yeşil yandığında araba hareket ediyor mu kontrolünü sağlamak amacıyla duyurusunu yap ve duyurusu yapıldığında komutları doğru biçimde eklenmiştir.
7)Değişkenler Bloğu komutlarını kullanabilme				

Süre değişkenini ekleyebilme ve doğru şekilde kullanabilme	Süre değişkeni eklenmemiştir.	hiç	Süre değişkeni eklenmiş ancak doğru biçimde kullanılmamıştır.	Süre değişkeni doğru biçimde eklenmiş ve yerleştirilmiştir.
Süre değişkenini 15 yap kodunu ekleyebilme ve 1 sn geçtikçe süre değişkenini azaltma kodunu ekleyebilme	Süre değişkenini 15 yap kodunu ekleyebilme ve 1 sn geçtikçe süre değişkenini azaltma kodları eklenmemiştir.	hiç	Süre değişkenini 15 yap kodunu ekleyebilme ve 1 sn geçtikçe süre değişkenini azaltma kodları eklenmiş ancak doğru biçimde kullanılmamıştır.	Süre değişkenini 15 yap kodunu ekleyebilme ve 1 sn geçtikçe süre değişkenini azaltma kodları doğru biçimde eklenmiş ve yerleştirilmiştir.

Görevi Tamamlayabilme	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlamış ve kodları doğru bir şekilde yerleştirmiştir. Yeşil bayrağa tıklandığında görsel olarak Trafik İşaretleri oyunu uygulaması başarılı bir şekilde çalıştırılmıştır.	Puan
	Görevi kazanımlara uygun olarak yerine getirmemiş, tasarımda ve kodlamada birtakım hatalar yapmış ve yeşil bayrak tıklandığında istenilen bir biçimde Trafik İşaretleri oyunu uygulaması çalıştırılmamıştır.	Puan

C.10 Hafta 10: Mevsimler Oyunu

Tablo 31: Hafta 10: Mevsimler Oyunu

Hafta10			
Ünite: Scratch ile Programlamaya Giriş			
Etkinlik: Mevsimler Oyunu Uygulaması			
Kazanımlar:	1.Sahne , Karakter ve Kostüm kullanabilme		
	2.Hareket bloğunu kullanabilme		
	3.Görünüm bloğunu kullanabilme		
	4. Ses bloğunu kullanabilme		
	5.Kontrol ve Olaylar bloğunu kullanabilme		
1)Genel Tasarım Öğelerini Düzenleme			
Nitelikler	Geliştirilmeli	Orta Düzeyde	Mükemmel

	(1)	(2)	(3)
Farklı karakterler (Farklı görsel mevsimleri güneş, deniz, ağaçtan dökülen yapraklar, kar yada kardan adam, kelebek, balerin, çiçekler vb. karakterler)	Kazanıma uygun hiçbir karakter eklenmemiştir. Sadece 1 karakter eklenmiş uygun olmayan karakterler eklenmiştir.	Amaca uygun olarak 2 den fazla karakter eklenmiştir.	Tasarıma ait bütün istenilen karakterler (Farklı görsel olarak mevsimleri güneş, deniz, ağaçtan dökülen yapraklar, kar yada kardan adam, kelebek, balerin, çiçekler vb..) doğru bir şekilde eklenmiştir.
İlkbahar, Yaz, Sonbahar ve Kış görsel tasarlayabilme ve ekleyebilme	İlkbahar, Sonbahar ve Kış butonları eklenmemiştir.	Yaz, İlkbahar, Sonbahar ve Kış butonları sadece bir kısmı eklenmiştir.	Yaz, İlkbahar, Sonbahar ve Kış butonları görsel olarak tasarlanmış ve başarılı şekilde eklenmiştir.
Sahne arkaplanını farklı mevsimlere uygun değiştirebilme	Sahne arkaplanı değiştirilmemiştir veya Sahneye arkaplan eklenmiş ancak hedefe uygun bir arkaplan seçilmemiştir.	Sahneye arkaplan eklenmiş ve hedefe uygunluğu kabul edilebilir nitelikte değildir.	Sahneye hedefe uygun olarak arkaplan eklenmiş ve aynı zamanda Sahne arka planı farklı mevsimlere göre başarılı biçimde eklenmiştir.
2)Hareket Bloğu komutlarını kullanabilme	Karakterlerin hareketini sağlayacak 10 adım git , kenara geldiyse sektör yada derece dön kodlarını ekleyebilme	Karakterlere hiçbir şekilde hareket etme özelliği kazandırılmamıştır veya Karakterlerden sadece 1 tanesine hareket eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere farklı hareketler eklenmiştir.
Karakterlere x y konumuna git kodunu ekleyebilme	Karakterlere x y konumuna git kodu eklenmemiş yada	Karakterlerden 2 ve 2 den fazla karaktere x y konumuna git kodu eklenmiştir.	Karakterlere x y konumuna git kodu düzgün biçimde eklenmiş ve başarılı

	dođru	şekilde	şekilde	çalışması
	eklenememiştir		sađlanmıřtır.	
Yaprakların hareketi yada kar tanelerinin düşmesi amacıyla 180 yönüne dođru yönel kodunu ekleyebilme	180 yönüne dođru yönel kodunu eklenmemiştir.	dođru kodu	180 yönüne dođru yönel kodu kullanılmıř ancak dođru şekilde çalışmamıřtır.	Yaprakların hareketi yada kar tanelerinin düşmesi amacıyla 180 yönüne dođru yönel kodu dođru biçimde karakterler üzerinde başarılı biçimde eklenmiştir.
3)Görünüm Blođu komutlarını kullanabilme				
Kelebeklerin yada kuř karakterlerinin uçması amacıyla kostüm kılıđına geç ekleyebilme ve yerleřtirebilme	Kostümler arası geçiř kodları dođru şekilde eklenmemiştir	veya	Karakterlerden 2 ve 2 den fazla karaktere kostümler arası geçiř kodları eklenmiř ancak dođru şekilde çalıştırılmamıřtır.	Tasarıma ait bütün istenilen karakterlere kostümler arası geçiř kodları eklenmiştir.
İlkbahar, Yaz, Sonbahar ve Kıř dekoruna geç kodunu ekleyebilme	İlkbahar, Sonbahar ve Kış dekoruna geç eklenmemiştir.	Yaz, Kış kodları	İlkbahar, Sonbahar ve Kış dekoruna geç kodlarının sadece bir kısmı eklenmiştir.	İlkbahar, Yaz, Sonbahar ve Kış dekoruna geç kodları dođru biçimde eklenmiştir.
4) Ses Blođu komutlarını kullanabilme				
Tasarıma uygun Gökgürültüsü yada yağmur efektini verecek ses kodlarını ekleyebilme	Tasarıma Uygun Ses Kodları kullanılmamıřtır	veya	Karakterlerden 2 ve 2 den fazla karaktere ses kodları birbirine karıřmayacak şekilde eklenmiştir.	Tasarıma uygun Gökgürültüsü yada yağmur efektini verecek ses kodları eklenmiştir.
5) Kontrol ve Olaylar Blođu komutlarını kullanabilme				
Yeřil Bayrak Tıklandığında kodunu ekleyebilme	Yeřil Bayrak tıklandığında blođu kullanılmamıřtır	veya	Karakterlerden 2 ve 2 den fazla karaktere yeřil bayrak tıklandığında kontrol kodu eklenmiştir.	Tasarıma ait bütün istenilen karakterlere yeřil bayrak tıklandığında kontrol kodu eklenmiştir.
	Karakterlerden sadece 1 tanesine yeřil bayrak tıklandığında kontrol kodu eklenmiştir.			

Karakter Tıklandığında kodunu ekleyebilme	Karakter tıklandığında kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine karakter tıklandığında kontrol kodu eklenmiştir.	Karakter kodu tıklandığında kodu mevsim butonları için bazılarında eklenmiş bazılarında eklenmemiştir.	Tasarıma ait bütün istenilen karakterlere karakter tıklandığında kontrol kodu eklenmiştir.
Duyurusunu yap kodunu ekleyebilme	Duyurusunu yap kodu eklenmemiştir.	Duyurusunu yap kodu bazı yerlerde eklenmiş bazı yerlerde eklenmemiştir.	Duyurusunu yap kodu doğru biçimde eklenmiştir.
Duyurusu yapıldığında kodunu ekleyebilme	Duyurusu yapıldığında kodu eklenmemiştir.	Duyurusu yapıldığında kodu bazı yerlerde eklenmiş bazı yerlerde eklenmemiştir.	Duyurusu yapıldığında kodu doğru biçimde eklenmiştir.
Tekrar Sürekli Döngüsü olarak kodunu ekleyebilme	Sürekli döngü kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine sürekli döngü kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere sürekli döngü kodu doğru bir şekilde eklenmiştir.	Tasarıma ait bütün istenilen karakterlere sürekli döngü kodu doğru bir şekilde eklenmiştir.
Koşul Eğer ise öyle değilse kodunu ekleyebilme	Koşul döngü kod bloğu hiç kullanılmamıştır veya Karakterlerden sadece 1 tanesine koşul döngü kodu doğru bir şekilde eklenmiştir.	Karakterlerden 2 ve 2 den fazla karaktere koşul döngü kodu eğer ise öyle değilse doğru bir şekilde eklenmiştir.	Koşul döngüsü olarak eğer ise öyle değilse kodu başarılı bir şekilde eklenmiştir.
Dekor ilkbahar, yaz , sonbahar veya kış olarak değiştiğinde kodunu ekleyebilme	Dekor ilkbahar, yaz , sonbahar veya kış olarak değiştiğinde kodu eklenmemiştir.	Dekor ilkbahar, yaz , sonbahar veya kış olarak değiştiğinde kodu bazıları eklenmiş bazıları eklenmemiştir.	Dekor ilkbahar, yaz , sonbahar veya kış olarak değiştiğinde kodu başarılı bir şekilde eklenmiştir.
	Görevi bütün kazanımlara uygun olarak istenilen bir şekilde tasarlanmış ve kodları doğru bir		Puan

Görevi
Tamamlayabilme

şekilde yerleştirmiştir. Yeşil bayrağa
tıklandığında görsel olarak Mevsimler Oyunu
başarılı bir şekilde çalıştırılmıştır.

Görevi kazanımlara uygun olarak yerine Puan
getirmemiş, tasarımda ve kodlamada birtakım
hatalar yapmış ve yeşil bayrak tıklandığında
istenilen bir biçimde Mevsimler Oyunu
çalıştıramamıştır.

Ek D: Hedefe Dayalı Senaryolar

Hafta 1- Animasyonlu Davetiye Tasarlama



Şekil 30: Hafta 1- Animasyonlu Davetiye Tasarlama

Tablo 32: Hafta1: Animasyonlu Davetiye Tasarlama

Hafta 1

Ünite: Scratch ile Programlamaya Giriş

Etkinlik: Animasyonlu Davetiye Tasarlama

1)Öğrenme	Kazanımlar:
Hedefleri:	1.Sahne , Karakter ve Kostüm kullanabilme 2.Hareket bloğunu kullanabilme 3.Görünüm bloğunu kullanabilme 4.Ses bloğunu kullanabilme 5.Kontrol bloğunu kullanabilme
2)Misyön:	1)Genel Tasarım Öğelerini Düzenleme Renk efektini deęiřtirme Farklı Karakterler ekleyebilme (Balon, Çiçek, Zarf, İnsan) Yer, Tarih ve Saat bilgilerini ekleyebilme Karakterlere farklı kostümler ekleyebilme Sahne arkaplanını deęiřtirme 2)Hareket Bloęu komutlarını kullanabilme Karakterleri hareket ettirebilme (yürüme, 10 adım git gibi) Döndürme kodunu kullanabilme 3)Görünüm Bloęu komutlarını kullanabilme Kostümler arası geçiř kodunu verebilme 4) Ses Bloęu komutlarını kullanabilme Tasarıma uygun ses ekleyebilme 5) Kontrol Bloęu komutlarını kullanabilme Yeřil Bayrak Tıklandığında kodunu ekleyebilme Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme Görevi Tamamlayabilme 33

3)Ana Hikâye:	Eren ana karakterdir. Ezgi, Eren'in annesidir. Eren ve Ezgi oturma odasında konuşmaktadırlar. Erenin annesi Ezgi, oğluna sürpriz doğum günü partisi hazırlamayı düşünmektedir. Eren'e sürpriz doğum günü partisi yapacağı için Erenin haberi yokken herkesi davet etmek istemektedir. Ancak herkesi nasıl çağırabileceğini düşünmektedir. Erenin annesi doğum günü sürpriz partisine çağıracağı kişilere davetiye kartı vermeyi düşünmüştür. Ancak partide herkesi çağırarak davetiyeleri nasıl hazırlanacağını düşünmektedir. Elinde o kadar kağıt ve kalem olmadığı aklına gelir. Ayrıca fazla kağıt ve mürekkep harcamanın doğaya zarar vereceğini ve israf olacağını düşünür. Bir müddet ne yapacağını düşündükten sonra evin oturma odasında bilgisayarı görünce aklına bir fikir gelir. Tüm davetiyeleri bilgisayardan hazırlayabileceği aklına gelir. Böylece kağıda ve kaleme ihtiyaç kalmayacaktır.
4) Görev (Role):	Animasyonlu davetiyede; Müzik , Hareket eden çiçekler Uçan balonlar, Dönen şapkalar Animasyonlu Davetiye mektubu zarfı Dans eden balerin Yer, Tarih, Zaman bilgileri (Yer:Rüya balon Salonu, Tarih:17 Mart 2016 Saat :17:00)
5)Senaryo Hareketleri:	Öncelikle Scratch senaryosunda bulunan Eren'in Annesinin hazırlayacağı sürpriz doğum günü partisi davetiyesi için sahne arka plan tasarımını oluştur. Farklı renkler ve şekiller kullanabilirsin. Hareket eden nesne olarak farklı Karakterler olarak Balon, Çiçek, Zarf, İnsan karakterlerinden ekle. Bu karakterlere renk efektini değiştirme ve döndürme efekt kodlarını ekle. Bunun için öncelikle yeşil bayrak tıklandığında kodunu ekle, sonra sürekli döngü kodunu ekledikten sonra renk efektini değiştir kodunu ekle. Aynı işlemi bütün karakterler için tekrarla. Daha sonra dans eden balerin için farklı kostümler oluştur. Sürekli döngü kodunu ekle. Kostümler arası geçiş kodlarını ekle. Zarf için açılıp kapanma görüntüsünü oluşturmak amacıyla açık ve kapalı olmak üzere farklı kostümler oluştur. Sürekli döngü kodunu ekle. Kostümler arası geçiş kodlarını ekle. En son kısımda ise Yer: Rüya mahallesi, Tarih:17 Mart 2016 Saat: 17.00 bilgilerini ekle. Tasarıma uygun olarak müzik eklemek amacıyla sesini çal kodunu ekle.
6) Kaynaklar (Sources):	Powtoon programıyla hazırlanmış Animasyonlu Davetiye senaryosu, Çalışma Kâğıdı
7)Geribildirim (Feedback):	-Gayet güzel ilerliyorsun, Bu kodu burada eksik kullanmışsın, Tasarımında renk efekti uygulayabilirsin,

Hafta 2- Etkileşimli Piyano Oyunu



Şekil 31: Hafta 2- Etkileşimli Piyano Oyunu

Tablo 33: Hafta2: Etkileşimli Piyano Oyunu Uygulaması

Hafta2	
Ünite: Scratch ile Programlamaya Giriş	
Etkinlik: Piyano Oyunu Uygulaması	
1)Öğrenme Hedefleri:	Kazanımlar: 1.Sahne , Karakter ve Kostüm kullanabilme 2.Hareket bloğunu kullanabilme 3.Görünüm bloğunu kullanabilme 4.Ses bloğunu kullanabilme 5.Kontrol bloğunu kullanabilme
2)Misyon:	1)Genel Tasarım Öğelerini Düzenleme Renk efektini değiştirme FarklıKarakterlerekleyebilme (Balon, Nota, Şapka , İnsan, Mikrofon,)

Piyano notalarını ekleyebilme ve yerleştirebilme

Karakterlere farklı kostümler ekleyebilme

Sahne arkaplanını değiştirme

2)Hareket Bloğu komutlarını kullanabilme

Karakterleri hareket ettirebilme

Döndürme kodunu kullanabilme

3)Görünüm Bloğu komutlarını kullanabilme

Kostümler arası geçiş kodunu verebilme

4) Ses Bloğu komutlarını kullanabilme

Tasarıma uygun her notaya vuruş ve nota belirtilerek farklı ses ekleyebilme

5) Kontrol Bloğu komutlarını kullanabilme

Yeşil Bayrak Tıklandığında kodunu ekleyebilme

Karakter Tıklandığında kodunu ekleyebilme

Bekleme kodunu kullanabilme

Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme

Görevi Tamamlayabilme

3)Ana Hikâye:

Erenlerin evinde doğum günü partisi yapılacaktır. O yüzden Erenin annesi partiye hazırlık yapmayı planlamaktadır. Öncelikle evde temizlik yapmayı düşünmektedir. Daha sonra Erenin arkadaşları parti için Rüya Mahallesindeki Erenlerin evine gelmek için yola çıkmışlardır.Erenin arkadaşlarından birisi evlerine geldiklerinde “Haydi kapıyı çalalım” diye seslenir.Erenin annesi Ezgi, Eren’in arkadaşlarını Erenin doğum günü partisine hoş geldiniz diye karşılar. Arkadaşları eve gelmez Erenin nerede olduğunu sorarlar. Annesi Eren’in parti için hazırlandığını ve hemen çağıracağını söyler. Eren’e seslenir: “Eren bak oğlum , kimler gelmiş , arkadaşların seni bekliyor.”Eren bir anda annesi ve arkadaşlarını görünce çok şaşırarak mutlulukla havalara uçar. Eren annesine ve arkadaşlarına çok mutlu olduğunu ve bugün onun için anlamlı bir doğum günü olduğunu söyler.Erenin arkadaşları Eren ‘e çok güzel hediyeler almış ve bütün evi annesi balon ve konfetilerle süslemiştir.Eren hediyeleri görünce çok mutlu olur ve arkadaşlarına teşekkür eder. Daha sonra partinin devamında Erenin arkadaşları “Haydi dans edelim, Müzik zamanı diye düşünürler. Eren bunun için annesinden izin alıp kendi piyanosunu çalacağını söyler arkadaşlarına. Daha sonra Erenin annesi Erene piyano çalması için izin verir.Eren arkadaşlarının

karşısında piyano çalmaya hazırlanmaktadır. Ancak piyanonun tuşlarına basıldığında ses bir türlü çıkmadığını ve piyanonun çalışmadığını fark eder. Erenin arkadaşı Mert neden çalışmadığını sorar ve piyanonun çalışmadığına ilişkin akıl yürütür. Eren ise arkadaşlarına piyanoya sürekli bastığını ancak herhangi bir sesin çıkmadığını söyleyerek piyanonun çalışmadığını görünce çok üzümüştür. O yüzden arkadaşlarından doğum günü partisinde eğlenmek için yardım istemektedir. Arkadaşlarına doğum günü partisinde piyano çalmak ve arkadaşlarıyla eğlenmek istediğini ancak ne yazık ki evlerinde bulunan piyanonun çalışmadığını açıklar. Daha sonra aklın a bilgisayarda piyano oyunu tasarlamak gelir. Bilgisayarda hazırlayacağı piyanoda her tuştan farklı notalardan farklı sesler çıkarmasını ve her tuşa bastıkça tuşa basma görünümü kazandırarak daha gerçekçi ve animasyonlu bir piyano tasarlamak istediğini söyleyerek yardım istemektedir.

4) Görev (Role):

- *Piyano çalmaya uygun sahne tasarımı yapılmalı
- *13 Notadan oluşmalı
- *Her nota farklı ses çıkarmalı
- *Animasyonlu dönen, rengi yanıp sönen nota işaretleri olsun.
- *Piyano çalan bir karakter oluşturun.
- *Piyanonun yanında dans eden karakterler veya mikrofon ile şarkı söyleyen karakterler oluşturun.

5) Senaryo Hareketleri:

Öncelikle Scratch senaryosunda bulunan Eren'in doğum günü partisinde çalacağı piyano ve sahne ortamı için sahne arka plan tasarımını oluştur. Farklı renk ve şekillerle tasarlayabilirsin. İlk önce piyano tuşları için 8 beyaz 5 siyah olmak üzere toplam 13 tuş tasarla ve her tuşa basma görünümü kazandırmak amacıyla farklı kostümler ekle oluştur. Piyano tuşlarına sırasıyla Karakter tıkladığında kodunu ekle. Farklı kostümler arası geçiş kodları için kostüme geç kodunu ekle. Ve her tuş için farklı nota seslerini vermek amacıyla ...notasını ...vuruş çal kodunu ekle. Mikrofon karakteri ekle ve animasyon oluşturmak amacıyla renk efektini değiştir kodunu ekle. Nota işaretleri tasarla ve bunlara dönme efekti için 15 derece dön kodunu, renk efekti vermek için renk efektini ... değiştir kodunu ekle. kendiniz tasarlayarak yada hazır karakterler kullanarak piyanoyu çalan bir karakter oluşturun. Piyanonun yanında dans eden karakterleri oluştur. Ve dans etme görünümü sağlamak

amacıyla kostüm deęiřtirme kodunu ekle. Ve mikrofon ile řarkı söyleyen karakter oluřturabilirsin.

- 6)Kaynaklar (Sources): Powtoon programıyla hazırlanmıř Piyano Oyunu senaryosu, alıřma Kaęıdı
- 7)Geribildirim (Feedback): -Gayet gzel ilerliyorsun, Bu kodu burada eksik kullanmıřsın, Tasarımında renk efekti uygulayabilirsin, Piyano tuřlarına bastıęında ses çıkmıyor, nota seslerini ekleyebilirsin.

Hafta 3- Labirent Oyunu



řekil 32: Hafta 3- Labirent Oyunu

Tablo 34: Hafta3: Labirent Oyunu Uygulaması

Hafta3	
Ünite: Scratch ile Programlamaya Giriş	
Etkinlik: Labirent Oyunu Uygulaması	
1)Öğrenme	Kazanımlar:
Hedefleri:	1.Sahne , Karakter ve Kostüm kullanabilme 2.Hareket bloğunu kullanabilme 3.Görünüm bloğunu kullanabilme 4.Ses bloğunu kullanabilme 5.Kontrol bloğunu kullanabilme 6.Algilama bloğunu kullanabilme
2)Misyona:	1)Genel Tasarım Öğelerini Düzenleme Farklı Karakterler ekleyebilme (Farklı renklerde hediye paketleri, İnsan) Labirent (İç içe 3 Halka Duvar)düzenini ekleyebilme ve yerleştirebilme Karakterlere farklı kostümler ekleyebilme Sahne arkaplanını değiştirme 2)Hareket Bloğu komutlarını kullanabilme Karakterleri hareket ettirebilme (10 Adım git) Belirli yönüne dön (Derece olarak doğru belirleyebilme) x , y konumuna git kodunu ekleyebilme Döndürme kodunu kullanabilme 3)Görünüm Bloğu komutlarını kullanabilme Kostümler arası geçiş kodunu verebilme Yanlış Balona geldiğinde “Bu Doğru Balon Değil” uyarı mesajını ekleyebilme Doğru Balona geldiğinde “Tebrikler ” uyarı mesajını ekleyebilme 4) Ses Bloğu komutlarını kullanabilme Tasarıma uygun karakterin duvarlara çarptığında uyarı sesini ekleyebilme Tasarıma uygun karakterin doğru hediye paketine ulaştığında alkış sesini ekleyebilme 5) Kontrol Bloğu komutlarını kullanabilme Yeşil Bayrak Tıklandığında kodunu ekleyebilme Yön tuşlarına basıldığında kodunu ekleyebilme Bekleme kodunu kullanabilme Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme Koşul Döngüsü olarak Eğer ise Sürekli kodunu ekleyebilme 6) Algılama Bloğu komutlarını kullanabilme Karaktere değişiyor mu kodunu kullanabilme Rengine değişiyor mu kodunu kullanabilme

Görevi Tamamlayabilme

3)Ana Hikâye: Eren doğum günü partisinde arkadaşlarının getirdiği hediyelere çok sevinmiştir. Eren arkadaşlarına bu hediyeler için çok mutlu olduğunu söylemiş ve teşekkür etmiştir. Eren hediyeleri çok merak etmiş ve mutlulukla hediyeleri açmak için sabırsızlanmıştır. Tüm hediyelerini heyecanla teker teker açıp bakmıştır arkadaşlarının yanında. Arkadaşları parti bittikten sonra gitmek için Eren ve annesine veda ederek ayrılmışlardır.Eren ve annesi arkadaşlarına geldikleri için teşekkür etmiştir.Erenin arkadaşları evden gittikten sonra , Eren'e annesi evin her yerinin hediye paketleriyle dolduğunu ve evin bütün odalarında dağınık bir şekilde durduğunu söylemiştir.Eren ise annesine tüm hediyeleri toplayacağına söz vermiş ve evin farklı odalarına dağılan tüm hediyeleri biran önce toplamaya başlamıştır. Eren, arkadaşı Nil ay'ın ona aldığı hediye için çok merak etmektedir. Ancak onun aldığı kırmızı hediye paketini bir türlü bulamamaktadır. Evlerinin labirent gibi çok karışık olduğunu söylemiş ve tüm odaların hediye paketleriyle dolduğundan kırmızı hediye paketini bulamadığı için kırmızı hediye paketini bulmada yardım istemektedir.Erenin aklına bir fikir gelir. Labirent şeklinde olan evlerinin tüm odalarını gezen bir kahraman ona yardım edebilir ve kırmızı hediye paketini bulabilir. Kırmızı hediye paketine ulaştığında da ona ödül vereceğini söyler Hatta Tebrikler mesajını da vereceğini dile getirir. Ayrıca kahraman evin duvarlarına çarptığında da uyarı sesi gelmektedir.

*Labirent şeklinde bir ev ortamı oluşturun.

4) Görev (Role): *Labirent şeklinde olan evin tüm odalarını gezen bir kahraman oluşturun.
*Labirentin çeşitli noktalarına farklı renklerde hediye paketleri oluşturun. (Kendiniz tasarlayın yada hazır karakterlerden kullanabilirsiniz.)
*Kırmızı hediye paketine (Nilay'ın aldığı hediye paketine) ulaştığında "Tebrikler" mesajını oluşturun.
*Kahraman karakter labirent duvarlara çarptığında uyarı sesi oluşturun.
*Kahraman karakter yanlış hediye paketine ulaştığında "Bu doğru hediye paketi değil" mesajını oluşturun.

5)Senaryo Hareketleri: Öncelikle Scratch senaryosunda bulunan Eren'in doğum günü partisinde hediyeleri toplaması amacıyla kendiniz labirent şeklinde ev ortamının tasarımını oluşturun. Farklı renkler ve şekillerle tasarlayabilirsiniz. Labirent ortamını tasarlamak amacıyla kendin çizerek iç içe olacak şekilde oluşturun. İlk önce evin tüm odalarını gezen bir kahramanı kendin tasarla ya da hazır karakter olarak ekleyebilirsiniz. Labirentin çeşitli noktalarına ister kendin tasarlayarak isterse hazır karakterlerden kullanarak farklı renklerde veya farklı şekillerde hediye paketleri tasarla ve oluşturun. Karakterin labirent üzerinde klavye okları ile gezmesini sağlayan yön tuşlarına tıkladığında hangi yöne gidecekse o

yönüne dön kodlarını ekle. Böylece karakter klavye tuşlarıyla , yönlendirdiğimiz biçimde labirentin içinde dolaşabilecektir. Kırmızı hediye paketine (Nilay'ın aldığı hediye paketine) ulaştığında, kırmızı hediye paketine değişiyor mu kodunu ve “Tebrikler” mesajını oluşturmak amacıyla diye konuş kodunu ekle. Kahraman karakter labirent duvarlara çarptığında uyarı vermesi için labirent rengine değişiyor mu yada labirent duvarlarına değişiyor mu kodlarını ekle ve uyarı sesi oluşturmak için sesini çal kodunu ekle. Kahraman karakter yanlış hediye paketine ulaştığında ise “Bu doğru hediye paketi değil” mesajını oluşturmak için diye konuş kodunu ekle.

6)Kaynaklar
(Sources):

Powtoon programıyla hazırlanmış Labirent Oyunu senaryosu, Çalışma Kağıdı

-Gayet güzel ilerliyorsun, Bu kodu burada eksik kullanmışsın, Tasarımında renk efekti uygulayabilirsin,

7)Geribildirim
(Feedback):

Hafta 4- Hediye Toplama Oyunu



Şekil 33: Hafta 4- Hediye Toplama Oyunu

Tablo 35: Hafta4: Hediye Toplama Oyunu Uygulaması

Hafta4	
Ünite: Scratch ile Programlamaya Giriş	
Etkinlik: Hediye Toplama Oyunu Uygulaması	
1)Öğrenme	Kazanımlar:
Hedefleri:	1.Sahne , Karakter ve Kostüm kullanabilme 2.Hareket bloğunu kullanabilme 3.Görünüm bloğunu kullanabilme 4.Ses bloğunu kullanabilme 5.Kontrol bloğunu kullanabilme 6.Algılama bloğunu kullanabilme 7) Değişkenler Bloğu komutlarını kullanabilme
2)Misyona:	1)Genel Tasarım Öğelerini Düzenleme Farklı Karakterler ekleyebilme (Farklı renklerde hediye paketleri, farklı nesnelere, Sepet, İnsan) Puan ve Zaman değişkenlerini ekleyebilme ve yerleştirebilme Karakterlere farklı kostümler ekleyebilme 2)Hareket Bloğu komutlarını kullanabilme Sahne arka planını değiştirme Karakterleri hareket ettirebilme (10 Adım git) Belirli yönüne dön (Derece olarak doğru belirleyebilme) x , y konumuna git kodunu ekleyebilme Döndürme kodunu kullanabilme 3)Görünüm Bloğu komutlarını kullanabilme Göster/Gizle kodlarını kullanabilme Sahne Arkaplanı geçiş kodlarını kullanabilme 4) Ses Bloğu komutlarını kullanabilme Tasarıma uygun ses ekleyebilme 5) Kontrol Bloğu komutlarını kullanabilme Yeşil Bayrak Tıklandığında kodunu ekleyebilme Boşluk tuşuna basıldığında kodunu ekleyebilme Bekleme kodunu kullanabilme Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme , Koşul Döngüsü olarak Eğer ise kodunu ekleyebilme Bütün blokları durdur komutunu ekleyebilme 6) Algılama Bloğu komutlarını kullanabilme Sağ ok tuşuna basılı ise kodunu kullanabilme Sol ok tuşuna basılı ise kodunu kullanabilme Karakter değişiyor mu kodunu kullanabilme 7) Değişkenler Bloğu komutlarını kullanabilme Puan ve Zaman değişkenini ekleyebilme Puan değişkenini ve Zaman değişkenini düzenleyebilme 8) Operatörler Bloğu komutlarını kullanabilme Belirli iki sayı arasında rastgele sayı tut tahmin kodlarını kullanabilme Büyük, Küçük, Eşittir operatörlerini kullanabilme , Görevi tamamlayabilme
3)Ana Hikâye:	Eren Doğum günü partisindeki hediyelerden Nilay'ın almış olduğu hediyeyi bulmuş ve çok mutlu olmuştur. Nilayın hediyesini açmış ve arabayı çok

beğenmiştir. O heyecanla evin diğer odalarındaki hediyeleri toplamamış ve ev dağınık haldedir. Erenin annesi gelip Eren'e evin her yerinde hediyelerin dağınık olduğunu ve biran önce toplaması gerektiğini söylemiştir. Eren ise Evin her yerine dağılmış olan tüm hediyeleri nasıl toplayacağını bilememektedir ve ne yapacağı konusunda düşünmektedir. Eren arkadaşlarından doğum günü partisindeki hediyeleri bir sepet yardımıyla toplama konusunda yardım beklemektedir. Bütün hediye paketlerini topladıkça annesi tarafından her hediye paketini topladıkça puan kazanacağını ve bu işlem için belirli bir süresi olduğunu söylemektedir. Bu nedenle o kısıtlı sürede sepet yardımıyla tüm hediyeleri toplaması nedeniyle yardım beklemektedir.

4) Görev (Role): * Öncelikle Scratch senaryosunda bulunan Eren'in doğum günü partisinde gelen hediyeleri toplamak amacıyla sahne ortamı için sahne arka plan tasarımını oluştur.

*Sahnede tüm hediyeleri toplamak amacıyla bir sepet veya bir insan tasarla.

*Sahnenin çeşitli noktalarına farklı renklerde hediye paketleri oluşturun. (Kendiniz tasarlayın yada hazır karakterlerden kullanabilirsiniz.)

*Hediye paketlerini topladıkça her seferinde puanını artması için puan değişkeni oluştur.

*Belirli sürede bu oyunun tamamlanması amacıyla zaman skoru oluştur. Zaman değişkeni 60 dan geriye doğru gitmesi için kodla.

*Hediye paketleri sepete ulaştığında puan değişkenini 10 değiştir, eğer sepete ulaşmadığında ise puan değişkenini -10 değiştir.

5)Senaryo Hareketleri: Öncelikle Scratch senaryosunda bulunan Eren'in doğum günü partisinde gelen hediyeleri toplamak amacıyla sahne ortamı için sahne arka plan tasarımını oluştur. Sahne arka planını ister hazır olarak ister kendiniz tasarlayarak oluşturabilirsiniz. Sahnede tüm hediyeleri toplamak amacıyla bir sepet veya bir insan tasarla. Sahnenin çeşitli noktalarına farklı renklerde hediye paketleri oluşturun. (Kendiniz tasarlayın ya da hazır karakterlerden kullanabilirsiniz.) Sepet yardımıyla tüm hediye paketlerini toplamak amacıyla klavye tuşlarıyla sağ ok ve sol ok tuşunu kullanmasını algılamak amacıyla algılama bloğundan yön tuşuna basılı kodunu ekle. Hediye paketlerini topladıkça her seferinde puanının artması için puan değişkeni oluştur. Belirli sürede bu oyunun tamamlanması amacıyla zaman skoru oluştur. Zaman değişkeni 60 dan geriye doğru gitmesi için zaman değişkenini 60 yap ve geriye doğru gitmesi amacıyla zaman değişkenini -1 değiştir kodunu ekle. Hediye paketlerini topladıkça yani hediye paketleri sepete ulaştığında puan değişkeninin artması amacıyla puan değişkenini 10 değiştir, eğer sepete ulaşmadığında ise puan değişkenini -10 değiştir kodlarını ekle.

- 6) Kaynaklar Powtoon programıyla hazırlanmış Hediye Toplama Oyunu senaryosu, Çalışma (Sources): Kağıdı
- 7) Geribildirim Aferin, Tasarımını güzel yapmışsın, kodlamada hataların var, dikkat etmen (Feedback): gerekir şeklinde olmuştur.

Hafta 5- Bilgi Yarışması Oyunu



Şekil 34: Hafta 5- Bilgi Yarışması Oyunu

Tablo 36: Hafta5: Bilgi Yarışması Oyunu Uygulaması

Hafta5	
Ünite: Scratch ile Programlamaya Giriş	
Etkinlik: Bilgi Yarışması Oyunu Uygulaması	
<hr/>	
1)Öğrenme	Kazanımlar:
Hedefleri:	1.Sahne , Karakter ve Kostüm kullanabilme 2.Görünüm bloğunu kullanabilme 3.Kontrol bloğunu kullanabilme 4.Algılama bloğunu kullanabilme 5. Değişkenler Bloğu komutlarını kullanabilme 6. Operatörler Bloğu komutlarını kullanabilme
2)Misyön:	1)Genel Tasarım Ögelerini Düzenleme Farklı Karakterler ekleyebilme (Farklı sorulara görsel farklı seçenekler, Doğru yada Yanlış işareti gösteren True False Simgeleri İnsan) Görsel seçenekleri ekleyebilme ve yerleştirebilme True ve False simgelerinden her soru için eşit sayıda ekleyebilme ve yerleştirebilme Sahne arkaplanını değiştirme 2)Görünüm Bloğu komutlarını kullanabilme Göster/Gizle kodlarını kullanabilme Tebrikler , bildiniz; yada Bilemediniz, Yanlış uyarı mesajlarını doğru biçimde ekleyebilme 3) Kontrol Bloğu komutlarını kullanabilme Yeşil Bayrak Tıklandığında kodunu ekleyebilme Karakter tıklandığında kodunu ekleyebilme Duyurusunu yap kodunu ekleyebilme Duyurusu yapıldığında kodunu ekleyebilme Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme Koşul Döngüsü olarak Eğer ise öyle değilse kodunu ekleyebilme Bu blokları durdur komutunu ekleyebilme 4) Algılama Bloğu komutlarını kullanabilme Soruyu sormak amacıyla -diye sor ve bekle kodunu kullanabilme Yanıt değişkenini ekleyebilme ve doğru şekilde kullanabilme 5) Operatörler Bloğu komutlarını kullanabilme Eşittir operatörünü kullanabilme Görevi Tamamlayabilme
3)Ana Hikâye:	Eren ve arkadaşları, sabah okula gittiklerinde öğretmenleriyle karşılaştılar. Öğretmenleri ile konuşurlar. Öğretmenleri Öğrencilere haftaya Okullar arası

Bilgi Yarışmasının düzenleneceğini söyler. Ayrıca onlara çok çalışmaları gerektiğini ve bilgi yarışmasında kazananın ödül kazanacağını söyler. Öğrenciler de bilgi yarışmasının olacağını öğrenince çok fazla çalışmaları gerektiğini düşünürler. Öğrenciler Bilgi Yarışmasında kazanana verilecek olan ödülün ne olduğunu merak ederler. Öğretmenleri ise onlara bilgi yarışmasında kazanan öğrenciye Tablet Bilgisayar hediye edeceğini söylemiştir. Eren, bunu duyunca heyecanlanır ve kendisinin tablet bilgisayara ihtiyacı olduğunu, bu yarışmayı mutlaka kazanmaları gerektiğini düşünmektedir. Daha sonra Öğretmenlerinin aklına bir fikir gelir. Bu hafta içinde kendi okullarında hazırlık amaçlı Bilgi Yarışması düzenlenebileceği önerisinde bulunur. Öğrenciler de bunun iyi bir fikir olduğunu hem de faydalı olabileceğini ifade ederler. Öğretmen bunun için hazırlık amaçlı olarak Bilgi Yarışmasını öğrencilere bilgisayar ortamında hazırlama görevi verir. Mert ise bunu duyunca kendi evlerinde bilgisayar olduğunu söyler ve akşam Eren ile birlikte yapma kararı alırlar. Erenin aklına Bilgi Yarışmasını Oyun ortamında tasarlandığında daha akılda kalıcı ve eğlenceli olacağı fikri gelir. Ancak Eren bilgisayar ortamında nasıl Bilgi Yarışması oyununu tasarlayacağını bilememekte ve soruları nasıl hazırlayacağını düşünmektedir. Bu konuda Eren Bilgi Yarışması oyununu hazırlamada yardım istemektedir. Hazırlanacak olan Bilgi Yarışmasında Öğretmen tarafından bazı görevler belirlenmiştir. Öğretmen 4 sorudan oluşan, çoktan seçmeli ve görsel olarak 4 seçeneği bir bilgi yarışması oyunu hazırlamalarını ister.

4) Görev (Role):

- * Öncelikle Bilgi yarışması oyunu senaryosu için Sahne arkaplanını tasarla.
 - * Bilgi Yarışmasını yönetecek olacak farklı karakterler yada insan
 - * Farklı sorulara farklı görsel seçenekler ekle ve yerleştir.
 - * Soruların doğru yada yanlış olduğuna dair bildirimler vermesi amacıyla true yada false simgelerini her soru için eşit sayıda ekle.
 - * Her soruyu duyurusunu yaparak kodlarını oluştur.
 - * Sorulara geridönüt vermek amacıyla Tebrikler , bildiniz; yada Yanlış Bilemediniz, ve Yanlış uyarı mesajlarını doğru biçimde ekle.
 - * Bilgi Yarışması sorularını oluşturmak amacıyla Algılama Bloğundan Sor ve bekle kodlarını ekle.
 - * Soruların cevaplarını kontrol etmek amacıyla eğer yanıt=..... ise kodunu ekle.
 - * Soruda doğru seçenek üzerinde True simgesi, Yanlış seçenekler üzerinde False işaretinin oluşması için gerekli kodları tasarla.
-

5)Senaryo Hareketleri:	<p>Bilgi Yarışması oyunu senaryosunda öncelikle Bilgi yarışması oyunu senaryosu için Sahne arka planını tasarla. Bilgi Yarışmasını yönetecek olacak farklı karakterler yada insan ve farklı sorulara farklı görsel seçenekler ekle ve yerleştir.</p> <p>Soruların doğru yada yanlış olduğuna dair bildirimler vermesi amacıyla true yada false simgelerini her soru için eşit sayıda ekle. Her soruyu duyurusunu yaparak kodlarını oluştur.</p> <p>Sorulara geri dönüt vermek amacıyla Tebrikler , bildiniz; yada Yanlış Bilemediniz, ve Yanlış uyarı mesajlarını doğru biçimde ekle.Bilgi Yarışması sorularını oluşturmak amacıyla Algılama Bloğundan Sor ve bekle kodlarını ekle.</p> <p>Soruların cevaplarını kontrol etmek amacıyla eğer yanıt=..... ise kodunu ekle. Soruda doğru seçenek üzerinde True simgesi, Yanlış seçenekler üzerinde False işaretinin oluşması için gerekli kodları (Görünüm bloğundan Göster/Gizle) tasarla.Bilgi Yarışması Soruları olarak,</p> <ol style="list-style-type: none">1)Bilgisayarın elle tuttuğumuz gözle görebildiğimiz parçaların genel adı nedir?2) Hangisi Donanım Parçası değildir?3) Hangisi Bilgisayar üzerinden yazılan yazıları çıktı almamızı sağlayan donanım aracıdır?4)Hangisi Sosyal Paylaşım Sitelerine Örnek değildir? şeklinde sırasıyla kod yazarak oluştur.
6) Kaynaklar (Sources):	<p>Powtoon programıyla hazırlanmış Bilgi Yarışması Oyunu senaryosu, Çalışma Kağıdı</p>
7) Geribildirim (Feedback):	<p>Aferin, Tasarımını güzel yapmışsın, kodlamada hataların var, dikkat etmen gerekir şeklinde olmuştur.</p>

Hafta 6- Balon Patlatma Oyunu



Şekil 35: Hafta 6- Balon Patlatma Oyunu

Tablo 37: Hafta6: Balon Patlatma Oyunu Uygulaması

Hafta 6: Balon Patlatma Oyunu	
1)Öğrenme	Kazanımlar:
Hedefleri	1.Sahne, Karakter ve Kostüm kullanabilme 2.Hareket bloğunu kullanabilme 3.Görünüm bloğunu kullanabilme 4.Ses bloğunu kullanabilme 5.Kontrol bloğunu kullanabilme 6.Algılama bloğunu kullanabilme 7) Değişkenler Bloğu komutlarını kullanabilme

2)Misyon:	<p>1)Genel Tasarım Öğelerini Düzenleme</p> <p>Farklı Karakterler ekleyebilme (Farklı renklerde balonlar, balonları patlatacak nesne)</p> <p>Oyunun kolaylık derecesini belirleyecek olan kolay orta ve zor gibi buton seçeneklerini ve oyuna başla butonunu tasarlayıp ekleyebilme ve yerleştirebilme</p> <p>Sahne arka planını değiştirme</p> <p>2)Görünüm Bloğu komutlarını kullanabilme</p> <p>Göster/Gizle kodlarını kullanabilme</p> <p>Kolay, Orta , Zor kostümüne geç kodlarını doğru biçimde ekleyebilme</p> <p>3) Kontrol Bloğu komutlarını kullanabilme</p> <p>Yeşil Bayrak Tıklandığında kodunu ekleyebilme</p> <p>Karakter tıklandığında kodunu ekleyebilme</p> <p>Duyurusunu yap kodunu ekleyebilme</p> <p>Duyurusu yapıldığında kodunu ekleyebilme</p> <p>Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme</p> <p>Koşul Döngüsü olarak Eğer ise öyle değilse kodunu ekleyebilme</p> <p>Bu blokları durdur komutunu ekleyebilme</p> <p>4) Algılama Bloğu komutlarını kullanabilme</p> <p>Soruyu sormak amacıyla -diye sor ve bekle kodunu kullanabilme</p> <p>Skor değişkenini ekleyebilme ve doğru şekilde kullanabilme</p> <p>x i yap, y yi yap kodlarını doğru biçimde ekleyebilme</p> <p>5) Operatörler Bloğu komutlarını kullanabilme</p> <p>Eşittir operatörünü kullanabilme</p> <p>İki sayı arasında rastgele sayı üret kodunu ekleyebilme.</p> <p>Görevi Tamamlayabilme</p>
3)Ana Hikâye:	<p>Öğretmen, Okullar arası düzenlenen bilgi yarışmasında Eren ve Mertlerin grubunun 1. olduğunu söyleyince bu duruma Erenlerin grubu çok sevinmiş fakat diğer arkadaşları üzülmüştür. Öğretmen Eren ve grubunu Bilgi Yarışmasını kazandıkları için tebrik etmiştir.Yarışmanın sonunda Erenlerin Tablet Bilgisayar kazandıklarını belirterek onlara Tablet Bilgisayar hediye etmiştir. Mert, Eren 'e evde Bilgisayar ortamında tasarladıkları oyunun işe yaradığını, Eren ise daha öncesinde böyle bir oyunu yapmalarının işe yaradığını söyleyerek çok sevinir. Öğretmen Bilgi Yarışması Oyununun kazananını açıkladıktan sonra bunun için yarın bir kutlama yapma önerisinde bulunur. Sınıfı süslemek için renkli balonlar ve grafiti süsler aldığını söyler. Çocuklar bu duruma çok sevinir.Öğretmen partinin sonunda da yine bir oyun oynayıp yarışma yapacağını söyler.Bunu duyunca Eren ve Mert ne yarışması</p>

yapacaklarını merak ederler.Öğretmen yarışmanın kurallarını açıklar.Bu yarışmada , partinin sonunda bahçede balonlar uçarken onları patlatma oyunu oynayacaklarını ve onları patlatma oyunu oynayacaklarını ve onları evden getirecekleri bir şeyle patlatabileceklerini söyler.Mert Bilgisayarında buna benzer bir oyun olduğunu söyleyince Öğretmen onlara bu akşam oradan alıştırmayı yapabileceklerini söyler. En çok balon patlatana dondurma hediyesi olduğunu ve çok güzel bir oyun olduğunu söyler. Bunu duyan Eren ve Mert öğretmenlerine sürpriz olarak Bilgisayarda balon patlatma oyunu tasarlama planı yaparlar ve oyunu tasarlamak için yardım istemektedirler.

4) Görev (Role):

* Öncelikle Balon Patlatma oyunu için sahne arka planını tasarla.

*Balon patlatma oyununa başlamak için oyun seviyesi belirlemek amacıyla kolay, orta ve zor butonunu tasarla ve yerleştir.

* Farklı renklerde balonlar tasarla ve yerleştir.

*Balonları patlatacak bir nesne oluştur.

* Puan skoru oluştur ve balonları patlattıkça puanın artmasını sağlayacak kodları oluştur.

*Balonlara patlatıcı nesne değiştiğinde balonun kaybolmasını sağlayacak göster gizle kodlarını ekle.

*En son oyun bittiğinde toplam skorunu gösterecek olan skor karakterini oluştur.

5)Senaryo

Hareketleri:

Öncelikle Balon Patlatma oyunu için sahne arka planını tasarla. Balon patlatma oyununa başlamak için oyun seviyesi belirlemek amacıyla kolay, orta ve zor butonunu tasarla ve yerleştir. Orta ve zor butonları için yapım aşamasında olduğuna dair uyarı mesajları oluştur. Farklı renklerde balonlar tasarla ve yerleştir. Balonları patlatacak bir nesne oluştur.Puan skoru oluştur ve balonları patlattıkça puanın artmasını sağlayacak kodları oluştur.Balonlara patlatıcı nesne değiştiğinde balonun kaybolmasını sağlayacak göster gizle kodlarını ekle.Balonların kaybolduktan sonra tekrar farklı konumlarda yeniden oluşmasını sağlayacak iki sayı arasında rastgele sayı üret kodunu oluştur. X ve y konumlarını rastgele sayılar ata. En son oyun bittiğinde toplam skorunu gösterecek olan skor karakterini oluştur. Ve Oyunun 10 saniyede bitmesini sağlamak amacıyla zaman değişkeni oluştur yada bekleme kodunu tasarlayabilirsin.

6) Kaynaklar
(Sources):

Powtoon programıyla hazırlanmış Balon Patlatma Oyunu senaryosu , Çalışma kağıdı

7) Geribildirim
(Feedback):

Aferin, Tasarımını güzel yapmışsın, kodlamada hataların var, dikkat etmen gerekir şeklinde olmuştur.

Hafta 7-Matematik 4 İşlem Resimli Bulmaca Oyunu



Şekil 36: Hafta 7- Matematik 4 İşlem Resim Bulmaca Oyunu

Tablo 38: Hafta7: Matematik 4 İşlem Resim Bulmaca Oyunu Uygulaması

Hafta 7: Matematik 4 İşlem Resimli Bulmaca Oyunu	
1)Öğrenme	Kazanımlar:
Hedefleri:	1.Sahne , Karakter ve Kostüm kullanabilme
	2.Hareket bloğunu kullanabilme
	3.Görünüm bloğunu kullanabilme
	4.Ses bloğunu kullanabilme
	5.Kontrol bloğunu kullanabilme
	6.Algılama bloğunu kullanabilme
	7)Değişkenler Bloğu komutlarını kullanabilme
	8)Operatörler Bloğu komutlarını kullanabilme
2)Misyön:	1)Genel Tasarım Öğelerini Düzenleme
	Farklı Karakterler ekleyebilme (Farklı renklerde soruların cevabı olacak sayılar, İnsan karakteri)

Oyundaki soruların cevaplarını belirleyecek olan sayı butonu seçeneklerini ve oyuna başla butonunu tasarlayıp ekleyebilme ve yerleştirebilme

Sahne arka planını değiştirme (Level1, Level2, Resim 1, Resim2, Game Over, Düşükbitiş, Ortabitış, İyi bitiş)

2)Görünüm Bloğu komutlarını kullanabilme

Göster/Gizle kodlarını kullanabilme

Sahneler arası kostümüne geç kodlarını doğru biçimde ekleyebilme

Soruları oluşturmak amacıyla diye konuş kodunu ekleyebilme ve soruları oluşturabilme

3) Kontrol Bloğu komutlarını kullanabilme

Yeşil Bayrak Tıklandığında kodunu ekleyebilme

Karakter tıklandığında kodunu ekleyebilme

Duyurusunu yap kodunu ekleyebilme

Duyurusu yapıldığında kodunu ekleyebilme

Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme

Koşul Döngüsü olarak Eğer ise öyle değilse kodunu ekleyebilme

Bu blokları durdur komutunu ekleyebilme

4) Algılama Bloğu komutlarını kullanabilme

Soruyu sormak amacıyla -diye sor ve bekle kodunu kullanabilme

Skor değişkenini ekleyebilme ve doğru şekilde kullanabilme

x i yap, y yi yap kodlarını doğru biçimde ekleyebilme

5) Operatörler Bloğu komutlarını kullanabilme

Eşittir ve veya operatörünü kullanabilme

İki sayı arasında rastgele sayı üret kodunu ekleyebilme.

Görevi Tamamlayabilme

3)Ana Hikâye:

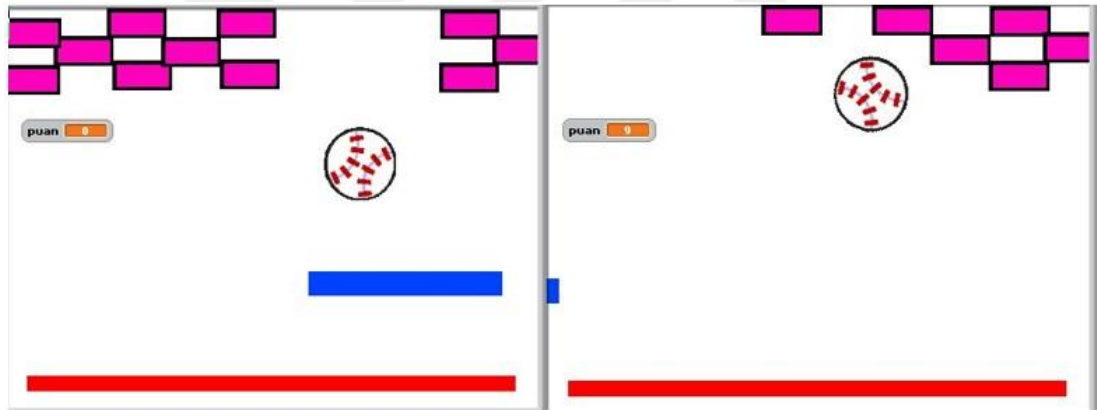
Eren ile Mert Okula gittiklerinde Öğretmen onlara Matematik dersinde işlemiş olduğu matematiksel işlemler ile ilgili bilgilerini pekiştirmek amacıyla onlardan matematiksel 4 işlemle resim bulmaca oyunu tasarlamalarını istemektedir. Bu tasarlanan oyuna göre her sorunun cevapları bir kutucuk şeklinde sayı ile gösterilmiştir. Buradaki soruyu doğru cevabını bildiğinde o seçenek gözden kaybolmaktadır. Ve 2 Level şeklinde tanımlanan bu oyunda level geçişleri kolaylık zorluk seviyesine göredir. Her leveli doğru olarak tamamlayan öğrenciler gizlenen resmi bulabilecektir. Bu oyunda her soru doğru olarak tamamlandığında 100 puan kazanacak , aksi takdirde kaybettiğinde ise yanlış cevap diye bildiri vermesi istenmektedir. Ayrıca 2 leveli de başarıyla tamamlandığında puan değişkeni 360 ve 360 dan büyükse ; Matematiğe biraz daha çalışmalısın şeklinde uyarı gelmesi, 1720 ye eşit yada büyükse Tebrikler Matematiğin çok iyi şeklinde uyarı, ortada bir değerdeyse Matematik 4 işlemde fena değilsin ama biraz daha çalışmalısın şeklinde uyarı gelmesi istenmektedir.

	<p>Her levelin sonunda bulmaca şeklinde tasarlanan sürpriz resimler soruları bildikçe , soru butonları kaybolacak ve resimler ortaya çıkacaktır.</p>
4) Görev (Role):	<p>*Öncelikle Matematik 4işlemlerle resim bulmaca oyunu için sahne arka planını tasarla.</p> <p>*Matematik 4 işlemlerle resim bulmaca oyunu için sahne arka plan tasarımında Başlangıç, Level1'e geçiş, Level2'ye geçiş, Oyun1, Oyun2, Kazanma, Kaybetme ve Orta dereceyle bitirme sahneleri olmak üzere 8 farklı sahne oluştur.</p> <p>*Daha sonra levellerin zorluk seviyelerine göre her 2 level için farklı 4 sorunun cevabı olacak şekilde cevap sayılarını tasarla.</p> <p>* Oyun 1 sahnesinin üzerine bulmaca resim ve 4 farklı sayı gelecek, Oyun 2 Sahnesinin üzerine farklı bir resim ve yine 4 farklı sayı gelecek şekilde farklı sorulara farklı cevap sayılarını tasarla ve yerleştir.</p> <p>* Daha sonra Görünüm bloğundan uygun soruların uygun sahnelerde görünüp gizlenme kodlarını yerleştir.</p> <p>* Oyuna başlamak için bir başla butonu tasarla ve butona basıldığında oyunun başlaması ve farklı soruların gelmesi açısından uygun kodları tasarla.</p> <p>*Soruları tasarlamak amacıyla Algılama bloğundan sor ve bekle kodunu kullanabilirsin.</p> <p>*Sorulan sorulara doğru cevap olarak doğru sayıya tıkladığında sayının kaybolması ve arkadaki resmin ön plana çıkması amacıyla gerekli koşul kodlarını tasarla.</p> <p>*Sorulan sorulara yanlış cevap verildiğinde yanlış cevap diye bildiri vermesi için uygun kodları tasarla.</p> <p>*Her soruya doğru cevap verildiğinde puan değişkeninin 200 arttır.</p> <p>*Oyun bittiğinde toplam puan durumuna göre kazanma sahnesi, yada kaybetme sahnesi yada orta dereceyle oyunu bitirme sahnesiyle karşılaşmak amacıyla gerekli kodları oluştur.</p>
5)Senaryo	Matematik 4işlemlerle resim bulmaca oyunu için sahne arka planını kendin tasarla.
Hareketleri:	Matematik 4 işlemlerle resim bulmaca oyunu için sahne arka plan tasarımında Başlangıç, Level1'e geçiş, Level2'ye geçiş, Oyun1, Oyun2, Kazanma , Kaybetme ve Orta dereceyle bitirme sahneleri olmak üzere 8 farklı sahne oluştur.Daha sonra levellerin zorluk seviyelerine göre her 2 level için farklı 4 sorunun cevabı olacak şekilde cevap sayılarını tasarla. Oyun 1 sahnesinin üzerine bulmaca resim ve 4 farklı sayı gelecek, Oyun 2 Sahnesinin üzerine farklı bir resim ve yine 4 farklı sayı gelecek şekilde farklı sorulara farklı cevap sayılarını tasarla ve yerleştir. Daha sonra Görünüm bloğundan uygun soruların uygun sahnelerde görünüp gizlenme kodlarını yerleştir. Oyuna başlamak için bir başla butonu tasarla ve butona basıldığında oyunun başlaması ve farklı

soruların gelmesi açısından uygun kodları tasarla. Soruları tasarlamak amacıyla Algılama bloğundan sor ve bekle kodunu kullanabilirsin. Sorulan sorulara doğru cevap olarak doğru sayıya tıkladığında sayının kaybolması ve arkadaki resmin ön plana çıkması amacıyla gerekli koşul kodlarını tasarla. Sorulan sorulara yanlış cevap verildiğinde yanlış cevap diye bildiri vermesi için uygun kodları tasarla. Her soruya doğru cevap verildiğinde puan değişkenininin 200 arttır. Oyun bittiğinde toplam puan durumuna göre kazanma sahnesi, yada kaybetme sahnesi yada orta dereceyle oyunu bitirme sahnesiyle karşılaşmak amacıyla gerekli kodları oluştur.

- 6) Kaynaklar Powtoon Programı aracılığıyla hazırlanmış Matematik 4 İşlem Resimli (Sources): Bulmaca Oyunu Senaryosu
- 7) Geribildirim Aferin, Tasarımını güzel yapmışsın, kodlamada hataların var, dikkat etmen (Feedback): gerekir şeklinde olmuştur.

Hafta 8-Tuğla Kırma Oyunu



Şekil 37: Hafta 8- Tuğla Kırma Oyunu

Tablo 39: Hafta8: Tuğla Kırma Oyunu Uygulaması

Hafta 8: Tuğla Kırma Oyunu	
1)Öğrenme	Kazanımlar:
Hedefleri:	1.Sahne , Karakter ve Kostüm kullanabilme
	2.Hareket bloğunu kullanabilme
	3.Görünüm bloğunu kullanabilme
	4.Ses bloğunu kullanabilme
	5.Kontrol bloğunu kullanabilme
	6.Algılama bloğunu kullanabilme
	7) Değişkenler Bloğu komutlarını kullanabilme

	8)Operatörler Bloğu komutlarını kullanabilme
2)Miyon:	<p>1)Genel Tasarım Öğelerini Düzenleme</p> <p>Farklı Karakterler ekleyebilme (Farklı renklerde tuğla nesnelere, tuğlaları kırarak top karakteri, kırmızı yer çubuğu, topu yakalama çubuğu)</p> <p>Puan değişkenini ekleyebilme ve yerleştirebilme</p> <p>Sahne arka planını değiştirme</p> <p>2)Hareket Bloğu komutlarını kullanabilme</p> <p>Topun hareket etmesi amacıyla yönüne dön ve 10 adım git kodlarını kullanabilme.</p> <p>3)Görünüm Bloğu komutlarını kullanabilme</p> <p>Göster/Gizle kodlarını kullanabilme</p> <p>4) Kontrol Bloğu komutlarını kullanabilme</p> <p>Yeşil Bayrak Tıklandığında kodunu ekleyebilme</p> <p>Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme</p> <p>Koşul Döngüsü olarak Eğer ise öyle değilse kodunu ekleyebilme</p> <p>5) Algılama Bloğu komutlarını kullanabilme</p> <p>Rengine değişiyor mu kodunu kullanabilme</p> <p>Tuğlaların topa değdiğinde kaybolması amacıyla top karakterine değişiyor mu kodunu ekleyebilme</p> <p>Topu yakalayacak olan çubuğun fareyle birlikte hareket etmesi amacıyla farenin x koordinatı kodunu ekleyebilme.</p> <p>6) Operatörler Bloğu komutlarını kullanabilme</p> <p>Eşittir operatörünü kullanabilme</p> <p>İki sayı arasında rastgele sayı üret kodunu ekleyebilme.</p> <p>7)Değişkenler bloğu komutlarını kullanabilme</p> <p>Puan değişkenini ekleyebilme ve doğru şekilde kullanabilme</p> <p>Puan değişkenini azaltma ve artırma kodlarını doğru biçimde kullanabilme</p> <p>Görevi Tamamlayabilme</p>
3)Ana Hikâye:	Eren ile Mert Okula gittiklerinde Beden Eğitimi dersinde Sınıf Öğretmeni bir oyun oynayacaklarını söylemiştir. Bu oyunda farklı renklerde taş ya da tuğlaları dizip o tuğlaları küçük bir topla kırmaya yada düşürmeye çalışılacaktır. Tuğlaları kırdıkça puan kazanacak top yörünge dışına çıktığında ise puan kaybedilecektir. Bu oyunu kazanmak için herkes önceden alıştırmaya çalışmaktadır. Bu oyunu Erenin kazanması için ona bilgisayar ortamında bir oyun tasarlayıp yardım edebilir misiniz?
4) Görev (Role):	<p>*Öncelikle Tuğla kırmaca oyunu için sahne arka planını tasarla.</p> <p>*Farklı renklerde veya farklı şekillerde tuğlalar tasarla ve yerleştir.</p> <p>*Farklı bir top karakteri ekle ve yerleştir.</p> <p>*Topun düştüğünde puan kaybetmesi amacıyla yer karakteri tasarla ve yerleştir.</p>

	<p>*Topun yere düşmesini önlemek amacıyla topu yakalayacak yakalama çubuğu tasarla ve yerleştir.</p> <p>*Puan değişkeni tasarla ve sahnenin bir yerine yerleştir.</p> <p>* Topun yere geldiğinde puan kaybetmesi amacıyla puan değişkenini azaltma ve tuğla kırdıkça puan kazanması amacıyla puan değişkenini arttırma kodunu doğru biçimde kullan.</p> <p>*Topu yakalayacak çubuğun fare imleci ile birlikte hareket etmesini sağlamak amacıyla farenin x koordinatı kodunu doğru biçimde kullan.</p> <p>* Topun yere düştüğünde ve topu yakalayacak çubuğa geldiğinde yukarıya doğru hareket etmesini sağlamak amacıyla uygun hareket bloğu kodlarını doğru biçimde kullan ve yerleştir.</p>
5)Senaryo Hareketleri:	<p>Öncelikle Tuğla kırmaca oyunu için sahne arka planını tasarla. Farklı renklerde veya farklı şekillerde tuğlalar tasarlayıp yerleştirmelisin.</p> <p>Farklı bir top karakteri kendin tasarla yada hazır olarak ekleyebilirsin ve yerleştirmelisin. Topun düştüğünde puan kaybetmesi amacıyla yer karakteri tasarlayıp yerleştirmelisin. Topun yere düşmesini önlemek amacıyla topu yakalayacak yakalama çubuğu tasarlayıp yerleştirmelisin Puan değişkeni tasarla ve sahnenin bir yerine yerleştirmelisin. Topun yere geldiğinde puan kaybetmesi amacıyla puan değişkenini azaltma ve tuğla kırdıkça puan kazanması amacıyla puan değişkenini arttırma kodunu doğru biçimde kullanmalısın. Topu yakalayacak çubuğun fare imleci ile birlikte hareket etmesini sağlamak amacıyla farenin x koordinatı kodunu doğru biçimde kullanmalısın. Topun yere düştüğünde ve topu yakalayacak çubuğa geldiğinde yukarıya doğru hareket etmesini sağlamak amacıyla uygun hareket bloğu kodlarını doğru biçimde yerleştirmelisin. Topun tuğlalara geldiğinde tuğlaların kırılması amacıyla toplar için karakterine değişiyor mu kodlarını ve görünüm bloğundan göster/gizle kodlarını doğru biçimde kullanmalısın.</p>
6) Kaynaklar (Sources):	<p>Powtoon Programı aracılığıyla hazırlanmış Tuğla Kırm Oyunu Senaryosu</p>
7) Geribildirim (Feedback):	<p>Aferin, Tasarımını güzel yapmışsın, kodlamada hataların var, dikkat etmen gerekir şeklinde olmuştur.</p>

Hafta 9-Trafik Kuralları Oyunu



Şekil 38: Hafta 9- Trafik Kuralları Oyunu

Tablo 40: Hafta9: Trafik Kuralları Oyunu Uygulaması

Hafta 9: Trafik Kuralları Oyunu	
1)Öğrenme	Kazanımlar:
Hedefleri:	1.Sahne , Karakter ve Kostüm kullanabilme 2.Hareket bloğunu kullanabilme 3.Görünüm bloğunu kullanabilme 4.Ses bloğunu kullanabilme 5.Kontrol bloğunu kullanabilme 6.Algılama bloğunu kullanabilme 7) Değişkenler Bloğu komutlarını kullanabilme 8)Operatörler Bloğu komutlarını kullanabilme
2)Misyona:	1)Genel Tasarım Öğelerini Düzenleme Farklı Karakterler ekleyebilme (Farklı renklerde arabalar, İnsan karakteri, Trafik Polisi, Başla butonu, Kurallar butonu, Lunapark, Trafik anayol) Oyunun kurallarını açıklayacak olan sayfaya yönlendirmek amacıyla kurallar butonunu ve oyuna başla butonunu tasarlayıp ekleyebilme ve yerleştirebilme Sahne arka planını değiştirme Sahne arka planı için Oyuna Giriş sahnesi, Kuralların açıklandığı kurallar sahnesi,

Kazanma ve kaybetme sahnesi ve arabaların trafik hareketinin olduğu anayol oyun sahnesi

1)Hareket Bloğu komutlarını kullanabilme

Arabaların trafikte hareketini sağlamak amacıyla 10 adım git, yada belirli saniyede x y konumuna git kodunu ekleyebilme,

İnsan karakterinin her yöne gitmesini sağlayan yönüne dön kodunu kullanabilme

2)Görünüm Bloğu komutlarını kullanabilme

Göster/Gizle kodlarını kullanabilme

Kırmızı, Sarı ve Yeşil kostümüne geç kodlarını doğru biçimde ekleyebilme

İnsan karakterinin yürümesi ve araba çarptığında kırmızı çağrı işareti gelecek şekilde kostümlerine geç kodlarını ekleyebilme

Arka planı değiştir kodunu ekleyebilme

3) Kontrol Bloğu komutlarını kullanabilme

Yeşil Bayrak Tıklandığında kodunu ekleyebilme

Karakter tıklandığında kodunu ekleyebilme

Duyurusunu yap kodunu ekleyebilme

Duyurusu yapıldığında kodunu ekleyebilme

Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme

Koşul Döngüsü olarak Eğer ise öyle değilse kodunu ekleyebilme

Süre olarak bekleme kodunu ekleyebilme

Bu blokları durdur komutunu ekleyebilme

4) Algılama Bloğu komutlarını kullanabilme

Arabaya geliyor mu yada Lunapark a geliyor mu kodunu ekleyebilme

x i yap, y yi yap kodlarını doğru biçimde ekleyebilme

Kırmızı sarı yada yeşil arabaya geliyor mu kodlarını ekleyebilme

5) Operatörler Bloğu komutlarını kullanabilme

Eşittir operatörünü kullanabilme

6)Değişkenler Bloğu komutlarını kullanabilme

Süre değişkenini ekleyebilme ve doğru şekilde kullanabilme, Süre değişkenini 15 yap kodunu ekleyebilme ve 1 sn geçtikçe süre değişkenini azaltma kodunu ekleyebilme

Görevi Tamamlayabilme

-
- 3)Ana Hikâye: Eren ile Annesi Lunaparka gitmek istemektedir. Lunaparkın oradan karşıdan karşıya geçmek zordur. Hem trafik ışığı koyup hem de geçmelerini sağlayabilirsiniz. Trafik ışıklarında kırmızı ışık yandığında beklemesi, sarı ışık yandığında hazır olması ve yeşil ışık yandığında geçmesi gerekmektedir. Eğer ışıklara uymadan geçtiğinde arabalara çarptığında trafik kazası geçirebilir , ve Eren için tehlikeli bir durum oluşturabilmektedir.Eren Trafik Işıklarına uygun arabalara çarpmadan karşıdan karşıya geçip lunaparka ulaştığında kazanacaktır.
- 4) Görev (Role): Eren'in ve annesinin güvenli bir şekilde Lunapark'a gidebilmelerini sağlamak amacıyla Scratch programını kullanarak Trafik Işıklarını oluşturmak ve karşıdan karşıya geçebilmelerini sağlayacak bir oyun tasarlamak. Bu oyun için öncelikle sahne arka plan tasarımlarını kazanma, kaybetme, oyuna giriş sahnesi ve oyunun kurallarını açıklayan kurallar sahnesi ve trafik anayolunda trafik akışının olacağı oyun sahnesini tasarla ve oluştur.Daha sonra bu oyundaki farklı renklerde araba karakterlerini kendin tasarlayabilirsin yada hazır olarak ekleyebilirsin.Lunaparkı da eklemelisin. Daha sonra Trafik ışıklarının kırmızı sarı ve yeşil olarak yanması için 3 farklı kostüm oluşturmalısın. İnsan karakteri arabaya çarptığında ve yürürken farklı adımlarda olacağı için 3 farklı kostüm oluşturmalısın. İnsan karakteri arabalara değdiğinde üzerine kırmızı çarpı işareti şeklinde kostüm oluşturmalısın. Eğer arabalara çarpmadan lunapark a ulaşırsa oyunu kazandınız tebrikler sahnesiyle karşılaşmaktadır. Oyun için 15 sn süre ayrılmıştır. Eğer verilen süreden önce insan karakteri lunapark a ulaşamazsa, oyunu kaybettiniz şeklinde uyarı mesajıyla karşılaşması amacıyla kazanma ve kaybetme sahnelerini de oluşturmalısın.
- 5)Senaryo Hareketleri: Bu oyunda Eren Trafik lambasının durumuna göre 15 sn içerisinde karşıdan karşıya geçmekte ve lunapark a ulaşmaya çalışmaktadır. Eren arabaya çarptığında Eren ölür ve oyun biter. Eren klavyedeki yön tuşlarıyla hareket etmektedir.Lunapark a geldiğinde Tebrikler mesajıyla karşılaşmaktadır. Oyunun başlangıç sahnesinde Başla ve Kurallar butonu olmak üzere 2 farklı buton tasarlamalısın. Kurallar butonuna tıkladığında kuralları açıklayacak olan sahne karşımıza çıkmaktadır. Başla butonuna tıkladığında ise Trafik şeridinin olduğu oyun sahnesine geçiş yapılmaktadır. Arabalar karşılıklı olarak hareket etmektedir. Trafik ışıkları kırmızı sarı ve yeşil olarak kostüm geçişleri yapılmaktadır. Trafik ışıklarının durumuna göre karşıdan karşıya geçerken arabaya çarptığında ölmektedir. Bu nedenle erenin yürümesi ve arabaya çarptığında farklı durumda olması için farklı kostümler oluşturmalısın. Eren karakterini karşıdan karşıya geçmesi için klavye tuşlarıyla hareket etmektedir.
-

-
- Bunun için uygun kodları oluşturmalsın. Eren in lunapark a ulaştığında tebrikler mesajıyla karşılaşması için de uygun sahneleri ve uygun kodları oluşturmalsın.
- 6) Kaynaklar Powtoon Programı aracılığıyla hazırlanmış Trafik Kuralları Oyunu Senaryosu
(Sources):
- 7) Geribildirim Aferin, Tasarımını güzel yapmışsın, kodlamada hataların var, dikkat etmen
(Feedback): gerekir şeklinde olmuştur.
-

Hafta 10-Mevsimler Oyunu



Şekil 39: Hafta 10-Mevsimler Oyunu

Tablo 41: Hafta 10: Mevsimler Oyunu

Hafta 10: Mevsimler Oyunu	
1)Öğrenme Hedefleri:	<p>Kazanımlar:</p> <ol style="list-style-type: none">1.Sahne , Karakter ve Kostüm kullanabilme2.Hareket bloğunu kullanabilme3.Görünüm bloğunu kullanabilme4.Ses bloğunu kullanabilme5.Kontrol ve Olaylar bloğunu kullanabilme6.Algılama bloğunu kullanabilme
2)Misyona:	<ol style="list-style-type: none">1)Genel Tasarım Öğelerini Düzenleme Farklı Karakterler ekleyebilme (Farklı görsel olarak mevsimleri niteleyecek güneş, deniz, ağaçtan dökülen yapraklar, kar yada kardan adam, kelebek, balerin, çiçekler vb. karakterler) İlkbahar, Yaz, Sonbahar ve Kış butonlarını görsel olarak tasarlayabilme ve ekleyebilme Sahne arkaplanını farklı mevsimlere uygun değiştirebilme2)Hareket Bloğu komutlarını kullanabilme Karakterlerin hareketini sağlayacak 10 adım git , kenara geldiye sektir yada derece dön kodlarını ekleyebilme Karakterlere x y konumuna git kodunu ekleyebilme Yaprakların hareketi yada kar tanelerinin düşmesi amacıyla 180 yönüne doğru yönel kodunu ekleyebilme3)Görünüm Bloğu komutlarını kullanabilme Kelebeklerin yada kuş karakterlerinin uçması amacıyla kostüm kılığına geç kodunu ekleyebilme ve yerleştirebilme İlkbahar, Yaz, Sonbahar ve Kış dekoruna geç kodunu ekleyebilme Tasarıma uygun Gökğürültüsü yada yağmur efektini verecek ses kodlarını ekleyebilme4) Kontrol ve Olaylar Bloğu komutlarını kullanabilme Yeşil Bayrak Tıklandığında kodunu ekleyebilme Karakter Tıklandığında kodunu ekleyebilme Duyurusunu yap kodunu ekleyebilme Duyurusu yapıldığında kodunu ekleyebilme Tekrar Döngüsü olarak Sürekli kodunu ekleyebilme Koşul Döngüsü olarak Eğer ise öyle değilse kodunu ekleyebilme

Dekor ilkbahar, yaz , sonbahar veya kış olarak deęiřtięinde kodunu ekleyebilme

Görevi Tamamlayabilme

3)Ana Hikâye: Eren ve arkadaşları okuldaki resim derslerinde resim yapmaktadırlar. Resim öęretmenleri sınıf içerisinde en güzel resmi yapan öęrenciyi ödüllendirecektir. Bu yüzden herkesin evlerinde örnek bir resim yapmasını ve çalışmasını istemiřtir. Eren ile Mert ise Bilgisayar üzerinde resim olarak farklı mevsimler tasarlayacaklardır. Bilgisayar ortamında resim yapabilecekleri bir oyun olsa hiç fena olmazdı aslında, onlara yardım edebilir misiniz?

4) Görev (Role): Farklı görsel olarak mevsimleri niteleyecek güneř, deniz, ağaçtan dökülen yapraklar, kar yada kardan adam, kelebek, balerin, çiçekler vb. karakterleri eklemek ve İlkbahar, Yaz, Sonbahar ve Kış butonlarını görsel olarak tasarlamak ve eklemek

Sahne arkaplanını farklı mevsimlere uygun deęiřtirebilmek. Öncelikle farklı mevsim geçiřlerinin kontrolünü sağlayacak butonları tasarlamalısın. Daha sonra her mevsimi yansıtacak güneř, deniz, ağaçtan dökülen yapraklar, kar yada kardan adam, kelebek, balerin, çiçekler vb. karakterleri eklemelisin. Yaprak düşmesi, kar tanelerinin düşmesi, kelebeęin uçması, denizde insanların yüzmesi gibi yaratıcı animasyonlar eklemelisin. Buton geçiřlerini yapmalısın. Butonlara tıkladıęında ait olduęu mevsimle ilgili animasyon geçiři oluřturmalısın.

5)Senaryo Hareketleri: Öncelikle Mavsimler oyununu tasarlamak amacıyla farklı mevsimleri yansıtacak arkaplanları eklemelisin. Daha sonra her mevsimi yansıtacak güneř, deniz, ağaçtan dökülen yapraklar, kar yada kardan adam, kelebek, balerin, çiçekler vb. karakterleri eklemelisin. Yaprak düşmesi, kar tanelerinin düşmesi, kelebeęin uçması, denizde insanların yüzmesi gibi yaratıcı animasyonlar eklemelisin.Mevsim geçiřlerini saęlamak amacıyla butonları tasarlamalısın ve sahnede uygun biçimde yerleřtirmelisinButonlara tıkladıęında o mevsimle ilgili arkaplan geçiřini saęlamalısın ve o mevsimle ilgili karakterlere gösterme, dięer karakterlere gizleme kodları yerleřtirmelisin. Her mevsime ait farklı ve yaratıcı animasyonlar oluřturmalısın.Örneęin tasarıma uygun gökgürültüsü efekti veya yaęmur animasyonu yada kar tanelerinin düşme animasyonu gibi.

6) Kaynaklar (Sources): Powtoon Programı aracılıęıyla hazırlanmıř Mevsimler Oyunu Senaryosu

7) Geribildirim (Feedback): Aferin, Tasarımını güzel yapmıřsın, kodlamada hataların var, dikkat etmen gerekir řeklinde olmuřtur.

Ek E: Gözlem Formları

Tablo 42: Hafta 1: Animasyonlu Davetiye Tasarlama Gözlem Formu

Öğrenci	Adı-Soyadı:	Süre	Başarı			Notlar
Gözlem						
Tarihi:						
Saat:						
Değerlendirme Kriterleri	Kısa	Normal	Uzun	Kötü	Orta	İyi
Scratch'te sahneyi değiştirebilir.						
Scratch'te yeni karakter ekleyebilir.						
Scratch'te bir karaktere farklı kostümler ekleyebilir						
Scratch'te karakterin boyutunu değiştirebilir.						
Hareket kod bloğunu kullanabilir.						
Kontrol ve Olaylar kod bloklarını kullanabilir.						
Görünüm kod bloğunu kullanarak, eklenen karakterlere kostüm değişikliği yaptırabilir.						
Ses kod bloğunu kullanabilir.						

Tablo 43: Hafta 2: Etkileşimli Piyano Oyunu Gözlem Formu

Öğrenci	Adı-Soyadı:	Süre	Başarı			Notlar
Gözlem						
Tarihi:	Saat:					
Değerlendirme Kriterleri	Kısa	Normal	Uzun	Kötü	Orta	İyi
Scratch'te sahneyi değiştirebilir.						
Scratch'te yeni karakter ekleyebilir.						

Scratch'te bir karaktere farklı kostümler ekleyebilir
Scratch'te karakterin boyutunu değiştirebilir.
Hareket kod bloğunu kullanabilir.
Kontrol ve Olaylar kod bloklarını kullanabilir.
Görünüm kod bloğunu kullanarak, eklenen karakterlere kostüm değişikliği yaptırabilir.
Ses kod bloğunu kullanabilir.

Tablo 44: Hafta 3: Labirent Oyunu Gözlem Formu

Öğrenci Adı-Soyadı:	Süre	Başarı			Notlar
		Kısa	Normal	Uzun	
Gözlem Tarihi:					
Saat:					
Değerlendirme Kriterleri					
Scratch'te sahneyi değiştirebilir.					
Scratch'te yeni karakter ekleyebilir.					
Scratch'te bir karaktere farklı kostümler ekleyebilir					
Scratch'te karakterin boyutunu değiştirebilir.					
Hareket kod bloğunu kullanabilir.					
Kontrol ve Olaylar kod bloklarını kullanabilir.					

Görünüm kod bloğunu kullanarak, eklenen karakterlere kostüm değişikliği yaptırabilir. Ses kod bloğunu kullanabilir. Algılama kod bloğunu kullanabilir.

Tablo 45: Hafta 4: Hediye Paketi Toplama Oyunu Gözlem Formu

Öğrenci	Adı-Soyadı:	Süre	Başarı			Notlar	
Gözlem Tarihi:	Saat:						
Değerlendirme Kriterleri		Kısa	Normal	Uzun	Kötü	Orta	İyi
Scratch'te sahneyi değiştirebilir.							
Scratch'te yeni karakter ekleyebilir.							
Scratch'te bir karaktere farklı kostümler ekleyebilir							
Scratch'te karakterin boyutunu değiştirebilir.							
Hareket kod bloğunu kullanabilir.							
Kontrol ve Olaylar kod bloklarını kullanabilir.							
Görünüm kod bloğunu kullanarak, eklenen karakterlere kostüm değişikliği yaptırabilir.							
Ses kod bloğunu kullanabilir.							
Algılama kod bloğunu kullanabilir. Değişkenler kod bloğunu kullanabilir.							
Operatörler kod bloğunu kullanabilir.							

Tablo 46: Hafta 5: Bilgi Yarışması Oyunu Gözlem Formu

Öğrenci	Adı-Soyadı:	Süre	Başarı			Notlar	
Gözlem							
Tarihi:		Saat:					
Değerlendirme Kriterleri		Kısa	Normal	Uzun	Kötü	Orta	İyi
Scratch'te sahneyi değiştirebilir.							
Scratch'te yeni karakter ekleyebilir.							
Scratch'te bir karaktere farklı kostümler ekleyebilir							
Scratch'te karakterin boyutunu değiştirebilir.							
Hareket kod bloğunu kullanabilir.							
Kontrol ve Olaylar kod bloklarını kullanabilir.							
Görünüm kod bloğunu kullanarak, eklenen karakterlere kostüm değişikliği yaptırabilir.							
Ses kod bloğunu kullanabilir.							
Algılama kod bloğunu kullanabilir.							
Operatörler kod bloğunu kullanabilir.							

Tablo 47: Hafta 6: Balon Patlatma Oyunu Gözlem Formu

Öğrenci	Adı-Soyadı:	Süre	Başarı			Notlar	
Gözlem							
Tarihi:		Saat:					
Değerlendirme Kriterleri		Kısa	Normal	Uzun	Kötü	Orta	İyi
Scratch'te sahneyi değiştirebilir.							
Scratch'te yeni karakter ekleyebilir.							
Scratch'te bir karaktere farklı kostümler ekleyebilir							

Scratch'te karakterin boyutunu deęiřtirebilir.
Hareket kod bloęunu kullanabilir.
Kontrol ve Olaylar kod bloklarını kullanabilir.
Görünüm kod bloęunu kullanabilir.
Ses kod bloęunu kullanabilir.
Algılama kod bloęunu kullanabilir.
Operatörler kod bloęunu kullanabilir.

Tablo 48: Hafta 7: Matematik 4 İşlem Resim Bulmaca Oyunu Gözlem Formu

Öęrenci	Adı-Soyadı:	Süre	Bařarı			Notlar	
Gözlem Tarihi:	Saat:						
Deęerlendirme Kriterleri		Kısa	Normal	Uzun	Kötü	Orta	İyi
Scratch'te sahneyi deęiřtirebilir.							
Scratch'te yeni karakter ekleyebilir.							
Scratch'te bir karaktere farklı kostümler ekleyebilir							
Scratch'te karakterin boyutunu deęiřtirebilir.							
Hareket kod bloęunu kullanabilir.							
Kontrol ve Olaylar kod bloklarını kullanabilir.							
Görünüm kod bloęunu kullanarak, eklenen karakterlere kostüm deęiřiklięi yaptırabilir.							
Ses kod bloęunu kullanabilir.							
Algılama kod bloęunu kullanabilir.							

Operatörler kod bloğunu kullanabilir.

Tablo 49: Hafta 8: Tuğla Kıırma Oyunu Gözlem Formu

Öğrenci	Adı-Soyadı:	Süre	Başarı	Notlar		
Gözlem	Tarihi:	Saat:				
Değerlendirme Kriterleri	Kısa	Normal	Uzun	Kötü	Orta	İyi
Scratch'te sahneyi değiştirebilir.						
Scratch'te yeni karakter ekleyebilir.						
Scratch'te bir karaktere farklı kostümler ekleyebilir						
Scratch'te karakterin boyutunu değiştirebilir.						
Hareket kod bloğunu kullanabilir.						
Kontrol ve Olaylar kod bloklarını kullanabilir.						
Görünüm kod bloğunu kullanarak, eklenen karakterlere kostüm değişikliği yaptırabilir.						
Ses kod bloğunu kullanabilir.						
Algılama kod bloğunu kullanabilir.						
Operatörler kod bloğunu kullanabilir.						
Değişkenler kod bloğunu kullanabilir.						

Tablo 50: Hafta 9: Trafik Kuralları Oyunu Gözlem Formu

Öğrenci	Adı-Soyadı:	Süre	Başarı			Notlar	
Gözlem							
Tarihi:		Saat:					
Değerlendirme Kriterleri		Kısa	Normal	Uzun	Kötü	Orta	İyi
Scratch'te sahneyi değiştirebilir.							
Scratch'te yeni karakter ekleyebilir.							
Scratch'te bir karaktere farklı kostümler ekleyebilir							
Scratch'te karakterin boyutunu değiştirebilir.							
Hareket kod bloğunu kullanabilir.							
Kontrol ve Olaylar kod bloklarını kullanabilir.							
Görünüm kod bloğunu kullanarak, eklenen karakterlere kostüm değişikliği yaptırabilir.							
Ses kod bloğunu kullanabilir.							
Algılama kod bloğunu kullanabilir.							
Operatörler kod bloğunu kullanabilir.							
Değişkenler kod bloğunu kullanabilir.							

Tablo 51: Hafta 10: Mevsimler Oyunu Gözlem Formu

Öğrenci	Adı-Soyadı:	Süre	Başarı			Notlar	
Gözlem							
Tarihi:	Saat:						
Değerlendirme Kriterleri		Kısa	Normal	Uzun	Kötü	Orta	İyi
Scratch'te sahneyi değiştirebilir.							
Scratch'te yeni karakter ekleyebilir.							
Scratch'te bir karaktere farklı kostümler ekleyebilir							
Scratch'te karakterin boyutunu değiştirebilir.							
Hareket ve Ses kod bloğunu kullanabilir							
Kontrol ve Olaylar kod bloklarını kullanabilir.							
Görünüm kod bloğunu kullanarak, eklenen karakterlere kostüm değişikliği yaptırabilir.							

Ek F: Etik Kurul Raporu



T.C.
ONDOKUZ MAYIS ÜNİVERSİTESİ
SOSYAL VE BEŞERİ BİLİMLER ETİK KURUL KARARLARI

KARAR TARİHİ	TOPLANTI SAYISI	KARAR SAYISI
22.04.2016	4	2016/ 55

KARAR NO: Eğitim Bilimleri Enstitüsü Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü
2016 – 55 yüksek lisans öğrencisi Emine KADİN'in "Hedef Bazlı Senaryoların Kullanılmasının Öğrencilerin Programlama Performanslarının Gelişimi Üzerine Etkisi" konulu anket çalışması okunarak görüşüldü.

Eğitim Bilimleri Enstitüsü Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü yüksek lisans öğrencisi Emine KADİN'in "Hedef Bazlı Senaryoların Kullanılmasının Öğrencilerin Programlama Performanslarının Gelişimi Üzerine Etkisi" konulu anket çalışmasının kabulüne oybirliği ile karar verildi.

ASLI GİBİDİR.

Ek G: Meb İzin Onayı



T.C.
SAMSUN VALİLİĞİ
İl Millî Eğitim Müdürlüğü

Sayı : 27485554-605.01-E.21114570
Konu : Emine KANDIN'in Uygulama İzni Hk.

06.11.2018

DAĞITIM YERLERİNE

- İlgi : a) Millî Eğitim Bakanlığı Yenilik ve Eğitim Teknolojileri Genel Müdürlüğünün 22/08/2017 tarihli ve 35558626-10.06.01-E. 12607291 - 2017/25 sayılı Genelgesi,
b) Samsun, Ondokuzmayıs Üniversitesi Eğitim Bilimleri Enstitüsü Müdürlüğünün 15.10.2018 tarih ve 42301062-100-E.23141 sayılı yazısı.

Ondokuzmayıs Üniversitesi Eğitim Bilimleri Enstitüsü Müdürlüğü Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı Yüksek Lisans Programı Öğrencisi Emine KANDIN'in ilimiz, ██████ İlçesinde bulunan ██████ Ortaokulunda okuyan öğrencilere yönelik " Kodlama Öğretiminde Hedef Bazlı Senaryo Kullanımı " başlıklı tez çalışması yapmak istediğine ilişkin ilgi (b) yazı ve ekleri, ilgi (a) genelgeye göre incelenmiş ve komisyon tarafından uygun görülmüştür.

Söz konusu çalışmanın komisyon kararı doğrultusunda, anket çalışma sonuçlarını çalışmayı yapan kişi tarafından raporlanarak, Müdürlüğümüz Ar-Ge Birimine gönderilmesine dikkat edilerek, Türkiye Cumhuriyeti Anayasası, Millî Eğitim Temel Kanunu ile Türk Millî Eğitiminin genel amaçlarına uygun olarak, ilgili yasal düzenlemelerde belirtilen ilke, esas ve amaçlara aykırılık teşkil etmeyecek şekilde, duyurusu ve denetimi ilçe millî eğitim müdürlüğünüz tarafından gerçekleştirilmek üzere okul müdürlüğü sorumluluğunda, eğitim-öğretimi aksatmadan gönüllük esasına bağlı olarak yapılmasının sağlanması hususunda;

Bilgilerinizi ve gereğini rica ederim.

Coşkun ESEN
Vali a.
İl Millî Eğitim Müdürü

- Ekler :
1- İlgi (b) dilekçe ve ekleri (1 sayfa)
2-06.11.2018 tarihli komisyon kararı (13 sayfa)

DAĞITIM:

Gereği:
Atakum İlçe Kaymakamlığına
(İlçe Millî Eğitim Müdürlüğü)

Bilgi:
19 Mayıs Üniversitesi
Eğitim Bilimler Enstitüsü Müd.

Ek H: 5. Sınıf Güncellenen Bilişim Teknolojileri ve Yazılım Dersi müfredatı kazanımları

Tablo 52: 5. Sınıf Güncellenen Bilişim Teknolojileri ve Yazılım Dersi müfredatı kazanımları (<http://www.e-mufredat.meb.gov.tr>)

5. ÜNİTE: BT.5.5. PROBLEM ÇÖZME VE PROGRAMLAMA 5.SINIFLAR		
	Kazanımlar	Konular
BT.5.5.1.1.	Günlük hayatta karşılaştığı problemlere çözüm önerileri getirir.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.2.	Verilen bir problemi uygun adımları kullanarak çözer.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.3.	Problem çözmede temel kavramları tanımlayarak problem türlerini açıklar.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.4.	Problem çözme sürecinde takip edilmesi gereken adımları fark eder.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.5.	Verilen bir problemi analiz eder.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.6.	Problemi çözmek için gerekli değişken, sabit ve işlemleri açıklar.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.7.	Problem çözümünde kullanılabilecek operatörlere örnek verir.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.8.	Problem çözümünde ifade ve eşitliklere örnek verir.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.9.	Problem çözümünde işlem önceliğine örnek verir.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.10.	Verilen bir problemin çözümünde operatörleri kullanır.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.11.	Verilen bir problemde ifade ve eşitlikleri kullanarak çözüm üretir.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.12.	Algoritma kavramını açıklar.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.13.	Bir problemin çözümü için algoritma geliştirir.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.14.	Akış şeması bileşenlerini ve işlevlerini açıklar.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.15.	Bir algoritma için akış şeması çizer.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları

Tablo 52: 5. Sınıf Güncellenen Bilişim Teknolojileri ve Yazılım Dersi müfredatı kazanımları (<http://www.e-mufredat.meb.gov.tr>) (devamı)

BT.5.5.1.16.	Bir algoritmayı test ederek hataları ayıklar.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.1.17.	Matematik ve bilgisayar bilimi arasındaki ilişkiyi tespit eder.	BT.5.5.1. Problem Çözme Kavramları ve Yaklaşımları
BT.5.5.2.1.	Programlamayla ilgili temel kavramları açıklar.	BT.5.5.2. Programlama
BT.5.5.2.2.	Blok tabanlı programlama aracının arayüzünü ve özelliklerini tanıır.	BT.5.5.2. Programlama
BT.5.5.2.3.	Blok tabanlı programlama ortamında sunulan hedeflere ulaşmak için doğru algoritmayı oluşturur.	BT.5.5.2. Programlama
BT.5.5.2.4.	Doğrusal mantık yapısını açıklar.	BT.5.5.2. Programlama
BT.5.5.2.5.	Doğrusal mantık yapısını kullanan algoritmalar geliştirir.	BT.5.5.2. Programlama
BT.5.5.2.6.	Karar yapısını ve işlevlerini açıklar.	BT.5.5.2. Programlama
BT.5.5.2.7.	Karar yapıları içeren algoritmalar geliştirir.	BT.5.5.2. Programlama
BT.5.5.2.8.	Döngü yapısını ve işlevlerini açıklar.	BT.5.5.2. Programlama
BT.5.5.2.9.	Döngü yapısı içeren algoritmalar oluşturur.	BT.5.5.2. Programlama
BT.5.5.2.10.	Farklı yapılar için oluşturduğu algoritmaların sonucunu yordayarak hatalarını ayıklar.	BT.5.5.2. Programlama

ÖZGEÇMİŞ

Emine KANDİN 06.06.1991 tarihinde Eskişehir’de doğdu. Bilecik /Bozüyük Mustafa Şeker Anadolu Lisesi’ni bitirdikten sonra Ondokuz Mayıs Üniversitesi Eğitim Fakültesi’nden 2013 yılında mezun oldu. 2014 yılında OMÜEBE Bilgisayar ve Öğretim Teknolojileri Eğitimi Yüksek Lisans programına başladı. Mezuniyetinden bu yana Samsun Kavak ilçesinde Rıdvan Çelikel Fen Lisesinde Bilişim Teknolojileri Öğretmeni olarak görev yapan KANDİN, orta derecede İngilizce bilmektedir. Temel ilgi alanları Bilişim Teknolojileri, Programlama Dilleri, Robotik Matematik , Keman, Piyano, Spor ve Sanat olan KANDİN daha sonraki dönemde Bilgisayar Ve Öğretim Teknolojileri Eğitimi alanında yada Ölçme Değerlendirme alanında Doktora programına devam etmek istemektedir.

İletişim Bilgileri

E mail: emine.kandin@gmail.com

eminekandin@windowslive.com

Telefon : 05432138662