



**T.C.
ULUDAĞ ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ANA BİLİM DALI**

**ÖĞRETMEN ADAYLARININ ALİCE DENEYİMİ: 3B ORTAMDA
PROGRAMLAMA**

YÜKSEK LİSANS TEZİ

Emrah KAYABAŞI

BURSA

2016



T.C.
ULUDAĞ ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ANA BİLİM DALI

ÖĞRETMEN ADAYLARININ ALİCE DENEYİMİ: 3B ORTAMDA
PROGRAMLAMA

YÜKSEK LİSANS TEZİ

Emrah KAYABAŞI

Danışman
Yrd. Doç. Dr. Semiral ÖNCÜ

BURSA

2016

BİLİMSEL ETİĞE UYGUNLUK

Bu çalışmadaki tüm bilgilerin akademik ve etik kurallara uygun bir şekilde elde edildiğini beyan ederim.

Emrah KAYABAŞI

03/02/2016



YÖNERGEYE UYGUNLUK ONAYI

“3B Ortamda Programlama Öğretimi Hakkında Öğretmen Adayı Görüşlerinin Değerlendirilmesi” adlı Yüksek Lisans tezi, Uludağ Üniversitesi Eğitim Bilimleri Lisansüstü Tez Önerisi ve Tez Yazma Yönergesi’ne uygun olarak hazırlanmıştır.

Tezi Hazırlayan

Emrah KAYABAŞI

Danışman

Yrd. Doç. Dr. Semiral ÖNCÜ

Bilgisayar ve Öğretim Teknolojileri Eğitimi ABD Başkanı

Prof. Dr. Aysan ŞENTÜRK

T.C.
ULUDAĞ ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜNE

Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana bilim dalında 801390201 numaralı Emrah Kayabaşı'nın hazırladığı “Öğretmen Adaylarının Alice Deneyimi: 3B Ortamda Programlama” konulu Yüksek Lisans çalışması ile ilgili tez savunma sınavı, 03/02/2016 günü 11:00 – 12:00 saatleri arasında yapılmış, sorulan sorulara alınan cevaplar sonunda adayın tezinin (**başarılı / başarısız**) olduğuna (**oy birliği / oy çokluğu**) ile karar verilmiştir.

Üye
Tez Danışmanı ve
Sınav Komisyonu Başkanı
Yrd. Doç. Dr. Semiral ÖNCÜ
Uludağ Üniversitesi

Üye
Yrd. Doç. Dr. Rasim BAŞAK
Uludağ Üniversitesi

Üye
Yrd. Doç. Dr. Yavuz SAMUR
Bahçeşehir Üniversitesi

Önsöz

Herkesin hayatında yer etmiş, onu bir adım ileri götüren ve yanlış yönde gittiğini anlamadığı zamanlarda bile onu doğru yola sokan bir öğretmeni olmuştur. Bazıları bu konuda daha şanslı olur ve bu sayı bir kişiden onlarcasına çıkabilir. İlk olarak, yüksek lisans ve araştırma görevliliğim süresince bana desteğini esirgemeyen değerli öğretmenim Yrd. Doç. Dr. Semiral Öncü'ye teşekkürlerimi sunarım.

Başta kişiliği ve olaylara analitik bakış açısıyla her zaman saygı duyduğum Prof. Dr. Aysan Şentürk olmak üzere tüm Bilgisayar ve Öğretim Teknolojileri Eğitimi ailesine (akademisyenler ve öğrencilerine) teşekkür ederim. Beni ben yapan, bu yaşıma kadar eşitlik, özgürlük ve adalet duygularını bana aşıl原因 ve sayıları onları bulan tüm öğretmenlerime saygılarımı sunarım. Ayrıca, çalışmalarında bana destek olan Burcu Yılmaz'a, Esmâ Aksoy'a ve Cem Türe'ye teşekkür ederim. Değerli görüşlerini benden esirgemeyen, her zaman yanımda olan ve çalışmalarım süresince eğlenceli zamanlar geçirmemi sağlayan Araş. Gör. Adem Alemdar'a, Araş. Gör. Büşra Görkaş'a, Araş. Gör. Can Deniz Güngörmüş'e, Cem Bora Güngörmüş'e, Erdem Alemdar'a ve Murat Gencan'a minnetlerimi sunarım.

Bu çalışma, Türkiye'deki okullarda programlama öğretimine yeni bir vizyon açabileceğine inandığım Alice yazılımına ilişkin öğretmen adaylarının görüşlerinin değerlendirilmesi ve Alice ile onların öz yeterlilik algılarında oluşabilecek muhtemel değişimleri saptamak amacıyla gerçekleştirilmiştir. Uludağ Üniversitesi, Bilgisayar ve Öğretim Teknoloji Öğretmenliğinde yapılan araştırmanın sonuçlarının üniversitelerde ve ortaöğretimde verilen programlama derslerine alternatif bir öğretim teknolojisini sunma potansiyeline sahip olduğuna inanılmaktadır.

Emrah Kayabaşı

Özet

Yazar	: Emrah KAYABAŞI
Üniversite	: Uludağ Üniversitesi
Ana Bilim Dalı	: Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı
Bilim Dalı	: Bilgisayar ve Öğretim Teknolojileri Eğitimi Bilim Dalı
Tezin Niteliği	: Yüksek Lisans Tezi
Sayfa Sayısı	: XIII + 142
Mezuniyet Tarihi	: 09 / 02 /2016
Tez	: Öğretmen Adaylarının Alice Deneyimi: 3B Ortamda Programlama
Danışmanı	: Yrd. Doç. Dr. Semiral ÖNCÜ

ÖĞRETMEN ADAYLARININ ALİCE DENEYİMİ: 3B ORTAMDA PROGRAMLAMA

21. yüzyıl eğitim dünyasıyla birlikte gelen yeniliklere ve değişime ayak uydurma eğilimi programlama eğitimi alanını doğrudan etkilemektedir. Örneğin, öğretim teknolojilerindeki çeşitliliğin artması beraberinde bu yazılımların değerlendirilmesi ihtiyacını getirmektedir. Alice de programlama öğretiminde kullanılan bir yazılım olmakla birlikte, scratch, greenfoot ve khanacademy gibi platformlarda yapılan programlama öğretiminin aksine üç boyutlu bir ortamı kullanıcılarına sağlamaktadır. Bu çalışma, öğretmen adaylarının Alice yazılımına ilişkin deneyimlerinin değerlendirilmesi ve öz yeterlilik algılarındaki değişimin saptanması amacıyla gerçekleştirilmiştir.

Araştırmanın örneklemini Uludağ Üniversite'si, Bilgisayar ve Öğretim Teknolojileri Eğitimi bölümünde, araştırmaya katılmayı kabul eden 24 öğretmen adayı oluşturmaktadır. Gömülü (embedded) desenin kullanıldığı araştırmada, nitel ve nicel araştırma desenlerinden yararlanılmıştır. Katılımcıların görüşleri durum çalışması

deseni ile deęerlendirilmiř ve öz yeterlilik algılarındaki deęiřim basit deneysel desen kullanılarak saptanmıřtır.

Öęretmen adayları Alice ile yapılan programlama öęretimi sayesinde öęrencilerin 3B dünya tasarımı deneyimi geliřtireceęi, programlama derslerine olan ilgi ve katılımlarının artacaęı ve ders başarılarının olumlu yönde etkileneceęini düşünmektedir. Öęrenilen bilgilerin kalıcılıęı ve programlama derslerine tutumlarını da etkiledięi görölmektedir. Ayrıca, katılımcılar Alice’i kullananların liderlik, yaratıcı düşünme ve problem çözme gibi 21. yüzyıl becerilerinin geliřeceęini de düşünmektedir. Öęretmen adaylarına göre, Alice eğlenerek ve uygulayarak öęrenme imkânını kullanıcılarına sunmaktadır. İerisinde barındırdıęı üç boyutlu dünyası ve nesnelere ile programlamaya görsellik katmakla birlikte kullanıcılara hayal gücünün kullanımı – somutlařtırma imkânını da saęlar. Alice’de karřılařılabilecek güçlüklerle ve programlama öęretimine saęladığı katkılardan da bahseden öęretmen adaylarının ön test ve son test sonuçları karřılařtırıldıęında öz yeterlilik algılarında istatistiksel olarak anlamlı bir farklılık söz konusudur. Sonuç olarak, Alice bu haliyle özellikle programlamaya giriş derslerinde kullanılmak üzere Türkiye’de alternatif bir öęretim teknolojisi olma potansiyeline sahiptir.

Anahtar sözcükler: 3B dünya, Alice, animasyon, eğlenerek öęrenme, iřbirlikli öęrenme, öz yeterlilik, programlama

Abstract

Author : Emrah Kayabaşı
University : Uludag University
Field : Computer Education and Instructional Technology
Branch : Computer Education and Instructional Technology
Degree Awarded : Master
Page Number : XIII + 142
Degree Date : 09 / 02 / 2016
Thesis : Prospective Teachers' Experiences On Alice: Programming In 3D Environment
Supervisor : Assist. Prof. Semiral ÖNCÜ

PROSPECTIVE TEACHERS' EXPERIENCES ON ALICE: PROGRAMMING IN 3D ENVIRONMENT

Tendency to adaption to innovations and change in 21st century education affects the field of programming education directly. For instance, increase in the variety of instructional technologies brings with it the necessity to evaluate these software. Contrary to programming instruction platforms like scratch, greenfoot and khanacademy, Alice software provides the users a 3D environment. This study was conducted to evaluate prospective teachers' experiences related to Alice software and determine the change on their perceptions about self-efficacy.

The sample of this research consists of 24 prospective teachers from Computer Education and Instructional Technologies Department in Uludağ University accepting participation. Designed as an embedded research, this study was comprised of both qualitative and quantitative research designs. The opinions of participants were evaluated via case study design and the change on their self-efficacy perceptions were determined using weak experimental design.

Prospective teachers stated that the students will be able to develop experiences on 3D world design and Alice will provide opportunities to increase their interest and participation in programming lessons. Retention of knowledge and user's attitudes will also be affected positively. Moreover, participants think that 21. Century skills like creative thinking, leadership and problem solving of Alice users will improve.

According to participants, Alice gives its users opportunities like learning with fun and learning by doing. Moreover, Alice provides the users with using their own imagination- concretization by integrating visuals into programming through 3D world and items involved. In addition, prospective teachers stated their opinions on possible difficulties that may be encountered on Alice and its contributions on programming instructions. Statistical analyses were conducted and significant differences were found on participants' perception of self-efficacy about programming when their pre and post test results were compared. As a result, Alice stands as a potential to be used especially in the introductory programming courses in Turkey as an alternative instructional technology.

Keywords: 3D world, Alice, animation, collaborative learning, learning with fun, programming, self-efficacy

İçindekiler

	Sayfa No
ÖNSÖZ	iv
ÖZET	v
ABSTRACT	vii
İÇİNDEKİLER	ix
ŞEKİLLER LİSTESİ	xi
TABLolar LİSTESİ.....	xii
KISALTMALAR LİSTESİ.....	xiii
1. BÖLÜM: GİRİŞ.....	1
1.1. Problem Durumu	6
1.2. Amaç	9
1.3. Önem	10
1.4. Varsayımlar	11
1.5. Sınırlılıklar	11
1.6. Tanımlar	12
2. BÖLÜM: ALANYAZIN.....	13
2.1. Bilgisaymsal Düşünme	13
2.2. Programlamada Görsellik.....	16
2.3. Akılda Kalıcılık ve Akademik Başarı	18
2.4. Eğlenerek Öğrenme.....	19
2.5. Alice’de Karşılalabilecek Güçlükler	20
2.6. İşbirlikli Öğrenme	23
2.7. Programlamada Öz yeterlilik Algısı.....	25
3. BÖLÜM: YÖNTEM.....	28
3.1. Araştırmanın Modeli	28
3.2. Çalışma Grubu (Örnekleme).....	31
3.3. Bağlam	32
3.3.1. Alice’in özellikleri.....	33
3.3.2. Üç boyutlu programlama dersinin içeriği ve kapsamı.....	37
3.4. Veri Toplama Araçları	38
3.5. Verilerin Toplanması ve Çözümlemesi.....	41
3.6. Güvenirlik	46
4. BÖLÜM: BULGULAR.....	55

4.1. Okullarda Alice ile Programlama Öğretimi	55
4.1.1. Eğlenerek öğrenme.	57
4.1.2. Programlamaya görsellik katma	60
4.1.3. Uygulayarak öğrenme	62
4.1.4. Hayal gücünün kullanılması – somutlaştırma	64
4.1.5. Diğer görüşler	66
4.2. Alice Yazılımında Karşılaşılabilecek Güçlükler	68
4.2.1. Nesne konumlandırma	69
4.2.2. Dil	70
4.2.3. Performans sorunları	71
4.2.4. Kamera kullanımı	72
4.2.5. Nesne galerisinin zenginleştirilmesi	73
4.2.6. Metotlar	73
4.2.7. Diğer görüşler	74
4.3. Programlama Öğretiminde Alice Yazılımının Güçlü Yönleri	75
4.3.1. Görsellik	75
4.3.2. Sürükle-bırak yapısı	78
4.3.3. Programlama kavramları	80
4.3.4. Hazır kodların yazım hatalarını önlemesi.....	81
4.3.5. Grup / bireysel çalışma	82
4.4. Programlama Konusunda Nitelikli Öğretmen Yetiştirme.....	86
4.4.1. Alice ile programlama eğitimi	86
4.4.2. Diğer branşların durumu	87
4.4.3. Kalıcılık	89
4.5. Öz Yeterlilik Algısı	91
5. BÖLÜM: TARTIŞMA VE ÖNERİLER.....	96
5.1. Programlama Öğretiminde Üç Boyutlu Dünyaların Kullanımı	96
5.1.1. Alice ile programlama öğretimi.....	96
5.1.2. Alice’te karşılaşılabilecek güçlükler	101
5.1.3. Alice’in güçlü yönleri.....	103
5.1.4. Alice’in nitelikli öğretmen yetiştirmeye katkıları	105
5.2. Öğretmen Adaylarının Programlamaya İlişkin Öz Yeterlilik Algıları.....	106
KAYNAKÇA.....	109
EKLER.....	119
Ek 1: Görüşmeler ve Açık Uçlu Anket Uygulanırken Kullanılan Form.....	119

Ek 2: Programlamaya İlişkin Öz Yeterlilik Anketi.....	121
Ek 3: Kod Tablosu	123
Ek 4: İzin Yazısı.....	126
Ek 5: Etkinlik Ödevi 1-2-3-4-5	127
Ek 6: Ders İçeriği ve Öğrenme Kazanımları.....	140
ÖZGEÇMİŞ	141



Şekiller Listesi

<i>Şekil</i>		<i>Sayfa</i>
1	21. yüzyıl eğitim dünyasında yeni denge	5
2	Hata ekranı	21
3	Yürüme metodunun oluşturulmasına ilişkin örnek kodlar.....	22
4	Gömülü desene göre gerçekleştirilen araştırmanın aşamaları	30
5	3B dünyadaki balina nesnesi.....	35
6	Kedi nesnesinin özellikler paneli	36
7	Yazılan say kodu ile balina nesnesinden konuşma baloncuğu çıkması	37
8	Yeniden değerlendirme sonucunda ortaya çıkan kodlar ve aralarındaki ilişkiler	56
9	Alice’de karşılaşılabilecek güçlükler	69
10	Programlama öğretiminde Alice’in güçlü yönleri	75
11	Programlama konusunda nitelikli öğretmen yetiştirme	86

Tablolar Listesi

<i>Tablo</i>		<i>Sayfa</i>
1	Grup çalışması ile bireysel çalışmanın farklı değişkenlere göre kıyaslanması.....	25
2	Verilerin toplanması süreci	42
3	Verilerin içerisinde rastgele seçilen cevaplara ilişkin kodlar	44
4	Araştırmacılara gönderilen excel tablosunun örneği	47
5	Üniversitelerde Alice ile programlama öğretimi	48
6	İlk ve orta dereceli okullarda Alice ile programlama öğretimi.....	49
7	Alice yazılımında karşılaşılabilecek güçlükler	50
8	Programlama öğretiminde Alice yazılımının güçlü yönleri.....	51
9	Programlama öğretiminde nitelikli öğretmen yetiştirme	52
10	Katılımcıların grup veya bireysel çalışmaya ilişkin tercihleri	83
11	Programlamaya ilişkin öz yeterlilik ölçeği maddeleri	92
12	Katılımcıların farklı değişkenlere göre betimsel istatistikleri.....	93
13	Wilcoxon bağımlı iki örneklem testi: ön-test ile son-test sonuçlarının genel toplamları	94
14	Wilcoxon bağımlı iki örneklem testi: ön-test ile son-test sonuçlarının genel toplamların karşılaştırılması.....	94
15	Öz yeterlilik algılarındaki soru bazlı değişimler: Z ve p değerleri	95

Kısaltmalar Listesi

CMU: Carnegie Mellon Üniversitesi

EBA: Eğitim Bilişim Ağı

FATİH: Fırsatları Arttırma ve Teknolojiyi İyileştirme Hareketi

MEB: Milli Eğitim Bakanlığı

T.B.M.M: Türkiye Büyük Millet Meclisi

TÜBİTAK: Türkiye Bilimsel ve Teknolojik Araştırma Kurumu

3B: Üç Boyutlu



1. Bölüm

Giriş

Sanayi çağından bilgi çağına geçilirken, 21. yüzyılın eğitim dünyasında bir paradigma değişimi söz konusudur ve dünya üzerindeki pek çok ülke bu değişime adapte olmaya çalışmaktadır. Bu paradigma değişimi, dünya sanayisi, eğitimi politikası, insanların yeni bilgileri öğrenme şekli, öğrenilen bilginin doğası ve eğitim sistemlerindeki problemler gibi alt başlıklarda incelenmektedir. Bu değişim, beraberinde insanın yaşamını sürdürebilmesi için ihtiyaç duyduğu yeterlilikler ve becerilerde de bir değişimi beraberinde getirmektedir. Örneğin, eskiden emek gücüyle ailesini geçindirebilen bir insan, günümüzde çoğu kez bilgiye nasıl erişeceğini, onu nasıl kullanacağını ve gerekli hallerde edindiği bilgileri ne şekilde değiştireceğini bilmesi gerekmektedir. Bilgi ve iletişim teknolojilerinin kullanımı günlük ve ekonomik yaşamda önemli bir yere sahiptir. Trilling ve Fadel (2009), 21. yüzyılda insanların ihtiyaç duydukları becerileri dijital medya okuryazarlığı, yenilik getirme, yaratıcılık, kritik düşünme, problem çözme ve yansıtıcı düşünme becerisi şeklinde sıralamaktadır.

Ranguelov, Howard, Dalfrith ve Noorani'ye (2011) göre Avrupa'daki insanların ezici bir çoğunluğu, özellikle genç nesil, farklı amaçlarla da olsa günlük yaşamın bir parçası olarak bilgisayar kullanmaktadır. Ayrıca, Avrupa'daki ülkelerin düzenledikleri müfredatları ile okullarda dijital medya okuryazarlığı, öğrenme ve yenilik yapma becerisi, yaratıcılık, problem çözme, iletişim gibi becerilerin öğrencilere kazandırılması gerektiğinden de bahsetmektedir. Bu becerilerin kazandırılmasında ve geliştirilmesinde ise bilgi ve iletişim teknolojilerinin kullanımını araç olarak göstermektedir. Türkiye'deki ilkokullardan akademik derece veren üniversitelere kadar okullardaki öğrencilere bu becerilerin ne denli kazandırıldığı ise sürekli güncelliğini koruyan bir tartışma konusudur.

Trilling ve Fadel (2009) Amerika'da yaptıkları çalışmada şirketlerin 200 milyar dolar gibi rakamlar harcayarak, öğrenim süreleri boyunca zaten eğitilmiş olan işçilerini hizmet-içi

eđitim programlarında tekrar eđitmeye alıřtıkları sonucuna ulařmıřlardır. zellikle az geliřmiř ve Trkiye gibi geliřmekte olan lkeler aısından bakıldıđında bu ađır bir yk de beraberinde getirmektedir. Bu ykten kurtulabilmek iin, 21. yzyıl eđitim dnyasında, Trkiye’de de đrencilerin okullarda edinmesi gereken yeterlilikler revize edilmiřtir. Peki ya đretmenlerde bu becerileri đrencilerine kazandırmak iin bulunması gereken nitelikler?

đrencilerine 21. yzyıl yeterliklerini kazandıracak đretmenlerin durumuna iliřkin, Milli Eđitim Bakanlıđı (MEB) 2012 yılına ait faaliyet raporunda (2013), đretmen adaylarının niversitelerde nitelikli eđitimi konusunda kaygısını ortaya koymuřtur. Aynı raporda, đretmenlerin sınıf ortamında đretim teknolojilerini kullanabilme becerisine sahip olmadıklarını da belirtmiřtir.

Gagne (2013) đretim teknolojisini, etkili đretimin yapılabilmesi iin eđitmenlerin eldeki araları belirli bir dzen ierisinde kullanması ve eđitmen ile đrenenler arasındaki iletiřimin desteklenmesi sreci olarak tanımlamaktadır. Yani, teknoloji đretimin tasarlanmasını ve aktarımını kolaylařtırmaktadır. Bunun yanı sıra đretim teknolojisi alanının arka planında bilgisayar bilimlerindeki ilerlemeler, iletiřim zerine arařtırmalar, donanımların geliřtirilmesi ve biliřsel psikoloji gibi eđitim pedagojilerinin de katkısı yadsınamaz.

Bilgisayar bilimlerinde son yıllarda yařanan srekli deđiřim, beraberinde yeni bilgi alanları ortaya ıkarmıřtır. rneđin programlama alanı kritik neme sahip alanlardan birisidir ve mfredatı ile đretim yntemi dzenli olarak gncellenmektedir (Malliarakis, Satratzemi, & Xinogalos, 2014). Bu gncellemelerin bir boyutu da programlama đretiminde kullanılan alternatif yazılımlardır. Bunlardan biri olan Alice, bařlangı seviyesinde programlama eđitimi yapabilmek iin Carnegio Mellon niversitesindeki (CMU) Randy Pausch nderliđinde bir ekip tarafından geliřtirilmiř bir đretim teknolojisidir. niversite dzeyinde bilgisayar bilimleri alanında programlamaya giriř derslerinde kullanılan Alice, ilk ve ortaokullardaki programlama derslerinde Amerika’da yaklařık 1700 đretmen tarafından kullanılmaktadır

(Cooper, 2010). Çalışma kapsamında defaten Alice sözcüğü 3B programlama aracı anlamına gelecek şekilde kullanılmıştır. Ayrıca, Alice yazılımı hakkında tanıtıma yöntem bölümündeki bağlamın başlığından ulaşılabilir.

MEB'e bağlı Talim ve Terbiye Kurulu Başkanlığı (2012) tarafından bilişim teknolojileri ve yazılım dersi temel yeterliliklerini belirleyen çerçeve program yayımlanmıştır. Bu program öğrencilerin bilgi ve iletişim teknolojileri konularında kazanmaları gereken nitelikleri dört öğrenme alanı altında toplamıştır. Bu öğrenme alanları sırasıyla, (1) bilişim okur-yazarlığı, (2) bilişim teknolojilerini kullanarak iletişim kurma, bilgi paylaşma ve kendini ifade etme, (3) araştırma, bilgiyi yapılandırma ve işbirlikli çalışma ve (4) problem çözme, programlama ve özgün ürün geliştirme şeklindedir.

Stasko (1990) programlama kodlarının animasyonla canlandırılmış gösteriminin öğrencilere programı anlama, var olan programları değerlendirme ve yeni programlar üretme noktasında yardımcı olduğunu düşünmektedir. Buna ek olarak, piyasadaki programlama kitaplarının çoğu her adımı detaylı olarak göstermekte ve haliyle uzun yazılmaktadır. Alice yazılımı ile ise bağlam (context) temelli programlama öğretilir ve bir konu üzerinde animasyonları, öykü yapıklarını, oyun oluşturmayı kullanarak yapılan öğretim ile programlamayı ve problem çözme yeteneğinin kazandırılmasını sağlar (Cooper, 2010). Yani, şu ana kadar kullanılan programlama öğretimi araçlarından, üç boyutlu olması ile gerçek yaşama ait bir problem durumunu çözmek için programlama mantığını kullanabilmelerinde ve kendilerine özgü bir ürün geliştirmelerinde etkili bir öğretim aracı olabilir. Bu konuda Biju (2013), iki yıl süren araştırma ardından, geleneksel stilde yapılan programlama öğretimine nazaran Alice'in oldukça etkili bir öğretim aracı olduğu sonucuna ulaşmıştır.

Fırsatları Arttırma ve Teknolojiyi İyileştirme Hareketi (FATİH) çalıştay raporunda (Milli Eğitim Bakanlığı, 2012), akademik personele atıfta bulunularak projenin geliştirme ve değerlendirme süreçlerinde aktif rol almaları istenmektedir. Bu noktada, akademik personelin

ders içeriklerinin zenginleştirilmesi, öğretmenlere ders-içi uygulamalarda kullanabilecekleri alternatif uygulamaların sunulması konusunda araştırmalarının artırılması katkı sağlayabilir. Örneğin, fen alanında sınıf veya laboratuvar ortamlarında gerçekleştirilemeyecek veya gerçekleştirilmesi tehlike arz eden konu ve uygulamaların öğretiminde etkileşimli materyaller oluşturulmasına üç boyutlu programlama araçları ile katkı sağlanabilir. Tüm bunlar düşünülecek olursa, üç boyutlu programlamadan sadece farklı bir öğretim metodu olarak faydalanmakla birlikte hazırlanan materyaller, tablet bilgisayarlardan akıllı tahtalara kadar farklı platformda öğretmen ve öğrencilerin kullanımına sunulabilir.

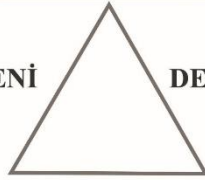
21. yüzyıl eğitim dünyasında problem çözme, yansıtıcı düşünme, yenilik getirme gibi becerileri öğrencilere kazandırabilmek için eğitim felsefesinde ve öğretim tekniklerinde arzulanan değişim Şekil 1’de gösterilmiştir. Örneğin, bu yeni öğrenme yaklaşımlarında değişmez gerçekler ve prensiplerin öğretimi yerine, sorgulanabilir problemler üzerinden dersler işlenmektedir. Buna ek olarak, doğrudan öğretimin yerini etkileşimli öğretime bırakması, sınıf ortamlarının rekabet yerine işbirliğini destekleyecek şekilde oluşturulması gibi prensiplerin önemi artmıştır (Trilling & Fadel, 2009).

Şekil 1

21. yüzyıl eğitim dünyasında yeni denge (Trilling & Fadel, 2009)

GELENEKSEL YAPI	ARZULANAN YAPI
Öğretmen Merkezli	Öğrenci Merkezli
Doğrudan Öğretim	Etkileşimli Öğretim
Bilgi	Beceri
İçerik	Süreç
Temel Beceriler	Uygulamalı Beceriler
Gerçekler ve Prensipler	Sorgulanabilir Problemler
Teori	Uygulama
Müfredat	Projeler
Zamanlanmış	İsteğe Göre
Standartlaştırılmış	Kişiselleştirilmiş
Rekabetçi	İşbirlikli
Sınıf Ortamı	Küresel Toplum
Metinsel	İnternet Tabanlı
Sonuç Değerlendirme	Süreç Değerlendirme
Okul İçin Öğrenme	Yaşam İçin Öğrenme

YENİ DENGİ



Biju'ya (2013) göre yeni nesildeki bireyler akıllı telefonlarla ve bilgisayar oyunları ile çok küçük yaşlarda etkileşim içerisine girmektedir. Bu açıdan bakıldığında, programlama derslerine ilgilerini çekebilmek için, oyunlara benzer bir ara yüz ve teknolojinin kullanımı ihtiyacı söz konusudur.

Lombardi (2007, s.2) öğrencilerin öğretmenlerini dinlemektense, yaparak (aktif) öğrenmeyi her zaman yeğlediklerini belirtmektedir ve “öğrencilerden Amerikan Sivil Savaşını canlandırmaları, tektonik plakaları bir araya getirerek bir volkanın patlamasını sağlamaları veya bir deprem için gereken koşulları oluşturmaları beklenebilir mi?” diye eklemektedir. Sonuç olarak, 21. yüzyılda artık öğrencilerin gerçek yaşamlarıyla doğrudan ilgili bu tip konuların öğretiminde teknoloji, görsellik (visualization) araçları ve farklı gösterim teknikleri fazlasıyla vardır. Deney yapmaktan gerçek yaşam problemlerini çözmeye kadar pek çok aktivite gerçekleştirilebilir hale gelmiştir.

Trilling ve Fadel (2009) alışkanlıklarımızın yanında standartlaştırılmış testlerin, 21. yüzyılda geliştirilmiştir teori ve kuramların uygulanmasına direnç gösteren noktalardan birisi olduğunu söylemektedir. Ve ekleyerek, Barack Obama ve Richard Riley'nin sözlerine yer vermektedir. Dünyanın önde gelen ülkelerinden birisinin lideri olan Barack Obama “ulusumuzun yöneticilerini ve bölgesel eğitim müdürlerini, öğrencilerin testleri yuvarlak içine aldıkları cevapları ölçen değil, problem çözme, kritik düşünme, girişimcilik, yaratıcılık gibi 21. yüzyıl becerilerini ölçebilecek standartlar ve değerlendirme yöntemleri geliştirmeye çağırıyorum” diyerek alternatif değerlendirme yöntemlerine olan ihtiyacı ortaya koymuştur. Benzer şekilde, eski Amerikan Milli Eğitim Bakanı Richard Riley ise “öğrencilerimizi şu an ortaya çıkmamış işlerde çalışmaları, henüz icat edilmemiş teknolojileri kullanmaları ve henüz varlığından bile haberdar olmadığımız problemleri çözmeleri için hazırlıyoruz” ifadesiyle öğrencileri değerlendirmek adına bazı hedefler çizmektedir. Değişikliklere ayak uydurabilecek nesilleri yetiştirebilmek amacıyla, şüphesiz ki öncelikle elimizde olan teknolojilerin kullanımının önemi büyüktür.

1.1. Problem Durumu

Günümüzdeki eğitimde paradigma değişiminden daha önce bahsedilmişti. Bu değişim, beraberinde getirdiği yeniliklere, değişen öğretim araçlarına, eğitim felsefelerine, teknolojik altyapıdaki yeniliklere ve güncellemelere uyum sağlama ihtiyacını da doğurmaktadır. Git gide artan değişimlere ve yeterliklere günümüz eğitimcilerinin ayak uydurmaları ve kendilerini güncel tutmaları için yeni yazılımlar ve öğretim yöntemlerinin sayısı da günden güne artmaktadır.

Fincher ve Utting'e (2010) göre üniversiteler programlamanın derinlemesine çalışıldığı yerlerdir ve bilgisayar eğitimi açısından 21. yüzyıl özel bir önem arz etmektedir. Yani, bilgisayarların yapabilecekleri işlemlerin kapasitesinin muazzam şekilde artması beraberinde, programlamanın sadece üniversitelerde öğretiminin kabullenilmemesi

gerekliliğini de getirmektedir. Eski bilgisayarlar yavaş, pahalı ve sayıca az idi. Günümüzde ise tam tersi bir eğilim göstererek hızlı, ucuz ve kolay ulaşılabilir hale gelmiştir. Wing'in (2006) de belirttiği gibi bugün işlemci hızı ve depolanabilir bilgi kapasitesinin artırılması ile birlikte, basit bir telefon bile 1970'lerdeki bir üniversitenin içindeki bilgisayarların tamamından daha gelişmiş hale gelmiştir. Naughton (2012) oyun geliştirme firmalarının daha çok programcı istihdam etmek istemesi, ülkelerin ileri teknoloji ürünler geliştirmek istemesi ve okullardaki bilgisayar laboratuvarlarının sayısının artırılması gibi taleplere dikkat çekmektedir. Bunun yanı sıra ekleyerek, bilgisayar bilimlerinin, bilgisayarca düşünmenin, programlama bilmenin ilköğretim düzeyinden başlayarak İngiltere'deki tüm okullarda öğrencilere kazandırılması gerektiğinden bahsetmektedir. Yani, tüm bu değişim ve gelişim süreci düşünülecek olursa ilkokul düzeyindeki öğrencilerden başlayarak programlama konusunda kendilerini geliştirebilecek ortamların ve programlama öğretiminin küçük yaşlardan itibaren yapılmasını elzem haline getirmiştir.

Programlamanın doğası ve sınırları hızlı bir şekilde değişmektedir. Yazılım geliştiricilerin sunduğu yeni öğretim teknolojileri, tasarımın ve kodlamanın doğasını gün ve gün değiştirmeye devam etmektedir. Blackwell (2002) bu alandaki en radikal değişimi, eskiden programlama işleri uzmanlar tarafından yerine getirilirken, artık programlama konularında uzman olmayan insanların da bu görevleri yapabilmesi olarak tanımlamaktadır. Örneğin, eskiden HTML kodu yazan bir programcının metni kalın hale getirme işlemini gerçekleştirmek için “ kalınlaştırılacak yazı ” şeklinde bir kod bloğunu metin kodu olarak girmeleri gerekmekteydi. Yeni nesil HTML kodları yazmaya imkân veren programların grafiksel ara yüzü sayesinde yazıyı seçip CTRL+K tuşuna basmak veya ara yüzdeki metin düzenleme araçlarından “K” butonuna tıklamak yeterli olmaktadır.

2000li yıllara gelindiğinde Nesne Tabanlı Programlama (Object Oriented Programming) gelişmeye başlamıştır ve şu an kullanılan programlama dillerinin genelde

nesne tabanlıdır. Doğası itibariyle problemlerin nesnelere ayrıştırılması ve o nesnelere üzerine kodlar yazılmasını içerir. Nesne tabanlı programlama sayesinde işletim sistemleri, karmaşık yazılımların geliştirilmesi, kullanıcı ara yüzlerinin oluşturulması gibi işlemler yapılabilmektedir (Elrad, Filman, & Bader, 2001). Peki böylesine karmaşık programlama dilinin deneyimsiz öğrencilere programlamaya ilk kez başladıklarında sunulması ne kadar doğru?

Programlama öğrenenlere karmaşık gelen geleneksel programlama dillerinin yapılarındaki bu sorunu gidermek için Scratch, Toontalk ve Microsoft small basic gibi uygulamalar kullanılmaktadır (Çatlak, Tekdal, Baz, 2015). Örneğin, Cooper, Dann, ve Pausch (2003a), Amerika’da yaptıkları çalışmada, programlamaya giriş dersinde geleneksel yöntemle yapılmış programlama öğretimiyle, Alice ile yapılanı karşılaştırmışlardır. Alice ile yapılan öğretimdeki öğrencilerin daha başarılı olduğu sonucuna ulaşılmıştır. Bu açıdan bakıldığında, programlama konusunda yetkin bireyler yetiştirebilmek için, Türkiye’de özellikle ilk ve orta dereceli okullarda kullanılan programlama öğretimi araçlarının zenginleştirilmesi fayda sağlayabilir.

Türkiye’de yapılan çalışmalara bakıldığında, Alice yazılımına ilişkin araştırmalar yok denecek kadar azdır. Bunun yanı sıra Amerika’da 1700’ün üstünde okulda kullanılan bu yazılımın ücretsiz olması ve çıktılarının platformdan bağımsız çalıştırılabilmesi (Cooper, Dann, & Pausch, 2010) beraberinde “Alice Türkiye’de programlama öğretiminde neden kullanılmıyor?” sorusunu getirmektedir. Bu çalışma tüm bu nedenlerden dolayı, geleceğin bilişim teknolojileri öğretmenlerinin, üç boyutlu dünyayı kullanıcılarına sunan Alice yazılımına ilişkin görüşlerinin değerlendirilmesi ve programlamaya ilişkin öz yeterlilik algılarındaki değişiklikleri saptamak amacıyla uygulamalı olarak gerçekleştirilmiştir.

1.2. Amaç

Bu çalışmanın amacı, öğretmen adaylarının programlama becerilerine ilişkin öz yeterlilik algılarını saptamak ve Alice yazılımının sağladığı üç boyutlu ortamda programlama öğretimine ilişkin öğretmen aday görüşlerini saptamaktır. Araştırma soruları şu şekilde listelenebilir:

1. Programlama kavramlarının öğretiminde üç boyutlu ortamların kullanımına ilişkin öğretmen adaylarının görüşleri nelerdir?
 - a. Bu çalışma kapsamında elde ettikleri programlama ve 3B dünya tasarımı becerilerinin gelecekte verebilecekleri derslere nasıl etki edeceğini düşünmektedirler?
 - b. Ortaöğretimde programlama öğretimi yapılırken Alice nasıl kullanılabilir? Ortaöğretim öğrencilerinin bu programla öğretimi nasıl karşılayacaklarını düşünüyorlar?
 - c. Alice ortamının kullanımında karşılaştıkları güçlükler nelerdir?
 - d. Alice yazılımının programlama öğretimine sağladığı kolaylıklar nelerdir?
 - e. Alice ortamı, nitelikli öğretmen yetiştirmeye ne gibi faydalar sağlayabilir?
2. Öğretmen adayları programlama konusunda kendilerini ne kadar yeterli hissediyorlar?

2.araştırma sorusuyla test edilen hipotezler aşağıda belirtilmiştir.

- H_0 : Öğretmen adaylarının ön-test ve son-testteki sonuçları arasında pozitif veya negatif farklar toplamı birbirine eşittir. Yani öz yeterlilik algılarında anlamlı bir farklılık yoktur.
- H_a : Öğretmen adaylarının ön-test ve son-testteki sonuçları birbirine eşit değildir. Negatif farkların toplamı, pozitif farkların toplamına göre büyük veya küçüktür. Yani, öz yeterlilik algılarında anlamlı bir farklılık vardır.

1.3. Önem

Türkiye’de son yıllarda eğitim alanında teknolojiye yatırım giderek artmaktadır. FATİH Projesiyle birlikte okullarda tablet bilgisayarların dağıtılması düşünülmektedir. Donanımına yapılan bu yatırımın, tabletlerde ve akıllı tahtalarda kullanılacak yazılımlarla da desteklenmemesi bu konuda bir zafiyet oluşturabilir. Nitekim, materyal ihtiyacını karşılamak amacıyla Eğitim Bilişim Ağı (EBA) her geçen gün kendisini geliştirmekte ve içerisinde Alice gibi yazılımların da yer aldığı alternatif yazılımlar sunulmaktadır. Buna rağmen, çalışmanın başladığı tarihte üç boyutlu programlama araçlarının ilköğretimden yükseköğretime kadar yeterince kullanılmadığından ve kullananların sayılarının az olduğundan söz edilebilir. Özellikle Amerika’da yapılan araştırmalarda Alice’in programlama öğretimine alternatif olduğu göze çarpmaktadır. Ülkemizde de platformdan bağımsız ve ücretsiz olarak sunulmasından dolayı yaygınlaştırılabileceği öngörülmektedir.

FATİH Projesi ve Türkiye Bilimsel ve Teknolojik Araştırma Kurumunun (TÜBİTAK) yürüttüğü çağrı programlarında da görülebileceği gibi nitelikli (teknolojiyi hayatına entegre etmiş) öğretmen yetiştirilmesinde üniversitelere büyük bir sorumluluk düşmektedir. Bu çalışmanın önemi öğretmen adaylarına bu nitelikleri kazandırmak konusunda yapılabilecek çalışmalar için bir ön veri oluşturma potansiyeline sahip olmasıdır.

Programlama öğretiminde, başarısız olan öğrencileri “problem çözmeyi bilmiyorlar” diyerek sınıflandırmak kolaydır. Ama, unutulmamalıdır ki, aynı öğrenciler aslında matematik, geometri gibi derslerde belirli bir seviyeye kadar problem çözmeye başarılıdırlar (Cooper, Dann, Pausch, 2000a). Bu noktada, programlama öğretiminde başarıya ilişkin eldeki öğretim araçlarının değerlendirilmesi önem arz etmektedir.

Üç boyutlu ortamda programlama öğretimine olanak sağlayan araçlardan biri olan Alice’e ilişkin Türkiye’de yapılan çalışmalar azdır. Alice, gerçek yaşam koşullarını üç boyutlu ortamda taklit edebilme becerisiyle hali hazırda kullanılan iki boyutlu Scratch,

Code.org, Greenfoot gibi yazılımların önüne geçmektedir. Özellikle ileride bilişim teknolojileri öğretmeni olma yetisine sahip olan bu araştırmanın katılımcıları yaşadıkları Alice deneyimine ilişkin görüşlerinin saptanması önem arz etmektedir. Yurtdışında yapılan araştırmaların da bu yazılım ile verilen programlama öğretiminde, öğrencilerin başarılarının (Herbert, 2010), derslerde aktifliklerinin (Mullins, Whitfield, & Conlon, 2009), tutumlarının olumlu yönde değiştiğini göstermesi (Cooper vd., 2003a), Alice'in Türkiye şartları açısından değerlendirilmesi potansiyelini gündeme getirmektedir.

1.4. Varsayımlar

Bu araştırmanın sonuçları aşağıdaki varsayımların kabul edilmesine bağlıdır:

- Araştırmanın katılımcıların görüşme formundaki sorulara ve programlama öz yeterlilik algısı ölçeğine samimi cevap vermişlerdir.
- Araştırmaya katılan öğretmen adayları Alice yazılımını değerlendirebilecek bilgi ve deneyime sahiptirler.
- Öğretmen adaylarının programlamaya ilişkin öz yeterlilik algıları 9 haftalık uygulamalar sonucunda değişmiştir.

1.5. Sınırlılıklar

Araştırma bulguları ve sonuçları, 2014-2015 eğitim öğretim yılı içerisinde Uludağ Üniversitesi Eğitim Fakültesi Bilgisayar ve Öğretim Teknolojileri Eğitimi bölümünde, son sınıfta okuyan 20, üçüncü sınıfta okuyan 5 bilişim teknolojileri öğretmen adayıyla sınırlıdır. Öğretmen adaylarının öz yeterlilik algıları anket sonuçlarına dayalı olarak belirlenmiştir. Bu kısıtlılığın dezavantajını en aza indirmek adına, daha önce geçerlik ve güvenilirlik çalışması yapılmış olan ölçme aracından faydalanılmıştır.

1.6. Tanımlar

Alice: Üç boyutlu dünyayı kullanıcılarına sunan, Carnegie Mellon Üniversitesinden (CMU) Randy Pausch önderliğinde geliştirilmiş programlama öğretimi için kullanılan yazılımdır.

FATİH Projesi: Türkiye çapındaki okulöncesi, ilköğretim ve ortaöğretim kurumlarını kapsayacak şekilde her bir dersliğe dizüstü bilgisayar ve etkileşimli tahta, öğrencilere ve öğretmenlere de tablet bilgisayar dağıtılmasını kapsayan projedir. Bunun yanı sıra eğitsel içerik geliştirilmesi ve yazılımların edinilmesini de içerisinde barındırır.

Sürükle-Bırak Ara yüz: Metin / yazı olarak kodların yazılmasını gerektirmeyen, tüm nesne ve hazır kodların ilgili alana sürüklenip bırakılması özelliğini içinde barındıran ara yüzüdür.

Yazım Hataları: Programlamada bir kod satırının yazımında dikkat edilmesi gereken noktalama işaretleri ve kod parçacığının doğru yazımı gibi eylemleri ifade eder.

2. Bölüm

Alanyazın

Araştırma sorularının ilki “Programlama kavramlarının öğretiminde üç boyutlu ortamların kullanımına ilişkin öğretmen adaylarının görüşleri nelerdir?” şeklindedir. Bu bölümde ilk olarak Alice’in genel tanıtımı yapıldıktan sonra, Alice ile yapılan programlama öğretiminin odak noktalarına ilişkin alanyazın taramasına yer verilmektedir. İlk araştırma sorusunun olası cevapları bu bölümdeki ilk altı başlık altında incelenmekte ve (1) Alice’in özellikleri (2) programlamada görsellik, (3) akılda kalıcılık ve başarı, (4) eğlenerek öğrenme, (5) Alice’de karşılaşılabilecek güçlükler ve (6) işbirlikli öğrenme olarak sıralanmaktadır. Son olarak programlamada öz yeterlilik algısı başlığında programlama öğretiminde öz yeterliliğin ne anlama geldiğine değinilmekte ve ikinci araştırma sorusu olan “Öğretmen adayları programlama konusunda kendilerini ne kadar yeterli hissediyorlar?” konusu ile ilintili yapılan çalışmalar sunulmaktadır. Bir başka deyişle, toplamda 7 başlık altında, bu çalışmada ortaya çıkabilecek muhtemel sonuçlara ilişkin alanyazın taramasına yer verilmektedir.

2.1. Bilgisayım Sal Düşünme

Bilgisayar bilimi, hesaplama, iletişim, eşgüdüm, otomasyon ve bellek gibi mekanik bilimi, basitlik, performans, güvenilirlik, güncellenme ve güvenlik gibi tasarım ilkelerini, programlama, mühendislik sistemi, modelleme, geç erleme ve yenileme gibi uygulamaları içeren ve uygun betimsel dilleri oluşturan disiplin olarak tanımlanmaktadır (Abelson & Sussman, 1985 akt. Barr, & Stephenson, 2011; Denning, 2009). Bu disiplinin uygulama boyutunda yer alan programlama, bireylerin bilişsel gelişimine olan katkısından dolayı son yıllarda özellikle üzerinde durulan bir konu haline gelmiştir (Akpınar & Altun, 2014). Nitekim Gibson (2012), bilgisayar biliminin bilgisayar uygulamalarının kullanımına ek olarak onların nasıl çalıştığı ve programlandığına ilişkin bilgileri de içerdiğini ifade etmektedir.

Programlamanın transfer edilebilir problem çözüme ve düşünme becerilerini geliştirmeye olan olumlu etkisinin (Mohamed, 1985; McCoy, & Dodl 1989; Dalton & Goodrum, 1991), programlama öğrenme ve öğretme hususunda dikkatleri topladığını söylemek mümkündür. Nitekim bireyler günlük yaşamlarında çeşitli durum ve problemlerle karşılaşabilmekte, bu problemlerin çözümünde bilişsel süreçlerini işleterek hedefledikleri sonuca ulaşmaya çalışmaktadırlar. Tam da bu noktada, teknolojinin hayatın adeta bir parçası haline geldiği günümüz dünyasında, bilgisayar ve dolayısıyla programlama devreye girmektedir. Karşılaşılan problemler karşısında, olası risklere karşı bireylerin ihtiyaç duyduğu koruma ve önleme, karmaşık durumlarla başa çıkmada soyutlama ve analizi kullanma, keşfedici düşünme ve tekrarlardan yararlanma gibi basamaklarda kendi bilişsel süreçleri ile bilgisayarların yapay zekâsını birleştirmeleri, programlamanın öğrenilmesine ilişkin önemli bir noktayı oluşturmaktadır (Naughton, 2012).

Programlama öğrenen öğrenciler ile yapılan çalışmalar (Papert, 1980; Clements, & Gullo, 1984; Hromokoviç, 2006) incelendiğinde, yansıtıcı ve yaratıcı düşünme, problem çözüme ve iletişim becerilerinin geliştirilmesi üzerinde olumlu yönde bir etkisinin olduğunu söylemek mümkündür. Ayrıca farklı alanlardaki (matematik, fen eğitimi vb.) kavramların öğretiminde programlamadan yararlanılmakta olduğu ifade edilebilir (Akpınar & Altun, 2014). Bu durum, bireylerin düşünce sistemleri ile bilgisayar kullanımları arasında bir bağın kurularak karşılaştıkları problemlerin çözümünde bilgisayardan yararlanmalarının söz konusu olabildiğini göstermektedir. Nitekim bu eğilim, bilgisayarlı düşünme (computational thinking) kavramının alanyazına girmesine katkıda bulunmuştur (Korkmaz, Çakır, Özden, Oluk, & Sarıoğlu, 2015).

Bilgisayarlı düşünme, Wing (2006) tarafından bilgisayar bilimine ait temel kavramlardan yararlanarak problem çözüme, sistemleri tasarlama ve insan davranışını anlama gibi noktaları içeren bir düşünme süreci olarak tanımlanmaktadır. Yani, bireyin günlük

hayatta karşılaştığı bir problem karşısında bir bilgisayar bilimcisi gibi düşünmesini içermektedir. Aho (2012) ise bilgisayarlı düşünmeyi bireylerin bir problemi formüle ederek çözümü için attıkları adımların adeta birer bilgisayar adımı gibi olması ve algoritma oluşturmaları olarak yeniden tanımlamıştır. Dolayısıyla bu kavram, problemlerin çözümünde Wing (2008)'in de belirttiği üzere matematiksel, analitik ve bilimsel düşünme biçimleri gibi bilişsel süreçleri kapsamakta, yaratıcı düşünme, algoritmik düşünme, eleştirel düşünme, problem çözme, işbirlikli öğrenme ve iletişim gibi becerileri içermektedir (ISTE, 2015).

Barr, Harrison ve Conery (2011), bilgisayarlı düşünmeyi bir problem çözme süreci olarak ele almaktadır. Bu sürecin basamaklarını ise şöyle ifade etmektedir:

- Problemleri bilgisayar ve diğer araçları çözümünde kullanacak şekilde formüle etme
- Verileri mantıklı bir şekilde organize ve analiz etme
- Modeller ve simülasyon gibi verileri soyutlayarak sunma
- Algoritmik düşünme yoluyla çözümleri otomatikleştirme
- En etkili ve verimli adım ve kaynakları kullanarak olası çözümleri tanımlama, analiz etme ve uygulama

Bu problem çözme süreçlerini farklı türdeki problemlere transfer etme ve genelleme basamakları incelendiğinde, bireylerin problem çözme becerilerini kullanmada bilişsel olarak izledikleri yol ile paralel olduğu görülmektedir. Bilgisayarlı düşünme ile problem çözümede bireylerin zihin gücüne ek olarak bilgisayarların gücünü kullanabilme yetisinin bir araya getirildiği söylenebilir. Basamakta yer alan algoritmik düşünme ise, bir problemi net bir şekilde belirleme, verilen problemi analiz etme, verilen problem için yeterli olan temel adımları bulma, bu temel adımları kullanarak doğru algoritmayı oluşturma ve algoritmanın etkililiğini artırma aşamalarından oluşan beceriler bütünüdür (Futschek, 2006).

Bilgisaymsal düşünmenin, farklı düşünme biçimlerini içinde barındırması ve problem çözmeye bu düşünce biçimlerini bilgisayar bilimi ile bütünleştirmesi açısından programlama öğrenimi ve öğretimi için önem taşıdığı söylemek mümkündür. Grover ve Pea (2013), bilgisayarlı düşünme içerisinde bulunan düşünme biçimlerini desteklemede programlamanın bir anahtar rolünde olduğunu ifade etmektedir. Nitekim alanyazında bilgisayarlı düşünmeyi ve içerdiği düşünme becerilerini (yansıtıcı, yaratıcı, algoritmik, analitik ve eleştirel) ayrı ayrı ele alarak bunları geliştirmek amacıyla yapılan çalışmalarda üç boyutlu programlama yazılımları ve oyunları gibi farklı araçların kullanıldığı görülmektedir (Werner, Denner, Campe, & Kawamoto, 2012; Lee, Louis, Ahn, & Bederson, 2012; Karadeniz, Samur, & Özden, 2014). Bunlardan en popüler olanları grafiksel programlama ortamları olan Scratch, Alice, Game Maker, Kodu ve Greenfoot; web temelli simülasyon yazarlığı araçlarından Agentsheets ve Agentcubes; fiziksel programlama platformlarından Arduino ve Gogo Boards'dır (Grover, 2013). Bunun yanı sıra Kodable, CTArcade gibi oyun temelli programlama araçları da vardır. Programlama yazılımlarından ve bu çalışmaya da konu olan Alice, Cooper, Dann ve Pausch (2000b) tarafından anında geri dönüt alma, eğlenerek öğrenme, işbirlikli öğrenme, sade ve anlaşılır bir dil sunma ve nesne temelli olma gibi özelliklerinden dolayı özellikle algoritmik düşünmeyi geliştirme açısından önerilmiştir.

2.2. Programlamada Görsellik

Algoritma ile program yazımında ve özellikle de programlamaya giriş derslerinde, görselliği ön plana çıkaran türden yazılımlar çeşitlenmiştir (Suryana & Husin, 2014). Görsellik; programlama esnasında kullanılan grafikler, iki ve üç boyutlu nesnelere ile tasarlanma, konumlandırma, düzenlemeden başlayarak yazılan algoritmaların çalıştırılması sonrasında ekrana yansıtılan görüntülere kadar uzanan hacimli bir kavramdır. Alanyazında, programlamada yararlanılan görselliğin çeşitli eğitsel faaliyetlere olumlu etki ettiğine dair çalışmalar bulunmaktadır. Örneğin, Suryana ve Husin (2014) yaptıkları deneysel çalışmada

3De-AlProV programı ile çalışan öğrencilerin, öğrenim kazanımlarını gerçekleştirmede geleneksel programlarla çalışan öğrencilere göre %13 daha başarılı oldukları sonucuna ulaşmışlardır. Programlamada görselliği ön plana çıkaran yazılımlar görselliğin katkısı anlamında 3 farklı açıdan sentezlenebilir: (1) kodlarla etkili çalışma, (2) hatayı önleme veya azaltma ve (3) gerçek yaşama yakın programlama ve sonuç alma deneyimi sunma. Aşağıda sırasıyla bunlara değinilmiştir.

Programlamada görselliğin katkılarında ilki kodlarla etkili çalışmadır. Üç boyutlu ve etkileşim gerektiren bir ortam olan Alice yazılımında hazırlanan kodların animasyonla canlandırılmış çıktıları, öğrencilerin hangi kodun ne amaçla kullanılacağını anlamalarını kolaylaştırır. Yani, görsellik, öğrencilerin bir problemi çözmek için parça parça yazılan kodların kendi başlarına ve hep birlikte nasıl çalışacaklarını anlamalarında kullanılabilir (Cooper vd., 2000a; Mullins vd., 2009). Alice ile öğretim öğrencilerin programlama konusunda özgüven sahibi olmalarını sağlamakta, temel programlama kavramlarının anlaşılmasını kolaylaştırmakta ve algoritmalarla Alice hikâyeleri arasındaki bağlantıları anlamalarına yardımcı olmaktadır (Howard, Evans, Courte, & Bishop-Clark, 2006).

İkinci olarak, Alice'te görsellik ile birlikte gelen sürükle-bırak ara yüzü hatayı önleme veya azaltma konusunda kullanıcılarına yardımcı olmaktadır. Öğrenciler nerede hata yaptıklarını ve nasıl düzelteceklerini anlamada bir sorun yaşamamaktadır (Cooper vd., 2000a; Mullins vd., 2009). Öğretmenler ve öğrenciler satır satır kod yazmayı gerektiren programlama araçlarından uzak durmaya çalışmaktadır (Malliarakis vd., 2014). Alice, metin tabanlı (satır satır, noktasına virgülüne kodların yazılması gereken) yazılımlara kıyasla, yazım hatalarından (syntax errors) bağımsız olması ile kavramsal yanlışların ortaya çıkmasını engeller. Görselliği sayesinde doğrudan istenileni uygulama ve sonuçları görmeyi sağlar (Al-Ahat, 2014; Mullins vd., 2009).

Gerçek yaşama yakın programlama ve sonuç alma deneyimi görselliğin katkılarına ilişkin alanyazında karşımıza çıkan son unsurdur. Alice'in görselliği programlama işinin öğretilmesinin yanı sıra yazılım geliştirme süreçlerinden nesne modellemeye kadar pek çok alana katkı yapacak potansiyele sahiptir. Üç boyutlu yapısı sayesinde gerçek yaşam deneyimlerine yakın nesnelere çalışma imkânı verir (Al-Ahat, 2014). İki ve üç boyutlu nesnelerin düzenlenmesi, konumlandırılması ve yazılan kodlarla hareketi ile bu iş gerçekleştirilir (Cooper vd., 2000a; Mullins vd., 2009). Çoklu ortam materyalleri geliştirilerek gerçek yaşam koşulları betimlenebilir (Al-Ahat, 2014).

2.3. Akılda Kalıcılık ve Akademik Başarı

Akılda kalıcılık kavramı öğrenilen bilgilerin ileriki zamanlarda hatırlanmasında (memorization) bir sorun ile karşılaşmama anlamında kullanılmaktadır. Bu bölümde öğrenilen bilginin akılda kalıcılığını etkileyen etmenlere ilişkin farklı alanlarda yapılan çalışmalardan çıkarımlar sunulmaktadır. Alice ile programlama öğretiminde akılda kalıcılık üzerine yapılan çalışmalara bakıldığında (Cooper, 2010; Powers, Ecott, & Hirshfield, 2007) öğrenilen bilginin ileri düzey programlama derslerine aktarılması sürecine odaklandığı gerektiğinden bahsettiği görülmektedir. Akılda kalıcılığı etkileyen faktörlerden ilki, metin tabanlı materyallerle yapılanın aksine görsel-işitsel materyallerle desteklenen öğretim, öğrencilerin bilgileri hatırlamasını/bilgilerin akılda kalıcılığını etkilemektedir (Choi & Johnson, 2005, 2007). Bunun yanı sıra, bilgisayar destekli olarak yapılan matematik derslerinde kullanılan dinamik yazılımlardaki görselliğin akılda kalıcılığı arttırdığı görülmektedir (Kutluca & Zengin, 2011; Zengin, Kağızmanlı, Tatar, & İşleyen, 2013). Tarih bilinci üzerine çalışan Sezginsoy ve Akkoyunlu (2011) ise yaptıkları araştırmada ezberlemek yerine bilginin öğrencilerin tarafından özümletilmesinin sağlanması gerektiğini, yani uygulamaların yapılmasını ve bu sayede akılda kalıcılığın artacağını düşünmektedirler. Yaptıkları görüşmelerde ders içindeki etkinliklerin (şiir, şarkı, bilmece, bulmaca gibi) kalıcılık puanı

üzerine etkisi olduğundan bahsetmektedir. Örneğin, Atatürk ve Türkiye Büyük Millet Meclisi (T.B.M.M) ile ilgili bir canlandırma yapmış ve katılımcıların bu tarz uygulamalardan hoşlandıklarını bulmuşlardır. Özetleyecek olursak akılda kalıcılığı etkileyen faktörler arasında görsellik, uygulayarak öğrenme ve her ne kadar doğrudan bir çıkarım yapılamasa da eğlenerek öğrenme sayılabilir. Son olarak, alanyazına bakıldığında Alice ile yapılan programlama öğretimi aynı zamanda öğrencilerin ders başarılarını da arttırmaktadır (Dann, Cosgrove, Slater, Culyba, & Cooper, 2012; Howard vd., 2006; Johnsgard & Mcdonald, 2008; Moskal, Lurie, & Cooper, 2004). Fakat literatürde Alice ile programlama öğretiminde ders başarısını etkileyen etmenler açıkça belirtilmemektedir.

2.4. Eğlenerek Öğrenme

Eğlenerek öğrenme tabirinin içerisine yapılan işten keyif duyma, oyun gibi görüp derslere ilgi gösterme ve memnuniyet gibi anlamlar yüklenebilir. Bu bölümde eğlenerek öğrenmeye ilişkin alanyazın taramasından örneklere yer verilmiştir.

Günümüz gençleri çok küçük yaştan itibaren, oyunlar ve çoklu ortam materyalleri sayesinde bilgisayarla içli dışlı olmaktadır. Etkileyici grafik ara yüzü ve özel efektleri olan ortamlarla etkileşimde bulunmak hoşlarına gitmektedir. Bu yüzden, alandaki yeni yetişen öğretmenler, öğrencilerinin hoşuna giden ve alışkın oldukları ortamlarda programlama öğretimi yapmaya çalışmaktadır. Yani, eğitimde etkileşimin olmadığı ortamlardan kaçınılması yönünde bir eğilim vardır (Malliarakis vd., 2014).

Alice ile programlama yapmak öğrencilere zevkli gelmektedir.(Al-Tahat, 2014; Bishop-Clark, Courte, Evans, & Howard, 2007; Howard vd., 2006; Mullins vd., 2009). Alice Carnegio Mellon Üniversitesi tarafından geliştirilmiş ve inanılmaz derecede eğlenceli üç boyutlu (3B) bir programlama ortamıdır (Al-Tahat, 2014). Howard ve diğerlerine (2006) göre çalışmalarına katılan öğrencilerin tamamına yakını (%96) Alice ile programlamayı sevmektedir.

Mullins ve diğlerleri (2009) Alice'in ara yüzü ve grafiksel yapısı bilgisayar bilimlerinde okuyan öğrencilerin yanı sıra alan dışından olan öğrencilerin derse ilgi duymalarını ve seçmelerini sağladığını belirtmektedir. Ayrıca, yine aynı çalışmada programlamaya giriş dersini alan öğrencilerin düzenli katılım ve ders başarılarına bakıldığında önceki senelere oranla arttığı gösterilmiştir. Aynı zamanda dersi seçen öğrenci sayısında da yüzde 10 oranında bir artış söz konusudur.

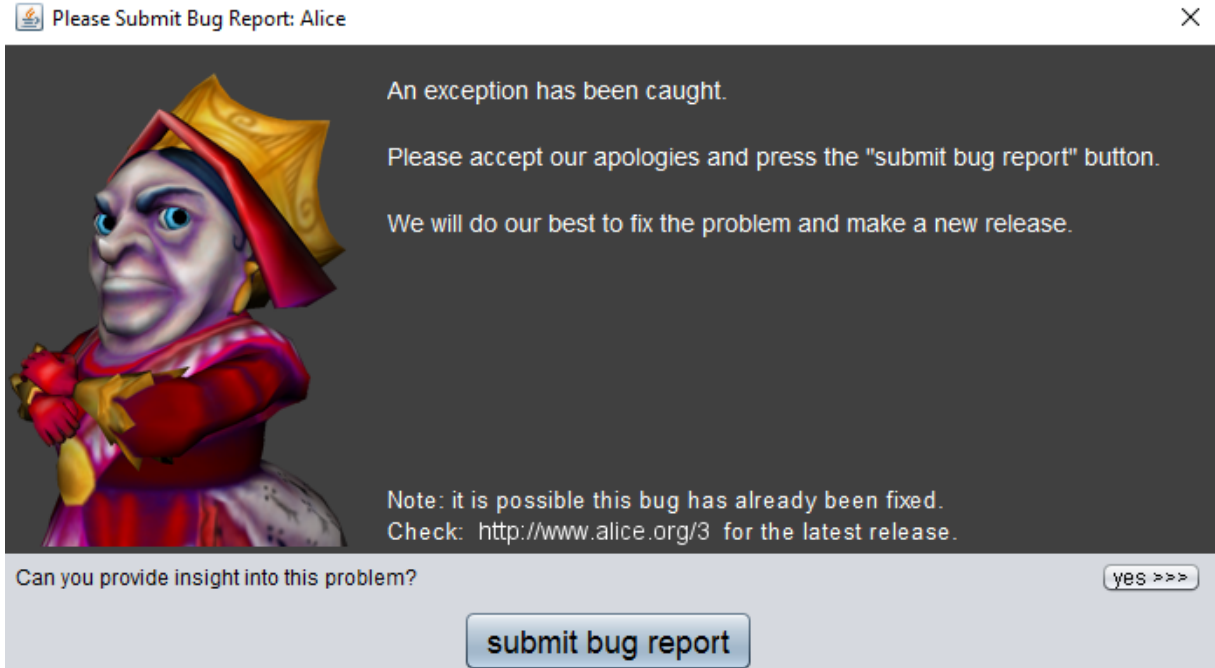
Skyes (2007) Alice'in programlama öğretiminde öğrencilere etkileşimli, eğlenceli, dikkat çekici ve kullanımı kolay bir ortam sağladığı belirtmektedir. Bunun yanı sıra, öğrencilere Alice ilgi çekici gelmekte, onların motivasyonlarını arttırmaktadır (Moskal vd., 2004; Skyes, 2007). Skyes (2007) yaptığı görüşmelerde bazı öğrencilerin ders dışında da evlerinde çalıştıklarından bahsetmektedir.

2.5. Alice'de Karşılabilir Güçlükler

Alice ile programlama öğretimi her ne kadar öğrencilere programlamaya giriş derslerinde kolaylık sağlayıp onların aktif olarak katılımlarını sağlasa da bir dizi problemle karşılaşmalarına da neden olmaktadır. Bu güçlükler şu şekilde listelenebilir:

1. Öğrenciler yazılımın hata (bug) ekranı ile uğraşmak zorunda kalmaktadır (Mullins vd., 2009; Skyes, 2007). Hata ekranına ilişkin örnek Şekil 2'de gösterilmektedir.

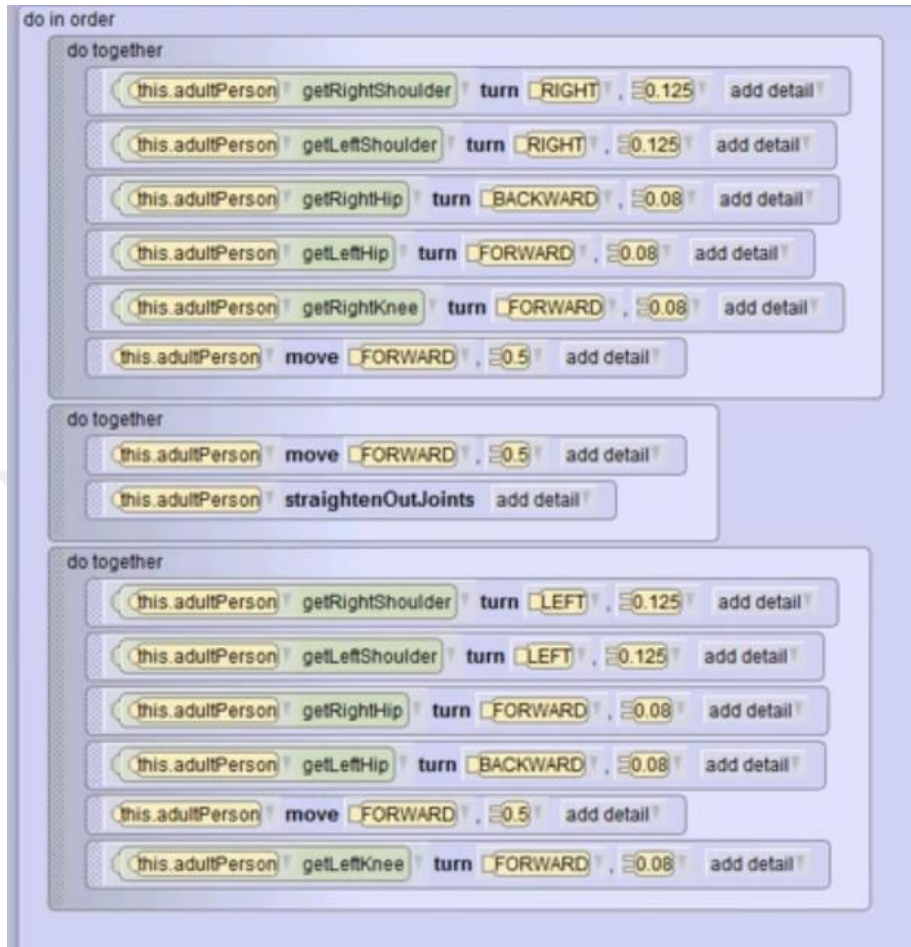
Şekil 2

Hata ekranı

2. Var olan metotların dışında yeni metotlar / olaylar yaratırken güçlüklerle karşılaşmaktadır (Mullins vd., 2009). Alice’de bazı prosedür ve fonksiyon bir eylemi gerçekleştirmede yetersiz kalmaktadır. Örneğin, zıplama eylemi için önce yukarı doğru hareket, daha sonra aşağı doğru hareket gerekir. Şekil 3’de Alice’de hazır olarak gelmeyen ve ihtiyacı karşılamak üzere yeni oluşturulan bir metotta (prosedür yada fonksiyon) yer alan kodlar gösterilmektedir.

Şekil 3

Yürüme metodunun oluşturulmasına ilişkin örnek kodlar



3. C++ ve Java gibi programlama dillerine geçişlerde, Alice'te yapılan programlama öğretiminin öğrencilerin nesne-tabanlı programlama kavramlarını anlamalarını zorlaştırmaktadır (Powers vd., 2007).
4. Alice'in donmasından kaynaklanan veri kayıpları ve çalışma da sorunlar ortaya çıkmaktadır (Skyes, 2007).
5. Yazılan kodların grafiksel çıktıları, öğrenenlerin gerçek hayattaki gibi hareketlerini beklmelerinden ötürü hayal kırıklığı yaratmaktadır (Powers vd., 2007).
6. Scratch ve Alice'in karşılaştırıldığı bir araştırmada bulunan sonuçlara göre, üç boyutlu nesnelere ve ara yüzle uğraşmaktaki güçlüklerin eğlenerek öğrenmesini

engellemektedir (Chang, 2014). Eđlenerek öğrenme konusuna bir önceki bölümde değinilmektedir.

7. Alice'in geneline bakıldığında öğrencileri sınırlaması ve istediklerini tam olarak yazdıkları kodlara yansıtamamaları sorun teşkil etmektedir (Bishop-Clark vd., 2007).

2.6. İşbirlikli Öğrenme

21. yüzyılda eğitim dünyasındaki değişimden bahsedilmişti. Eğitim kuramları veya öğrenme/öğretme yaklaşımları açısından bakıldığında da eğitimin amacı, işlevi, doğası ve sınırları konusunda çeşitlilik söz konusudur. Sanayi devrimiyle hayatlarımızda iyice ortaya çıkmış rekabetçi düzen ve eğitim sistemi bilgi çağına geçildiğinde yerini işbirliği içerisinde çalışmaya bırakmaya başlamıştır. İşbirlikli öğrenmenin önemi NASA, IBM, Microsoft veya Apple gibi dünya çapındaki kuruluşların çalışmalarında rahatlıkla görülebilir. İşbirlikli öğrenme bireysel olarak küçük ölçekte gerçekleştirilemeyecek veya başarılamayacak bir işi, farklı donanımlara sahip insanların bir araya geldiği gruplarla daha etkili ve büyük ölçekte gerçekleştirmeye olanak tanır. Ayrıca, genel anlamda işbirliğini düşünecek olursak, internet ve iletişim teknolojilerindeki ilerlemelerle artık dünya çapında gerçekleştirilebilir hale gelmiştir. Örneğin, Amerikan şirketleri Hintli mühendisleri, Hindistan'da yer alan müşteri hizmetleri bölümlerinde çalıştırarak müşterilerine hizmet vermektedirler. Bir başka örnek olarak da Avrupa birliği çapındaki projelerde farklı ülkelerdeki araştırmacıların ve/ya bilim insanlarının birlikte çalışmaları gösterilebilir.

Kozlowski ve Ilgen (2007) bugün pek çok şirket farklı deneyimlere sahip insanları birleştirerek gruplar halinde çalıştırdığından bahsetmektedir. Örneğin NASA Apollo 13'teki astronotların hayatlarını farklı alanlardan uzman kişilerin birlikte çalışmasını sağlayarak kurtarmıştır. Bu grubun içerisinde fizik ve matematik uzmanları, mühendisler, teknik elemanlar gibi farklı donanımlara sahip insanlar yer almıştır. Yani, her ne kadar eğitim

sistemlerinde okuma, yazma, matematik, edebiyat, fizik gibi konuların üzerinde duruluyor olsa da, 21. yüzyıl eğitim sisteminde, işbirliği içerisinde çalışma becerisi de tıpkı okuma, yazma, matematik gibi okullarda üzerinde durulması önerilen konulardan birisidir.

Roseth, D.W. Johnson ve R. T. Johnson (2008) yaptıkları çalışmada rekabetçi öğrenme yerine, işbirlikli öğrenme desteklenmesi gerektiğini savunmuşlardır. Rekabetçi eğitim sisteminde birkaç öğrencinin başarısı, diğerlerinin başarısızlığına neden olabilir. İşbirlikli öğrenme de ise karşılıklı yardımlaşma, grup üyeleri arasında iletişim, gerekli kaynakların üyeler arasında değişimi söz konusudur. Dolayısıyla, Roseth ve diğerlerine göre öğrencilerin akademik başarısı ve işbirlikli öğrenme arasında olumlu bir ilişki söz konusudur.

Denton (2012)'a göre işbirlikli öğrenmede uygulanması gereken prensipler şunlardır:

1. Ders amaçlarının açık bir şekilde öğrencilere ders özeti kullanılarak sunulması
2. Aktiviteler arası geçişlere dikkat edilmesi,
3. Öğretim etkinliklerine öğrencilerin katılımını sağlama,
4. Dikkat dağıtıcı materyallerin ve uyarıcıların öğretim ortamından uzaklaştırılması,
5. Gerekli görülen uygulamaların seçimi (Google Docs, Microsoft Office, Prezi gibi),
6. Öğretim etkinliklerinin düzenlenmesi

İşbirlikli öğrenme kuramına göre, derslerde öğrencilere grup projeleri verilebilir ve grup değerlendirme formları kullanılabilir. Öğrenciler dijital öyküleme felsefesi ile materyaller geliştirebilir, sınıf içi tartışmalara katılabilir ve işbirlikli yansımalar yazabilir. Bilginin farklı gösterim şekilleri (metin, ses, görüntü, vb.) ile üzerinde durulan konu hakkında derinlemesine anlama imkânı oluşturulabilir. Süreç değerlendirmesi ve tabular değerlendirme ölçütleri kullanılarak öğrencilerin akademik ders başarıları hesaplanabilir (Denton, 2012).

Howard ve diğerleri (2006), işbirlikli yaklaşımın programlama öğretimindeki avantajları/dezavantajlarına ilişkin bulgularını beş başlık altında toplamaktadır. Katılımcılarla yapılan görüşmeler neticesinde ortaya çıkan özet Tablo 1'de verilmektedir.

Tablo 1

Grup çalışması ile bireysel çalışmanın farklı değişkenlere göre kıyaslanması

Değişken	Frekans(%)
Öğrenmenin / başarının artışı	%95
Daha yaratıcı olma durumu (Farklı fikirler)	%67
Süreci yavaşlatması	%34
Hikâye hakkında katılımcıların farklı fikirlerinin olması	%38
Eş zamanlı çalışmaya imkân vermemesi ve aynı anda yalnızca bir kişinin proje üstünde çalışabilmesi	%29

Buna göre grup halinde (işbirlikli) çalışan katılımcıların tamamına yakını Alice'in programlama öğrenimini kolaylaştıracağını düşünmektedir. Farklı fikirlerin ortaya çıkmasını sağlayacağını düşünenlerde azımsanmayacak kadar çoktur. Süreci yavaşlatması, oluşturulacak hikâyeler hakkında grup üyelerinin farklı fikirlerin olması ve eş zamanlı çalışmaya imkân vermemesi grup çalışmasını zorlaştıran noktalardandır.

2.7. Programlamada Öz yeterlilik Algısı

Son yıllarda psikoloji ve eğitim alanlarında önemli bir yer tutan öz yeterlilik kavramı, alanyazında ilk olarak öncülüğünü Bandura'nın yaptığı Sosyal Öğrenme Kuramında karşımıza çıkmaktadır. Bandura (1994), öz yeterliliği bireyin hayatında önemli yer eden olaylar üzerinde etkisi olabilecek performanslara ilişkin sahip olduğu yeterliliklere inancı olarak tanımlamak ve bu inançların bireyin nasıl hissedeceği, düşüneceği ve davranacağı üzerinde belirleyici bir rol oynadığını ifade etmektedir. Nitekim kuramında, öz yeterliliğin bireyin yaşamındaki etkinliklerin seçimi, bunlar için harcadığı çaba, zorluklar karşısındaki dayanaklılığı ve de performansı üzerinde etkili olduğunu belirtmiştir. Dolayısıyla, öz yeterliliğin bilişsel, duyuşsal ve devinimler süreçlerle ilişkili olduğunu söylemek mümkündür.

Yüksek düzeyde öz yeterliliğe sahip olan bireylerin, karşılaştıkları zorlukları tehdit olarak değil başarılması gereken bir iş olarak gördüklerini ifade eden Bandura (1994), başarısızlıklarını yeterliliklerini yöneterek aşmaya çalıştıklarını belirtmektedir. Düşük öz yeterliliğe sahip bireylerin ise zorlukları kişisel tehditler olarak algıladıklarını böylece kendi yetersizlikleri ve dış engeller üzerine odaklandıklarını ifade etmektedir. Bu durum, Rotter'ın (1990) Denetim Odağı Kuramındaki bireylerin davranışlarının sonuçları üzerinde kendi yeterliliklerine olan inançlarına veyahut şans, kader gibi dış güçlerin etkisine olan inançlarına bağlamalarına karşılık gelmektedir. Davranış sonuçları üzerinde kendi çaba ve yeteneklerinin etkisine inanan bireyler içten denetimli olarak tanımlanırken sonuçların çevresel faktörler tarafından belirlendiğine inanan bireyler ise dıştan denetimli olarak tanımlanmaktadır.

Bandura (1977), öz yeterlilik algısının 4 faktörden etkilendiğini belirtmektedir. Bunlardan ilki bireyin bir beceriyi uyguladığı ustalık deneyimleridir. İkincisi, diğer bireylerin davranışlarının model alınmasını ifade eden dolaylı deneyimlerdir. Üçüncüsü sözel iknalaradır. Bireyler karşılaştıkları durumlar karşısında başkalarından aldıkları öneri, cesaretlendirme ve tavsiyeler ile hareket edebilmektedir. Dördüncü ve sonuncusu ise stres ve kaygı gibi duygusal durumlardır.

Öz yeterlilik algısı, bireylerin bir performansı gerçekleştirmek için gerekli şartları oluşturma ve performansı planlama-yürütme konusunda kendileri hakkındaki yargıları kapsadığından, belirli durum ve problem karşısında nasıl düşünüp hareket edeceğini etkilemesi açısından önem taşımaktadır (Bandura, 1994). Dolayısıyla öz yeterliliğin öğrenme-öğretme sürecinde önemli bir değişken olarak yer aldığını söylemek mümkündür. Nitekim Bandura (1977) bireylerin hem gerçekleştirecekleri eylemlerin seçiminde hem de gereken çabaların gösterilmesinde sahip olunan yeterliliklerin, öz yeterlilik algısının ve sonuç beklentisinin doğrudan etkisi olduğunu ifade etmektedir.

Öğrenme-öğretme sürecinde öz yeterliliğin rolüne ilişkin araştırma eğilimlerinin hız kazandığı alanlardan biri de bilgisayar bilimleridir. Teknoloji okuryazarlığının yetiştirilmesi 21. yüzyıl insanından beklenen yeterliliklerden biri haline gelmesi, bilgisayar bilimlerinin önemli bir alanı olan programlamaya ilişkin bilgi ve becerilerinin kazanılmasını beraberinde getirmiştir. Bu bilgi ve becerilerin edinilmesi sürecinin duyuşsal boyutlarından biri olan öz yeterlilik algısı, programlama öğreniminde önemli bir rol oynamaktadır. Alanyazın incelendiğinde, çalışmaların programlamaya ilişkin öz yeterlilik algısının tespit edilmesi (Aşkar ve Davenport, 2009; Busch, 1995; Owolabi ve Adegoke, 2014; Özyurt ve Özyurt, 2015) üzerine odaklandığı görülmektedir. Çalışmaların bir diğer odak noktası olarak öz yeterlilik algısının programlama üzerindeki etkisinin çeşitli değişkenler açısından yordanması (Altun ve Mazman, 2012; Jegede, 2009; Ramalingam ve Wiedenbeck, 1998) olarak sınıflandırılabilir. Fakat öğrenenlerin Alice ile programlamaya ilişkin öz yeterlilik algıları üzerine yapılan çalışmaların azlığı dikkat çekmektedir. Alanyazında yalnızca, Wang, Mei, Lin, Chiu ve Lin (2009), “Lise Öğrencilerine Alice ile Programlama Kavramlarının Öğretilmesi” isimli çalışmasına rastlanmaktadır. Çalışma grubunu 166 lise öğrencisinin oluşturduğu ön test-son test kontrol gruplu deneysel desende hazırlanan çalışmalarında, öz yeterlilik boyutu bulunan motivasyon ölçeği uygulanmış ve deney grubunun öz yeterlilik alt boyutunda ön test son test puanları arasında manidar bir farklılık olduğu belirlenmiştir.

3. Bölüm

Yöntem

Yöntem bölümünde Araştırmanın modeli, Örneklem (Çalışma Grubu), Veri Toplama Araçları ve Verilerin Toplanması ve çözümlenmesi süreçleri tanımlanmıştır.

3.1. Araştırmanın Modeli

Programlama eğitiminde üç boyutlu dünyaların kullanımına ilişkin öğrenci görüşlerinin değerlendirilmesi ve öz yeterlilik algılarının saptanması amacıyla, Uludağ Üniversitesi, Eğitim Fakültesi, Bilgisayar ve Öğretim Teknolojileri Eğitimi bölümünde gerçekleştirilen bu araştırmada karma araştırma yöntemi yaklaşımı kullanılmıştır.

Karma araştırma yöntemi, nitel ve nicel ile birlikte üç temel araştırma yönteminden birisidir (Creswell, 2009; Yıldırım ve Şimşek, 2013). Teorik veya uygulamaya ilişkin bilgiyi edinmek amacıyla kullandığı nitel ve nicel araştırma desenleriyle, üretilen bilgiye farklı noktalardan bakmayı sağlar. Yaklaşık 60 yıl önce karşımıza çıkan karma araştırma yöntemi ilk defa sosyolog ve antropologlar tarafından kullanılmıştır. Yani araştırmalarda kullanılan yeni bir yöntem değil, yeni bir yöneliş, eğilim olarak karşımıza çıkmaktadır (Johnson, Onwuegbuzie, & Turner, 2007)

Creswell'e (2009) göre sosyal bilimler ve sağlık bilimleri üzerine yapılan araştırmalarda nitel ve nicel araştırma yöntemleri tek başına bu bilimlerin problemlerinin karmaşık yapısını tanımlamada yetersiz kalmaktadır. Karma yöntem, bir problemi daha detaylı açıklamak için bu iki araştırma yaklaşımının tek başına elde edebileceği verileri kullanmak yerine, iki yaklaşımın elde ettiği sonuçların birleştirilmesi imkânını sunmaktadır. Bir diğer tanımda ise Creswell ve Clark (2007) birbirlerini desteklemek için kullanılan nitel ve nicel araştırma verilerinin birleştirilmesinden oluşan sonuçlar veya veri tabanlarını karma yöntem araştırması olarak tanımlamaktadır. Bu araştırmada, karma araştırma yönteminin

kullanılmasının nedeni problem durumlarını açıklamada daha detaylı verilere ulaşılmak istenmesinden kaynaklanmaktadır.

Karma araştırma yöntemi sırasında izlenmesi gerekenler adımlar altı başlık altında toplanabilir (Creswell & Clark, 2007) :

1. Araştırma sorularına dayalı olarak hem nitel hem de nicel verileri ikna edici ve titiz bir şekilde toplanması ve analiz edilmesi,
2. Aynı anda iki veri türünü, bu veri türlerinden birini diğerinin içine yerleştirerek veya sırasıyla birini diğerinin üstüne inşa ederek bütünleştirilmesi,
3. Araştırma vurgulanmak istenen veri türlerinden birine veya her ikisine öncelik verilmesi,
4. Bu prosedürleri tek bir çalışma içerisinde veya bir çalışma programının birden fazla aşamasında kullanılması,
5. Bu prosedürleri, felsefi dünya görüşleri ve kuramsal bakış açıları kapsamında çerçeve içine alınması,
6. Bu işlemleri, çalışma yürütme planını yönlendiren özel araştırma deseni ile birleştirilmesi

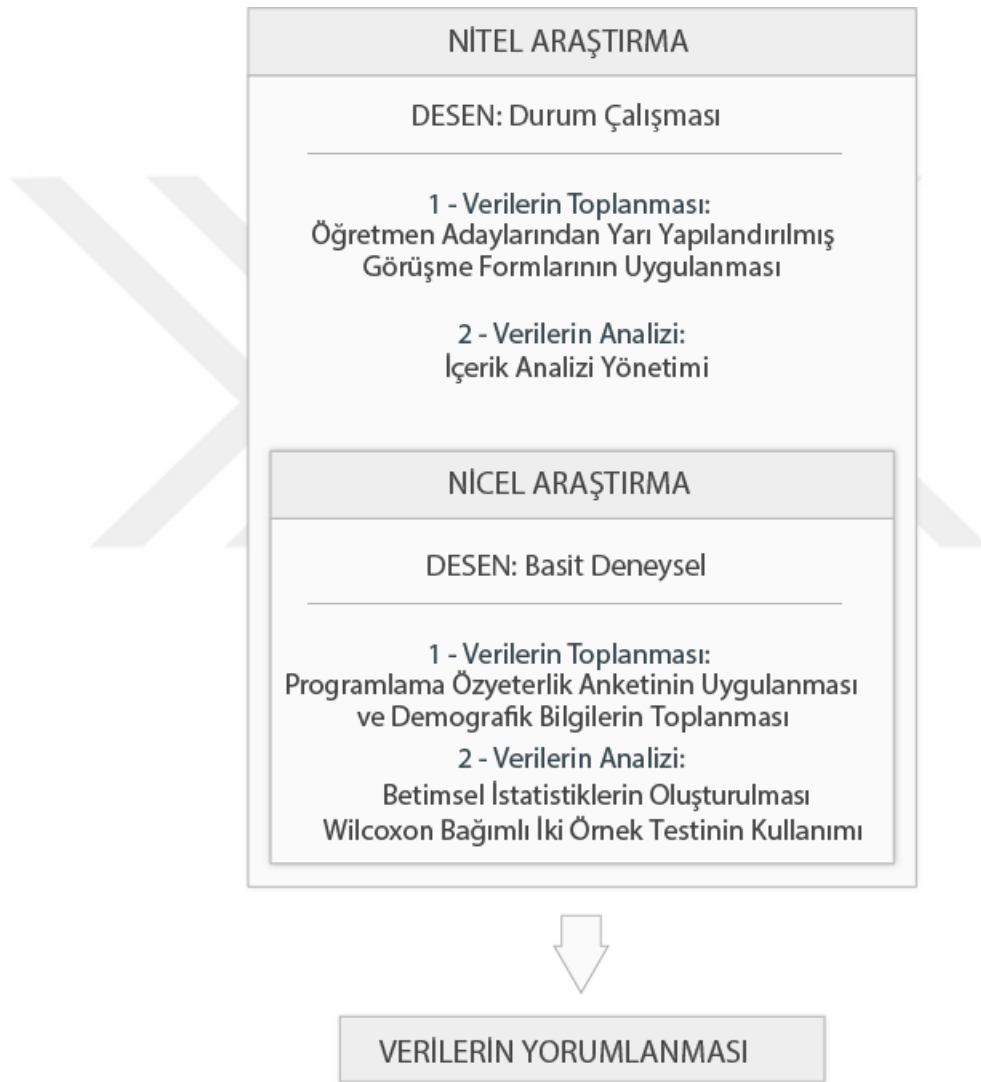
Bu araştırmada, karma yöntem araştırması modellerinden (desenlerinden) gömülü desen kullanılmıştır. Gömülü desen karma araştırma yöntemi türlerinden bir tanesidir. Nitel veya nicel bir grup verinin, kendisine göre daha büyük yapıdaki diğer formdaki veriyle desteklenmesi veya verinin içerisine yedirilerek sunulmasıdır (Creswell, 2009). Gömülü desende bu iki yöntemin bir tanesine ağırlık verilir. Yani araştırma soruları açısından düşünüldüğünde bir yöntemin ağırlığı sayıca daha fazladır (Yıldırım & Simsek, 2013).

Bu araştırmada kullanılan gömülü desen nitel araştırma ağırlıklıdır ve nitel araştırma deseni olarak durum çalışması kullanılmıştır. Nicel veriler kümesi incelenirken ön-test, son-test basit deneysel desen aracılığıyla oluşturulmuş ve bu küme nitel araştırma sorularına

derinlik kazandırabilmek amacıyla kullanılmıştır. Gömülü desenin bir sonraki adımı ise verilerin bütünlük içerisinde sunulması yani verilerin yorumlanması sürecidir. Bu araştırma gerçekleştirilirken izlenen sıra ve ağırlıkları aşağıdaki Şekil 4’de özetlendiği gibidir:

Şekil 4

Gömülü desene göre gerçekleştirilen araştırmanın aşamaları



Durum çalışması deseni ile araştırmacılar bir olay, durum, faaliyeti derinlemesine inceleme imkânına sahiptirler. Bunun yanı sıra disiplinler arasıdır yani psikoloji, hukuk, tıp gibi alanlardan da faydalanarak bir veya birden fazla bireyin karşılaştıkları durumlara ilişkin

düşünceleri incelenebilir. Gözlem ve mülakatlar gibi veri toplama kaynaklarıyla elde edilen veriler içerik analizi, betimsel analiz gibi yöntemlerle incelenebilir ve detaylı çıkarımlar yapılabilir (Creswell, 2013).

Basit deneysel yaklaşımın en önemli özelliği kontrol gruplarını barındırmamasıdır. Üç şekilde uygulanabilir: (1) bir gruba tek seferlik bir deneyin gerçekleştirilmesi, (2) ön-test son-test olacak şekilde deney grubundan ölçümler alınması ve (3) farklı grupların farklı testlere tabi tutulmasıdır (Fraenkel & Wallen, 2005). Bu çalışmada basit deneysel desende ön-test, son-testler bir gruba uygulanarak programlamaya ilişkin öz yeterlilik algıları saptanmaya çalışılmıştır. Bunun yanı sıra betimsel istatistiklerde kullanılarak çalışma grubunun demografik bazı özellikleri tanımlanmaya çalışılmıştır. Nicel veriler yukarıdaki şekil de de görüldüğü gibi ikincil düzeyde bir öneme sahiptir ve durum çalışmasından elde edilen nitel verileri desteklemek amacıyla kullanılmıştır.

3.2. Çalışma Grubu (Örneklem)

Araştırmanın evreni, Uludağ Üniversitesi Bilgisayar ve Öğretim teknolojileri eğitimi bölümünde okuyan, her sınıf düzeyinden öğretmen adaylarından oluşturmaktadır. Evrendeki toplam bireysel sayısı yaklaşık olarak 280 civarındadır. Çalışma grubu, bölümde seçmeli olarak verilen “3B Programlama” dersine kayıt yaptıran 5 adet 3.sınıf öğrencisi (%17.86) ve 23 adet 4.sınıf lisans öğrencisinden (%82.14) oluşmaktadır. Örneklemi oluşturan bireylerin 11’i erkek (%39.28), 17’si kadındır (%60.71).

Örnekleme yöntemi olarak amaçlı örneklem kullanılmıştır. Amaçlı örneklem de araştırmacı, sadece kolay ulaşılabilir örneklem gruplarıyla çalışmaz aynı zamanda bir veya birkaç kendi oluşturduğu ölçütü kullanarak ilgili durumu değerlendirmesi ve örneklemi ona göre oluşturması gerekir (Fraenkel & Wallen, 2005). Bu çalışmada genel gözleme dayalı olarak göze çarpan nokta olan, son sınıfa gelmiş öğretmen adaylarının programlama kavramlarına tam hâkim olmamaları, somutlaştıramamaları ve ileride ortaöğretim seviyesinde

öğretmen olduklarında güçlüklerle karşılaşacakları varsayımından yola çıkarak örneklem oluşturulmuştur.

Gönüllülük esasına göre çalışmanın nitel bölümüne katılmayı 28 öğrenciden 24 tanesi kabul etmiştir. Çalışmanın veri toplama aşamasının eğitim öğretim döneminin sonuna denk gelmesi ve ardından öğrencilerin tatile ayrılacak olmaları, veri toplama faaliyetlerini zaman açısından kısıtlamıştır. Öğretmen adaylarının yazılı olarak soruları cevaplamayı tercih etmesinden dolayı 19 adet katılımcının cevapları açık uçlu anket (Ek-1) aracılığıyla toplanmıştır. Böylelikle araştırmacı açısından görüşmelerin transkript edilmesinde karşılaşılabilecek olası zaman kaybı da ortadan kaldırılmıştır. Görüşmeler geri kalan beş kişi ile gerçekleştirilmiştir. Ön-test, son-testlerde ise hedef gruptan çalışmaya katılanların sayısı 24'tür. Analizler yapılırken bu iki testin ikisini de çözmeyenler veri bloğunun içerisinden çıkartılmıştır. Toplamda 24 öğrenci üzerinden nicel istatistikler oluşturulmuştur. Güvenirlik başlığında nicel verinin niteliği ve hedef gruptan çalışmaya katılan 24 kişinin verilerinin geçerliği ve güvenilirliği konularında istatistiksel analizlere yer verilmiştir.

3.3. Bağlam

Günümüzde küçük yaş gruplarında programlama öğretiminde kullanımı gittikçe artan Scratch, Code.org, Greenfoot gibi iki boyutlu yazılımların yanı sıra üçüncü boyutu kullanıcılarına sunan Alice de programlama öğretiminde bir alternatif olarak durmaktadır. Bu araştırma açık kaynak kodlu bir öğretim aracı olan, ücretsiz olarak edinilebilen ve bu araca ilişkin Türkiye'de sınırlı sayıda araştırma yapılmış Alice yazılımına ilişkin öğretmen adayları görüşlerini ve öz yeterlilik algılarını değerlendirmek üzere gerçekleştirilmiştir. Bilgisayar ve öğretim teknolojileri bölümünden mezun olacak olan bilişim öğretmeni adayları atandıktan sonra programlama derslerini yürüteceğinden, programlama öğretimine ilişkin öz yeterlilik algılarının tespit edilmesinin mezun olduktan sonraki durumlarını yordaması açısından önemli olduğu düşünülmektedir. Sonraki iki alt başlıktan biri olan Alice'in özellikleri bölümünde

yazılımın tanıtımına yer verilmiştir. Diğerinde ise üç boyutlu programlama dersinin içeriği ve kapsamına ilişkin bilgiler sunulmuştur.

3.3.1. Alice'in özellikleri. Alice yazılımı programlamaya giriş derslerinde kullanılmak üzere geliştirilmiş bir öğretim teknolojisidir ve diğer programlama yazılımlarına bakıldığında üç boyutlu oluşu ile ön plana çıkmaktadır (Cooper, 2010). Bu başlık altında Alice'in özelliklerine, ne amaçla kullanıldığına ve Alice'de yapılabileceklerle ilişkin örneklerle yer verilip ara yüzü tanıtılmaktadır.

Alice üç boyutlu dünyayı kullanıcılarına sunan bir öğretim teknolojisidir ve Alice'i kullanarak hikâye anlatımında animasyonlar oluşturulabilir, etkileşimli bir oyun tasarlanabilir veya internette yayımlamak üzere videolar oluşturulabilir. Bunun yanı sıra temel programlama kavramlarının öğrencilere kazandırılması için de Alice yazılımı kullanılmaktadır. Kullanıcılarına sağladığı üç boyutlu nesnelere (insanlar, hayvanlar, araçlar, mitolojik yaratıklar vs.) sayesinde kullanıcılar kendi dünyalarını oluşturulabilmekte ve hazır olarak sunulan blok kodlar ile programlama kavramları öğrenebilmektedir. Alice'in bir diğer amacı ise öğrencilerin programlama konularından zevk almalarını sağlamak ve onlara ilginç yöntemler sunarak derslerinde daha başarılı olmalarını sağlamaktır (Herbert, 2010).

Herbert'e (2010) göre Alice nesne tabanlı bir programlama aracı yerine nesnelere programlama aracı olarak tanımlanabilir. Alice yazılımı yüklendiğinde kullanılacak nesnelere ve onlara tanımlanmış metotları hazır olarak gelmektedir. Yani bunun anlamı nesnelere baştan tasarlanmasına veya klasik programlama yazılımlarında olduğu gibi (Visual Studio, Dreamviewer vs.) satır satır, noktası virgüle kadar kodların yazılmasına gerek kalmamasıdır. Buna ek olarak, nesnelere tıpkı Alice'deki dünyalar gibi üç boyutludur. Metotlar, fonksiyonlar ve olaylar ise bu nesnelere bir eylemi uygulayabilmek için kullanılmaktadır.

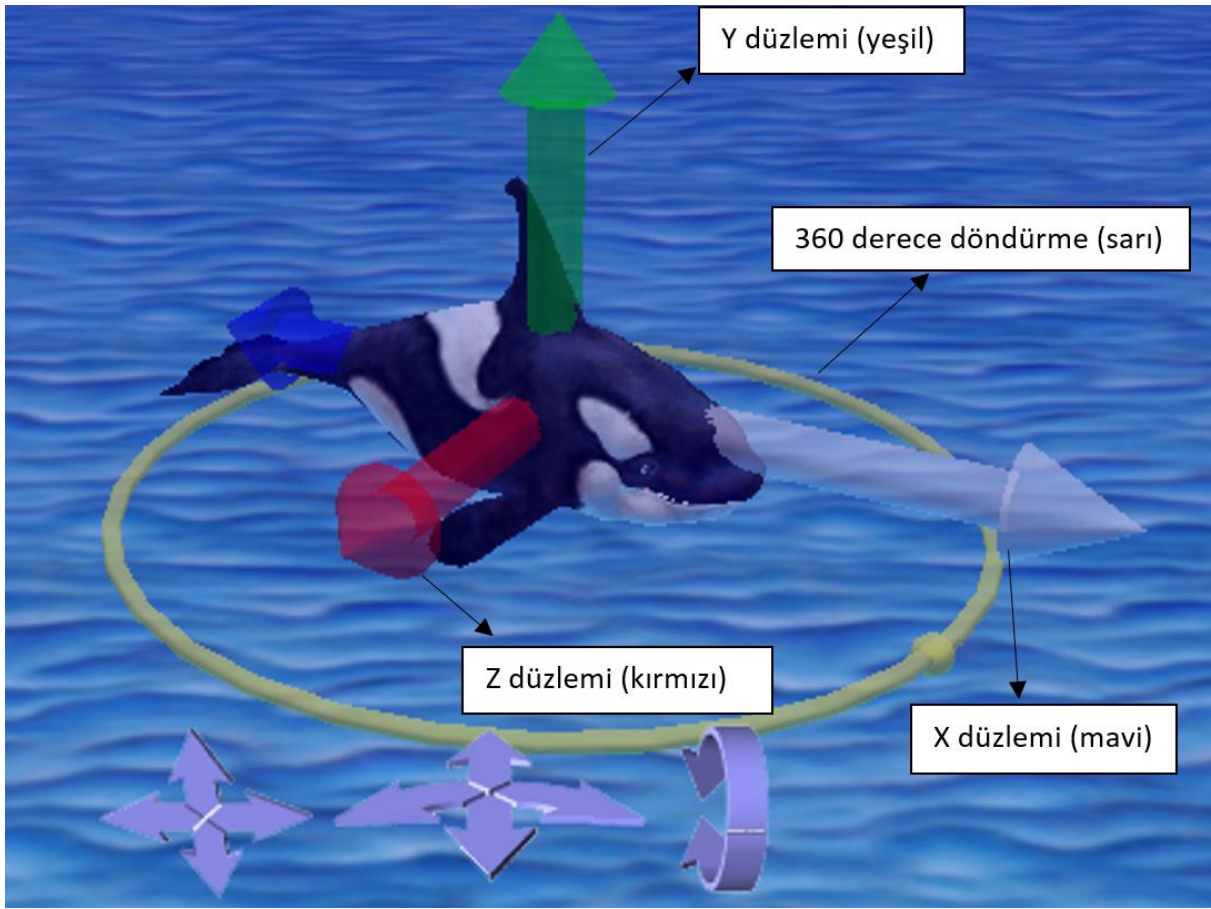
Al-Taht (2014) Alice yazılımının kullanıcılarına sağladıklarını aşağıdaki gibi sıralamaktadır.

- Sanal bir dünya sağlaması ve kontrolünün kullanıcılarında olması,
- Üç boyutlu nesnelere ve sıralandığı nesne ağacı
- Nesnelerin özelliklerinin değiştirilebileceği bir bölüm
- Nesne ağacından seçilen nesneye ilişkin metotlar ve fonksiyonlar
- Programlamanın yapıldığı ve sürükle-bırak yapısı ile kodların sıralandığı alan
- Olaylar bölümü ile kullanıcıların metotları kullanarak olay oluşturmalarına imkan sağlaması (fare tıklamasında bir dizi eylemin gerçekleştirilmesi gibi) şeklinde sıralanabilir.

Üç boyutlu dünyanın ve nesnelerin gösterildiği Şekil 2’de Alice yazılımından alınmış ve deniz temasında bulunan bir balina nesnesine ilişkin bir ekran görüntüsüne yer verilmiştir. Alice’de her bir üç boyutlu (3B) dünyanın kendine has teması bulunmaktadır. Örneğin, denizden, mars yüzeyinden veya buzullardan oluşan bir dünya oluşturulabilir.

Şekil 5

3B dünyadaki balina nesnesi

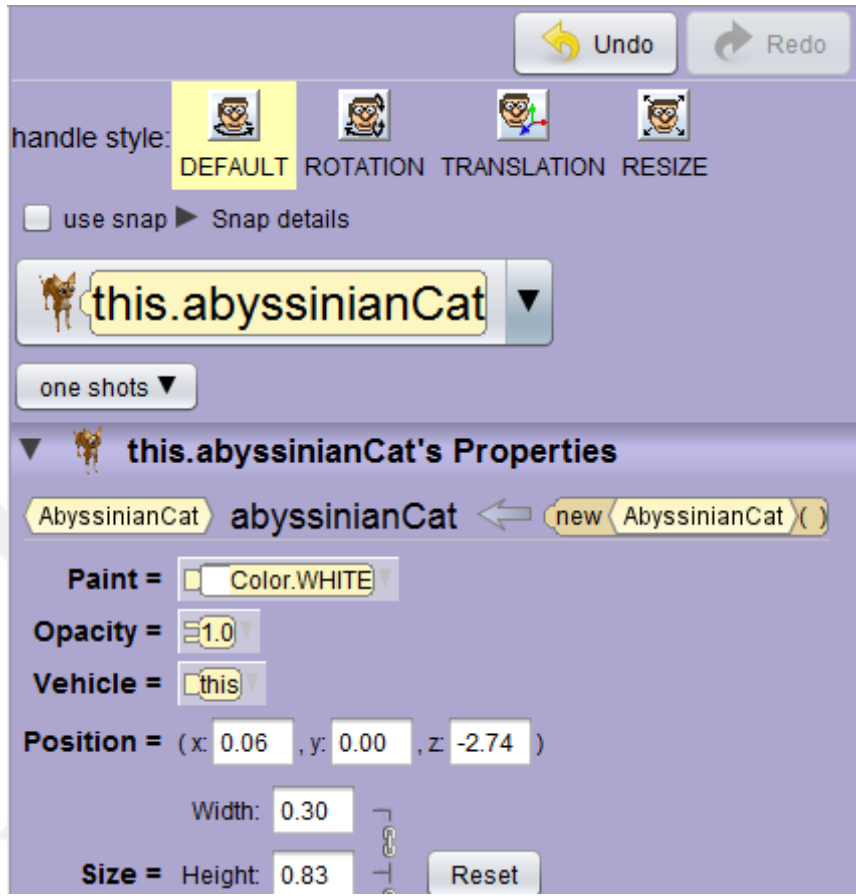


Şekil 5’de görülebileceği gibi 3B dünyada x (mavi), y (yeşil), z (kırmızı) düzlemlerinde balina nesnelere hareket ettirilip konumlandırılabilir. 360 derece döndürme işlemi (sarı) de yapılabilmesinin yanı sıra ileri konumlandırma işlemleri de Alice’in içerisindeki araçlarla gerçekleştirilebilir. Şekilde yer alan ve farklı yönleri gösteren oklar ise üç boyutlu dünyanın içerisinde hareket ettirilmesini sağlar.

Alice’de yer alan bir diğer özellik ise nesnelere özelliklerinin değiştirildiği paneldir. Şekil 6’te Alice’e eklenen 3B bir kedi nesnesinin, döndürülmesi, rengi, saydamlığı, konumu, büyüklüğü gibi özelliklerinin değiştirilebileceği ve yapılan düzenlemelerin geri alınması / yinelenmesi işlemlerinin yapılabilmesi panel gösterilmektedir.

Şekil 6

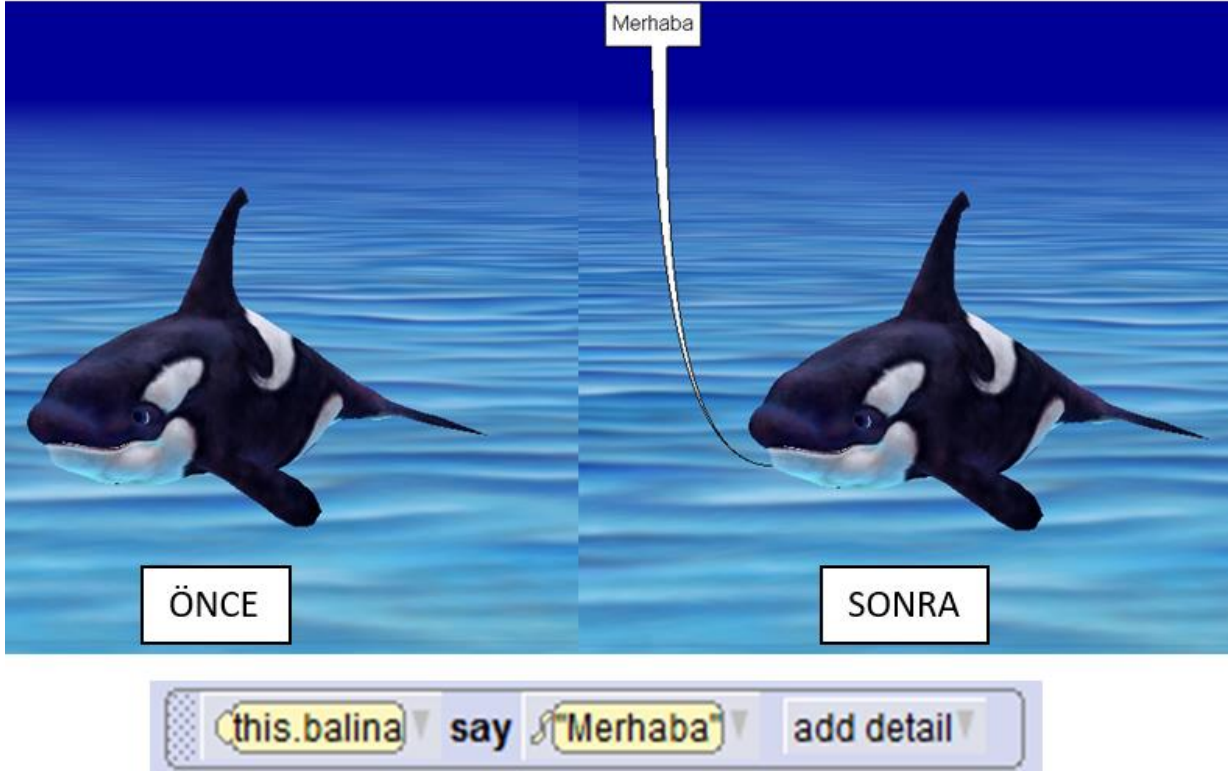
Kedi nesnesinin özellikler paneli



Alice’de farklı amaçlarla kullanılan pek çok hazır metot (fonksiyon ve prosedür) bulunmaktadır. Örneğin, her ne kadar kullanılacak metotlar nesneden nesneye çeşitlilik gösteriyor olsa da “say (söyle)” gibi bazı metotlar tüm nesnelere yer almaktadır. Eğer nesnenizden konuşma baloncuğu çıkartmak gerekiyorsa bu iki metot ve onlara ait parametreler kullanılabilir. Şekil 7’de balina nesnesine yazılan bir kod parçası ve hazırlanan proje çalıştırıldığında çıktısındaki ekran görüntüsüne ilişkin örnek verilmektedir.

Şekil 7

Yazılan say kodu ile balina nesnesinden konuşma baloncuğu çıkması



3.3.2. Üç boyutlu programlama dersinin içeriği ve kapsamı. Bu çalışma, Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümünde, 2013-2014 eğitim öğretim yılı güz döneminde verilen 3B Programlama dersi kapsamında 9 haftalık bir sürede uygulanmıştır. 3B programlama dersi son sınıfta verilen ve programlamayla alakalı olan iki zorunlu seçmeli dersten birisidir. Yani katılımcılar bu ders sonunda Ders sürecinde, ilk olarak Alice 3.1 yazılımına ilişkin tanıtım sunumları yapılmıştır. Ek olarak, katılımcılar beş etkinlik ödevi ve kendi isteklerine göre grup halinde veya bireysel olarak hazırladıkları bir dönem sonu projesi hazırlamıştır. Etkinlik ödevlerinin tamamı sıralı olarak Ek 5'te sunulmuştur. Üç boyutlu programlama dersine ilişkin bilgiler ve kazanımlara ise Ek 6'te yer verilmiştir. Bunun yanı sıra Alice'de hazırladıkları materyallerde kullanacakları eğitim/öğretim teorilerine ilişkin de tanıtıcı bilgiler verilmiş ve bölümlerinde aldıkları veya daha önce karşılaşmadıkları öğrenme

teorilerini hatırlamalarına yönelik küçük etkinlikler düzenlenmiştir. Haftalık olarak yapılan uygulamalara ilişkin bilgiler şu şekildedir:

1. Haftada Alice ara yüzünün incelenmesi, Etkinlik-1 ödevi ve programlama kavramlarına giriş,
2. Haftada metotlara (fonksiyon/prosedürler/döngüler), kamera kullanımı, Etkinlik-2 ödevi ve öğrenme kuramları,
3. Haftada olaylar, Etkinlik-3 ödevi ve kontrol araçları (do in order, do together vb.)
4. Haftada algoritma, Etkinlik-4 ödevi, film gösterimi ve ikili yükleme kuramı
5. Haftada mantıksal (boolean) operatörler, film gösterimi ve Etkinlik-5 ödevi
6. Haftada seslendirme, 2B ve 3B yazılar, yayımlama
7. Haftada dönem sonu projelerine (öykü/oyun/video) ilişkin bilgilendirme
8. Haftada sınıf içi tartışmalarla projelere ilişkin eleştiri ve dönütlerin verilmesi
9. Hafta projelerin teslimi ve sunulması

3.4. Veri Toplama Araçları

Bu araştırmada nitel veriler, yarı yapılandırılmış görüşme formları ile toplanmıştır. Görüşmeler, insanların bir konuya ilişkin ne düşündüklerini, ne hissettiklerini veya eğilimlerinin ne yönde olduğunu bulmak amacıyla gerçekleştirilir. Genellikle verilen cevapların karşılaştırılabildiği veya kıyaslanabildiği bilgilerin edinilmesi için yarı-yapılandırılmış görüşmelerden faydalanılabilir. Nitel araştırmada görüşmelerin yapılan çalışmaların son basamağında kullanılması en uygundur çünkü araştırmacının konuya ilişkin görüşlerini değiştirme eğilimi vardır (Fraenkel & Wallen, 2005). Görüşmelerde, araştırmacılar genel ve açık uçlu sorularını katılımcılara sorarak katılımcıların verdikleri yanıtları kayıt altına alırlar. Daha sonra bu kayıtların transkriptleri çıkarılarak veri analizi için hazır hale getirilir (Creswell, 2012). Yıldırım ve Şimşek'e (2013) göre nitel araştırmacıların

kullandıkları en yaygın veri toplama yöntemlerinden birisi görüşmelerdir ve görüşme formları hazırlanırken soruların anlaşılır ve açık uçlu olması, yönlendirmelerden uzak alt sorulara yer verilmesi ve mantıksal bir sıralamada hazırlanan soruların bir düzen içinde sorulmasına dikkat edilmelidir.

Öğretmen adaylarının üç boyutlu programlamaya ilişkin görüşlerini toplamak amacıyla görüşmeler gerçekleştirmek için yarı yapılandırılmış görüşme formu (Ek 1’de verilmiştir) hazırlanmıştır. Görüşme formları hazırlanırken şu adımlar izlenmiştir:

1. Yurt içi ve yurt dışı alanyazında Alice yazılımı, programlama öğretim araçları, sanal dünyalar ve üç boyutlu programlamaya ilişkin makale, kitap ve süreli yayınların incelenmiştir.
2. Yarı yapılandırılmış görüşme formunun taslağının oluşturulmuştur.
3. Yarı yapılandırılmış görüşme formunun nitel araştırma konusunda uzman 2 kişi tarafından incelenmesi ve soruların düzeltilmesi sağlanmıştır.
4. Ankara’da yer alan bir üniversitede Alice yazılımı üzerinde eğitim almış iki kişi üzerinde görüşmeler pilot çalışma olarak gerçekleştirilmiş ve elde edilen görüşler ışığında görüşme formunun düzeltilmesi sağlanmıştır.
5. Görüşme formuna son hali verilerek, mülakatlara uygun hale getirilmiştir.

Araştırmanın nicel kısmında veriler basit deneysel modele göre toplanmıştır. Nicel veriler, katılımcıların programlamaya ilişkin öz yeterlilik algıları ve demografik bilgileri olmak üzere iki başlık altında toplanabilir. Programlamaya ilişkin öz yeterlilik algısı, Ramalingam ve Wiedenbeck’in “Programlamaya ilişkin Öz yeterlilik Algısı Ölçeğinin” Türkçe ’ye uyarlanmış hali olan 9 maddelik ölçek kullanılarak elde edilmiştir (Altun ve Mazman, 2012). 152 katılımcı üzerinde uygulanan ölçeğin orijinal hali kısaltılarak toplam varyansın %80,81’ini açıklayan 9 madde ve 2 faktörden (basit programlama görevlerini gerçekleştirme ve karmaşık programlama görevlerini gerçekleştirme) oluşan anket formu elde

edilmiştir (Altun & Mazman, 2012). Ölçek Likert tipi sorulardan oluşmuş ve katılımcıların bu sorulara verdikleri cevaplar “Hiç Güvenmiyorum” ile “Tamamen Güveniyorum” arasında değişmektedir. Anket Ek 2’de verilmiştir. Bunun yanı sıra aynı ankette katılımcıların demografik bilgileri (cinsiyet, yaş, sınıf, bölüm), kaç yıldır program yazdıkları, hangi programlama dillerine hâkim olduklarına ilişkin de sorulara yer verilmiştir. Anketin ön-testleri öğrencilere Alice yazılımı tanıtılmadan önce uygulanmış, son-testlerse 9 hafta sonunda, görüşmelerden önce ve projelerinin teslimi aşamasında uygulanmıştır.

Creswell’e (2012) göre nicel veri toplarken gerçekleştirilmesi gereken adımlar şu şekildedir:

1. Çalışmanın katılımcılarının belirlenmesi: Çalışma grubuna ilişkin bilgiler örneklem bölümünde detaylıca açıklanmıştır.
2. Çalışma için izinlerin alınması süreci: Bu araştırma Uludağ Üniversitesi Eğitim Fakültesi Bilgisayar ve öğretim teknolojileri eğitimi bölümünde gerçekleştirilmiştir. Çalışmanın nitel ve nicel veri toplama için gereken izin yazısı Ek 4’te gösterildiği gibidir. Bu araştırmanın katılımcıları arasında 24 kişi hem ön-test hem de son-testi cevaplandırmış ve çalışmaya katılmaya gönüllü olmuştur.
3. Katılımcılara ilişkin bilgi toplarken ne tarz bilgilere ihtiyaç duyulacağıının saptanması: demografik bilgilerinin yanında programlamaya ilişkin öz yeterlilik algılarının belirlenmesine ihtiyaç duyulmuştur.
4. Veri toplamada kullanılacak aracın seçimi: Öz yeterlilik algılarını saptayabilmek amacıyla alanyazın taramasında elde edilen bir ankette yararlanılmıştır.
5. Nicel verinin toplanması süreci: Nicel veriler toplanırken etik unsuruna dikkat edilmelidir ve yukarıdaki dört maddeye uyulmalıdır. Öğretmen adaylarının kişisel bilgileri gizli tutulmuş, ön ve son test anketleri uygulanırken takma isimleri

kullanmaları istenmiştir. Bunun yanı sıra daha önceki bölümlerde açıklanan hususlara da dikkat edilmiştir.

Bu bölümde veri toplama araçlarına ilişkin bilgilere yer verilmiştir. Verilerin toplanması süreci bir sonraki bölümde örneklerle detaylı bir şekilde açıklanmıştır. Toplanan veriler çözümlenirken yapılanlar da yine aşağıdaki bölümde yer almaktadır.

3.5. Verilerin Toplanması ve Çözümlemesi

Görüşme aracılığıyla verilerin toplanması aşamasında, öncelikle, katılımcılara uyacak şekilde bir görüşme takvimi hazırlanmış ve görüşmeler iki haftalık bir süreçte, gerçekleştirilmiştir. 6 öğretmen adayının sorulara verdikleri cevapların ses kayıt alınmıştır. Üç Boyutlu Programlama seçmeli dersine kayıt yaptıran öğrencilerden 4 tanesi yazılı veya sözlü olarak görüşlerini açıklamak istemediklerini belirtmiş, bu yüzden çalışmanın dışında tutulmuşlardır. Görüşmeler iki aşamada gerçekleştirilmiştir. 24 katılımcıdan 19'undan soruların cevapları yazılı olarak alınmıştır. Yani, açık uçlu anket yöntemi ile toplanmıştır. Sonraki aşamada grubu temsil ettiği düşünülen 5 katılımcı ile derinlemesine görüşmeler gerçekleştirilmiştir. Katılımcılardan 1 tanesi 3. sınıf öğrencisi, 4 tanesi ise son sınıf öğrencisinden oluşmaktadır.

Görüşmeler gerçekleştirilmeden önce katılımcılara verdikleri cevapların sadece bu araştırmada kullanılacağı ve görüşmenin amacı ile ilgili bilgiler verilmiştir. Ses kaydı ile ilgili izinlerinin olup olmadığı sorularak onaylarının alınmasının ardından görüşmelere geçilmiştir. Ses kayıt süreleri 20-25 dakika arasında değişkenlik göstermektedir. Görüşmeler, Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü'nün yüksek lisans için kullandığı ve teknoloji odası olarak adlandırılan derslikte yapılmıştır. Bu derslik programının uygunluğu, sessiz olması ve dikkat dağınık etkenlerden uzak olması sebebiyle tercih edilmiştir. Açık uçlu anketlerin toplandığı ve görüşmelerin gerçekleştirildiği tarihler Tablo 2'de gösterildiği gibidir:

Tablo 2

Verilerin toplanması süreci

Etkinlik	Katılımcı İsmi (Takma)			Etkinlik Tarihi
	Saliha	Fethi	Ayşe	
	Deniz	Gaye	Ulaş	29.12.2014 ile
	Hüseyin	Leyla	Bahar	09.01.2015
Açık Uçlu	Ali	Eda	Sena	tarihleri arası
Anketlerin	Zeynep	Ela	Murat	
Uygulanması	Arif	Dilara		
	Sevda	Selahattin		
	Lara			11.01.2015
Görüşmelerin	Ayten			12.01.2015
Gerçekleştirilmesi	Abdullah			13.01.2015
	Fatma			14.01.2015
	Selcan			15.01.2015

Nicel veriler toplanırken kullanılan ve öz yeterlilik algısının ölçümünü ile demografik bilgileri içerisinde barındıran ölçek ön-test olarak 26.09.2014 tarihinde, son-test 26.12.2014 tarihinde basılı olarak öğretmen adaylarına dağıtılmıştır. Ön-testler Alice yazılımına ilişkin bilgiler verilmeden önce ilk dersin başında toplanmış, son-testler ise proje teslimatlarının gerçekleştirildiği gün uygulanmıştır.

Toplanan nitel veriler çözümlenirken içerik analizi yöntemi kullanılmıştır. Yıldırım ve Şimşek'e (2013) göre içerik analizi, elde edilen verilerin belirlenen kavramlar altında açıklanabilmesi ve verilerin arasındaki olası ilişkilerin ortaya konulması amacıyla kullanılır. Başka bir deyişle toplanan veriler kavramsallaştırılır, oluşan kavramlar düzenlenir ve eldeki

verileri açıklayan temalara ulaşılır. Verilerin analizi dört aşamada gerçekleştirilir: (1) Verilerin kodlanması, (2) temaların bulunması, (3) kodların ve temaların düzenlenmesi, (4) bulguların tanımlanması ve yorumlanması. Araştırma kapsamında bu dört aşamada gerçekleştirilen işlemler aşağıda açıklanmıştır:

- 1) Verilerin kodlanması:** Elde edilen görüşme verileri ilk sütunda katılımcı numarası, ikinci sütunda katılımcıların sorulara verdikleri yanıtların mantıksal düşünce birimlerine ayrıştırılmış hali ve üçüncü sütunda da ilgili kodların yer aldığı bir excel tablosu üzerinde kodlanmıştır. Katılımcıların verdikleri cevaplar ilk etapta varsa oluşturulan kodun altına alınıp alınmadığına bakılmış, alınamıyorsa yeni bir kod türetilmiştir. Kullanılan kodların bir kısmı okunan alanyazından alınmış, bir kısmı öğrencilerin vermiş oldukları cevaplardan türetilmiştir. Verilerin içerisinden rastgele seçilen kodlar ve katılımcıların cevaplarının hangi koda girdiğine ilişkin bilgi Tablo 3’de gösterildiği gibidir:

Tablo 3

Verilerin içerisinde rastgele seçilen cevaplara ilişkin kodlar

Katılımcı İsimleri	Katılımcıların Sorulara Verdikleri Cevaplar	Kod
Leyla	3B dünya tasarımını kullanarak öğrencilerimin konuya olan ilgisini arttırabilirim	3B Dünya Kabiliyeti
Eda	Şöyle ki biz ortaöğretim döneminde sürekli kodlarla uğraşıyorduk ve hatalarla karşılaşılırdık bu bize çok sıkıcı geliyordu. Ancak Alice gördüğüm programlama dillerinden hem sağladığı görsellik açısından programlama öğretimini daha eğlenceli ve kalıcı hale getirebilir.	Görsellik
Arif	Yazılım anlamında herhangi bir güçlük yaşamadım. Gayet açık ve amaca hizmet eden bir yazılım, sadece bazı performans sorunları var. Örneğin çalışan proje başka bir bilgisayarda açılmayabiliyor, kasma, donma meydana gelebiliyor. Sonra cüceyi karşımızda görüveriyoruz. Ama tabi bunlar hep kullanıcı tarafından bakılanlar, hiçbir yazılım 4-4lük değil. İlla ki hatalar olacak. Ki bu masaüstü programlar için daha da tolere edilebilir bir durum.	Performans Sorunları

- 2) Temaların bulunması:** Yıldırım ve Şimşek (2013) temaları kodların kategorize edilmesi, ilişkilerini ortaya koyulan kodların daha genel bir başlık altında toplanması süreci olarak tanımlamaktadır. Bu çalışmada kullanılan temalar mülakat sorularına göre organize edilmiş ve beş başlıkta toplanmıştır: (1) Üniversitelerde Alice ile programlama öğretimi (2) İlk ve orta Dereceli okullarda

Alice ile programlama öğretimi, (3) Alice yazılımında karşılaşılabilecek güçlükler, (4) Programlama öğretiminde Alice yazılımının güçlü yönleri ve (5) Programlama öğretiminde nitelikli öğretmen yetiştirme

- 3) Kodların ve temaların düzenlenmesi:** İlk kodlamanın sonucunda, toplamda beş başlık altında 47 adet kod türetilmiştir. Daha sonra ikinci bir kodlama yapılarak sayıları az olan kodlar veya tek bir kod altında toplanabilen alt kodlar birleştirilerek birbirinden farklı 21 koda indirilmiştir. Yukarıda listelenen beş temada ise toplamda 27 kod bulunmaktadır. Oluşan bu kodlara bulgular başlığı altından ulaşılabilir. Bu işlem yapılırken dikkat edilen hususlar, kodların ve temaların sade ve anlaşılır olması, kodların düşünce birimlerini açıklayıcı olması ve temaların kodları genel başlıklar altında toplayabilir olması olarak listelenebilir.
- 4) Bulguların tanımlanması ve yorumlanması:** Bulguların tanımlanması ve yorumlanmasında, nitel araştırmada daha önce bahsedilen içerik analizi kullanılmıştır. Yıldırım ve Şimşek'e (2013) göre bulguların betimlenmesi ve yorumlanmasında kullanılan içerik analizinde, araştırmacının ilk etapta yapması gereken bulgularının ayrıntılı bir şekilde betimlenmesi işlemidir. Daha sonra araştırmacı yorumlarını katarak bulgu ile ilgili yorum ve açıklamalarını sunabilir. Yine, bu bölümde araştırma problemleri ile bağlantının da net olarak ortaya konması gerekmektedir.

Ön-test ve son-test olmak üzere iki aşamada toplanan nicel verilerin analizinde IBM SPSS 21 yazılımından faydalanılmıştır. Parametrik olmayan testlerden biri olan Wilcoxon bağımlı iki örneklem testi ile öğretmen adaylarının öz yeterlilik algıları değerlendirilmiştir. Bu testin seçilmesindeki temel neden örneklemdaki bireylerin sayısındaki azlıktır. Örneğin, Ross (2004), Baykul (1999), Çepni (2014), katılımcı sayılarının 30'dan az olduğu durumlarda parametrik testlerin kullanılmamasını önermektedir. Wilcoxon bağımlı iki örneklem testi bu

öneriler dikkate alınarak (Baykul, 1999; Çepni, 2014; Ross, 2004) verilerin normal dağılım gösterip göstermediğine bakılmaksızın uygulanmıştır. Wilcoxon testi uygulanırken, SPSS yazılımına geçirilen verilerdeki ön-test ve son-test puanları toplanarak aralarında anlamlı bir fark olup olmadığına bakılmıştır.

3.6. Güvenirlilik

Bulguların tanımlanması ve yorumlanması aşamasına geçilmeden önce organize edilen veriler içerik analizi, güvenilirliği sağlamak için iki yardımcı araştırmacıya daha gönderilmiştir. Yardımcı araştırmacılardan verileri ilk kodlamadan bağımsız olarak kodlamaları istenmiştir. Yardımcı araştırmacılara gönderilen ve onlar tarafından kodlanan Excel tablosunun örnek bir kesiti Tablo 4’te verilmiştir. Yardımcı araştırmacılardan biri bilgisayar ve öğretim teknolojileri alanı mezunu olup 4 yıldır bilişim öğretmenliği yapmaktadır. Diğeri ise aynı bölümde okuyan 4. sınıf lisans öğrencisidir. Katılımcı isimleri gönderilen tablodan kaldırılmış, oluşturulan kodlar ikinci sütundan başlayacak şekilde, ilk satıra yan yana sırasıyla yazılmıştır. Katılımcıların cevapları ise ilk sütunda alt alta rastgele sıralanmıştır. İlk Excel satırı dondurularak her veri satırı (katılımcı cevabı) için yardımcı araştırmacılardan oluşturulan kodlardan yalnızca bir tanesini seçmeleri istenmiştir. Yardımcı araştırmacılar kodlamayı, katılımcıların verdiği cevap hangi koda uyuyorsa ilgili satır ve sütunun kesiştiği hücreye “1” yazarak gerçekleştirmiştir. Katılımcıların verdikleri cevaplar yalnızca bir koda uyacak şekilde düzenlenmiştir. Bunun yanı sıra kodlamayı kolaylaştırmak için bir kod kitapçığı hazırlanmış ve ilgili kodlara ilişkin tanımlamalar, örnekler ve kod isimleri bu kitapçıkta gösterilmiştir. Yardımcı araştırmacılar için hazırlanan kod kitapçığı Ek 3’te gösterilmiştir.

Tablo 4

Arařtırmacılara gnderilen excel tablosunun rneęi

Veri	Kullanım Biimi	Kalıcılık	Grsellik
Bir de renmede kalıcılık saęlanmış olur. ğrenci o bilgiyi hatırlamada ok sorun yařamaz.		1	
Alice olduka geniř bir grsel ktphaneye sahip ve ğrencilere birok hikaye oluřturma olanaęı saęlıyor.			1
Alice'deki grsellik imkanları ile programlamanın mantıęını eęlenceli ve kolay bir řekilde kavramalarını saęlar.			1
Algoritmayı normal kaęıt zerinde oluřturmak yerine byle grsel bir program sayesinde daha anlamlı hale getirebilirim ğrencilere diye dřnyorum.	1		
Ortağretim ğrencileri bu iřlere meraklı ğrencilerdir.			
Belirli oyunlar tasarlamayı gsteririm. Mesela bir oyun stnden dersime bařlarsam ve onların da yapmasını saęlarsam ilgilerini ekecektir.	1		

Yardımcı arařtırmacılar verilerin kodlaması iřlemini tamamladıęında arařtırmanın gvenirlięini ortaya koymak zere Cohen Kappa Katsayısından faydalanılmıştır. Bu ařamada, arařtırmacının kodları satırlara yardımcı arařtırmacıların kodları stunlara gelecek řekilde ikili matris ıkartılmıştır. Her iki yardımcı arařtırmacı iin ayrı ayrı olarak bu iřlem tekrarlanmış ve Tablo 5, Tablo 6, Tablo 7, Tablo 8, Tablo 9'da gsterilmiştir.

Tablo 5

Üniversitelerde Alice ile programlama öğretimi

Araştırmacı	Yardımcı Araştırmacı 1				Yardımcı Araştırmacı 2					
<u>Kodlar</u>	<u>3B Dünya Tasarımı Deneyimi</u>	<u>Derse İlgili ve Katılım</u>	<u>Kalıcılık</u>	<u>Ders Başarısı</u>	<u>Diğer Görüşler</u>	<u>3B Dünya Tasarımı Deneyimi</u>	<u>Derse İlgili ve Katılım</u>	<u>Kalıcılık</u>	<u>Ders Başarısı</u>	<u>Diğer Görüşler</u>
3B Dünya Tasarımı Deneyimi	15	5	1		1	12	1		4	7
Derse İlgili ve Katılım	4	24		1	1		24		4	4
Kalıcılık	2	1	25		1			28		1
Ders Başarısı				11					11	
Diğer Görüşler	2				6				1	8

Tablo 6

İlk ve orta dereceli okullarda Alice ile programlama öğretimi

Araştırmacı	Yardımcı Araştırmacı 1					Yardımcı Araştırmacı 2						
	<u>Kullanım Biçimi</u>	<u>Kalıcılık</u>	<u>Görsellik</u>	<u>Öğrenci Gelişimi</u>	<u>Öğrencilerin Derse Tutumu</u>	<u>Diğer Görüşler</u>	<u>Kullanım Biçimi</u>	<u>Kalıcılık</u>	<u>Görsellik</u>	<u>Öğrenci Gelişimi</u>	<u>Öğrencilerin Derse Tutumu</u>	<u>Diğer Görüşler</u>
<u>Kodlar</u>												
Kullanım Biçimi	26			2		1	20			2		7
Kalıcılık		4					2		1	1		
Görsellik		1	23		1			22	1	2		
Öğrenci Gelişimi	1			7		1		2	4	2		1
Öğrencilerin Derse Tutumu	2		1		24	3		1		29		
Diğer Görüşler	1					4						5

Tablo 7

Alice yazılımında karşılaşılabilecek güçlükler

Araştırmacı	Yardımcı Araştırmacı 1						Yardımcı Araştırmacı 2								
	<u>Nesne Konumlandırma</u>	<u>Dil</u>	<u>Performans Sorunları</u>	<u>Nesne Galerisinin</u>	<u>Zenginleştirilmesi</u>	<u>Kamera Kullanımı</u>	<u>Metotlar</u>	<u>Diğer Görüşler</u>	<u>Nesne Konumlandırma</u>	<u>Dil</u>	<u>Performans Sorunları</u>	<u>Nesne Galerisinin</u>	<u>Zenginleştirilmesi</u>	<u>Kamera Kullanımı</u>	<u>Metotlar</u>
<u>Kodlar</u>															
Nesne Konumlandırma	11							7					1		2
Dil		10					1		11						
Performans Sorunları			16				1			16					
Nesne Galerisinin															
Zenginleştirilmesi				3	12	1			2			14	1		
Kamera Kullanımı						7							7		
Metotlar							7							7	
Diğer Görüşler		1	5	1		2	3		1				3	9	

Tablo 8

Programlama öğretiminde Alice yazılımının güçlü yönleri

Araştırmacı	Yardımcı Araştırmacı 1					Yardımcı Araştırmacı 2								
<u>Kodlar</u>	<u>Görsellik</u>	<u>Sürükle-Bırak Yapısı</u>	<u>Kamera Kullanımı</u>	<u>Programlama Kavramları</u>	<u>Hazır Kodlar ve Yazım Hataları</u>	<u>Grup / Bireysel Çalışma</u>	<u>Diğer Görüşler</u>	<u>Görsellik</u>	<u>Sürükle-Bırak Yapısı</u>	<u>Kamera Kullanımı</u>	<u>Programlama Kavramları</u>	<u>Hazır Kodlar ve Yazım Hataları</u>	<u>Grup / Bireysel Çalışma</u>	<u>Diğer Görüşler</u>
Görsellik	23						1	21		1	1	1		3
Sürükle-Bırak Yapısı		25					3		24		2	1		
Kamera Kullanımı			1				1			1	1			
Programlama Kavramları	2			21	2			2	1		20	1		
Hazır Kodlar ve Yazım Hataları					8							7		1
Grup / Bireysel Çalışma			1	1		30	1	1					29	2
Diğer Görüşler	1	1		2	4		4	1			1	3		7

Tablo 9

Programlama öğretiminde nitelikli öğretmen yetiştirme

Araştırmacı	Yardımcı Araştırmacı 1			Yardımcı Araştırmacı 2		
<u>Kodlar</u>	<u>Kullanım Biçimi</u>	<u>Kalıcılık</u>	<u>Görsellik</u>	<u>Kullanım Biçimi</u>	<u>Kalıcılık</u>	<u>Görsellik</u>
Kullanım Biçimi	37	2	1	37		3
Kalıcılık	1	25		1	24	1
Kalıcılık	1	1	24			26

Matrislerin çıkartılması işleminden sonra, Cohen'in kapa katsayısı hesaplanmıştır. Her tablo bir ana temayı gösterecek şekilde hazırlanmıştır. Bu temalar, (1) Üniversitelerde Alice ile programlama öğretimi (2) İlk ve orta dereceli okullarda Alice ile programlama öğretimi, (3) Alice yazılımında karşılaşılabilecek güçlükler, (4) Programlama öğretiminde Alice yazılımının güçlü yönleri ve (5) Programlama öğretiminde nitelikli öğretmen yetiştirme olmak sıralanmış ve yukarıdaki tablolarda gösterilmiştir. Koyu renkle verilmiş olan rakamlar eşleşme değerlerini göstermekte, eşleşmeyen durumlarda ise tablodaki ilgili yerlere araştırmacıların ne düşündüğüne ilişkin rakamlar eklenmiştir. Örneğin, Tablo 1'e bakılacak olursa derse ilgi ve katılım konusunda 24 düşünce birimi (idea unit) konusunda bir eşleşmeden söz edilebilir. Ayrıca, araştırmacının derse ilgi ve katılım dediği 4 düşünce biriminin yardımcı araştırmacı 1 tarafından "3B Dünya Tasarımı Deneyimi" olarak işaretlediği görülebilir. Sim ve Wright'a (2005) göre eldeki verilere (katılımcıların

cevaplarına) bu şekilde matrisler oluşturulduktan sonra aşağıdaki adımlar uygulanarak Cohen Kappa katsayısı hesaplanabilir:

1. Her bir satır ve sütundaki sayılar toplanmıştır.
2. Tablodaki tüm sayılar toplanarak toplam cevapların sayısı (N) elde edilmiştir.
3. Tablolarda kalın harflerle verilen eşleşmeleri (agreement) gösteren sayılar toplanmış ve a simgesi ile beşinci adımda gösterilmiştir.
4. Tablolarda kalın harflerle verilen eşleşmeleri gösteren sayılara ilişkin değerler (ef) şu şekilde hesaplanmıştır:

$$ef = \frac{\text{satır toplamı} * \text{sütun toplamı}}{\text{tablodaki tüm sayıların toplamı (N)}}$$

5. Daha sonra her bir kod için oluşan ef değerlerin toplamı hesaplanmıştır
- $$eftoplam = ef1 + ef2 + \dots + efn$$
6. Cohen Kappa katsayısı (K) hesaplanırken kullanılan formül ise şu şekildedir:

$$K = \frac{a - eftoplam}{N - eftoplam}$$

7. Hesaplanan Cohen Kappa katsayısı ile ilgili yorumlar yapılırken, katılımcıların cevaplarındaki eşleşmenin düştüğü aralığa göre değerlendirme yapılır.
 - a. $K < 0$ ise ilişki yoktur (poor).
 - b. $K 0.01 - 0.20$ arası ise zayıf bir eşleşme (slight) vardır.
 - c. $K 0.21 - 0.40$ arası ise vasat bir eşleşme (fair) söz konusudur.
 - d. $K 0.41 - 0.60$ arası ise ortalama bir eşleşmeden söz edilebilir.
 - e. $K 0.61 - 0.80$ arası ise önemli derecede eşleşme (substantial) vardır.
 - f. $K 0.81 - 0.99$ arası ise iyi bir eşleşme (almost perfect) vardır.

İlk yardımcı araştırmacının 5 ana temadaki cohen kappa katsayıları 0.75, 0.82, 0.78, 0.81, 0.90'dır. İkinci yardımcı araştırmacının ise 0.76, 0.74, 0.84, 0.79, 0.90 çıkmıştır.

Sonuçlara bakıldığında ilk yardımcı araştırmacı için 2, 4 ve 5. Temalarda iyi bir eşleşme söz

konusuyken, 1. ve 3. Temalarda önemli derecede uyuşma vardır. İkinci yardımcı araştırmacı için ise 3. ve 5.temada iyi bir eşleşme varken, 1. , 2. Ve 4. temalarda önemli derecede kodlamalarda uyuşma olduğu ortaya çıkmıştır.

Araştırmanın nicel kısmında uygulanan ölçeğe ilişkin, Mazman ve Altun (2012) uyarladıkları programlamaya ilişkin öz yeterlilik ölçeğinin tümü için Cronbach Alpha katsayısını 0,928 olarak hesaplamıştır ve bu ölçeğin güvenilir olduğunu savunmaktadır. Bu ölçeğin iki alt düzeyi söz konusudur. Basit programlama görevleri düzeyinde ek-1’de yer verilen anketin ilk üç sorusu, karmaşık programlama görevleri düzeyi ise son 6 sorudan oluşmaktadır. Bu iki faktör için güvenilirlik katsayıları sırasıyla 0,907 ve 0,943’dür.

Bu çalışmada ön testte tüm ölçek için güvenilirlik katsayısı 0,904 olarak ortaya çıkmıştır. Son-test sonuçlarının güvenilirliğine ilişkin yapılan analiz sonucunda tüm ölçeğin güvenilirlik katsayısı 0,945 olarak hesaplanmıştır. Ön-testte basit programlama görevleri için 0,876, karmaşık programlama görevleri için 0,882 olarak hesaplanmıştır. Son-testte ise bu değerler basit ve karmaşık programlama görevleri için sırasıyla 0,884 ve 0,932’dir.

4. Bölüm

Bulgular

Bir önceki bölümde verilerin toplanması ve veri toplama araçlarına ilişkin bilgilere yer verilmiştir. Bahsi geçen süreçte elde edilen nitel ve nicel veriler, bu bölümde 5 ana başlık altında incelenmektedir. Öğretmen adaylarının görüşlerindeki benzerliklerin çok olmasından dolayı ilk iki tema olan (1) üniversitelerde Alice ile programlama öğretimi ile (2) ilk ve orta dereceli okullarda Alice ile programlama öğretiminin birleştirilmesiyle ilk ana başlık oluşturulmuştur. Son dört başlık ise sırasıyla, Alice yazılımında karşılaşılabilecek güçlükler, programlama öğretiminde Alice yazılımının güçlü yönleri, programlamaya ilişkin nitelikli öğretmen yetiştirme konusunda öğretmen adayı görüşleri ve öz yeterlilik algısıdır. Her bir başlık altında kodlara ilişkin bilgilere yer verilmiş, öğretmen adaylarının görüşlerinden alıntılar yapılmış ve araştırmacının konuya ilişkin yorumlarına yer verilmiştir.

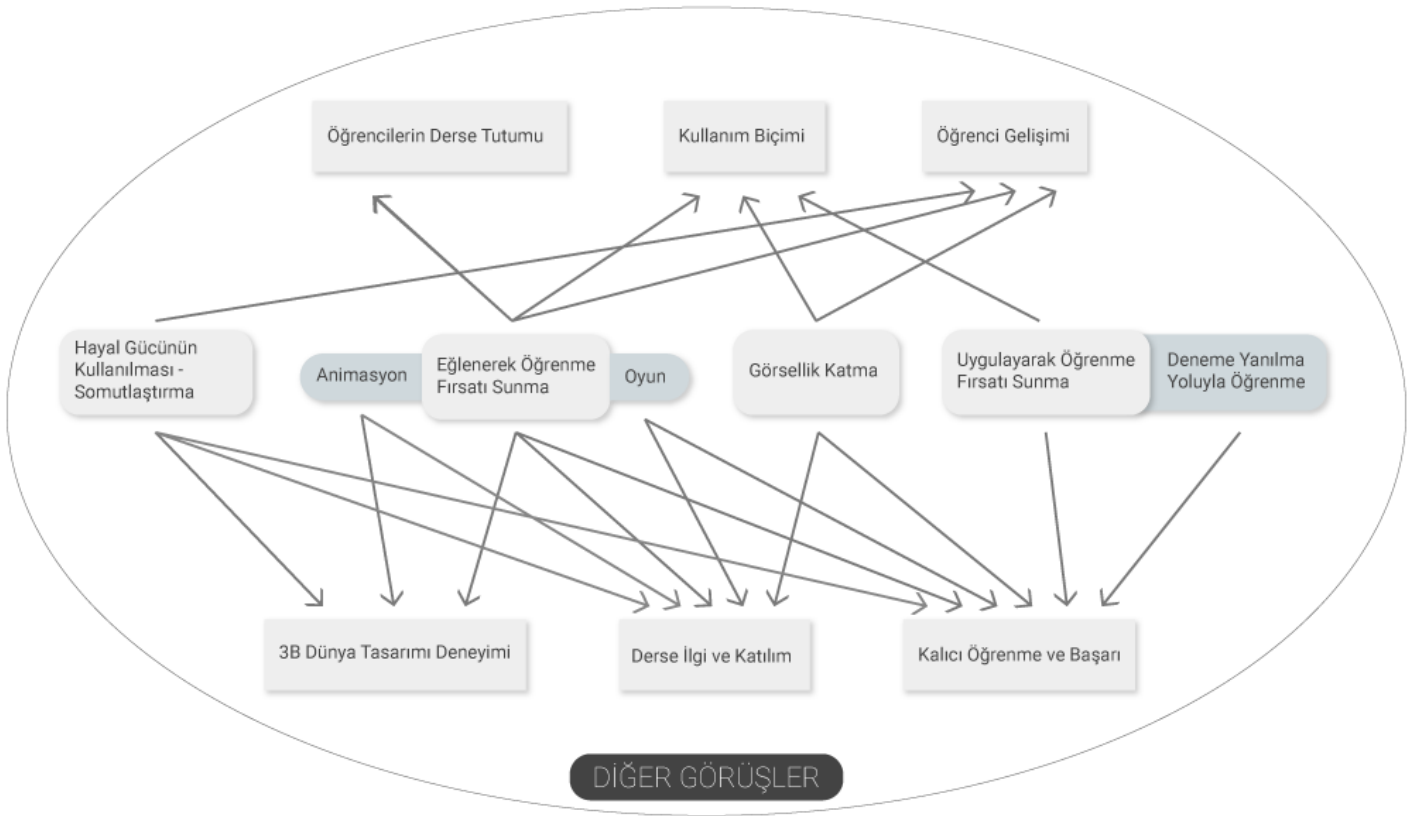
4.1. Okullarda Alice ile Programlama Öğretimi

Bu bölümde araştırma sorularından “Programlama kavramlarının öğretiminde üç boyutlu ortamların kullanımına ilişkin öğretmen adaylarının görüşleri nelerdir?” sorusunun iki alt sorusuna yanıt aranmaktadır. Bu bölüm Bulgular bölümünün başında bahsi geçen “üniversitelerde programlama öğretimi” ile “ilk ve orta dereceli okullarda Alice ile programlama öğretimine” ilişkin görüşlerin birleştirilmesiyle oluşturulmuştur. Birleştirilmesindeki sebep olarak, öğretmen adaylarının kendilerini müstakbel öğrencilerinin yerine koymaları ve üniversitelerde programlama öğretimi ile ilk ve ortaöğretimdeki programlama öğretimini özdeşleştirmeleri gösterilebilir. Bu konu hakkındaki öğrenci görüşleri Şekil 8’de verilen nedenler temel alınarak beş başlık altında toplanmıştır: (1) eğlenerek öğrenme, (2) programlamaya görsellik katma, (3) uygulayarak öğrenme, (4) hayal gücünün kullanılması– somutlaştırma ve (5) diğer görüşler. Bu bölümde cevap aranan araştırma sorusunun alt soruları ise şu şekilde listelenebilir:

- a. Bu çalışma kapsamında elde ettikleri programlama ve 3B dünya tasarımı deneyimlerinin gelecekte verecekleri derslere nasıl etki edeceğini düşünmektedirler?
- b. Ortaöğretimde programlama öğretimi yapılırken Alice nasıl kullanılabilir? Ortaöğretim öğrencilerinin bu programla öğretimi nasıl karşılayacaklarını düşünüyorlar?

Şekil 8

Yeniden değerlendirme sonucunda ortaya çıkan kodlar ve aralarındaki ilişkiler



Şekil 8’de ortada yer alan beş temayla, öğretmen adaylarının düşüncelerinin birimlere ayrılması sonucu oluşan kodlar arasındaki ilişkiler tek yönlü oklarla gösterilmektedir. Örneğin görsellik katma konusuyla derse ilgi ve katılım, kalıcı öğrenme ve başarının, kullanım biçimi ve öğrenci gelişimi temaları arasında bu tarz bir ilişki olduğu söylenebilir. Yani, daha

özelleştirecek olursak öğretmen adayları üç boyutlu yazılımların programlamaya görsellik katmasından bahsederken aynı zamanda derse ilgi ve katılıma da değinmiştir. Öğrencilerin derse tutumu ile birlikte bahsedilen yalnız bir tema bulunurken, kalıcı öğrenme ve başarı ile birlikte olan beş farklı unsur vardır. Öğretmen adaylarının görüşleri doğrultusunda oklar bir temadan diğerine gönderilmiş ve ilişkiden bahsedilmeyen temalarla ortadakiler ilişkilendirilmemiştir. Sıradaki alt başlıklarda temalar arası ilişkilere dair bulgulara detaylı olarak yer verilmiştir.

4.1.1. Eğlenerek öğrenme. Öğretmen adayları eğlenerek öğrenme konusunda Alice'in 3B dünya tasarımı deneyimini, derse ilgi ve katılımı, kalıcı öğrenme ve başarıyı, kullanım biçimini, programlamaya karşı olumlu tutum geliştirmeyi etkileyeceğini düşünmektedir.

Katılımcılar üç boyutlu dünya tasarımı deneyimlerinin dersleri daha eğlenceli hale getireceğini düşünmektedir. Ayşe bu konuda “Tabi görsel içerikli bir program gördüklerinde de hemen merak edip kullanmak istiyorlar. Bu 3B programı kullanmaları onlar için bir ders olarak değil de bir oyun olarak görülebilir.” demiştir. Lara ise “Bu derste edindiğim beceriler, görsellik olduğu için ve 3B animasyon tarzı olduğu için, programlama derslerini alışılmışın dışında ve eğlenceli hale getireceğini düşünüyorum.” diyerek düşüncelerini dile getirmiştir.

Eğlenerek öğrenme konusunda, öğretmen adaylarının üzerinde durduğu bir diğer husus derse ilgi ve katılım konusudur. Örneğin, Ulaş “Güzel animasyonlarla dersi eğlenceli kılarak eğlenerek öğrenmelerinin katılımlarını arttıracığını düşünüyorum.” derken, Leyla ise “Programlamayı öğrencilere yaş düzeyine uygun, renkli ve eğlenceli bir dünya olarak sunabileceğimi düşünüyorum.” diyerek görüşlerini gerekçelendirmiştir. Katılımcılara göre eğlenerek öğrenme derse ilgi katılımın sebebidir.

Murat “Alice içerisinde barındırdığı görsellik ve eğlence ile öğrencilere programlama kavramlarını anlama konusunda çok etkili olacaktır. Öğrendiklerimizin kalıcılığını etkiler.”

diyerek görselliğe dikkat çekmiş ve görselliğin bulguların genelinde olduğu gibi programlama öğretiminde önemli bir yere sahip olduğunu ortaya koymuştur.

Deniz öğrencilerin eğlenerek öğrenmesine ilişkin ilginç bir görüşten bahsetmiş ve “Öğrencilerin ders başarısını olumlu yönde etkiler çünkü eğlenerek öğrenir programı hatta programlama öğrendiğinin farkında bile olmayabilir” diyerek eğlenerek öğrenmenin başarıyı arttıracaklarını savunmuştur. Yani, eğlenerek öğrenme ile öğrencilerin derslerinde başarılı olması arasında bir bağ olduğu düşünülebilir.

Kullanım biçimine ilişkin en çok üstünde durulan konu Alice’in özellikle programlamaya giriş amacıyla kullanılabileceğidir. Alanyazın taramasında da yer verildiği üzere, Alice’in tasarımcıları bu yazılımı programlamaya giriş amaçlı ve ilk/orta dereceli okullarda programlama öğretimi amacıyla kullanımını öngörmektedir. Bu açıdan öğretmen adaylarının görüşleri daha önce yapılan çalışmalar ve yazılımın doğası ile paralellik göstermektedir. Örneğin, Arif:

Gördüğüm ve okuduğum kadarıyla öğrencilere, özellikle de ortaöğretim öğrencilerine karmaşık kod blokları haliyle karışık geliyor. (Haklılar) Bundan kaynaklı az önce belirttiğim gibi çocuklara programlamayı oyunla, animasyonla öğreten çeşitli platformlar ortaya çıkmaya başladı. Eğer aktarılmak istenen güzel bir senaryo ile aktarılabilirse programlamaya giriş amaçlı Alice ortamı kullanılabilir.

diyerek programlamaya giriş amaçlı kullanılabileceğinden ve programlama öğretiminde yeni trendlerden bahsetmiştir. Gaye “Öğrencilere programlamanın temelini verilmesinde kullanılabilecek bir program.” derken, Dilara “Alice programlama öğretilmeden giriş amaçlı kullanılabilir bu sayede yazılım mantığının temelini hem keyif alarak hem de çok zorlanmadan kavramış olurlar.” diyerek Alice’i nasıl kullanacaklarını belirtmişlerdir.

Öğretmen adaylarının bu görüşleri incelendiğinde, Alice’teki eğlenerek öğretim potansiyeli

ile Türkiye’deki her düzeyde okulda, Alice’in özellikle programlamaya giriş derslerinde kullanılabileceği göze çarpmaktadır.

Alice’in yapısal olarak animasyon tarzına sahip olması derse ilgi ve katılımı arttıran sebeplerden bir diğeridir. Mayer ve Moreno’ya (2002) göre animasyon, hareket, gösterim ve resimlerin peş peşe sıralanması olarak tanımlamaktadır. Alice’de ise animasyonlar hazırlanırken çoğunlukla resim gibi iki boyutlu nesnelere yerine üç boyutlu nesnelere yararlanılır (Herbert, 2010). Animasyonlar konusunda Bahar “Bir bilişim teknolojisi öğretmeni olarak, bilgisayarın bulunduğu bir ortamda öğrencileri zapt etmek oldukça zor fakat hazırlanan yaratıcı animasyonlar öğrencilerin derse olan ilgisini arttıracaklarını düşünüyorum.” diyerek öğretmenlerin animasyonları sınıfta göstererek öğrencilerin ilgisini çekebileceğinden bahsetmiştir. Ulaş ise “Öğrencilere öğretmek istediğim konular hakkında animasyonlar hazırlatarak öğrencilerin öğrenmesini kolaylaştırmak için kullanabilirim.” diyerek animasyonların önemine dikkat çekmiştir. Katılımcıların görüşleri incelendiğinde, eğlenerek öğrenme grubu altında incelenebilecek animasyon tarzının programlamayı anlama, ilgilerini çekme ve öğrenmeyi kolaylaştırma açısından önemli bulunduğu görülmektedir.

Hüseyin ise “Stajda 6. Sınıf öğrencilerden birinin teneffüs esnasında laboratuvarında yanıma gelip öğretmenim bilgisayarda oyun oynayabilir miyim diye sorarak SketchUp programında uğraşmasını örnek olarak verebilirim. Öğrenci Alice gibi programları bir oyun olarak görüyor ve programda uğraşıyor.” diyerek ara yüzü itibarıyla öğrenciler tarafından oyun olarak algılanabilecek öğretim teknolojilerinin önemini vurgulamıştır. Özellikle genç neslin çok küçük yaşlarda bilgisayar ve multimedya materyallerle karşılaşmaya başlaması, oyunlarla öğretim noktasında bir arayışa gidilmesine yol açmıştır. Öğretmen adayları da bu konuda son yıllarda geliştirilen yazılımlarda bu açıdan önemli değişiklikler olduğunu vurgulamıştır.

Son olarak katılımcılar (n=20), öğrencilerin programlamaya karşı olumlu tutum geliştirilmesi ile eğlenerek öğrenme arasında bir ilişki kurmuştur. Tutumun ne anlama geldiği konusunda Senemoğlu (2012, s.410) üç farklı düzeyden bahsetmektedir

Birincisi din, dil, ırk, cinsiyet farkı gözetmeksizin insanlara yardım etmek, saygılı olmak, hoşgörülü olmak gibi sıralanabilir. İkincil düzey tutumlar, çeşitli durumlarla ilgili olumlu tercihleri yansıtmaktır. Öğrenmekten hoşlanmak bu düzeye örnek olarak verilebilir. Üçüncü düzey tutum gurubu ise, vatani sevmek, toplumun ihtiyaçları ve amaçları ile ilgilenmek bu grupta sayılabilir.

Bu bölümde öğrencilerin derse tutumu ikincil düzey olarak değerlendirilmiş ve sorulan sorulara ilişkin görüşler bu çerçevede incelenmiştir. Saliha “Programlama zevkli hale gelir. Bizde de aynı şey oldu.” demiştir. Abdullah ise “Bir kişinin anlatmasını dinlemek yerine öğrencilerin hepsi aynı anda görsel bir ortamda uygulama yapsa daha rahat ve zevkli hale gelir yani.” diyerek görselliğin ve uygulama imkanının dersi daha zevkli hale getireceğinden bahsetmiştir. Bu konuda görüş bildiren öğretmen adaylarının neredeyse hepsi “... bu sayede olumlu tutum geliştirirler.” İfadesi ile cevaplarını sonlandırmıştır. Yani, katılımcılar programlamaya karşı olumlu tutum geliştirme konusunda eğlenerek öğrenmenin önemli bir unsur olduğunu belirtmektedir.

4.1.2. Programlamaya görsellik katma. Görsellik konusu giriş bölümünde detaylı bir şekilde tanıtılmıştı. Kısaca hatırlatmak gerekirse görselliği ön plana alan (visualization) programlama araçları, kodlarla etkili çalışmayı, hatayı önlemeyi veya azaltmayı, gerçek yaşama yakın programlama ve sonuç alma deneyimi sunmayı sağlar. Bu sayede kod bloklarını anlamlandırma işi kolaylaşmaktadır. Alice programlamaya görsellik kattığından dolayı, katılımcılar kullanıcıların derse ilgi ve katılımlarının artacağı, akılda kalıcılığa ve

başarıya etkisi olacağı ve Alice’i derslerde kullanım biçimini etkileyeceği yönünde görüşler ortaya koymuşlardır. Aşağıda sırasıyla bunlar incelenmiştir.

Görsellikle en çok ilişkilendirilen konu derse ilgi ve katılımdır. Ali “Görsel programlamanın öğrencilerin ilgisini çekmesi açısından yararlı olacağını düşünüyorum.” derken, Hüseyin “Sırf yazı satırlarından oluşan kod satırları itici, sıkıcı gelebiliyor. Alice işin içerisine görselleri de katarak kod yazmanın sıkıcılığını ve iticiliğini ortadan kaldırıyor.” diyerek programlamaya ilgi konusunda görselliğe vurgu yapmaktadır. Sevda ise “Görsel ve içerik açısından birçok öğrencide merak duygusu uyandıracığına bunun sonucunda derse katılım oranının artacağına eminim.” diyerek görselliğin derse katılımı olumlu yönde etkileyeceğinden bahsetmiştir.

Akılda kalıcılığın nedenlerine ilişkin görüşlerin başında da görsellik gelmektedir. Örneğin Ela “Evet [görsellik] programlama kavramlarının akılda kalmasını sağlar. Çünkü sözel olarak anlatınca unutulması daha kolay olur. Görseller üzerinden ilgi çekici karakterler ile bağlantı kurarsa öğrenci daha zor unutulur.” diyerek görselliğin kalıcılık üzerindeki önemine dikkat çekmektedir. Ayşe ise “Programlama dersi yalnız sözel ezberle sınırlı kalmayıp görsellerle de birleşince anlatım daha meraklandırıcı ve öğretim kalıcı olur.” diyerek öğrencileri ezberden kurtarabileceğinden ve ilgi çekici anlatımların yapılabileceğinden bahsetmiştir.

Birçok öğretmen adayı başarının sebepleri arasında göstermiştir. Ela “Sözel olarak ders anlatımının görsel işitsel öğeler kullanılarak hazırlanmış uygulamalar aracılığıyla anlatmanın daha etkili ve verimli olacağını ve başarının artacağını düşünüyorum.” diyerek ders başarısının görsel-işitsel öğelerin yoğunlukta olduğu Alice sayesinde artacağı görüşünü savunmuştur. Saliha “Sonuçta görsellikler ön planda olduğu için, ders başarılarının ilgilerine bağlı olarak artacağını düşünüyorum.” diyerek görselliğin öğrencilerin derse olan ilgisini çekeceğini ve bunun da ders başarılarını arttıracığını söylemiştir.

Öğretmen adaylarına göre görsellik aynı zamanda Alice'in programlama öğretiminde kullanım biçimini doğrudan etkileme potansiyeline sahiptir. Programlamaya giriş konularından bir tanesi algoritmalar ve onların kullanımı konusudur. Algoritma pek çok alanda farklı anlamlara gelmekle birlikte, özetle bir probleme çözüm yolu bulunması için izlenilebilecek yol olarak tanımlanabilir. Öğretmen adayları özellikle programlamanın temeli olan algoritma öğretimi konusunda Alice'in kullanılabileceğinden bahsetmişlerdir. Örneğin, Abdullah "Programlama mantığını anlamak açısından, algoritma oluşturuyoruz ya mesela, algoritmayı normal kâğıt üzerinde oluşturmak yerine böyle görsel bir program sayesinde daha anlamlı hale getirebilirim öğrencilere diye düşünüyorum." diyerek Alice'in görselliğine ve algoritma öğretiminde kullanılabileceğine vurgu yapmıştır.

4.1.3. Uygulayarak öğrenme. Programlamayı uygulayarak yapma şansına sahip olduklarından dolayı öğretmen adayları, kalıcı öğrenme ve ders başarısının, kullanım biçiminin etkileneceğinden bahsetmişlerdir.

Bahar "Sadece çocuklar değil bizlerde sayfalar dolusu kod yazma zorunluluğu olmadan görsel bir şekilde hem nesnel kullanıyoruz hem de o nesnelere hareketlerini kendimiz kullandığımız kodlarla belirliyoruz. Bu yüzden akılda kalıcılığı sağlayacağını düşünüyorum" diyerek uygulayarak öğrenmenin ve derslerde aktif olmanın önemine vurgu yapmıştır. Sena "Eğer etkinliklere gereken önem verilirse programlama kavramlarının kalıcılığı sağlanır. Çünkü üzerinde durulan her şey öğrenilir. Yapılan her örnekte deneme yanılma yoluyla öğrenmeleri de artar. Kendi yaptığımız faaliyetler, bir başkasının yaptığını anlamaya çalışmalarından daha öğreticidir." şeklinde düşüncelerini aktarmıştır. Etkinlik ve faaliyetler kelimeleri ile bu çalışma esnasında yaptıkları uygulamaları kast ettiği görülebilmektedir. Tabi bunun yanı sıra bireysel çalışmanın da önemini altını çizmiştir. Yine Deniz bu konudaki düşüncelerini "Evet sağlar çünkü çocuk ezber yapmaz uygulayarak

öğrendiği için akılda kalır.” diyerek gerekçelendirmektedir. Deniz’e göre yapılan uygulamalar ezberin önüne geçip kalıcı öğrenmeyi sağlayabilir.

Sena ise ders başarısına uygulamaların etkisi açısından bakmış ve “Ders başarısı için de yapacakları işi yeterince öğrenmiş olmaları gerekir. Bu yüzden sınıf içi ve sınıf dışı etkinliklere fazlasıyla yer veririm. Öncelikle daha iyi kavrayabilmeleri için etkinlikleri birlikte ve sınıfta yaparız. Sonrasında buna benzer başka şeyleri evde yapmalarını isterim.” demiştir.

Fatma Alice’i uygulama amaçlı kullanacağından ve kendi hazırladığı/var olan örneklerle anlatımı zenginleştirebileceğinden bahsetmektedir:

Ortaöğretimde etkinlik düzeyinde kullanırım. Bir program ya da bir proje yapmalarını isterim. En başta kendi yaptığımı gösteririm, bunun yanında diğer yapılan örnekleri de gösteririm. Ne kadar basitleri de var, güzelleri de var şeklinde. Bunları gösterdiğim zaman çocuklar da kendi düzeylerini görebilecekler ve ona göre etkinlikleri yapabilecekler.

Fatma’nın sözlerindeki *basit* ve *güzel* kelimeleri görüşme esnasındaki ses tonundan derste başta kolay uygulamalarla başlayarak zora doğru gideceğine ve bu durumun öğrencilerin Alice’de kendi seviyelerini tespit etmelerine etki edeceğine değinmektedir. Ayten ise “İlk olarak mantığı öğretmeye çalışırım. Bir kedi ilerlerken ses çıkarması istiyorsam şu kod bloğunu kullanmamız gerekir şeklinde onları yönlendiririm. Yani ilk ben gösteririm ve daha sonra onların kullanmasını isterim.” diyerek Fatma’ya benzer sözler sarf etmiş ve hem gösterim esnasında hem de öğrencilere uygulamalar yaptırılarak Alice’in ortaöğretimde kullanılabileceğini belirtmiştir. Uygulamaların ne şekilde yapılabileceğine ilişkin Deniz “Bizim yaptığımıza nazaran daha basit şekilde uygulamalar yapılarak öğretilir. Öğrenciler tasarım kısmını beğenir ancak kod kısmına geldi mi iç içe döngüler artarsa zorlanabilirler. Basitten başlayarak zora doğru öğretim yapılabilir.” diyerek ortaöğretim öğrencilerine programlama öğretimi yapılırken üniversite öğrencilerine nazaran daha basit uygulamalar

yapılmasını ve aşamalı bir şekilde öğretimin gerçekleştirilmesini savunmuştur. Sena ise “Alice’e başlamadan önce program içinde bulunan kodların mantığını açıklamak, ne işe yaradığını söylemek, İngilizce kelimelerin karşılıklarını vermek öğrenmelerini hızlandırır.” diyerek görüşlerini dile getirmiştir. Sena’ya göre önce teorik olarak programlama kavramları öğretilmeli, daha sonra uygulamalara geçilmelidir. Yani bu durum, Alice kullanıcılarının hazır bulunuşluk düzeylerini etkileyebilir.

4.1.4. Hayal gücünün kullanılması – somutlaştırma. Katılımcıların Alice ile programlamaya ilişkin görüşlerinden bir diğeri de öğrencilerin hayal gücünün kullanılmasıdır. Hayal gücünden bahsederlerken, bazıları önceki temalarla örtüşen beş farklı hususa değinmişlerdir.

Bu hususlardan ilki 3b dünya tasarımı deneyimidir. Örneğin, Zeynep 3B dünya tasarımı deneyiminin hayal gücünün aktif kullanımına etki ettiğini düşünmektedir:

Bir tema (deniz, kara, buzul gibi) seçerek üzerinde oluşturulan dünya, nesnelere, kodlar gelecekte verebileceğim derslere çok büyük etki edeceğini düşünüyorum. Çünkü 3B dünya tasarımı [deneyimi] demek bence öğrencilerin hayal dünyasının gerçek hayata aktarılması demek. Öğrenci hayal dünyasında oluşturduklarını, 3B programlama dersi sayesinde gerçek hayata aktarabiliyor.

Zeynep’in bu amaçla, Alice’in içerisindeki üç boyutlu nesnelere ve temalara başvurduğu anlaşılmaktadır. Her ne kadar Zeynep Alice kullanıcılarının hayal dünyalarını canlandırabileceklerini düşünse de bazı öğretmen adayları nesnelere yetersizliğinden yakınmaktadır. Bu konu bir sonraki bölümde Alice’te yaşanan güçlükler başlığında detaylıca incelenecektir.

Arif ise 3B dünya tasarımı deneyiminin asıl anahtar ögesi olan soyut/somut dönüşümünü ön plana çıkarmaktadır. “Öğrencilerin zaten derslerde yeterince soyut bilgiler ile donatıldığını düşünüyorum. 3B Programlama dersinde öğrendiklerim (3B Dünya Tasarımı)

ile sınıfımdaki öğrencilerin zihnindeki soyut bilgileri somut kavramlara dönüştürebilirim. Nasıl? Örneğin bilgisayar donanım parçalarını, uygun bir kurguyla, animasyonlu ya da etkileşimli biçimde kurgulayabilirim.” diyerek 3B dünya tasarımı deneyimlerinin soyut kavramları somutlaştırmada etkili olduğuna dikkatimizi çekmektedir.

İkinci husus, derse ilgi katılım, Alice’te hayal güçlerinin gerçeğe yansıtılabilmesi konusunda öğretmen adaylarının üzerinde durduğu bir başka konudur. Mesela, Eda’ya göre “Kodlar basit, anlaşılabilir düzeyde ve 3B nesnelere kullanıldığından ve üç boyutlu dünyada çalışıldığından öğrencilerin dersi eğlenceli hale getirip, öğrencilerin ilgisini çekerek derse katılımı sağlar.” Zeynep ise derse ilgi katılımının artması ile birlikte yine derse katılım ve başarının artacağını savunmuştur.

Üçüncü husus, akılda kalıcılık ve başarı potansiyelidir. Alice’in sağladığı akılda kalıcılık, gerçek hayata yakın bir öğretim aracı olması özelliğiyle gerekçelendirilmiştir. Bu durumu en güzel özetleyen görüşlerden biri Ayten’in söyledikleridir: “Programlama kavramlarının akılda kalmasını sağlar. Sonuçta bir araba kullanımıyla algoritma mantığını birleştirebilir. Bir ampul patladığında takılması ve çıkarılması aşamalarını yazıp gerçekle bağdaştırarak öğrenebilir.”

Son husus ise öğrenci gelişimi ile hayal gücünün kullanımı arasında kurulan bağıdır. Öğrenci gelişimi kodunun altında öğretmen adayları görüşlerine değinmeden önce gelişme ve gelişim kavramları üzerinde durmak gerekebilir. “Gelişme kavramına insanların büyüme, olgunlaşma ve öğrenmenin etkileşimiyle sürekli olarak ilerleme kaydeden değişmesidir. Gelişmeyi ürün olarak ele aldığımızda gelişimi bu ürünün süreç yönü olarak tanımlayabiliriz. Gelişim, olgunlaşma ve öğrenme olmadan gelişim sağlanamaz.” (Senemoğlu, 2012, s.3). Örneğin, bir öğrencinin Alice’de programlama yapabilmesini, bir ürün olarak düşünüp, gelişme olarak, Alice’de programlamayı öğrenmesi sürecini de gelişim olarak adlandırabiliriz. Öğrencilerin kendini geliştireceğinden bahseden Sevda “Farklı tasarımlar ortaya koymakla

birlikte kendi hayal dünyalarını kolayca yansıtacaklar bunu yaparken de farkında olarak veya olmayarak programlama mantığını geliştireceklerdir.” diyerek programlama öğretiminde hayal dünyalarının yansıtılmasına değinmiştir. Seveda'nın *farkında olarak veya olmayarak* kelimelerini kullanmasını Alice'i öğrencilerin bir oyun/eğlence gibi görmesi ile açıklayabiliriz. Yani, Seveda'ya göre nasıl bilgisayar oyunları ile bazı İngilizce kelimelerle (up, down, skill vb.) karşılaşan insanlar, o sözcüklerin Türkçe karşılıklarını da farkında olmadan öğreniyorsa, Alice'de de programlama bu şekilde öğrenilebilir.

4.1.5. Diğer görüşler. Öğretmen adaylarının ilk beş başlık altında toplanamayan görüşlerine bu bölümde yer verilmiştir. Diğer görüşlerin altında incelenen konular, akılda kalıcılık ve başarı, 21. yüzyıl becerileri, spesifik örnekler ve eleştiriler olarak sınıflandırılabilir. Mesela, akılda kalıcılık ve başarı konusundan bahsedilmiş, ama bu konu “eğlenerek öğrenmeyle”, “programlamaya görsellik katmayla”, “uygulayarak öğrenmeyle” ve “hayal gücünün kullanılması – somutlaştırmayla” ilişkilendirilememiştir. Bu bölümde akılda kalıcılık ve başarıda olduğu gibi ilişkilendirilemeyen konular bu başlık altında incelenmiştir.

21. yüzyıl becerilerine giriş bölümünde değinmiş idik. Problem çözmeye üç boyutlu dünya tasarımı deneyiminin etkisi konusunda Leyla “3B dünya tasarım becerilerimin öğrencilerimin en çok problem çözme becerilerinin gelişmesine katkı sağlayacağını düşünüyorum.” diyerek Alice ile programlama öğretiminin problem çözmeye ilişkisinden bahsetmiştir. Bahar öğrencilerin yaratıcılıklarının gelişeceğinden bahsetmiş, Buna ek olarak, Arif “Sağladığı görsellik imkânları ile Liderlik vasfı gelişir. Çünkü özgüven kazanır.” diyerek özgüven kazanmalarının liderlik vasfı kazanmalarına/geliştirmelerine etki edeceğini düşünmektedir.

Programlama öğretiminde Alice'in kullanımına ilişkin öğretmen adayı görüşlerinin bazıları spesifik konulara odaklanmakta ve bir bütünlük göstermemektedir. Bu bölümde bu görüşlere ilişkin örneklere yer verilmiştir. Lara “Ben açıkçası lisede bu şekilde bir

programlama dersi almadım. Üniversitenin başında bir EBİT dersi aldım ve kaldım. Keşke ben de bu şekilde bir dersi o zamanlarda alsaydım.” demiş ve Alice ile yapılan öğretimden memnun kaldığından ve beklentisinden bahsetmiştir. Arif:

Aslında öğretmen öğrenciye programlama öğreteceğim desin. İnanın o kadar çok kaynak var ki. Dün bir tane daha denk geldi. khanacademy.org tamamen rastlantı sonucu ulaştım. Çok da hoşuma gitti açıkçası. Scratch öyle, yine code.org var. Bir araştırmada buna benzer 60’a yakın platform olduğunu okumuştum. Hepsi de faal durumda.

diyerek programlama öğretiminde farklı öğretim teknolojilerinden bahsetmiş ve bu işin programlamayı öğretecek öğretmende bittiğine vurgu yapmıştır. Ayrıca,

Türkiye’deki eğitim sistemi ve bu eğitim sisteminde yer alan öğretmenlere ilişkin eleştiriler diğer görüşler başlığı altında incelenen yorumların sonucusudur. Sevda:

Şöyle bir gerçekte var ki, ortaöğretimde haftada iki saat bilgisayar dersi olan öğrencilerin Alice gibi bir programı takibi ne kadar mümkün olur? Demek istediğim bir hafta önce aldığı gmail adresini veya şifresini unutan öğrenciler Alice kodları üzerinde ne kadar yoğunlaşabilir bilemiyorum.

diyerek ortaöğretimdeki bilişim dersinin ders saatinden doğabilecek sıkıntılardan bahsetmiştir. Arif’in görüşleri ise diğerlerine nazaran daha sert eleştiriler olmuş ve:

Okullara baktığımız zaman büyük çoğunluğunda bilgisayar laboratuvarı var. Şimdi aslında asıl önemli konu. Öğretmenlerin tutumu. Şeker var, un var, yağ var. O zaman neden helva yapılmıyor derler adama. Burada öğretmen tutumları karşımıza çıkıyor. Staj yapan diğer arkadaşarımdan da duyduğumuz kadarıyla öğretmenler okullarda çocuklara programlama eğitim vermiyor, hatta oyun oynatanlar bile var imiş.

diyerek ortaöğretimdeki bilişim öğretmenlerinin derslerine ve öğrencilerine yeteri kadar ilgi göstermemesinden yakınmıştır. Ve ekleyerek:

İşte siz üniversite öğrencilerini programlama derslerinden soğutursanız, yeterli yazılım dersi vermezseniz, sistemin kölesi yapıp sınavla sınırlarsanız, o öğrenci öğretmen olduğunda öğrencilere de bir şey katmaz. Alice ortaöğretim düzeyince çok iyi bir alternatif olur, tabi etkili kullanılabilirse, tabi bunu kullanacak öğretmen bulunabilirse de!

demıştır. Özellikle son yıllarda Türkiye’de tartışılan programlama öğretiminin ilkökul düzeyine indirgenmesi yönünde yapılan tartışmalar da incelendiğinde, bu eleştiriler önemli bir noktaya parmak basıyor olabilir. Bunun yanı sıra, okullardaki bilgisayar laboratuvarlarının fiziksel durumunu ve bir konunun anlatımında kullanılmadan önce öğrenim kazanımlarına göre revize edilmesi gerekliliğinden de bahsetmişlerdir.

4.2. Alice Yazılımında Karşılaşılabilecek Güçlükler

Bu bölümde araştırma problemlerinden “Alice ortamını kullanımında karşılan güçlükler nelerdir?” sorusunun cevabı aranmaktadır. Daha önce Alice ile programlama öğretimi başlığı altında verilen görüşlere nazaran bu ve bir sonraki bölümde daha spesifik konulara odaklanılmıştır. Görüşme formuyla toplanılmış olan öğretmen adayı görüşleri toplamda 7 temada incelenmiştir. Bu temalar Şekil 9’da gösterildiği gibidir.

Şekil 9

Alice’de karşılaşılabilecek güçlükler



4.2.1. Nesne konumlandırma. Nesne konumlandırma teması, iki ve üç boyutlu nesnelerin üç boyutlu ortamda yerleştirilmesi ve boyutlarının ayarlanması gibi işlemlerde güçlük çeken öğretmen adaylarının görüşlerinin birleştirilmesi sonucu ortaya çıkmıştır. Nesne konumlandırmayla ilgili dört temel problem göze çarpmaktadır (1) nesnelerin üç boyutlu ortamda kaybolması, (2) döndürme sorunu, (3) küçük nesnelerle çalışmanın zorluğu, (4) nesne sayısı arttıkça kontrolünün zorlaşması. Özellikle nesnelerin üç boyutlu ortamda kaybolması ve x, y, z düzlemlerinde döndürülmesi problemi bunlardan öne çıkanlardır. Örneğin, Eda “Eklediğimiz nesneler 3B dünya olduğu için diğer nesnelerin arkasına saklandı ve bu nesneler ile ilgili düzenleme yapmaya çalışırken nesneleri bulmakta sıkıntı yaşadık.” diyerek nesnelerin bulunduğu yerin tespitinde güçlüklerle karşılaştıkları belirtmişlerdir. Deniz ise “İlk başta programa hâkim olmadığımızdan dolayı Alice’deki bazı nesnelerin yerini bulamamıştık.” demiş ve başta sorunla karşılaşmasına rağmen daha sonra alıştığından bahsetmiştir. Döndürme problemiyle ilgili olarak ise Ayten “Döndürme de sorunlar yaşıyordum hani nesneyi sağ tarafa doğru döndüreceğim ama o ileri gidiyordu ya da

gömülüydü.” demiş ve nesnenin yönünün belirlenmesinde sorun yaşadığını aktarmıştır.

Fethi ise bu probleme çözüm önerisi olarak “Programda nesnelerin yön ve konumunun veri olarak girilememesi bir sorundu.” Şeklinde görüşünü dile getirmiştir. Bir diğer problem ise küçük nesnelere kullanıldığında onları konumlandırmada yaşanan güçlüklerle ilişkindir.

Hüseyin “Zorluk olarak bazen programda ufak nesnelere olunca o nesnelere sürüklemek ve döndürmek gibi yerlerde sıkıntı çıkarabiliyor.” diyerek bu konudaki görüşünü dile getirmiştir.

Öğretmen adayları içerisinde yalnızca Sevda, sahnedeki nesne sayısının artmasının programda performans sorunları yarattığından bahsetmiş ve “Ayrıca nesne sayısı arttıkça sahnedeki nesnelerin kontrolü oldukça yavaş oluyor hatta bazen seçmiyor, değişikliğe izin vermiyor.” diyerek nesnelerin üç boyutlu dünyaya yerleştirilmesinde yaşadığı problemleri dile getirmiştir.

4.2.2. Dil. Bu araştırmanın yapıldığı Uludağ Üniversitesinin eğitim dili Türkçedir.

Öğretmen adaylarının yaklaşık üçte biri (n=8), her ne kadar zorunlu ders olarak İngilizce dersi alıyor olsalar da Alice yazılımını kullanırken karşılaştıkları kodları anlamada ve anlamlandırmada sorun yaşadığından bahsetmiştir. Örneğin, Lara “Dil konusu Alice’de beni çok zorladı. Türkçe desteği olsaydı farklı şeylerde yapabilirdim. Özellikle prosedürleri ve fonksiyon çeşitlerini dilden dolayı anlamakta zorluk çektim.” diyerek bu konuda yaşadığı güçlüğü dile getirmiştir. Gaye “3 boyutlu bir program olduğu için cazip gelecektir. İngilizce olması programlama mantığını anlamak için iyi olabilir fakat İngilizcesi olmayanları programlamaya ön yargılı bakmalarına sebep olabilir.” demiş ve İngilizceleri yeterli olmayanların olumsuz tutum takınabileceklerinden bahsetmiştir. Özellikle programlamaya giriş derslerinde kullanılmak amacıyla geliştirilen Selcan “Kodlama yapılırken öğrencilerin İngilizce bilmesi gerekiyor. Orada sıkıntı çekebilirler. İngilizce olmasından dolayı daha rahat kullanamadık.” diyerek diğer katılımcılara paralel söylemlerde bulunmuştur. Aslında Alice programında sadece basit İngilizce terim ve komutlara rastlanmaktadır. Say (söyle), move

(taşı), up (yukarı), down (aşağı) gibi İngilizce öğretimi yapılırken kullanılan temel sözcükleri anlamada sorun yaşayan öğretmen adayları, “acaba ilk ve orta dereceli okullarda İngilizceyi doğru bir şekilde öğretebiliyor muyuz?” sorusunu da beraberinde getirmektedir. Elbette ki bu konu mevcut araştırmanın kapsamı dışında olduğu için, ilgili alan uzmanlarının takdirindedir ama İngilizce’ye hakim olmadan da program kullanılabilir. Ela ise İngilizce olarak kullanılan Alice yazılımının devlet okullarında, özellikle ortaöğretim öğrencilerine programlama öğretirken bir sorun teşkil ettiğini dile getirmiştir. Katılımcıların bir bölümü Alice’i kullanmaya başlamadan önce tek tek hazır kodların anlamlarının ve Türkçe anlamlarının ortaöğretim öğrencilerine öğretilmesi gerektiğini vurgulamıştır. Bu sayede bu problemin önüne bir nebze geçilebilir.

4.2.3. Performans sorunları. Performans sorunları temasının altında, öğretmen adaylarının Alice yazılımından veya bilgisayarın donanım özelliklerinin yetersizliğinden kaynaklanan donmalara, hatalara ve güçlüklerle ilişkin görüşleri toplanmıştır. Bilgisayar/yazılımdaki donmalar ilk alt kod olarak düşünülebilir. Dilara “Oluşturduğum animasyonların süresi uzadığı”, Ali “Ondan fazla nesne eklediği” ve Lara “Çok fazla nesne ve kod yazdığı” durumlarda Alice yazılımında bilgisayar tabiriyle kasma/donma gibi eylemlerle karşılaştığından bahsetmiştir. Bu tür sorunların bazen oluşturulan projelerde veri kayıplarına ve bilgisayarın (işletim sisteminin) tamamen donmasına neden olduğunu da dile getiren öğretmen adayları olmuştur.

İkincil olarak, Öğretmen adayların birçoğu “Submit Bug Report” yani hata raporunu gönder uyarısının da Alice yazılımının akışını bozduğunu dile getirmiştir. Örneğin, Eda “Sıkla karşılaştığımız bir submit bug report hatası veriyor ve bu da programın çalışmasını bozuyor. Genelde karşılaştığımız sorun bu oldu. Onun dışında büyük bir sorunla karşılaşmadık.” demiştir. Arif ise kısa ve öz bir şekilde “Israrla çıkan bug gönder ekranı var.” diyerek bunun bir sorun olduğunu dile getirmiştir. Alice yazılımının yaratıcıları

kullanıcılardan gelen dönütler ve onların taleplerine yönelik yeni özellikler eklemeye çalışmaktadır. Özellikle yazılım içerisinde yer alan yeni özellik talep et (Request a New Future) sayesinde bunu kısmen de sağlamış görünmektedirler. Yazılımın ilk çıktığı Alice 2.0'dan bu yana 2.0, 2.1, 2.2, 2.3, 2.4, 3.0, 3.1, 3.2 gibi 8 farklı sürümü kullanıcılarına sunmuştur. Yani, sürekli kendini güncelleyen bir yazılım söz konusudur ve ileride bu hata ekranlarının en aza indirgenmesi beklenebilir. Arif ayrıca “tabi bunlar hep kullanıcı tarafından bakılanlar, hiçbir yazılım 4-4lük değil. İlla ki hatalar olacak. Ki bu masaüstü programlar için daha da tolere edilebilir bir durum.” diyerek karşılaşılabilecek bir sorun olduğunu da eklemiştir.

Son olarak, Öğretmen adaylarının küçük bir kısmı (n=2), evlerinde ve okulda kullandıkları Alice sürümlerinin 32 Bit ve 64 Bitlik yazılım olarak ikiye ayrılmasından sorunlarla karşılaştıklarından bahsetmişlerdir. Bunun nedeni ise, bu çalışmaya başlamadan önce, katılımcıların böyle bir ayırım olmaksızın Alice yazılımını edinebilmeleri olarak gösterilebilir.

4.2.4. Kamera kullanımı. Giriş bölümünde de açıklandığı üzere, Alice'teki üç boyutlu dünyanın farklı bölgelerinde çalışılabilmektedir. Yazılımda bir bölgeden diğerine geçişi kolaylaştırmak için kamera nesneleri geliştirilmiştir. Öğretmen adaylarının az da olsa bazıları kamera kullanımını konusunda sorunla karşılaşmıştır. Yaşadıkları sorunların iki alt tema altında toplandığı söylenebilir. Bunlardan ilki sahneler arası geçişlerde karşılaştıkları güçlüklerdir. Saliha “Alice kullanırken bir sahneden diğer sahneye atlarken kamerada kaymalar meydana geldi.” demiş ve diğer öğretmen adayları da buna benzer görüşlerini tekrarlayarak geçişlerde zorluk yaşadıklarını dile getirmişlerdir. İkincisi ise üç boyutlu dünyanın bir bölgesini işaret eden kamera açısını ayarlarken karşılaştıkları sorunlara ilişkindir. Kamera açısını değiştirirken ilk başta deneyimsiz oldukları için, konuşma baloncuklarının ve seslerin sürelerinde sorun yaşadıklarını dile getiren öğretmen adayları

olmuştur. Örneğin, Ayten “Düşünce baloncuklarını koydum ama kamerayı başka bir yöne çevirdiğimde onlar tekrar çıkmaya başladı yani sürekli ekranda ben müdahale etmeden karşıma geliyordu ya da ses kayıtları birbirine giriyordu.” demiştir.

4.2.5. Nesne galerisinin zenginleştirilmesi. Alice’de üç boyutlu dünyanın yanı sıra bu dünyadaki tüm nesnelere de hazır olarak kullanıcılara sunulur. Örneğin, yazılımın yüklenmesi ile birlikte ağaç, araç, kuş, balık gibi nesnelere üç boyutlu, konuşma baloncukları, üçgen, dikdörtgen gibi şekillerde iki boyutlu olarak hazır halde gelir. Öğretmen adaylarının yaklaşık üçte birlik bir bölümü uygulamalar esnasında farklı nesnelere amaçları dışında kullanmak zorunda kaldığından, nesnelere sayısındaki sınırlıklardan bahsetmiş ve zenginleştirilmesi gerektiğini belirtmiştir. Her ne kadar bu bölümdeki ortalamanın üstünde bir sayı olsa da söyledikleri benzerlik göstermekte ve genel olarak nesne yetersizliği ve bunun sonuçlarından bahsetmişlerdir. Örneğin Saliha “Nesnelere bulmakta çok zorluk çektik. Hatta bu yüzden sahnelerimizin bazıları çok sade oldu.” ve Hüseyin “Nesnelere kesinlikle yetersiz. Belli başlı birkaç senaryo ya göre düzenlenmiş sanki. O senaryoların dışına çıkmaya kalktığımızda gerekli nesnelere bulamıyoruz.” diyerek nesnelere azlığından dolayı, nesne galerisinin tam olarak beklentilerini karşılamadığından bahsetmiştir. Dilara ise “Alice kısmında yer alan nesnelere çok çeşitli olmasına rağmen en basit nesnelere içerisinde yer almadığını görmek beni hayal kırıklığına uğrattı.” demiş ve bu durumun kendisini olumsuz etkilediğinden bahsetmiştir. Sena “Alice içindeki nesne sayısı çok az. Bir saçmalıkta 2.0 sürümünde olan nesnelere 3.1’e yükseltildikten sonra çıkartılmış olması. Bazı basit nesnelere bulamıyoruz. Kendimiz dışarıdan nesne ekleyemiyoruz.” diyerek derslerde 3.1 kullanılmasına rağmen 2.0 sürümünü de incelemiş olduğundan bahsetmiştir. Gerçekten de bakıldığı zaman eski sürümdeki bazı temel nesnelere yeni sürümde çıkartıldığı görülmektedir.

4.2.6. Metotlar. Metot bir programın içerisinde yer alan kod/kod topluluğunu ifade eder. Bu bölümde öğretmen adaylarının ufak bir kısmı metotların azlığından dolayı

zorlandıklarını iletmiştir. Genellikle metot sayısının az olmasından yakınan öğretmen adayları, olaylara ilişkinde değerlendirmelerde bulunmuştur. Örneğin Abdullah “Zıplama fonksiyonunu kullanamadım mesela. Zıplama olarak yoktu bir metot. Kendim oluşturmak zorunda kaldım. O metotların artması faydalı olabilir.” demiş ve metot sayısının artırılması gerekliliğinden bahsetmiş. Ortaöğretim öğrencilerine uygulanacaksa özellikle yürüme, zıplama gibi metotların artırılması işe yarayabilir. Sevda yine aynı konuda

Daha sonra seçilen nesnelere yürütülmesi için çok fazla kod yazılmakta.

Keşke Alice dünyasının içerisinde (yürüme, uçma vs) gibi hazır kodlar

bulunsaydı demekten alamadım kendimi. Ayrıca nesnelere yürüdüklerinde çok yavaşlar. Yapılan projede tüm olayların hızlı yapıldığını düşünürsek, yürüme komutunun projeye göre fazla yavaş kaldığını görmekteyiz.

demiştir. Örnek olarak, yürüme eylemi üzerinden giden Sevda, yavaş bir şekilde ancak yürütebildiklerinden bahsetmiştir. Bu uygulamalar sırasında karşılaştığımız, yürüme eylemini kullanılan projelerde özel bir problem olarak göze çarpmaktadır. İç içe döngülerin sayısı arttıkça kodların çalıştırılması esnasında problemler ile karşılaştıklarından da bahsedenler olmuştur.

4.2.7. Diğer görüşler. Öğretmen adaylarının yaşadıkları Alice’le ilgili problemlerden yukarıdaki temel başlıklara uymayan ve sıkça karşılaşılmayan görüşleri burada “diğer görüşler” teması altında toplanmıştır. Bunlar dışarıdan eklenen ses ve resim dosyaları ile çalışma, yazılar ile çalışma ve kısayollar (shortcuts) şeklinde listelenebilir. Dışarıdan eklenen ses dosyalarına ilişkin Saliha “Programın tüm sesleri desteklemesi gerekir bence.” demiştir. Dilara ise “Kendi animasyon ödevim için kullandıklarımdan söz edecek olursam ses dosyası atarken türleri kısıtlaması Alice programının eksik yönlerinden biri çünkü kaydettiğim sesleri türlerini bir daha değiştirmek durumdan kaldım.” diyerek Alice’in bazı ses dosyalarını kullanırken sorun çıkardığından bahsetmiştir. Resim dosyalarının kullanımına ilişkinse iki

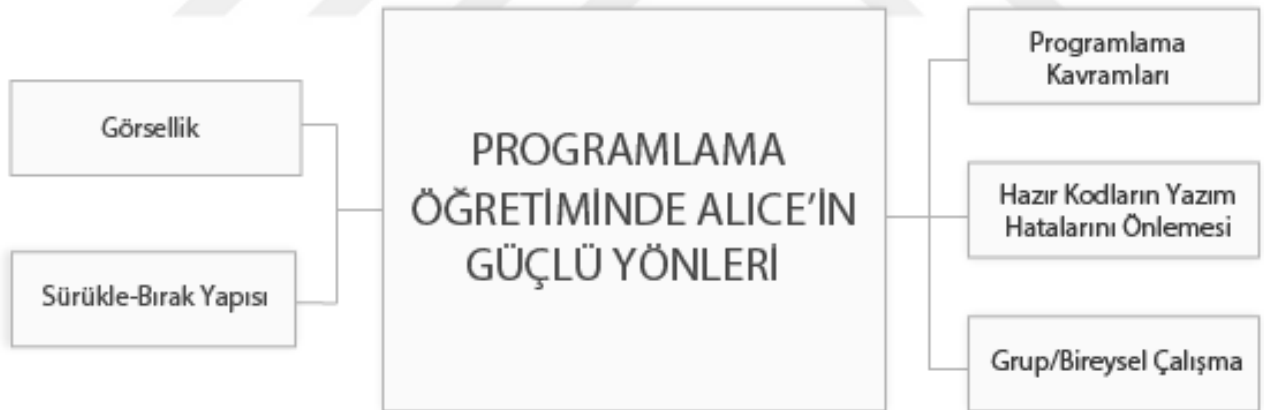
boyutlu ve gif formatlı (hareketli resim ekleme) olan bir resmi Alice'e yüklediğinde çalıştıramadığından bahseden bir katılımcı da vardır. Bunun dışında Ayten “Yazı yazmada büyütüp küçültme yapamıyorduk. Yazı tipini değiştiremiyorduk.” diyerek bir eksikliği vurgulamıştır. Fatma ise kısayolların yetersizliğini, birçok programda tam ekran yapmayı sağlayan F5 kısayolu ile örneklendirmiştir. Buraya kadar bahsi geçen güçlükler incelendiğinde Alice'in hep zorluklar çıkaran bir program gibi algılanması yanlış olur. Alice'in güçlü yönlerine ilişkin görüşleri ayrı bir bölümde aşağıda açıklanmıştır.

4.3. Programlama Öğretiminde Alice Yazılımının Güçlü Yönleri

Bu bölümde araştırma sorularından Alice yazılımının programlama öğretimine sağladığı imkânlarla ve programlama öğretimindeki güçlü yönlerine ilişkin öğretmen adayları görüşleri değerlendirilmiş olup Şekil 10'da görüldüğü gibi 5 alt başlıkta özetlenmiştir.

Şekil 10

Programlama öğretiminde Alice'in güçlü yönleri



4.3.1. Görsellik. Katılımcıların daha önceki bölümlerde görsellik konusuna bir hayli önem verdikleri değerlendirilmişti. Alice yazılımını görsellik açısından katılımcılara göre birçok farklı özelliğinden dolayı tercih edilebilir bir programlama öğrenme ortamı sağlamaktadır. Bu bölümde öğretmen adaylarına yöneltilen “Alice'in güçlü yönleri sizce nedir?” sorusuna daha önce bahsedilen düşüncelerden de yararlanılarak görsellik teması altında cevap aranmaya çalışılacaktır. Katılımcıların tamamı (n=24) görselliğin önemine dikkat çekmiş, bunların

içerisinden ufak bir bölümü ise (n=4) görselliğinin geliştirilmesi konusuna değinmiştir. Toplanan görüşlerin Alice'in güçlü yanı ile alakalı olanları üç alt tema altında toplanarak incelenebilir. Bunlar, görselliğın programlama öğretimının (1) kolay bir şekilde yapılmasına, (2) ilgi çekici olmasına, (3) gerçekçi olmasına yardımcı olduğudur. Bu başlık, yukarıda listelenen üç tema altında öğretmen adayı görüşlerinin değerlendirilmesi yapıldıktan sonra görselliğın olası diğer etkileri ve katılımcıların görsellik hakkındaki eleştirileriyle sonlandırılmıştır.

Katılımcıların büyük bir bölümü Alice'in görselliğının programlama öğretimine kolaylık sağladığından bahsetmiştir. Katılımcılara göre görsellik “basit arayüzü” (Ali), “üç boyutlu olması” (Leyla), “nesneleri galeriden aratarak kolaylıkla ulaşılabilmesi” (Eda), “anlaşılabilir ve karmaşık olmayan yapısı” (Zeynep), ve “ gerek prosedür ve fonksiyon bölümü, gerek kod yazma alanının vs. açık bir şekilde programda yer alması” (Sevda) ile programlama öğretimine kolaylık sağlamaktadır.

Öğretmen adayları, görselliğın etkileri arasında ilgi çekici olmasının üzerinde yine programlama öğretimine kolaylık sağlaması kadar vurgu yapmıştır. Ela “Görsel olarak karakter seçme, değiştirme gibi özellikleri programı çekici kılmaktadır.” diyerek görselliğın ilgi çekici olmasından bahsetmiştir. Ali ise kullanıcıların ilgilerini çekeceğinden ve bu sayede programlamayı gözünde büyütmesini engelleyeceğinden bahsetmiştir. Murat “Görsel dünyanın olması ilgi çekiyor ve kod yazmaktan çok animasyon oluşturma havası veriyor.” diyerek animasyon yapar gibi programlama öğrenebileceğinden bahsetmiştir. Sena da görselliğın her alanda son yıllarda ön planda tutulan bir özellik olduğundan ve programlama öğretiminde konuya daha istekli yaklaşan bir kitle yakalanabileceğini söylemiştir.

Görselliğın gerçekçi bir ortam sağlaması ve programlama mantığını anlamada kullanıcılara yardımcı olması da yine üzerinde durulan hususlardandır. Örneğın Ayten “Gerçekle bağlantı kurmamıza katkı sağladı. Mesela siber zorbalığı anlattık ve 3B bir ortamda

anlattık. Bu da hikâyenin gerçekliğini arttırdı.” diyerek aslında üç boyutlu bir evrende yaşarken buna yakın bir ortamda programlama öğrenmenin faydasından bahsetmiştir. Önceki bölümlerde yer verilmiş somutlaştırma konusu da yine gerçekçi oluşuna bağlanılabilir. Mesela, Saliha bu hususta üç boyutlu ortam sayesinde donanım parçalarının çalışma prensiplerinin bir kurgu ile rahatlıkla anlatılabileceğinden bahsetmiştir. Programlama mantığını anlamada yardımcı olması konusunda ise Selcan “Programlama kavramlarını önceden öğrenmiştik ama görsellik sayesinde kafamda daha rahat şekillendirdim.” daha önce programlama dersi aldığını ama bağlantıları Alice ile daha rahat kurabildiğini iletmiştir.

Yukarıda değerlendirilen görüşlerin dışında, katılımcılara göre öğrencilerin kodlama eylemine olan dikkatlerini arttırabileceği gibi öğrenilen bilgilerin akılda kalıcılığına etkisi de söz konusudur. Görsellik sayesinde yine yazılan kodlardaki hataların tespiti ve düzeltilmesi de kolaylaşmaktadır. Bu da yazılan kodların görsel çıktıları yani videolar sayesinde gerçekleştirilebilir. Örneğin, Selcan “Görsel olarak iki adım gideyim dediğin vakit, direk görsel olarak iki adım gittiğinde ya da daha karmaşık bir şekilde kodları düzenlediğinde yaptığını görsel olarak görebilme şansın var. Yaptıramadığını gördüğün vakit geriye dönüp düzeltme şansın var.” diyerek görüşlerini dile getirmiştir. Bunların yanı sıra kullanılan nesnelere üç boyutlu olması sayesinde çok yönlü düşünmeyi destekleyebileceği ve problemlere farklı çözüm yolları getirilebileceğinden de daha önce bahsedilmişti. Kameraların bile üç boyutlu nesnelere manipüle edilebilmesi Alice’te görselliğin bir avantajı olarak gösterilebilir.

Öğretmen adayları her ne kadar olumsuz gibi görünse de görselliğinin geliştirilmesine yönelik eleştirilerini de aktarmıştır. Sevda “Nesne içeriği açısından zengin olmamakla birlikte, nesnenin amacı dışında kullanılabilir olması da artık bir yönü olduğunu düşünmekteyim (halıdan yol yapmak gibi, ya da mangodan tohum yapmak gibi).” demiş ve nesnelere sayılarının arttırılması gerekliliğinden bahsetmiştir. Bunun yanı sıra bu şekilde programlama yapmanın öğrencileri alternatif yollar üretmeye ittiğini de iletmiştir. Aksi yönde

düşünen katılımcılarda söz konusudur. Örneğin, Bahar “Alice oldukça geniş bir görsel kütüphaneye sahip ve öğrencilere birçok hikâye olanağı sağlıyor.” diyerek bu konuda düşüncelerini dillendirmiştir.

4.3.2. Sürükle-bırak yapısı. Sürükle bırak yapısının sağladığı avantajlara Fatma'nın söyledikleri ile başlayalım: “Alice, programlama öğretiminde nesnelere sürükle-bırak yöntemini kullanarak ekliyoruz bu yönden çok avantajlı. Çocuğun çünkü sürekli program yazımını durdurması gerekmeden, aklına bir şey geldikçe ekleyecek ekleyecek sonra bir bakacak ki program donuyor (gülüşmeler). Kalacak öyle”. Daha önceki bölümde bahsedildiği gibi sürükle-bırak eylemini ilgilendiren bazı performans sorunlarının olduğu göze çarpmaktadır. Bu da istenileni tam olarak gerçekleştirilememesi ile sonuçlanmaktadır. Fatma'nın projesinde 100'ün üzerinde üç boyutlu görselle çalıştığından dolayı bir takım sorunlar ortaya çıkmıştır. Konuya dönecek olursak sürükle bırak yapısı öğretmen adaylarının tamamına yakını tarafından olumlu karşılanmış (n=23) ama birkaç tanesi sürükle-bırak eyleminin bazı noktalarda sorunlara neden olduğundan (n=2) bahsetmiştir. Katılımcıların sürükle bırak yapısının avantajlarına ilişkin görüşleri dört başlık altında incelenebilir: (1) kolaylık sağlanması, (2) zamandan tasarruf, (3) satır satır kod yazma güçlüğü ortadan kaldırması, (4) sıkılmadan çalışabilmeye imkan tanınması

Katılımcıların büyük bir çoğunluğu sürükle-bırak eyleminin programlama yaparken kolaylık sağladığından bahsetmiştir. Bazıları ise özellikle sahnelere nesnelere eklenmesi aşamasında bu durumun sorun yarattığından bahsetmiştir. Gaye hazır kodların ve nesnelere sürüklenip bırakılarak eklenebilmesini bir kolaylık olarak görmüştür. Selahattin ise Alice'i “Programlamaya Turbo pascal ile başlayan bizim jenerasyon için turbo devrim niteliğinde” diyerek nitelendirmektedir. Ela rahatlıkla istediği kodu ve nesneyi eklenebildiğinden, Bahar ise “İstediğim kodu istediğim şekilde kod ekranına sürükleyebiliyorum ve kodların sırasını rahatlıkla değiştirebiliyorum. Bu da bana çok büyük kolaylık sağlıyor. Üstelik kodları

rahatlıkla iç içe yerleştirebiliyorum.” diyerek sıralamada ve iç içe döngülerde kolaylık sağladığından bahsetmiştir. Lara ise hiç programlama bilmeyen birinin bile rahatlıkla çalışabileceğinden bahsetmiştir. Daha önce bahsettiğimiz bazı noktalarda sorun yaşadığından bahseden öğretmen adaylarının görüşleri bu noktada devreye girmektedir. Örneğin, Selcan “Sürükle bırak yapısı bazen kolaylaştırdı. Bazen de zorlaştırdı. Kod boyutunda rahatlık sağlıyor, elle yazmaman, götürüp taşıman gibi. Ama sahnelere eklediğimiz bazı görsellerde farklı bir yere gitme de ve konumlandırma da sıkıntı yaşıyorsun.” demiş ve üç boyutlu dünyaya nesnelere eklerken bunun bir dezavantaj olduğundan bahsetmiştir.

Zamandan tasarruf sağlamasına ilişkin öğretmen adayları görüşleri de bir hayli çoktur. Sevda “Her komutu ayrı ayrı menülerin, açılan pencerelerin içerisinde aramak yerine bulunduğu noktadan sürükle bırak yöntemiyle kullandık. Bu sayede hem zamandan tasarruf ettik hem de programı zorluk çekmeden kullandık.” diyerek çoğu programda menülerdeki seçeneklerle uğraşmaktan kurtulduğunu ve zamanını çalmadığını dile getirmiştir. Ayşe ve Deniz ise özellikle yanlış yazılan kodların kaldırılması noktasında sürükle-bırakın kolaylık sağladığını ve hızlıca istediklerini gerçekleştirmede işe yaradığını dile getirmiştir.

Sürükle-bırak yapısının satır satır kod yazma güçlüğü ortadan kaldırması da Alice’in bu konuda belirtilen en önemli özelliklerinden birisidir. Ali bu konuda “Bir visual basic programı gibi tek tek yazmamamı sağladı. Hataları daha kolay görmemi ve anlamamı sağladı.” diyerek Alice’i visual basic ile karşılaştırmış ve hem rahatlığından hem de anlaşılır olmasından bahsetmiştir. Lara ise “Ortaöğretim öğrencilerine öğretim yaparken sürükle-bırak ile hazırlanan Alice’i tek tek kod yazdırmaya tercih ederdim.” diyerek özellikle ortaöğretim öğrencileri için uygun olduğundan bahsetmiştir.

Bu yapı sayesinde kullanıcıların programlamayı sıkılmadan yapabilmesi ise son başlıktır ve diğerlerine nispeten az sayıda bir grup öğretmen adayları tarafından belirtilmiştir. Sena “Sürükle-bırak yapısı ile bir kere yanlış yazılan kodlarla uğraşmaktan kurtuldum.

Böylece mantığı doğru da oluştursam sırf küçük eksiklikler için yanlış yaptığımız zannederek uğraşmıyorum. Sıkılmadan yapıyorum. Sıkılıysaydım uğraşmazdım zaten.” diyerek sıkılmadan yaptığını belirtmiştir. Gerekçe olarak ise yanlış yazımları kolaylıkla kaldırabilmesini göstermiştir.

Bunların dışında sürükle-bırak yapısının, kodları ezberlemek yerine onları kullandırtmayı sağladığından, özellikle bilgisayar ve öğretim teknolojileri eğitimi bölümüne meslek liseleri dışından gelen öğretmen adaylarına programlama öğretiminde işe yarayabileceğinden ve programlamanın genel mantığını kavratma açısından ortaöğretim öğrencilerine kullandırabileceğini belirtilmiştir.

4.3.3. Programlama kavramları. Öğretmen adaylarının görüşleri programlama kavramlarını anlamalarına yardımcı olma konusunda, programlamanın mantığını anlama ve hakim olma dışında çeşitlilik göstermektedir. Programlama mantığını anlamalarını yardımcı olduğunu belirten öğretmen adaylarından Dilara “Alice yazılımı programa öğretiminde bana ve meslektaşlarıma pek çok kolaylık sağlayacaktır. Bunlardan en önemlisi; kodlama mantığını en kolay şekilde kavramak olacaktır.” demiştir. Sevda ise “Alice’in içerisindeki programlama mantığı oldukça açık ve anlaşılır. Bu da programı kolay kullanılabilir yapmasıyla birlikte programlama mantığının da iyi bir şekilde anlaşılmasını sağlamaktadır. Eklenen her prosedürün, o nesneye ait fonksiyonun belirlenmesi (eklenebilmesi, otomatik çıkması) ayrı bir avantaj.” diyerek örneklendirmelerde bulunmuş ve mantığı anlamada yardımcı olacağından bahsetmiştir.

Diğer öğretmen adaylarının görüşleri çeşitlilik göstermektedir. Örneğin Sena “Programlama kavramlarını sürükleyip kod ekranında deneme yanılma yoluyla uğraşırken öğrenmeme yardımcı oldu. Ne işe yaradıklarını daha iyi kavrayabildim.” diyerek sürükle-bırak ara yüzünün programlama kavramlarının öğretimine etkisine değinmiştir. Zeynep ise:

Programlama kavramlarını öğrenmemizde çok büyük etkisi oldu. Kod kısmının anlaşılır, düzenli bir şekilde sunulması programlama kavramının

daha basit bir şekilde öğrenmemize etkili oldu. Örneğin; bir tavuğu kendi etrafında döndürmek için hangi bölümden, hangi kod bloğunu kullanacağımızı daha rahat bir şekilde ulaşıp, kullanabiliyoruz. Bu da bizim programlama kavramını, kavramlarını daha basit bir şekilde kalıcı olarak öğrenmemizi sağlıyor.

demiş ve kalıcı bir şekilde öğrenebileceğini düşündüğünden bahsetmiştir.

Hayatımızdaki pek çok durumun analizini de kolaylaştıracağından bahseden Fatma

Alice programlama öğretiminde, renk uyumunu, algoritmayı kavratır, ilişkilendirmeleri kurmayı kolaylaştırır. Örneğin, Ben seninle konuşabiliyorum ama aynı zamanda sizin bana dönmeniz de gerekiyor. Bu tarz ilişkileri kurabilir. Günlük hayatta herkes fark etmeden kullanıyor aslında bu tip ilişkileri ama incelikleri düşünmüyoruz. Bu ince ayrıntıları düşünmemizi sağlıyor. Başını eğip, kolunu kaldıracak, aynı zamanda konuşacak gibi.

diyerek görüşlerini dile getirmiştir. Ayşe ise programlama kavramlarını anlamak için önce konuya ilgi duymak gerektiğinden ve Alice'in bunu gerçekleştirebileceğinden bahsetmiştir.

4.3.4. Hazır kodların yazım hatalarını önlemesi. Bu bölümde hazır kodların yazım hatalarını önlemesine ilişkin öğretmen adaylarının görüşlerine yer verilmiştir. Katılımcılar, hazır kodlar hususunda Alice'in güçlü yönleri arasında daha önce sürükle-bırak yapısı bölümünde de yer verilen görüşlerle paralel düşüncelerinden bahsetmiştir. Örneğin, Zeynep:

Programlama kısmının en zorlanılan yerlerinden bir tanesi de el ile yazılan kodların sadece bir virgülünün eksik olması ya da bir kelimenin yanlış yazılmasının bütün programın çalışmasına engel olmasıdır. Fakat Alice programında böyle bir eksiklikten söz edilemez. Sadece sürükle bırak yöntemi

ve istenilen yerlere istenilen deęerlerin girilmesi ile kod bloęu hazır hale getirebiliyoruz.

diyerek yazım hatalarının önüne geçtięini dile getirmiştir. İstenilen yerlere istenilen deęerlerden kastı ise metotların ilgili parametrelerindeki düzenlemelerdir. Örneęin, bir insanın yürütürken kaç saniyede yürütüleceęinin girilmesi bir parametre sayesinde gerçekleştirilir ve bu parametrenin ismi duration'dır (süre). Selcan ise ortaöğretim öğrencilerindeki hiperaktif olma durumlarından bahsetmiş ve onları zapt edebilme için bir koddaki bir harfi bile yanlış yazdığında çalışmayan programlamanın bir işe yaramayacağından bahsetmiştir. Selcan'a göre Alice bunu engelleyerek programlamada önemli bir alternatif olarak göze çarpmaktadır. Sena ise bu konuda "Sonuna, başına getirmediğim noktalama işaretleri için hata da vermiyor." demiş ve aslında daha önceki programlama derslerinde kullandığı yazılımlara hayıflanmıştır. Deniz, Saliha ve Gaye de hazır kodların ve nesnelerin kullanılmasının avantajına ve kod kalabalığıyla uğraşmak zorunda kalmadıklarına dikkat çekmiştir.

4.3.5. Grup / bireysel çalışma. Alice ile programlama öğrenme yönteminin grup çalışmasına mı yoksa bireysel çalışmaya mı uygun olduğuna dair görüşler Tablo 10'da gösterilmiştir. Buna göre katılımcıların %52'lik bir bölümü Alice'in bireysel çalışmaya, %32'lik bir bölümü ise grup çalışmasına elverişli olduğunu düşünmektedir. Hem bireysel hem de grup çalışmasına elverişli olduğunu düşünen öğretmen adaylarının oranı %16'lık bir dilimi oluşturmaktadır.

Tablo 10

Katılımcıların grup veya bireysel çalışmaya ilişkin tercihleri

Katılımcıların Tercihi	n	Yüzde
Bireysel Çalışma	13	52
Grup Çalışması	8	32
Her ikisi de yapılabilir	4	16

Öğretmen adaylarının büyük bir bölümü Alice ile yapılan programlama öğretiminde bireysel çalışmaya daha uygun olduğunu belirtmiştir. Bunun nedeni olarak ise katılımcılar, grup halinde çalışırken parçaların birleştirilememesi probleminden, toplantı için zaman ayıramama, grup üyelerinin sadece kendi yaptıkları bölümü ve uyguladıkları becerileri öğrenebildiklerine kadar farklı görüşleri savunmuştur. Örneğin Abdullah “Bireysel çalışmayı seçme sebebim, gruplarda insanların çokta yardımcı olmaması birbirlerine. Grup çalışmasına elverişli ama benim yapıma uygun değil.” diyerek bazı öğrencilerin tercihlerinin bu yönde olabileceğini, grup üyeleri arasındaki yardımlaşmanın beklendiği gibi olmadığını belirtmiştir. Final projesini grup olarak yapmayı tercih eden Ayten ise “Bireysel çalışma için daha uygun. Çünkü yaptığımız projeleri birkaç parçaya bölüp birleştiremiyoruz.” diyerek Alice’te parçalar halinde yaptıkları projelerini birleştiremediklerinden bahsetmiştir. Her ne kadar video halinde çıktı alarak onları birleştirme yoluna gitseler de bu konuda güçlük yaşadıkları da gözlemlenmiştir. Hüseyin ise farklı bir noktaya parmak basmış ve “Bence grup çalışmasına pek uygun değil çünkü program görüntü ve koddan ibaret. Kodlarda ne yazdığını bilmiyorsan nesnelere nereye koyduğunun bir önemi kalmıyor. Grup çalışması yapılacaksa gruptaki kişilerin kodlara ve nesnelere hâkim olması lazım. Bireysel olarak daha iyi olabilir.” demiştir. Yani, sahne tasarımları (nesnelere seçilmesi yerleştirilmesi, düzenlenmesi) ile yazılan kod bloklarının (nesnelere hareketi, seslendirilmesi, şekil/konum/düzeylerindeki değişiklikler vs.)

arasında doğrudan bir bağlantı olduğundan ve bu yüzden her iki eylemde de ustalaşmış kişilerin grup çalışması yapabileceğine değinmiştir. Leyla ise zaman probleminden yakınmış ve “Grup arkadaşım Zeynep’le ne kadar ayrılmaz ikili olsak da projemizi tamamlamamız için okul dışında da vakit ayırmamız gerekiyor. Okul dışında görüşünce de proje yapılmıyor maalesef.” demiştir. Ve ekleyerek

En önemlisi 3B dünya tasarımı bireysel çalışmaya daha uygun olduğunu düşünüyorum. Çünkü bu dünyada hayal ettiğini ya da herhangi bir konuyu istediğin şekilde yansıtmaya fırsatın var. Ve iki kişinin hayal ettiği şeyler ya da ele aldığı bir konuyu yansıtmaya şekli çok az bir oranda kesişiyor. Çalışmalar bireysel olduğunda her zaman (gecenin 3’ünde) yapabilirsin. Hem grup arkadaşların tarafından hayal kısıtlaması olmadan proje yapılabilirdi.

demiş ve az önce bahsedilen zaman problemine tekrar değinerek, hayal gücünü istediği gibi çalışmalarına aktaramadığını dile getirmiştir. Sena ise “Grup çalışması verildiğinde çoğu zaman birlikte yapılamadığı için, iş bölümü yapıldığında bir projenin her aşamasında yer almayan öğrenci yalnızca kendi yaptığı bölümleri daha iyi öğrenir.” diyerek istenilen öğrenme kazanımlarının her bireyde gerçekleştirilmesini istiyorsak grup çalışmasının dezavantajına değinmiştir. Ve ekleyerek “Verilen ödevin durumuna göre tek başlarına da yapmalarını isteyebilirim. Yapamadıkları bir şey olduğunda zaten ya bana ya da arkadaşlarına sorarak öğrenmeye çalışırlar. Bu sayede projede onu öyle yapalım bunu böyle yapalım gibi olası tartışmalara da girmezler.” şeklinde görüşünü dile getirmiştir.

Grup çalışmasına elverişli olduğunu düşünen katılımcıların oranı da azımsanmayacak kadar çoktur. Alice ile grup çalışması yaptırılarak programlama öğretilmeli diyen öğretmen adaylarının tamamına yakını, grup üyelerinin farklı beceri ve düşüncelerinin istenilen işi yapmayı kolaylaştırdığını belirtmiştir. Arif “Etkili kullanılabilirse, bu ortamların grup çalışmalarına elverişli olduklarını düşünüyorum. Çünkü aşamalı birkaç bölümden oluşuyor.

Tasarım, yazılım, entegrasyon gibi. Her bireyin yatkın olduğu bir alan vardır. Ben her şeyi yapabiliyorum diyen biri aslında hiçbir şeyi iyi yapamıyordur.” diyerek tasarım, yazılım, entegrasyon gibi farklı becerilere sahip insanların bir araya gelerek bir bütünü oluşturabileceklerini belirtmiştir. Deniz

Bazı öğrenciler programın kod kısmında daha iyi iken bazıları ise tasarım kısmında daha iyidir. Grup çalışması olduğunda tasarımı iyi olan daha çok tasarımla ilgilenir programlaması iyi olan ise kod kısmıyla ilgilenir ve ortaya çok güzel ürünler çıkabilir. Ancak elimizde geldiğince öğrencilere hem tasarım hem de programlama kısmını öğretmeliyiz.

demiş ve grup üyelerinin birbirilerinin açığını kapatabileceğini belirtmiştir. Bir başka deyişle, Deniz ve bu konuda görüş bildirmiş diğer öğretmen adaylarına göre, grup çalışmasıyla tasarım ve kodlama kısımlarından birine daha yatkın olan (tasarım veya kodlama) öğrenciler diğerinin eksikliklerini o konuda kapatabilir.

Öğretmen adaylarının küçük bir bölümü Alice ile öğrencilerin grup halinde veya bireysel olarak çalışmalar yapabileceği görüşünü savunmuştur. Murat “İnsanın kendi hayal gücünü geliştirebileceği bir programlama tabanıdır. Grup çalışmasında kullanışlı bir programda olabilir ama bireysellik konusunda daha geliştirici olacağını düşünüyorum.” demiş ve bireyselliğin aslında daha ağır bastığından bahsetmiştir. Selcan ise “Grup çalışmasına elverişli çünkü farklı fikirler ve düşünceler çıkıyor. Ortaya daha değişik ürünler koyabiliyoruz. Bireysel olarak da uygulamalar yapılabilir ama uğraş gerektiriyor. Daha fazla çalışmamız gerekebilir.” diyerek bireysel çalışmada daha çok zaman gerektiğinden ve grup çalışmasının farklı fikirleri yansıtmada ve kullanmada daha elverişli olduğundan bahsetmiştir.

4.4. Programlama Konusunda Nitelikli Öğretmen Yetiştirme

Bu bölümde öğretmen adaylarının programlama öğretimi ve materyal geliştirme konusunda nitelikli olarak yetişmeleri konularına ilişkin görüşleri değerlendirilmiş, Şekil 11’de görülen üç başlıkta incelenmiştir.

Şekil 11

Programlama konusunda nitelikli öğretmen yetiştirme



4.4.1. Alice ile programlama eğitimi. Öğretmen adaylarına “Bilgisayar ve öğretim teknolojileri mezunu olan öğretmen adayları programlama eğitimlerinin başlangıcında Alice ile almış olsalar veya ileride programlama derslerini Alice ile verseler ne gibi faydalar sağlardı?” şeklinde soru yöneltildiğinde görüşlerinin çeşitlendiği görülmektedir. Önceki bölümlerde de ekseriyetle üzerinde durulan programlama kavramlarının daha iyi anlaşılması, programlamayı sevmelerine neden olması, üç boyutlu ortamın faydaları, eğlenceli öğretimin yapılabilmesi, öğrencilerin ilgisini çekmesi, programlama kavramlarının ve becerilerinin kalıcılığını arttırması hususuna değindikleri görülmüştür. Bunların dışında, örneğin Selcan, üniversitelerde programlama öğretimi yapılırken Alice ile başlanması gerektiğini dile getirmiştir. Murat “Alice yazılımıyla programlamanın temeli atılabilir bu yazılımla öğretmen adaylarına meslek hayatındaki yazılım bilgisi verilebilir.” diyerek öğrencilerde

programlamanın temellerinin atılabileceğinden bahsettiği ve Selcan gibi düşündüğü görülmüştür. Abdullah ise:

Alice'i öğrenirken bazı kuramları da öğrenmiş olduk. Kuramlar daha çok kafada yer etti. Normalde birisine sorduğunda ikili yükleme kuramı nedir diye iki saat düşünür. İkili yükleme kuramı işte görsel işitsel yollarla öğrenmesi, Alice'te bunu mantıken uyguladık yani. Hem görsel olarak görüyor hem de seslendirmeler yaptık. Bu yönden kuramları uygulayarak öğrenmiş olduk.

demmiştir. İkili yükleme kuramı öğretmen adaylarına materyallerini geliştirirken ilk defa tanıştırmış ve genel bilgi verdikten sonra araştırma yapıp Alice ile uygulamalarını geliştirmeleri istenmiştir. İkili yükleme kuramına ilişkin detaylı bilgiler giriş bölümünde de verilmiştir. Abdullah burada kuramın doğrudan Alice ile ilintili olduğunu ve uygulayarak öğrendiğini belirtmiştir.

4.4.2. Diğer branşların durumu. Öğretmen adaylarının tamamı matematik, fen ve sosyal bilgiler gibi diğer branşlardaki öğretmenlerin de Alice ile hazırladıkları materyalleri derslerinde kullanabileceğini belirtmiştir. Katılımcıların %60'ı diğer branşlardaki öğretmenlerin kullanabileceklerini düşünürken %40'ı kullanabilmelerini şarta bağlamıştır.

Kullanımı şarta bağlamayanlardan Saliha "Evet matematik gibi alanlardaki hocalar kullanabilirler. Çocukların hem görsel hem işitsel duyularına hitap etmiş olurlar. Program eğlenceli olduğu için çocukların ilgisini çeker. Çocuklar derste sıkılmamış olur veya derse nefret ederek gelmezler." demiş ve görsel-işitsel öğelerin yoğunluğu sayesinde öğrencileri derse çekebileceklerini belirtmiştir. Fethi " üç boyutlu yapısı sayesinde Alice anlatılan dersin sıradanlaşmasını engeller. Bazı derslerin daha iyi kavratılmasını sağlar" demiş, Bahar ise örneklendirerek "Matematik alanında yetişen bir öğretmen Alice ile derslerinde yol-hız problemlerini anlatabilir" demiştir.

Öğretmen adayları branş farkı gözetmeksizin diğer öğretmenlerin Alice'ten faydalanabileceklerini düşünmektedir. Örneğin Abdullah “Özellikle ilkokulda sınıf öğretmenleri kullanabilir. Atıyorum çarpım tablosunu öğretirken Alice üzerinde bir uygulama geliştirebilir. Ben kendi hikâyeme bakarsam bende ilkokul öğrencilerinin kişilik gelişimini daha iyi oturtması amacıyla oluşturduğum bir hikâye oldu. Kullanılabilir.” diyerek öğrencilerin kişilik gelişimini sağlayacağını düşündüğü bir materyal geliştirdiğinden ve sınıf öğretmenlerinin kullanabileceğini belirttiği görülmüştür. Selahattin ise “Kullanabilirler özellikle fen alanında küçük animasyonlar olsun fen konuları bunun için çok uygun ışık kaldırma basınç vb. her konuyla ilgili basit kodlarla güzel sonuçlar çıkabilir.” demiş ve fen konuları için uygunluğundan bahsetmiştir. Dilara kullanıcı ile etkileşime olanak sağladığından, Ali tarihi olaylar ve fizik gibi konuların anlatımında geliştirilen materyallerin kullanılabileceğini, Hüseyin slaytla anlatmaktansa Alice ile daha görsel ve etkili sunumlar yapılabileceğini belirtmiştir.

Katılımcıların yarıya yakını diğer branş öğretmenleri için materyal hazırlama konusunu olumlu görmekle beraber çekincelerini de dile getirmiştir. Ulaş “Diğer alanlardaki öğretmenler tarafından kullanılabilir fakat görselleri biraz daha zenginleştirip programın eksiklerinin giderilmesi gerektiğini düşünüyorum.” diyerek Alice'te yaşanan güçlükler bölümünde de belirtilen hususlardaki eksikliklerin giderilmesi gerektiğinden bahsetmiştir. Fatma ise materyal geliştirme kısmında bilgisayar öğretmenlerinden faydalanılmasının doğru olduğunu ve matematik gibi alanlardaki öğretmenlerin derslerine odaklanmaları gerektiğini belirtmiştir. Selcan yeni yetişen öğretmenlere Alice'in öğretilabileceğini ama hali hazırda öğretmen olan ve okullarda kendi tarzına göre eğitim vermeye alışmış öğretmenlerin bundan vazgeçmesinin imkânsız olduğunu söylemiştir. Arif ise “Alice bu alanlar için kullanılmayacak bir program değil. Gayet güzel de kullanılır. Ama BÖTE bölümü mezunları bunları kullanmıyorken, fen, matematik zümre öğretmenlerinin kullanmalarını düşünmek hayal olur.

Bu jenerasyonda böyle bir durumun olmayacağını düşünüyorum.” diyerek çekincesini belirtmiş ve bilgisayar ve öğretim teknolojileri eğitimi mezunu öğretmenlerin bile çoğu yerde kullanmadığı bu tarz programların diğerlerinden kullanılmasını beklemenin ütopyik olduğuna vurgu yapmıştır.

4.4.3. Kalıcılık. Kalıcılık konusundaki görüşler daha önce Alice ile ortaöğretimde ve üniversitelerde yapılacak programlama öğretimi ile ilgili başlıklar altında da dile getirilmişti. Öğretmen adayları görsellik, uygulayarak öğrenme, eğlenerek öğrenme ve gerçek hayata yakın bir ortam sağlamasından dolayı öğrenilen bilgilerin kalıcı olacağını düşünmektedir. Bunun yanı sıra, katılımcılar Alice ile programlama öğretimi yapılırken öğrencilerin gelişim özelliklerine dikkat edilmesi gerektiğinden de bahsetmişler, ezber gerektirmemesi, basit düzeyde sunulan kodların kullanımı ve ilgi çekici anlatımlar sayesinde de kalıcılığın artacağını belirtmişlerdir. Bu bölümde kalıcılık ile ilgili daha önceki bölümlerde değinilmemiş hususlar üzerinde durulmuş ve iki başlık altında toplanmıştır: (1) yazılan kodların sonuçlarını anlık olarak görebilme imkânı ve (2) pekiştireçler.

Alice’te yazılan kodların sonuçlarının anlık olarak görülmesi hadisesi, programın çalıştırılmasıyla gerçekleşmektedir. Farklı noktadaki kodların pasif/aktifleştirilmesi ile istenilen yerde bir hata olup olmadığı denetlenebilmektedir. Buda katılımcılara anlık olarak hazırlanan kodlara ve sahne tasarımlarına müdahale şansını beraberinde getirmektedir. Bu konuda Saliha:

[Öğrenilen bilgilerin] kalıcı olduğunu düşünüyorum. Çünkü yukarıda da belirttiğim gibi sadece sözel olarak kod mantığı anlatılmıyor. Kodları oluşturmanın mantığını Alice programı ile daha iyi anladım. Yazdığımız kodların sahneye nasıl etkisi olduğunu anında gördüğüm için kodlar kafamda oturdu yani ezberlememiş oldum.

diyerek hem daha önce bahsettiğimiz ezberden kurtarmasına hem de kalıcılığa değinmiştir. Saliha'ya göre bu programlamayı daha iyi anlamasına neden olmuştur. Sena ise:

Eğer biraz daha üzerinde dursaydım kalıcılığı artardı. Yani daha uzun süre boyunca göreceğim bir ders olsaydı ve daha fazla etkinlik oluştursaydım, öğrenmediğim ve eksik kalan yerler üzerinde de onları öğrenip kullanabilseydim daha güzel olabilirdi. Ancak bu şekilde de benim için oldukça fayda sağladı. Verdiğim hareketleri anında görebildiğim için hangi komutun ne işe yaradığını güzel bir şekilde öğrenebildim. Kendimde uygulamalar yaptığımdan öğrendiklerimin kalıcı olacağını düşünüyorum.

demiş ve öğretim sürecinin biraz daha uzun olması gerekliliğinden bahsetmiştir.

Bunun yanı sıra yazdığı kodları daha iyi anlamasını sağladığına ve uygulamalara değinmiştir.

Bu sayede öğrendiklerinin kalıcı olduğunu belirtmiştir. Lara “Eş zamanlı olarak yaptıklarımı görme şansım da oldu. Hata yaptığım zaman program çalışmadığında anında dönüt olarak gördüm. Düzelttiğim şekli de gördüm. Ve bu şekilde öğrenmem kalıcı hale geldi.” diyerek her ne kadar programlama yaparken hatalar yapsa da dönütleri anında gördüğünden ve öğrenmesinin kalıcı olduğundan bahsetmiştir.

Pekiştireçler yukarıda sıralanan diğer görüşlerin yanı sıra öğretmen adayların üzerinde durduğu bir başka husustur. Sevda bu konuda “Geçmişte aldığım algoritma mantığıyla aynı olsa da yönteminin farklı olması ve bunun görsellikle desteklenmesi kalıcılığı oldukça arttırdı. Sahneye eklediğimiz her objenin ilgili prosedürlerin eklenmesiyle birlikte hareket etmesi (olayın gerçekleşmesi) programlama kavramlarının da pekişmesinde etken rol oynadı.” diyerek pekiştireçlerden bahsetmiştir. Ayşe “Derste öğrendiklerimiz kalıcı oldu. Ki şöyle derste öğrendiklerimiz kapsamında haftalık ödevler hazırladık. Hazırladığımız ödevlerde öğrenmediklerimiz varsa araştırdık öğrendiklerimizi ise pekiştirmiş olduk.” demiş ve haftalık

etkinliklerde yaptıkları uygulamaların kendilerini arařtırmaya ittiđini ve bu sayede öğrenilenlerin pekiřtirildiđinden bahsetmiřtir.

4.5. Öz Yeterlilik Algısı

Bu bölümde öğrencilerin programlamaya ilişkin öz yeterlilik algılarının saptanması amacıyla uygulanan ön-test / son-test sonuçlarına yapılan istatistiksel analizler sonucunda elde edilen bulgulara yer verilmiřtir. Altun ve Mazman (2012) Türkçe'ye uyarladıkları “programlamaya ilişkin öz yeterlilik algısı ölçeđini” sadeleřtirerek 9 madde ve iki faktöre (basit programlama görevlerini gerçekteřtirme ve karmařık programlama görevlerini gerçekteřtirme) indirgemiřlerdir. Bunun yanı sıra cinsiyet, yař, sınıf, yıl bazında programlama deneyimi, alınan programlama ders sayıları ve kullanabildikleri programlama dillerine ait bilgiler de ölçek vasıtasıyla toplanmıřtır.

Ölçekte yer alan sorular ve bu soruların basit programlama görevlerini gerçekteřtirme ile karmařık programlama görevlerini gerçekteřtirme faktörlerinin hangisinde yer aldıđı Tablo 11'de gösterilmektedir.

Tablo 11

Programlamaya ilişkin öz yeterlilik ölçeği maddeleri (Altun ve Mazman, 2012)

Faktör	Madde
Basit	S1. “Merhaba Dünya” mesajının görüntülenebileceği bir program yazabilirim.
Programlama	S2. Üç sayının ortalamasını hesaplayan bir program yazabilirim.
Görevleri	S3. Verilen herhangi bir sayı dizisinin ortalamasını hesaplayan bir program yazabilirim.
	S4. İstenilenler açıkça tanımlandığında bir problemin çözümüne yönelik oldukça karmaşık ve uzun bir program yazabilirim
	S5. Yazacağım bir programı modüler bir biçimde organize edip tasarlayabilirim.
Karmaşık	S6. Yazdığım uzun ve karmaşık bir programdaki tüm hataları ayıklayabilir ve çalışabilir hale getirebilirim.
Programlama	S7. Uzun, karmaşık ve birden fazla dosya gerektiren bir programı kavrayabilirim.
Görevleri	S8. Bir programın daha okunabilir ve açık olması için uzun ve karmaşık kısımları yeniden yazabilirim.
	S9. Çevrede bir sürü dikkat dağıtıcı olsa bile programa odaklanma yollarını bulabilirim.

Araştırmaya katılmayı kabul eden öğretmen adaylara ait demografik bilgileri Tablo 12’de gösterilmiştir.

Tablo 12

Katılımcıların farklı değişkenlere göre betimsel istatistikleri

Değişken	Değer	Frekans(%)
Cinsiyet	Kadın	%41.70
	Erkek	%58.30
Sınıf	3. sınıf	%16.70
	4. sınıf	%83.30
Program yazılan yıl ortalaması	Hiç yazmadım (0)	%54.20
	1-3 yıl arası deneyim	%16.70
	4 yıl ve üzeri deneyim	%29.10
Alınan programlama dersi sayısı ortalaması	1-3 programlama dersi	%12.50
	4-6 programlama dersi	%75.00
	7 ve üzeri programlama dersi	%12.50

Katılımcıların öz yeterlilik algılarının saptanması amacıyla uygulanan ölçekten elde edilen verilere Wilcoxon bağımlı iki örneklem testi uygulanarak, Alice ile programlama öğretiminin öğretmen adaylarının öz yeterlilik algılarında bir değişime yol açıp açmadığına bakılmıştır. Öz yeterlilik algılarındaki değişime ilişkin hipotezler aşağıda listelenmiş ve sonuçları Tablo 13 ve Tablo 14’te gösterilmiştir:

Tablo 13

Wilcoxon bağımlı iki örneklem testi: ön-test ile son-test sonuçlarının genel toplamları

	N	Mean Rank	Sum of Ranks
Negative Ranks	6 ^a	6.92	41.50
Positive Ranks	18 ^b	14.36	258.50
Ties	0 ^c		
Total	24		

Tablo 14

Wilcoxon bağımlı iki örneklem testi: ön-test ile son-test sonuçlarının genel toplamların karşılaştırılması

Test Statistics ^a	
Z	ST - ÖT -3.102 ^b
Asymp. Sig. (2-tailed)	.002

a. Wilcoxon bağımlı iki örneklem testi

b. Negatif sıralamaya dayalı

Yukarıdaki iki tabloda ST: Son Test, ÖT: Ön Test'i ifade etmektedir. Katılımcıların ön test ve son test puanlarının karşılaştırılması amacıyla uygulanan Wilcoxon bağımlı iki örneklem testi neticesinde $Z = -3.102$ ve $p = .002 < .005$ 'dir. Yani, araştırma sorularının altında yer verilen hipotezlerde H_0 reddedilir ve H_a hipotezi kabul edilir. Alice ile yapılan programlama öğretimi, katılımcıların programlamaya ilişkin öz yeterlilik algılarını olumlu yönde etkilemiş ve istatistiksel olarak anlamlı bir fark bulunmuştur.

Katılımcıların toplam puanları karşılaştırıldıktan sonra, aynı zamanda öz yeterlilik algılarındaki soru bazlı değişimlere de Wilcoxon bağımlı iki örneklem testi ile bakılmış ve bu istatistikler Tablo 15 ve Tablo 16'da gösterilmiştir.

Tablo 15

Öz yeterlilik algılarındaki soru bazlı değişimler: Z ve p değerleri

	ST1 - ÖT1	ST2 - ÖT2	ST3 - ÖT3	ST4 - ÖT4	ST5 - ÖT5	ST6 - ÖT6	ST7 - ÖT7	ST8 - ÖT8	ST9 - ÖT9
Z	-1.589 ^b	-1.674 ^b	-1.969 ^b	-2.862 ^b	-3.520 ^b	-2.347 ^b	-2.696 ^b	-2.229 ^b	-2.476 ^b
Asymp. Sig. (2-tailed)	.112	.094	.049	.004	.000	.019	.007	.026	.013

5. Bölüm

Tartışma ve Öneriler

Öğretmen adaylarının üç boyutlu programlama öğretimine ilişkin görüşleri ve öz yeterlilik algılarına ilişkin bulgulardan yola çıkarak elde edilen verilerin çözümlenmesiyle ortaya çıkan sonuçlar bu bölümde yer verilmiştir. Bu bölümde ayrıca çalışmanın sınırlılıkları ve ileriki çalışmalarda odaklanılabilecek hususlar konusundaki öneriler de sunulmuştur.

5.1. Programlama Öğretiminde Üç Boyutlu Dünyaların Kullanımı

5.1.1. Alice ile programlama öğretimi. Öğretmen adayları Alice yazılımı sayesinde üç boyutlu dünya tasarımı becerilerinin geliştiğini düşünmekte ve bu doğrultuda bunun ileride verecekleri bilişim derslerinde işlerine farklı noktalarda yarayabileceğinden bahsetmektedir. Katılımcılara göre, bu becerileri sayesinde gelecekteki öğrencileri tıpkı kendileri gibi hayal güçlerini aktif olarak kullanabilecek, problem çözmede, programlama derslerini eğlenceli bir şekilde öğrenme ve anlama noktasında onlara yardımcı olacaktır. Alanyazına bakıldığında, bu bulguları destekleyen ve Alice kullanıcılarının üç boyutlu dünya ve onun tasarımı sayesinde yukarıda sıralanan noktalarda kendilerini geliştirebileceklerinden bahsedildiği görülmektedir. Örneğin, öğrenciler programlama öğretimi yapılırken üç boyutlu ve aynı zamanda interaktif dünya sayesinde problem çözme becerilerinin geliştirirler. Bir başka deyişle araştırmacılar problem çözmenin öğretilmesinde üç boyutlu dünyada hazırlanan animasyonlaştırılmış çıktıların önemine değinmektedir (Cooper vd., 2000a; Cooper vd., 2003a; Cooper, Dann, & Pausch, 2003b; Dann, Cooper, & Pausch, 2002). Alice ile yapılan programlama öğretiminde eğlenerek öğrenme gerçekleştirilebilir. Zevkli bir şekilde öğretim yapılabilmesi (Al-Tahat, 2014; Howard vd., 2006; Mullins vd., 2009), kullanıcıların programlamayı sevmeleri (Al-Tahat, 2014) ve bu sayede derse olan ilgi, katılım ve başarılarının arttığı (Mullins vd., 2009) sonuçlarına ulaşılmıştır. Alanyazındaki örnekler eğlenerek öğrenmenin olası etkilerini de ortaya koymakta ve bu sayede yapılabileceklere ilişkin ışık tutmaktadır. Programlama

derslerini anlama ve özellikle programlama kavramlarının öğretimi noktasında da çok sayıda çalışma yapılmıştır. Örneğin, Algoritma yada işlem basamaklarını anlamlandırma (Cooper vd., 2003a) ve eş zamanlı olarak yazılan kodların üç boyutlu bir ortamda gösterimi sayesinde kavranmasını kolaylaştırmaya (Dann vd., 2002) değinilmektedir. Fakat şu anki bilgimiz dâhilinde, literatürde soyut kavramların somutlaştırılması ve hayal gücünün aktif kullanımı hususunda bir çalışmaya rastlanmamaktadır. Katılımcıların görüşleri doğrultusunda ileriki çalışmalarda belki bu iki husus üzerine odaklanabilir.

Katılımcılar Alice'i kullananların, programlama derslerine ilgisinin ve katılımının artacağını düşünmektedir. Derse ilgi ve katılım konusunda eğlenerek öğrenmenin, görselliğin ve üç boyutun etkisi bu noktada yadsınamaz. Bu hususta dijital yerlilerden bahsetmek gerekebilir. Prensky (2001) öğrencilerin radikal bir şekilde değişim gösterdiklerini ve var olan eğitim yapısının sürdürülemeyeceğinden bahsetmektedir. Yeni nesli dijital yerliler olarak tanımlayan Prensky, öğretmenleri ise dijital göçmenler olarak sınıflandırmakta ve eski usul yöntemlerle artık dijital yerlileri eğitemeyeceklerini anlamaları gerektiğinden bahsetmektedir. Dijital yerliler, bebekliklerinden itibaren teknolojiyle iç içe yaşamakta, öğrenmenin eğlenceli olmasını istemekte, bilgiye internetten erişim, sosyal medya ile iletişim kurmaları, etkileşimli ortamları sevmeleri, hazırlanan ödevlerin dijital ortamda düzenlenmesi ve yönlendirmesi noktalarında dijital göçmenlerden farklılık göstermektedir (Colleen, 2015; Prensky, 2001). Yani, onları derse çekebilmek ve katılımlarını arttırmak için alışkın oldukları yöntemleri kullanmak en mantıklısı olacaktır. Bu noktada Alice görsellik, eğlenerek öğrenme ve üç boyutlu yapısı sayesinde, aynı zamanda kendileri de dijital yerliler olarak tanımlanabilecek katılımcıların tamamına yakınının (n=24) da belirttiği gibi ilgilerini çekebilir ve derse olan katılımlarını arttırabilir. Nitekim bu hususta alanyazında Alice'in derse ilgi ve katılımı arttırdığına ilişkin araştırmalarda bu görüşü desteklemektedir (Mullins vd., 2009).

Gerçekleştirilen görüşmeler esnasında üstünde durulan bir başka konu ise Alice'in programlama öğretiminde, öğrenilen bilgilerin kalıcılığına (retention of knowledge) ve ders başarısına olumlu yönde etkisi olduğuna ilişkindir. Katılımcıların tamamına yakını (n=21) bu konuya dikkat çekmekte, geri kalanları (n=4) ise Alice'de yaşanan güçlüklerin kalıcılığa herhangi bir etkisi olmayacağını düşünmektedir. Kalıcılık konusunda öğretmen adayları görselliği, uygulayarak ve eğlenerek öğrenme imkânını, gerçek hayata yakın bir ortam sağlamasını (üç boyutlu oluşu) gerekçe olarak göstermektedir. Alanyazına bakıldığında Alice ile yapılan programlama öğretiminin kalıcılığa etkisine ilişkin pek çok araştırma göze çarpmaktadır. Bu çalışmalar, öğrencilerin Alice kullanarak yapılan programlamaya giriş derslerinde edindikleri bilgileri, daha sonra aldıkları programlama derslerine aktarımı ve Alice ile programlamaya ilişkin geliştirdikleri olumlu tutumu diğer programlama derslerinde de sürdürmeleri üzerine odaklanmaktadır (Cooper vd., 2003a; Cooper, 2010; Kelleher & Pausch, 2007; Powers vd., 2007). Buna ek olarak, farklı yazılımlarla yapılan uygulamalara bakıldığında öğrenilen bilgilerin kalıcılığının artırılmasında görsellik (Choi & Johnson, 2005, 2007; Zengin vd., 2013), uygulayarak öğrenme (Sezginsoy & Akkoyunlu, 2011), eğlenerek öğrenme (Sezginsoy & Akkoyunlu, 2011) etkilidir. Son olarak, Alice gerçek yaşama yakın üç boyutlu bir ortamı kullanıcılarına sağlar (Al-Ahat, 2014) fakat bunun kalıcılığı etkileyip etkilemediğine ilişkin alanyazında bir bulguya rastlanmamaktadır. Powers ve diğerlerine (2007) göre öğrenenlerin gerçek hayattaki gibi hareketlerini beklemelerinden ötürü kullanıcıların morallerinin bozulup zaman zaman programlamadan uzaklaşmalarına neden olmaktadır. Fakat hali hazırdaki programlama öğretiminde kullanılan araçlara bakıldığında üç boyutlu bir ortamı sağlaması bile başlı başına bir avantaj olarak görülebilir. Bunun yanı sıra, bu araştırmadaki öğretmen aday görüşlerini bir tarafa koyacak olursak, akılda kalıcılık üstüne daha uzun süreli çalışmalar yapılmalıdır. Yani, ileride yapılacak çalışmalar uzun süreli yapılmalı ve periyodlar halinde değerlendirmeler yapılarak bilginin kalıcılığı ölçülmelidir.

Son olarak, Alice ile yapılan programlama öğretiminin akademik ders başarısını olumlu yönde etkilediğine ilişkin alanyazında pek çok çalışmaya rastlanmaktadır (Bishop vd., 2006; Dann vd., 2012; Johnsgard & McDonald, 2008; Moskal vd., 2004). Başarıyı nasıl arttıracığına ilişkin katılımcıların gerekçeleri ise kalıcı öğrenme ile paralellik göstermektedir.

Öğretmen adayları Alice'in özellikle programlamaya giriş derslerinde kullanılabileceğinden bahsetmiştir. Katılımcılar, Alice'in gerek üniversitelerde gerek ilk ve orta dereceli okullarda programlama öğretimi yapılırken, programlamanın temellerini verme açısından işe yarar bir öğretim teknolojisi olduğunu söylemiştir. Derslerde kullanım biçimine etki eden faktörlere bakıldığında katılımcılar görüşlerini uygulayarak öğrenme imkânı, eğlenerek öğrenme imkânı ve programlamaya görsellik katması olarak dile getirmektedir. Daha önce de bahsedildiği gibi bu üç faktörle ilgili alanyazında bir hayli araştırma bulunmakta ve Alice'in bunları sağlayabileceğine ilişkin görüşler sıralanmaktadır. Dann ve diğerleri (2012) her ne kadar Alice doğası itibariyle programlamaya giriş derslerinde kullanılmak üzere hazırlanan bir yazılım olsa da Alice ile öğrencilere kazandırılan programlama bilgisinin, verilecek ileri düzey programlama derslerinde de kullanılması üzerine odaklanmaktadır. Özellikle ileri düzey programlama yapılabilecek alanlardan birisi olan Java hakkında, Alice'te yazılan kodların Java çıktılarına da erişilebilir hale gelmesi bu konudaki en güzel örneği teşkil etmektedir.

Öğrencilerin programlama dersine olan tutumlarının değişeceğine ilişkin görüşler eğlenerek öğrenme üzerine odaklanmakta ve üç boyutlu ortamda yapılan programlama öğretiminin derse olan tutumu olumlu etkileyeceğinden bahsedilmektedir. Yani, katılımcılara göre öğrenciler programlama derslerinden zevk alarak, isteyerek ve gönüllü bir şekilde derslerini takip ederler.

Öğrencilerin kendini geliştirmesi ve bazı yeterliliklere sahip olması konusu üzerine görüşler, hayal güçlerinin aktif kullanımı – somutlaştırma ve 21. yüzyıl becerilerinin

kazanılmasına yönelik olanlar olmak üzere ikiye ayrılabilir. Katılımcılar somutlaştırma konusunda, üç boyutlu bir ortamın gerçek yaşama en yakın ortam olduğu görüşü savunmakta, bu sayede hayal güçlerinin kullanımı ve istenilenlerin gerçekleştirilmesinin daha kolay olduğunu söylemektedir. 21. yüzyıl becerileri son yıllarda üzerinde bir hayli araştırma yapılan alanların başında gelmektedir. Öğretmen adaylarına göre, Alice ile yapılan programlama öğretimi, problem çözme becerisi kazanma, yaratıcılığın geliştirilmesi ve liderlik vasfının kazanılmasında öğrencilere yardımcı olma potansiyeline sahiptir.

Öğretmen adaylarının, yukarıda tartışılan ve Şekil 8’de de yer alan ilişkilere dair görüşlerinden şu sonuçlar çıkarılabilir:

1. Hayal gücünün kullanılması – somutlaştırma ile öğrenci gelişimi ve 3B dünya tasarımı deneyimi arasında bir ilişki söz konusudur. Yani, katılımcılar üç boyutlu ortamda yapılan programlama öğretimi sayesinde öğrenenlerin hayal güçlerini aktif olarak kullanabileceğini ve soyut kavramları somutlaştırmada onlara yardımcı olacağını düşünmektedir. Bu sayede, öğrenciler kendilerini ve 3B dünya tasarımı deneyimlerini geliştirebileceklerdir.
2. Katılımcılara göre, eğlenerek öğrenme ve altında sunulan iki kod olan oyun ve animasyon imkanını sunan Alice ile programlama öğrenenler 3B dünya tasarımı deneyimi, derse ilgi ve katılım, kalıcı öğrenme ve başarı, derse tutum, Alice’in kullanım biçimi, öğrenci gelişimi konularında olumlu yönde etkilenecektir.
3. Katılımcılara göre, üç boyutlu programlama araçlarının, programlamaya görsellik katmasından dolayı öğrenenlerin derse ilgi ve katılımı artacak, kalıcı öğrenme ve başarı sağlanacak, Alice’in kullanım biçimi etkilenecek ve öğrenciler kendilerini farklı konularda daha fazla geliştirebilecektir.

4. Uygulayarak öğrenme veya deneme yanılma yoluyla öğrenme temasında öğretmen adayları programlamada kalıcı öğrenme ve başarının sağlanabileceği ve Alice'in kullanım biçiminin etkileneceğini düşünmektedir.

5.1.2. Alice'te karşılaşılabilecek güçlükler. Dünyanın en iyi teknolojisini kullanıyor olsak bile onunla sorunlar yaşamamamız içten bile değildir. Öğretmen adaylarına göre nesne konumlandırma, dil, performans sorunları, kamera kullanımı, nesne galerisinin zenginleştirilmesi ve metotlardaki sınırlılıklar karşılaşılan temel problemlerdir.

Chang'ın (2014) da belirttiği gibi nesnelerin üç boyutlu dünyada kaybolması ve yerleştirmede sorunlar yaşamaktadır. Fakat bizim durumumuz açısından bakıldığında nesne konumlandırma sorunları özellikle uygulamaların başlarında öğretmen adayları tarafından bir problem olmuş, Alice ile vakit geçirdikçe bu sorun ortadan kalkmıştır.

Türkçe eğitim yapan bir üniversite olan Uludağ Üniversitesinde İngilizce konusunda yetersiz olduklarını bildiren katılımcılar, bazı metotları anlamada ve ara yüzdeki araçları kullanmada güçlük çekmektedir. Özellikle Türkiye'deki okullarda hızla yaygınlaşan Scratch kullanımı düşünüldüğünde, Alice'in de Scratch gibi Türkçe ara yüz desteğiyle derslerde kullanılması gerekebilir. Fakat katılımcıların söylediklerinin aksine, öğrencilerin programlamada kendilerini gerçek manada geliştirmelerini istiyorsak İngilizce ortamda yapılan bir öğretimin olası olumlu etkilerini de incelemek gerekmektedir. Çünkü ileri düzey programlama yapılan dillere (Java, C++, C#) bakıldığında tüm kodlar genellikle İngilizce alt yapısından gelmektedir. Bunun yanı sıra bu tip öğretim araçları öğrencilere temel İngilizce terimlerini anlama noktasında yardımcı olabilir.

Bir başka problem olan yazılımdaki performans sorunlarıdır. Özellikle sıkça karşılaşılan hata ekranları (Mullins vd., 2009; Skyes, 2007) Alice'i kullananlarda programlamadan soğumaya neden olabilir. Bunun yanı sıra, Skyes'ın (2007) da belirttiği gibi, kullanıcılar Alice'in veya bilgisayarın donmasından kaynaklanan sorunlarla karşılaşmaktadır.

Bu durum veri kayıplarına neden olmakla birlikte tekrar aynı kodları yazmayı, nesnelere eklemeyi de beraberinde getirmektedir.

Kamera nesnesinin kullanımı ve mantığını anlama da güçlük çeken öğretmen adayları olmuştur. Fakat birçok katılımcı da bunun aksine, Alice’de kameranın bile nesne şeklinde eklenebiliyor ve açılarının değiştirilebiliyor olmasının yazılımın güçlü yanlarından biri olduğunu belirtmektedir.

Nesne galerisi Alice’te hazır olarak kullanıcılara gelmekte ve yeni üç boyutlu nesnelerin eklenmesine izin vermemektedir. Bu durum katılımcılar tarafından olumsuz bir yan olarak belirtilmiş, önceki sürümlerde yer alan bazı nesnelere ulaşamadıklarını belirtmiştir. Temel bazı nesnelerin olmamasından dolayı, istediklerini eklemeyen öğretmen adaylarında hayal kırıklığına neden olmaktadır. Dört aylık kısa bir süreçte kendisini yenileyen Alice’in, bu konuda kendisini geliştirmesi gerekebilir.

Nesnelerin hareketini sağlamada, onlara bir eylemin yaptırılması için metotlar ve prosedürler kullanılmaktadır. Bunların sayısındaki yetersizliklere değinen katılımcılar, hazır halde metotların kendilerini sunulmasını istemiştir çünkü yeni metotlar oluşturmada güçlük yaşamışlardır. Mullins ve diğerleri (2009) yaptıkları çalışmada yeni metot oluşturmada sorun yaşayan kullanıcıların olduğunu belirtmektedir. Bishop-Clark ve diğerlerine (2007) göre Alice bazı noktalarda öğrencileri sınırlandırmaktadır. Fakat bu noktada Alice geliştiricileri her şeyi kullanıcılara hazır olarak sunmak yerine, bir problemi çözerken kendi yöntemlerini oluşturmaları istiyor olabilir. Nitekim eldeki imkânları kullanarak bir problem durumunun çözülmesi problem çözme, yaratıcı düşünme ve eleştirel düşünme gibi bazı becerilerin gelişimine katkı sağlayabilir. Mesela, öğretmen adaylarının birçoğu insan veya hayvan figürlerini betimleyen nesnelerin yürütülmesinde sorun yaşamıştır. Ama bu sorunu çözmek için de farklı yollara başvurmuştur. Kimisi yeni bir metot tanımlayarak bu sorunu çözmüş, kimisi bir olay (event) içerisinde yazdıkları döngüyle işin üstesinden gelmiştir. Yani, bu

durum beraberinde “Öğrencilerin bir problemi çözmelerini belirli şartlara/eldeki imkânlarla bağlamadıktan sonra acaba programlama öğretimi gerçek anlamda amacına ulaşabilir mi?” sorusunu getirmektedir.

Alanyazına bakıldığında, katılımcıların belirtmediği bazı unsurlar da göze çarpmaktadır. Powers ve diğerleri (2007) yukarıda sıralananların yanı sıra Alice’ten C++ ve Java gibi programlama dillerine geçişte yazım kurallarının öğrenilmemesinden dolayı sorun yaşayanlar olduğunu vurgulamaktadır. Fakat yapısı itibarıyla programlamaya giriş derslerinde kullanılan Alice yazılımı, temelini verdikten sonra ne kadar yazım kuralları (syntax) konusunda suçlanabilir? Sonuçta, Alice syntax hatalarının yapılmasını engellemek üzere oluşturulmuştur. Bu durum ancak ileri düzey programlama derslerini veren öğretmenlerin halletmesi gereken bir sorun olarak göze çarpmaktadır. Yazılımın, doğasında yazım kurallarını öğretmek yoktur.

5.1.3. Alice’in güçlü yönleri

Katılımcılara göre Alice’in güçlü yönlerinin başında görsellik gelmektedir. Görsellik konusu ekseriyetle öğretmen adaylarının vurguladığı ve pek çok açıdan programlama öğretimine katkı sağladığını düşündüğü alanların başında gelmektedir. Öğretmen adayları görsellik kodunun altında sürükle-bırak yapısına, geniş bir görsel kütüphaneye (nesne galerisine) sahip olduğuna, hata tespitinin ve düzeltilmesinin kolaylaştığına ve programlama derslerinin zevkli geçtiğine değinmektedir. Ayrıca, Alice’teki görsellik sayesinde kolaylıkla programlama öğretiminin yapılacağına, dikkat çekiciliğini arttırdığına, 3B oluşu sayesinde soyut kavramların somutlaştırılabileceğine ve akılda kalıcılığa etkisinden bahsetmektedir. Nitekim bu sıralananların büyük bir bölümü hakkında alanyazında geniş bir çalışma olduğu görülmektedir. Örneğin tıpkı katılımcılar gibi, Malliarakis ve diğerleri (2014) satır satır kod yazımının öğrencilere sıkıcı geldiğinden ve onları programlamadan soğuttuğundan bahsetmektedir. Bunun yanı sıra, görselliğin hata tespiti ve düzeltilmesi işlemine sağladığı

katkı (Cooper vd., 2000a; Mullins vd., 2009) ve yazım hatalarını (syntax errors) önlemesi (Al-Ahat, 2014) bu tarzdaki bir ara yüzün olası faydalarındandır. Alice’de daha öncede bahsettiğimiz gibi tüm nesnelere üç boyutlu olarak hazırlanan dünyalara eklenmektedir. Katılımcıların ufak bir bölümü, kameranın bile üç boyutlu nesne olarak eklenebilmesi ve rahatlıkla kamera açılarıyla oynanmasını Alice’in başka bir güçlü yanı olarak görmektedir. Yani sonuç olarak programlama öğretiminin kolaylaşmasında, zevkli hale gelmesinde, öğrenilen bilginin kalıcılığı ve ders başarısının artmasında Alice etkili bir öğretim aracı olma potansiyeline sahiptir.

Öğretmen adaylarının yarısından fazlası, bu çalışma süresince alanyazın bölümünde işbirlikli öğrenmeye ilişkin verilen vizyon çerçevesinde çalışmalarını sürdürmüştür. Grup çalışması veya bireysel çalışmaya ilişkin görüşleri sorulduğunda, katılımcıların büyük bir bölümünün (%52) bireysel çalışmayı tercih ettiği görülmektedir. Bunun temel sebeplerine bakacak olursak, Alice yazılımında parçalar halinde çalışmada yaşanan güçlükler ve Howard’ın (2006) da belirttiği gibi toplantı için grup üyelerinin zaman ayıramaması veya ortak bir zaman dilimi bulunamaması gelmektedir. Bunun yanı sıra, katılımcılara göre eğer grup çalışması yapılacaksa, grup üyelerinin sadece kendi yaptıkları bölümü (tasarım, kodlama gibi) öğreneceğinden, öğrencilerine kazandırılacak kazanımların ne olduğuna ve ne ölçüde kazandırılacağına dikkat edilmelidir (Denton, 2012). Kozlowski ve Ilgen’e (2007) göre grup çalışması farklı beceri ve düşüncelere sahip grup üyelerinin birleşimi sayesinde yapılacak işi kolaylaştırmaktadır. Gerçekten de bakacak olursak, bir Türk atasözünde de geçtiği gibi “ Bir elin nesi var, iki elin sesi var.” mantığı burada devreye girmektedir. Yani güçlerini birleştiren grup üyeleri muhtemelen daha rahat öğrenecektir. Tüm bunlar gösteriyor ki, öğretmen adayları Alice ile programlama eğitiminin bireysel olarak verilmesinden yana olsa da, grup çalışmasını savunan öğretmen adaylarının görüşleri de dikkatle incelenmeli ve mümkünse Alice yazılımı bu çerçevede kendini yeni bir forma evirmelidir. Mesela, bir projenin

parçalarının birleştirilmesinin kolaylaştırılması veya hızlı bir şekilde yapılması üzerinde çalışılabilir. Bir bölüm öğretmen adayı (%16) ise hem grup çalışması hem de bireysel çalışmanın yapılabileceğine dikkat çekmektedir.

5.1.4. Alice'in nitelikli öğretmen yetiştirmeye katkıları

Diğer branşlara Alice ile programlama öğretimi yapılması veya farklı branşların derslerinde Alice ile hazırlanan materyallerin kullanımı konusu da diğer önemli bir husustur. Katılımcılara göre farklı branşlarda Alice'in kullanımı konusunda öğretim yapılabilir. Ayrıca, derslerinde Alice ile hazırladıkları materyalleri kullanarak daha rahat bir öğretim yapabilirler. Öğretmen adaylarının matematik, fen, sosyal bilgiler, sınıf öğretmenliği alanlarına ilişkin örneklendirmelerde bulunduğu görülmektedir. Örneğin, Alice ile bir tarihi olay veya Fen bilimlerinden bir konu (kaldıraçlar, sürtünme vb.) rahatlıkla hikâyeler oluşturularak anlatılabilir. Fakat bu noktada, bir şerh düşerek, katılımcıların da belirttiği gibi materyal hazırlama ve kullanma konusunda eski öğretmenlerde sorun yaşanabileceğine değinmek gerekmektedir. Dijital göçmenler yeni öğretim teknolojilerinin kullanımı ile ilgili gerçekten sıkıntı yaşayabilirler ve direnç gösterebilirler (Colleen, 2015; Prensky, 2001). Bunun yanı sıra an itibariyle çoğu üniversitede öğretmen adaylarına Alice'in gösterilmemesi de büyük bir sorun olarak ortada durmaktadır. Yaygınlaştırılmasına ilişkin bu çalışma bir ön veri olarak düşünülebilir ve bu konuda çalışmalar yapılabilir.

Programlama öğretiminin öğrenenlerin problem çözme becerilerinin gelişmesi konusunda yararlı olabileceğini ifade eden öğretmen adayları, programlamanın özünü oluşturan algoritma öğreniminin ve dolayısıyla algoritmik düşünmenin Alice'in sunduğu imkânlar ile sağlanabileceğini düşünmektedirler. Nitekim Naughton (2012), 21. yüzyıl bireylerinden beklenen en önemli yeterliliklerden biri olan problemleri çözme becerisinin, bireylerin kendi düşünme biçimleri ile bilgisayarların birleşmesiyle daha da gelişebileceğinden bahsetmektedir. Bu durum, günümüzde bireylerin farklı düşünme

biçimlerinin farkında olması, kullanması ve problem çözümüne giden yolda bilgisayar mantığıyla kararlar almasının, yani, bilgisayarlı düşünmesi gerekliliğini ortaya koyduğunu söylemek mümkündür. Araştırma sonucunda elde edilen, katılımcıların programlama öğretiminde kullanılan Alice yazılımının çok yönlü düşünmeye ve alternatif çözüm yollarının oluşturulmasına katkıda bulunabileceğine ilişkin görüşleri, Alice yazılımının bilgisayarlı düşünmeyi bu açılarından destekleyebileceğini göstermesi açısından önemlidir. Fakat alanyazında Alice yazılımının bilgisayarlı düşünme üzerindeki etkilerine ilişkin çalışmaların sınırlı olduğu görülmektedir.

5.2. Öğretmen Adaylarının Programlamaya İlişkin Öz Yeterlilik Algıları

Öğretmen adaylarının programlamaya ilişkin öz yeterlilik algıları ön-test ve son-testler uygulanarak basit deneysel desene göre ölçülmüştür. Buna göre, Tablo 13 ve Tablo 14 incelenecek olursa, Alice ile programlama eğitiminden geçen öğretmen adaylarının öz yeterlilik algılarında istatistiksel olarak anlamlı bir farklılık bulunmaktadır ($Z = -3.102$ ve $p = .002 < .005$). Katılımcıların “Programlamaya ilişkin öz yeterlilik algısı anketine” verdiklerine cevaplar doğrultusunda basit ve karmaşık düzeyde programlama görevlerini gerçekleştirme konusunda öz yeterlilik algıları artmıştır. Buna rağmen, Tablo 15 ve Tablo 16’da görülebileceği üzere soru bazında Wilcoxon bağımlı iki örneklem testi uygulandıktan sonra, “basit programlama görevlerini gerçekleştirme” faktörünün birinci ($p = .112$) ve ikinci ($p = .094$) sorusunda istatistiksel olarak anlamlı bir farklılık olmadığı görülmüştür. Yani öğretmen adayları “Merhaba dünya” mesajını içeren ve üç sayının toplamını bulan bir program yazma konusunda bir değişim söz konusu değildir. Ayrıca, basit programlama görevlerini gerçekleştirme faktörünün bu iki sorusunda ilgili sonuçların ortaya çıkmasında, öğretmen adaylarının daha önce aldıkları derslerin etkisi olduğu söylenebilir. Sonuçta, aldıkları programlama dersi sayılarına bakıldığında 4 ila 6 tane programlama dersi alanların oransal olarak çoğunlukta olduğu (%75) göze çarpmaktadır.

Bandura'nın (1994) da ifade ettiđi gibi öz yeterlilik algısı yüksek olan bireyler bir zorlukla karşılaştıklarında onu çözmeye çalışacaklar ve bu sayede daha başarılı olacaklardır. Rotter (1990) ise öz yeterliđi düşük bireyleri çevresel faktörlerle davranışlarının sonuçlarını değerlendiren/açıklayan bireyler olarak tanımlamaktadır. İlk araştırma sorusuna cevap ararken, öğretmen adaylarının Alice ile eğlenerek öğrenme imkânı, uygulayarak öğrenme imkânı, olumlu tutum geliştirilmesi, hayal gücünün kullanılması ve görsellik gibi etkenlerin odaklandıkları görülmektedir. Dolayısıyla bu gibi faktörlerin öz yeterlilik algıları üzerinde etkisi ile ilgili ileride araştırmalar yapılabilir. İkincil olarak, Bandura'nın (1977) da belirttiđi gibi bireylerin öz yeterlilik algıları kaygı ve stres gibi durumlardan etkilenmektedir. Yukarıda sayılan etkenlere imkân vermesiyle Alice'in kaygı ve stresin önüne geçtiđi söylenebilir. Alice'de hata tespiti ve düzeltilmesinin kolay olması, yazım hatalarından bağımsız bir arayüzü kullanıcılara sunması, çoğunlukla grup çalışması yapan bireylerin grup arkadaşlarından etkilenmeleri ve yazılımdaki görsel dönütlerin fazlalığı da yine bu etkenler arasında sayılabilir. Bilgimiz dahilinde Alice ile öz yeterlilik konusunda alanyazında Wang ve diğerlerinin (2009) araştırması bu çalışmanın bulgularıyla paralellik göstermektedir.

Bu çalışmanın bilgisayar ve öğretim teknolojileri eğitimi alanına çeşitli açılardan katkı sağladığı ve sağlayacağı düşünülebilir. 21. yüzyılda derslerde öğrencilerin ve öğretmenlerin kullanabilecekleri teknolojik araçlar ve yazılımlar zenginleşmiştir. Küresel dünyada artık Dünya ülkelerinin birinde üretilen bir yazılımın bütün ülkelere eş zamanlı olarak dağıtılması imkânı mevcuttur. Bu çerçeveden bakıldığında, ilk olarak Amerika'da programlama becerilerinin edinilmesi amacıyla geliştirilmiş olan Alice yazılımının Türkiye'deki okullarda da programlama eğitiminde kullanımı yaygınlaştırılabilir. Çalışmanın sonuçlarının, Alice programının okullarda kullanımı noktasında bu potansiyele sahip olduğunu ortaya koymakla birlikte ileride yapılacak araştırmalara ön veri oluşturduđunu söylemek mümkündür. İkinci olarak, Alice programının üç boyutlu dünyalarda programlama öğretiminin yanı sıra

programlama derslerini daha önce bahsedilen sebeplerden dolayı etkilediği ortaya çıkarılmıştır. Sonuçlar bölümde programlama öğretiminde üç boyutlu dünyaların kullanımı başlığı altında öğretmen adaylarının görüşleri değerlendirilirken bu unsurlar belirtilmiştir. Son olarak, bilgisayar ve öğretim teknolojileri bölümünden mezun olan bilişim öğretmeni adayları atandıktan sonra programlama derslerini yürütecek potansiyele öğretmenlerdir. Bu adayların programlama öğretimine ilişkin öz yeterlilik algılarının tespit edilmesinin mezun olduktan sonraki durumu öngörmek açısından önemli olduğu söylenebilir. Bu araştırma ile öğretmen adaylarının programlama konusundaki öz yeterlilik algılarına bakılmış, mezuniyet durumuna gelen katılımcıların Alice ile programlama eğitimlerini almadan önce öz yeterlilik algılarının düşük olduğu sonucuna ulaşılmıştır. Bu eğitimleri aldıktan sonra ise öz yeterlilik algılarında istatistiksel olarak anlamlı ve olumlu yönde bir farklılık söz konusu olmuştur.

Kaynakça

- Aho, A. V. (2012). Computation and computational thinking. *The Computer Journal*, 55(7), 832-835.
- Akpınar Y. & Altun A. (2014). Bilgi toplumu okullarında programlama eğitimi gereksinimi. *İlköğretim Online*, 13(1), 1-4.
- Altun, A., & Mazman, S. G. (2012). Programlamaya ilişkin öz yeterlilik algısı ölçeğinin Türkçe formunun güvenirlik ve geçerlik çalışması. *Eğitimde ve Psikolojide Ölçme ve Değerlendirme Dergisi*, 3(2), 297-308.
- Al-Tahat, K. (2014). An innovative instructional method for teaching object-oriented modelling. *International Arab Journal of Information Technology (IAJIT)*, 11(6), 540-549.
- Aşkar, P. & Davenport, D. (2009). An investigation of factors related to self-efficacy for java programming among engineering students. *The Turkish Online Journal of Educational Technology*, 8(1), 26-32.
- Bandura, A. (1994). Self-efficacy. In V. S. Ramachaudran (Vol. 4, ss. 71-81), *Encyclopedia of Humanbehavior*. New York: AcademicPress.
- Bandura, A. (1977). Self-efficacy: Toward a unifying theory of behavioral change. *Psychological Review*, 84(2), 191-215.
- Barr, D., Harrison, J., & Conery, L. (2011). Computational thinking: A dijital age skill for everyone. *Learning & Leading with Technology*, 38(6), 20-23.
- Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community?. *Acm Inroads*, 2(1), 48-54.

- Baykul, Y. (1999). *İstatistik: Metodlar ve uygulamalar*. Ankara: Anı Yayıncılık.
- Bishop-Clark, C., Courte, J., Evans, D., & Howard, E. (2007). A quantitative and qualitative investigation of using Alice programming to improve confidence, enjoyment, and achievement among non-majors. *Journal of Educational Computing Research*, 37(1), 193- 207.
- Biju, S. M. (2013). Taking advantage of Alice to teach programming concepts. *E-Learning and Digital Media*, 10(1), 22-29.
- Blackwell, A. (2002). What is programming. In J. Kullis, L. Baldwin, & R. Scoble (Eds.), *14th Workshop of the Psychology of Programming Interest Group* (pp. 204-218). London: Brunel University.
- Busch, T. (1995). Gender differences in self-efficacy and attitudes toward computers. *Journal of Educational Computing Research*, 12(2), 147-158.
- Chang, C. (2014). Effects of using Alice and Scratch in an introductory programming course for corrective instruction. *Journal Of Educational Computing Research*, 51(2), 185-204. doi:10.2190/EC.51.2.c
- Choi, H., & Johnson, S. D. (2005). The effect of context-based video instruction on learning and motivation in online courses. *The American Journal of Distance Education*, 19(4), 215-227.
- Choi, H. J., & Johnson, S. D. (2007). The effect of problem-based video instruction on learner satisfaction, comprehension and retention in college courses. *British Journal of Educational Technology*, 38(5), 885-895.

- Clements, D. H., & Gullo, D. F. (1984). Effects of computer programming on young children's cognition. *Journal of Educational Psychology*, 76(6), 1051-1058.
- Cooper, S., Dann, W., & Pausch, R. (2000a). Alice: a 3-d tool for introductory programming concepts. *Journal of Computing Sciences in Colleges*, 15(5), 107-116.
- Cooper, S., Dann, W., & Pausch, R. (2000b). Developing algorithmic thinking with Alice. In D. Colton, J. Caouette, & B. Raggad (Eds.), *Information Systems Educators Conference* (pp. 506-539). Philadelphia, Pennsylvania: AITP Education Foundation.
- Cooper, S., Dann, W., & Pausch, R. (2003a, February). Teaching objects-first in introductory computer science. *In ACM SIGCSE Bulletin*, 35(1), 191-195.
- Cooper, S., Dann, W., & Pausch, R. (2003b). Using animated 3d graphics to prepare novices for CS1. *Computer Science Education*, 13(1), 3-30.
- Cooper, S. (2010). The design of Alice. *ACM Transactions on Computing Education*, 10(4), 1-16.
- Creswell J. W., Plano Clark, V.L. (2007) *Designing and conducting mixed methods research*. Thousand Oaks, California: Sage.
- Creswell, J. (2009). *Research design: Qualitative, quantitative, and mixed method approaches* (3rd ed.). Thousand Oaks, California: Sage Publications.
- Creswell, J. (2012). *Educational research: Planning, conducting, and evaluating quantitative and qualitative research* (4th ed.). Boston: Pearson.
- Creswell, J. W. (2013). *Nitel araştırma yöntemleri: Beş yaklaşıma göre nitel araştırma ve araştırma deseni* (3. Baskı). Ankara: Siyasal Kitabevi.

- Çatlak, Ş., Tekdal, M., & Baz, F. Ç. (2015). Scratch yazılımı ile programlama öğretiminin durumu: Bir doküman inceleme çalışması. *Journal of Instructional Technologies & Teacher Education*, 4(3).
- Çepni, S. (2014). *Arastirma ve proje çalışmalarına giriş*. Trabzon: Celepler Matbaacılık.
- Dalton, D.W. & Goodrum, D.A. (1991). The effects of computer programming on problem-solving skills and attitudes. *Journal of Educational Computing Research*, 7(4), 483-506.
- Dann, W., Cooper, S., & Pausch, R. (2002). Using 3-d interactive animation to provide program visualization as a gentle introduction to programming in preparing students for CS1. In M. Ben-Ari (Ed.), *Proceedings of the Second Program Visualization Workshop* (pp. 90-92). Aarhus, Denmark: University of Aarhus.
- Dann, W., Cosgrove, D., Slater, D., Culyba, D., & Cooper, S. (2012). Mediated transfer: Alice 3 to java. In R. McCauley (Ed.), *Proceedings of the 43rd ACM technical symposium on Computer Science Education* (pp. 141-146). New York: ACM.
- Denning, P. J. (2003). Great principles of computing. *Communications of the ACM*, 46(11), 15-20.
- Denton, D. W. (2012). Enhancing instruction through constructivism, cooperative learning, and cloud computing. *TechTrends*, 56(4), 34-41.
- Elrad, T., Filman, R. E., & Bader, A. (2001). Aspect-oriented programming: Introduction. *Communications of the ACM*, 44(10), 29-32.
- Eurydice (2011). *Key data on learning and innovation through ICT at school in Europe*. Brussels: Education, Audiovisual and Culture Executive Agency.
- Fincher, S., & Utting, I. (2010). Machines for thinking. *ACM Transactions on Computing Education (TOCE)*, 10(4), 1-7.

- Flottemesch, K. (2013). Learning through narratives: The impact of digital storytelling on intergenerational relationships. *Academy of Educational Leadership Journal*, 17(3), 53-60.
- Fraenkel, J., & Wallen, N. (2005). *How to design and evaluate research in education* (6th ed.). New York: McGraw-Hill.
- Futschek, G. (2006). Algorithmic thinking: The key for understanding computer science. In V. Dagiene (Ed.), *Informatics education—the bridge between using and understanding computers* (pp. 159-168). Vilnius, Lithuania: Springer-Verlag Berlin Heidelberg.
- Gagne, R. M. (2013). *Instructional technology: Foundations*. New York: Routledge.
- Gibson, J. P. (2012). Teaching graph algorithms to children of all ages. In *Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education* (pp. 34-39). New York: ACM.
- Grover, S., & Pea, R. (2013). Computational thinking in k–12 a review of the state of the field. *Educational Researcher*, 42(1), 38-43.
- Guzdial, M. (2008). Education paving the way for computational thinking. *Communications of the ACM*, 51(8), 25-27.
- Herbert, C. (2010). *An introduction to programming using Alice 2.2* (2nd ed.). Boston, Massachusetts: Course Technology.
- Howard, E.V., Evans, D., Courte, J. & Bishop-Clark, C. (2009). A qualitative look at Alice and pair-programming. *Information Systems Education Journal*, 7(80), 3-10.
- Hromkovič, J. (2006). Contributing to general education by teaching informatics. In R. T. Mittermeir (Ed.), *Informatics Education - The Bridge between Using and Understanding Computers* (pp.25-37). Berlin: Springer.

- ISTE (2015). *CT leadership toolkit*. Retrieved December 10, 2015, from <http://www.iste.org/docs/ctdocuments/ct-leadershipt-toolkit.pdf?sfvrsn=4>.
- Jegede, P.O. (2009). Predictors of java programming self efficacy among engineering students in a Nigerian university. *International Journal of Computer Science and Information Security*, 4(1&2).
- Johnsgard, K., & McDonald, J. (2008). Using alice in overview courses to improve success rates in programming. In H. Saiedian, & L. Williams (Eds.), *Twenty-first Conference on Software Engineering Education and Training* (pp. 129-136). New York: ACM.
- Johnson, D. W., & Johnson, R. (1989). *Cooperation and competition: Theory and research*. Edina, Minnesota: Interaction Book Company.
- Johnson, R. B., Onwuegbuzie, A. J., & Turner, L. A. (2007). Toward a definition of mixed methods research. *Journal of mixed methods research*, 1(2), 112-133.
- Karadeniz, Ş., Samur, Y., & M. Y. Özden (2014). Playing with algorithms to learn programming: A case study on 5 years old children. In D. Tien (Ed.), *The 9th International Conference on Information Technology and Applications* (pp. 1-3). Sydney, Australia: ICITA.
- Kelleher, C., & Pausch, R. (2007). Using storytelling to motivate programming. *Communications of the ACM*, 50(7), 58-64.
- Kirk, C. P., Chiagouris, L., Lala, V., & Thomas, J. D. (2015). How do digital natives and digital immigrants respond differently to interactivity online?. *Journal of Advertising Research*, 55(1), 81-94.
- Korkmaz, Ö., Çakır, R., Özden, M. Y., Oluk, A., & Sarıoğlu, S. (2015). Bireylerin bilgisayarca düşünme becerilerinin farklı değişkenler açısından incelenmesi. *Ondokuz Mayıs Üniversitesi Eğitim Fakültesi Dergisi*, 34(2), 68-87.

- Kozlowski, S. W., & Ilgen, D. R. (2007). The science of team success. *Scientific American Mind*, 18(3), 54-61.
- Kruck, S. E., & Teer, Faye P. (2009). Interdisciplinary student teams projects: A case study. *Journal of Information Systems Education*, 20(3), 325-330.
- Kutluca, T., & Zengin, Y. (2011). Matematik öğretiminde GeoGebra kullanımını hakkında öğrenci görüşlerinin değerlendirilmesi. *Dicle Üniversitesi Ziya Gökalp Eğitim Fakültesi Dergisi*, 17, 160-172.
- Lombardi, M. M. (2007). Authentic learning for the 21st century: An overview. *Educause Learning Initiative*, 1, 1-12.
- Lee, T. Y., Louis, M., Ahn, J., & Bederson, B. B. (2012). CTArcade: Computational thinking with games in school age children. *International Journal of Child-Computer Interaction*, 2(1), 1-10.
- Malliarakis, C., Satratzemi, M., & Xinogalos, S. (2014). Designing educational games for computer programming: A holistic framework. *Electronic Journal of e-Learning*, 12(3), 281-298.
- Mayer, R. E., & Moreno, R. (2002). Animation as an aid to multimedia learning. *Educational psychology review*, 14(1), 87-99.
- McCoy, L. P., & Dodl, N. R. (1989). Computer programming experience and mathematical problem solving. *Journal of Research on Computing in Education*, 22(1), 14-25.
- Milli Eğitim Bakanlığı (2013). *2012 faaliyet raporu*.
http://sgb.meb.gov.tr/meb_iys_dosyalar/2013_07/31123903_meb_faaliyet_raporu_2012.pdf'den alınmıştır. Erişim Tarihi:15.10.2015

Milli Eğitim Bakanlığı (2012). *Bilişim teknolojileri ve yazılım dersi öğretim programı*.

<http://ttkb.meb.gov.tr/program2.aspx/program2.aspx?islem=2&kno=196>'den alınmıştır. Erişim Tarihi: 15.10.2015.

Milli Eğitim Bakanlığı (2012). *Fatih projesi çalıştay raporu* (2012).

<http://fatih.inetd.org.tr/Calistay/Fatih-calistay-rapor.pdf>.'den alınmıştır. Erişim Tarihi: 15.10.2015

Mohamed, M.A. (1985). *The effects of learning logo computer language upon the higher cognitive processes and the analytic/global cognitive styles of elementary school students (man-machine interaction)* (Doctoral dissertation). Available from ACM Digital Library. (UMI No. AAI8600638)

Moskal, B., Lurie, D., & Cooper, S. (2004). Evaluating the effectiveness of a new instructional approach. *SIGCSE Bulletin*, 36(1), 75-79.

Mullins, P., Whitfield, D., & Conlon, M. (2009). Using Alice 2.0 as a first language. *Journal of Computing Sciences in Colleges*, 24(3), 136-143.

Naughton, J. (2012). Why all our kids should be taught how to code. *Observer New Review*, 31(3), 12-13.

Owolabi, J. & Adegoke, B. A. (2014). Multilevel analysis of factors predicting self efficacy in computer programming. *International Journal on Integrating Technology in Education*, 3(2), 19-29.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Cambridge: Basic Books.

Powers, K., Ecott, S., & Hirshfield, L. M. (2007). Through the looking glass: teaching CS0 with Alice. *SIGCSE Bulletin*, 39(1), 213-217.

- Prensky, M. (2001). Digital natives, digital immigrants part 1. *On the horizon*, 9(5), 1-6.
- Ramalingam V. & Wiedenbeck S. (1998). Development and validation of scores on a computer programming self efficacy scale and group analyses of novice programmer self-efficacy. *Journal of Educational Computing Research*, 19(4), 365-379.
- Rodger, S. H. (2002). Introducing computer science through animation and virtual worlds. *SIGCSE Bulletin*, 34(1), 186-190.
- Roseth, C. J., Johnson, D. W., & Johnson, R. T. (2008). Promoting early adolescents' achievement and peer relationships: The effects of cooperative, competitive, and individualistic goal structures. *Psychological bulletin*, 134(2), 223.
- Ross, M. R. (2004). *Introduction to probability and statistics for engineers and scientists*. San Diego: Elsevier Academic Press.
- Rotter, J. B. (1990). Internal versus external control of reinforcement: A case history of a variable. *American psychologist*, 45(4), 489-493.
- Senemođlu, N. (2012). *Geliřim öğrenme ve öğretim: Kuramdan uygulamaya*. Ankara: Pegem Yayıncılık.
- Sezginsoy, B., & Akkoyunlu, B. (2011). Sosyal bilgiler dersinde tarih bilinci oluřturmada dizgeli öğretim etkinliđi. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 41(1), 411-422.
- Sim, J., & Wright, C. C. (2005). The kappa statistic in reliability studies: use, interpretation, and sample size requirements. *Physical Therapy*, 85(3), 257-268.
- Sykes, E. (2007). Determining the effectiveness of the 3D Alice programming environment at the computer science I level. *Journal of Educational Computing Research*, 36, 223-244.

- Stasko, J. T. (1990). Tango: A framework and system for algorithm animation. *Computer*, 23(9), 27-39.
- Suryana, N., & Husin, B. (2014). Effectiveness of integrated algorithm-program visualization: A case study with the 3De-AIProV. *Advanced Science Letters*, 20(1), 304-308.
- Trilling, B., & Fadel, C. (2009). *21st century skills: Learning for life in our times*. San Francisco: John Wiley & Sons.
- Yildirim, A., & Simsek, H. (2013). *Sosyal bilimlerde nitel arastirma yontemleri*. Ankara: Seçkin Yayıncılık.
- Wang, T. C., Mei, W. H., Lin, S. L., Chiu, S. K., & Lin, J. M. C. (2009). Teaching programming concepts to high school students with alice. In J. Froyd (Ed.) *Proceedings of the 39th IEEE international conference on Frontiers in education conference*. (pp. 955-960) New Jersey: IEEE Press Piscataway.
- Werner, L., Denner, J., Campe, S., & Kawamoto, D. C. (2012). The fairy performance assessment: measuring computational thinking in middle school. In R. McCauley (Ed.), *Proceedings of the 43rd ACM technical symposium on Computer Science Education* (pp. 215-220). New York: ACM.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society of London : A Mathematical, Physical and Engineering Sciences*, 366(1881), 3717-3725.
- Zengin, Y., Kağızmanlı, T. B., Tatar, E., & İşleyen, T. (2013). Bilgisayar destekli matematik öğretimi dersinde dinamik matematik yazılımının kullanımı. *Mustafa Kemal Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 10(23), 167-180.

Ekler

Ek 1: Görüşmeler ve Açık Uçlu Anket Uygulanırken Kullanılan Form

PROGRAMLAMA ÖĞRETİMİNDE ÜÇ BOYUTLU DÜNYALARIN KULLANIMINA İLİŞKİN GÖRÜŞME FORMU

Tarih: ___/___/___

Saat(Başlangıç/Bitiş): _____/_____

GİRİŞ

Merhaba, ismim Emrah Kayabaşı, Uludağ Üniversitesi Eğitim Fakültesi Bilgisayar ve Öğretim Teknolojileri Anabilim Dalında hem yüksek lisans öğrencisiyim hem de aynı anabilim dalında araştırma görevlisiyim. Bölümümüzde verilmekte olan Üç Boyutlu Programlama dersindeki deneyimleriniz ile ilgili bir araştırma yapıyorum. Bu ders Bilgisayar ve öğretim teknolojileri alanında yetiştirilen öğretmenlerin, programlama konularında nitelikli olarak yetiştirilmesi amacıyla müfredatta yer almaktadır. Üç Boyutlu programlama derslerinde yaptığınız etkinliklerin etkililiği ve verimliliğini ortaya çıkarmayı ve bu konuda öneriler ortaya koymayı amaçlayan bu çalışmanın, hem bu okul hem de diğer özel okullar açısından yararlı olacağını ümit ediyorum. Bu araştırma kapsamında siz öğretmen adayları ile görüşmeler yapıyorum.

Görüşmemize geçmeden önce, yaptığım tüm görüşmelerde verilen bilgiler sadece bu araştırmada kullanılacak ve kişisel bilgiler kesinlikle gizli tutulacaktır. Görüşmenin yarım saat süreceğini tahmin ediyorum ve izin verirseniz görüşmeyi kaydetmek istiyorum. Bu şekilde hem zamanı daha etkili kullanabiliriz hem de sorulara vereceğiniz yanıtların kaydını daha ayrıntılı tutma fırsatı elde edebilirim.

Bu araştırmaya katılmayı kabul ettiğiniz için şimdiden teşekkür ederim. Eğer sizin bana görüşmeye başlamadan önce sormak istediğiniz bir soru varsa, önce bunu yanıtlamak istiyorum.

SORULAR

1. Bu derste elde ettiğiniz programlama ve 3B dünya tasarımı becerilerinin gelecekte verebileceğiniz derslere nasıl etki edeceğini düşünüyorsunuz?
 - a. Öğrencilerinizin performanslarını (ders başarısı ve derse katılım) nasıl etkileneceğini düşünüyorsunuz?

- b. Bu beceriler, Programlama kavramlarının akılda kalmasını sağlar mı?
 - c. 3B dünya tasarımı becerileriniz derslerine ne yönde etki eder?
2. Ortaöğretimde programlama öğretimi yapılırken sizce Alice nasıl kullanılabilir? Ortaöğretim öğrencilerinin bu programla öğretimi nasıl karşılayacağını düşünüyorsunuz?
 - a. Öğrencilerinizin programlama dersine olan tutumları nasıl etkilenir?
 - b. Sağladığı görsellik (visualization) imkânları ile programlama öğretiminde ne gibi faydalar sağlayabilir?
3. Alice yazılımını kullanırken karşılaştığınız güçlükler nelerdir?
 - a. Alice ara yüzünde karşılaştığınız güçlükleri sıralayabilir misiniz?
 - b. Alice yazılımındaki araçlara ilişkin görüşleriniz nelerdir?
4. Alice yazılımı sizce programlama öğretiminde ne gibi kolaylıklar sağlamaktadır?
 - a. Görsellik ile ilgili düşüncelerinizi söyleyebilir misiniz?
 - b. Sürükle bırak yapısı ile hazırlanan Alice yazılımı, bu özelliğiyle işlerinizi kolaylaştırdı mı? Nasıl?
 - c. Programlama kavramlarını öğrenmenize bir etkisi oldu mu? Ne şekilde?
 - d. Grup çalışmasına elverişli bir program olduğunu düşünüyor musunuz? Neden?
5. Alice yazılımı öğretmen adaylarının özellikle materyal geliştirme ve kullanma konusunda nitelikli olarak yetişmelerine ne gibi faydalar sağlayabilir?
 - a. Matematik, fen gibi alanlardan yetişen öğretmenlerin Alice yazılımını ders materyallerini geliştirirken kullanabileceğini düşünüyor musunuz? Neden?
 - b. Bilgisayar ve öğretim teknolojileri mezunu olan öğretmen adayları programlama eğitimlerinin başlangıcında Alice ile almış olsalar veya ileride programlama derslerini Alice ile verseler ne gibi faydalar sağlardı?
 - c. Bu ders kapsamında edindiğiniz programlama ve 3B dünya becerilerinin kalıcılığı konusunda düşünceleriniz nelerdir? Neden kalıcı olacağını/olmayacağını düşünüyorsunuz?

Ek 2: Programlamaya İlişkin Öz Yeterlilik Anketi

Programlamaya İlişkin Öz Yeterlilik Ölçeği

Merhaba, bu anket sizin programlama görevlerinde kendinize ilişkin öz yeterlilik algınızı belirlemek üzere hazırlanmıştır. Programlama dili olarak, kullandığınız tüm programlama dillerini (C, C++, Visual Basic, Java vb.) ya da spesifik herhangi birini esas alabilirsiniz. Araştırmaya olan katkınızdan dolayı şimdiden teşekkür ederiz.

Cinsiyetiniz: Erkek Kadın

Yaşınız: 18-25 26-35 35-41 41+

Sınıf: 1. 2. 3. 4. Lisansüstü

Bölüm:

Kaç yıldır program yazıyorsunuz?

Bugüne kadar programlamaya ilişkin ya da programlama içeren kaç ders aldınız?

Hangi programlama dillerini kullanabiliyorsunuz?

Aşağıdaki programlamaya ilişkin verilen görevleri yaparken kendinize olan güveninizi 1 ile 7 arasında derecelendirerek belirtiniz.

"1= Hiç Güvenmiyorum", "2=Genellikle Güvenmiyorum", "3= Biraz Güveniyorum", "4= %50/50"
"5= Oldukça Güveniyorum", "6=Genellikle Güveniyorum", "7=Tamamen Güveniyorum"

	Hiç güvenmiyorum	Genellikle güvenmiyorum	Biraz güveniyorum	50/50	Oldukça güveniyorum	Genellikle güveniyorum	Tamamen güveniyorum
	1	2	3	4	5	6	7
1. "Merhaba Dünya" mesajının görüntülenebileceği bir program yazabilirim..							
2. Üç sayının ortalamasını hesaplayan bir program yazabilirim.							
3. Verilen herhangi bir sayı dizisinin ortalamasını hesaplayan bir program yazabilirim..							
4. İstenilenler açıkça tanımlandığında bir problemin çözümüne yönelik oldukça karmaşık ve uzun bir program yazabilirim.							
5. Yazacağım bir programı modüler bir biçimde organize edip tasarlayabilirim							
6. Yazdığım uzun ve karmaşık bir programdaki tüm hataları ayıklayabilir ve çalışabilir hale getirebilirim.							
7. Uzun, karmaşık ve birden fazla dosya gerektiren bir programı kavrayabilirim.							
8. Bir programın daha okunabilir ve açık olması için uzun ve karmaşık kısımları yeniden yazabilirim.							
9. Çevrede bir sürü dikkat dağıtıcı olsa bile programa odaklanma yollarını bulabilirim.							

Ek 3: Kod Tablosu

Bu materyal İçerikAnalizi.xlsx dosyasındaki düşünce birimleri / cümlelerin kodlanması işlemini gerçekleştirmek amacıyla hazırlanmıştır.

1. Kodlamalar yapılırken, her bir cümle aşağıda verilen temalardan yalnızca birine uyacak şekilde düzenlenmiştir. Excel dosyasındaki cümlelerin bulunduğu satırdaki ilgili olan tema hangisiyse “1” rakamı yazılmalıdır.
2. Tema “başlık”, Tanım “ilgili temanın açıklamasını/ne anlama geldiğini” ifade etmektedir. Örneklerle de daha iyi anlamanız amaçlanmıştır.

Alice yazılımı üç boyutlu bir programlama yazılımıdır. Özellikleri incelendiğinde 3B animasyonlar, hikâyeler, oyunlar oluşturmak için kullanılabilir.

1.SORU İÇİN TEMALAR

Tema	Tanım	Örnek
3B Dünya Tasarımı Deneyimi	Alice kullanıcılarına üç boyutlu bir platform (3B dünya, 3B nesnelere) sunmaktadır. Nesnelere uyumlu bir şekilde oluşturulan dünyada yerleştirilmesi, boyutlarının ayarlanması, renklendirmeler gibi tasarım becerileri olarak tanımlanabilir.	Farklı bir boyutta da çalışarak gerçekliğe daha yakın materyaller geliştirebileceğimi düşünüyorum.
Derse İlgi ve Katılım	Kullanıcıların derse ilişkin motivasyonlarını ve aktif katılımcı olup/olmadıklarına ilişkin görüşler bu başlık altında listelenebilir.	Alice sayesinde derslerde daha çok eğleneceğimden daha aktif olacağımı ve derslere katılacağımı düşünüyorum.
Kalıcılık	Öğrenilen bilgilerin veya edinilen becerilerin gelecekte de sürdürülebilmesi durumudur.	Öğrendiklerimin, görsellikten dolayı aklımda kalacağını düşünüyorum.
Ders Başarısı	Bir öğrencinin dersten aldığı akademik not olarak düşünülebilir. Performans veya sınav sonuçlarının değerlendirilmesi sonucunda belirlenir.	A, B, C nedeninden dolayı ders başarılarının artacağını düşünüyorum.
Diğer Görüşler	Yukarıdaki tanımlamalara uymayan tüm görüşler diğer görüşler olarak kodlanmalıdır.	

2.SORU İÇİN TEMALAR

Tema	Tanım	Örnek
Kullanım Biçimi	Alice yazılımının ders esnasında ne şekilde, ne sırada, hangi amaçla kullanılacağını tanımlamak için oluşturulmuş bir temadır.	Programlama öğretirken Alice’i uygulama yaparken kullanırım.
Kalıcılık	Öğrenilen bilgilerin veya edinilen becerilerin gelecekte de sürdürülebilmesi durumudur.	Öğrencilerin öğrendikleri 3B dünyanın avantajlarından dolayı akılda kalır.
Görsellik	Üç boyutlu nesnelerin, renklendirmelerin, konuşma baloncukları gibi alanlara eklenen yazılar gibi görsel öğelerin bulunması durumudur.	Sağladığı görsellik imkânları ile öğrencilerin ilgisini çekebileceğini düşünüyorum.
Öğrenci Gelişimi	Bilgi ve becerilerin artırılması amacıyla kullanılmıştır.	Öğrencilerin yaratıcılıklarının ve hayal dünyalarının gelişeceğini düşünüyorum.
Öğrencilerin Derse Tutumu	Tutum bireylerin bir duruma karşı takındığı tavır demektir. Öğrencilerin Alice ile yapılan programlama eğitimine nasıl bir tutum içerisinde bulunacaklarına ilişkin görüşler bu tema altında toplanmıştır.	Dersler daha eğlenceli geçeceğinden olumlu bir tutum içerisinde (eğleneceklerini, seveceklerini, beğeneceklerini, zevk alacaklarını) bulunacaklarını düşünüyorum.
Diğer Görüşler	Yukarıdaki tanımlamalara uymayan tüm görüşler diğer görüşler olarak kodlanmalıdır.	

3.SORU İÇİN TEMALAR

Tema	Tanım	Örnek
Nesne Konumlandırma	Nesnelerin üç boyutlu ortamda yerleştirilmesi ve boyutlarının ayarlanması durumunu ifade eder.	Nesneleri 3B dünyaya eklerken/yerleştirirken sorunlar yaşadım.
Dil	Türkçe Dil desteğinin olmaması durumu için “Dil” teması kullanılmıştır.	İngilizce dilinden dolayı zorluk yaşadım.
Performans Sorunları	Alice yazılımından veya bilgisayarın donanım özelliklerinin yetersizliğinden kaynaklanan donmalar, hatalar vs. bu temanın altında yer alır.	Bilgisayarım dondu. Zaman zaman Alice’e çok nesne eklediğimde Alice dondu.
Nesne Galerisinin zenginleştirilmesi	Daha fazla üç boyutlu nesnenin Alice ile kullanıcılara sunulması durumunu ifade eder.	Daha fazla nesne olması gerektiğini düşünüyorum. Aradıklarımı tam olarak bulamadım.
Kamera Kullanımı	Kamera kullanımına ilişkin kaynaklanan güçlükleri tanımlamak için kullanılır.	Sahne geçişleri esnasında sorunlar yaşadım. Kamera açısını ayarladığımda bazı nesnelerin kaybolduğunu gördüm.
Metotlar	Metot bir programın içerisinde yer alan kod/kodları ifade eder, Olaylar ise klavyeye basıldığında zıplama işlemi gerçekleştirmek gibi daha karmaşık işlemler için kullanılır.	Olayların ve Metotların artırılması gerektiğini düşünüyorum.
Diğer Görüşler	Yukarıdaki tanımlamalara uymayan tüm görüşler diğer görüşler olarak kodlanmalıdır.	

4.SORU İÇİN TEMALAR

Tema	Tanım	Örnek
Görsellik	Üç boyutlu nesnelere, renklendirmeler, 3B dünya, animasyonlar, konuşma baloncuklarına eklenen yazılar gibi görsel öğelerin bulunması durumudur.	Görselliğin dersleri olumlu yönde etkileyeceğini düşünüyorum. Özellikle 3B sayesinde öğrenciler programlama kavramlarını daha anlamlı şekilde öğrenebilirler.
Sürükle-Bırak Yapısı	Alice’de nesnelere, yazılan kodlar sürükle-bırak mantığına dayanmaktadır. Yani 3B bir nesne zıplama (move up) komutunun sürüklenip kod düzenleme alanına bırakılması yeterlidir.	Sürükle-bırak yapısı hem hataları önler, hem de algoritma mantığını öğretmede etkilidir.
Programlama Kavramları	Metot, fonksiyon, algoritma gibi programlama kavramlarının öğretiminde Alice’in kullanımına ilişkin görüşler bu tema altında toplanmıştır.	Programlama kavramlarını öğrenirken özellikle algoritma mantığının kazanılmasında Alice’in etkili bir araç olduğunu düşünüyorum.
Hazır Kodların Yazım Hatalarını Önlemesi	Hazır kodlardan kastedilen Visual Basic gibi programlarda satır satır metinsel olarak yazılan kodların nesnelere öntanımlı olarak eklenmiş olmasıdır. Yazım hataları da noktalı virgül, tırnak, parantez gibi ifadelerden Alice’in bağımsız olmasıdır.	Visual Basic’te olduğu gibi bir noktalı virgülden unutulunca tüm programın çalışmaması gibi bir durum Alice’de yok. Hazır kodlar bu haliyle daha kullanışlı olmasını sağlıyor.
Grup/Bireysel Çalışma	Alice yazılımında grup çalışması veya bireysel çalışmaya ilişkin tüm görüşler bu tema altında toplanmıştır.	Grup çalışmasına elverişli bir program olduğunu düşünüyorum çünkü... Bireysel çalışmaya elverişli bir program olduğunu düşünüyorum çünkü...

5.SORU İÇİN TEMALAR

Tema	Tanım	Örnek
Alice ile Programlama Eğitimi	“Programlama Eğitimleri Alice ile alınmış/verilmiş olsaydı neler değişirdi?” veya “Alice ile programlama eğitiminin getirileri ve götürüleri nelerdir?” gibi soruların cevapları bu tema altında toplanmıştır.	Alice ile programlama eğitimi yapılırken, 3B dünya sayesinde programlama kavramlarını daha iyi öğretebileceğimi düşünüyorum.
Diğer Branşların Durumu	Müzik, Resim, Matematik gibi BÖTE dışındaki alanlarda, hem öğretmen adayları hem de var olan öğretmenlerin Alice’i kullanımına ilişkin görüşler bu temanın altında toplanmıştır.	Matematik, fen gibi alanlarda mezun olan öğretmenlerin Alice’i kullanamayacağını düşünüyorum çünkü...
Kalıcılık	Öğrenilen bilgilerin veya edinilen becerilerin gelecekte de sürdürülebilmesi durumudur.	Uygulamalı olarak derslere hazırlandığım için öğrendiklerimin kalıcı olacağını düşünüyorum.

Ek 4: İzin Yazısı

T.C.
EĞİTİM FAKÜLTESİ DEKANLIĞI
Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölüm Başkanlığı



Sayı: 79377886-900/1644

27/04/2015

Konu: Emrah KAYABAŞI'nın Anket Uygulaması

EĞİTİM FAKÜLTESİ DEKANLIĞINA

İlgi : 24.04.2015 tarihli ve 90661511-900/1632 sayılı yazınız.

Eğitim Bilimleri Enstitüsü Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı Yüksek Lisans öğrencisi Emrah KAYABAŞI, "3B Ortamda Programlama Öğretimi Hakkında Öğretmen Adayı Görüşlerinin Değerlendirilmesi" konulu tezine veri toplamak amacıyla Bölümümüz öğrencilerine anket uygulama isteği uygundur.

Gereğini arz ederim.

Doç. Dr. Adem UZUN
Bölüm Başkanı

U.Ü. Eğitim Fakültesi Görükle Kampusu 16059 Nilüfer/BURSA

Tel :0224 29 42249

Faks: 0224 294 21 99

e-posta :yildizsenay@uludag.edu.tr

Elektronik Ağ: <http://egitim.uludag.edu.tr>

Ayrıntılı Bilgi

Şenay Y., Memur

Bu belge UDOS ile hazırlanmıştır. Teyit için: https://udos.uludag.edu.tr/teyit/?Uzmqbze_nUSgGEHywrKbIQ

Ek 5: Etkinlik Ödevi 1-2-3-4-5

Ek Görevler

Görev 3: Nesne Düzenleme Araçlarını (Default, Rotation, Translation, Resize) kullanarak ne işe yaradıklarını not ediniz.

Görev 4: Geri al/Yinele (Undo/Redo) tuşlarını kullanarak ne işe yaradığını not ediniz.

Görev 5: Galerî bölümünü kullanarak sahnenize iki yeni nesne ekleyiniz.

Görev 6: Sahnenize eklediğiniz nesnelere birini nesne ağacından seçerek, Özellikler (Properties) bölümünde yer alan en az 3 özelliği kullanıp değişikliklerini not ediniz.

Görev 7: File, Edit, Project, Run, Window menülerini inceleyiniz ve ne tarzda işlemler yapıldığını not ediniz.

Görev 8: Nesne ve kodları silmek için neler yapılması gerektiğini not ediniz.

8

Araş. Gör. Emrah Kayabaşı



Lisans:
Orta Doğu Teknik
Üniversitesi
Bilgisayar ve Öğretim
Teknolojileri Eğitimi
Eylül 2007- Haziran
2012

Yüksek Lisans:
Uludağ Üniversitesi
Bilgisayar ve Öğretim
Teknolojileri Eğitimi
Mart 2013 - ...

Uludağ Üniversitesi

Eğitim Fakültesi, Bilgisayar ve Öğretim
Teknolojileri Eğitimi Bölümü, BURSA



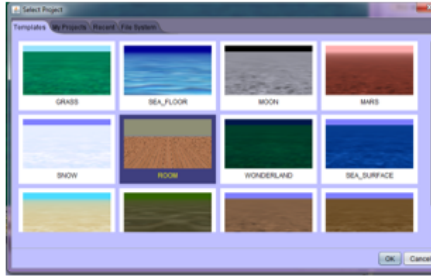
9

Alice 3,1'e Giriş

Etkinlik 1:

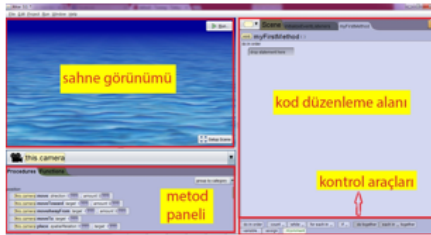
- Alice 3,1 Ara yüzünün Tanıtılması ve İncelenmesi
- Örnek Bir Alice Dünyasının Yüklenmesi ve Oynatılması
- Kendi Dünyanızın oluşturulması

1



Tema Seçimi

Alice ilk çalıştırıldığında karşınıza gelen ekranda ne tarz bir 3B dünya oluşturmak istiyorsanız ona göre verilen temalardan birini seçersiniz. OK/TAMAM butonuna basıldığında Alice ara yüzüne erişmiş olursunuz. Amacınız doğrultusunda bir tema seçiniz.



Alice Ara yüzü

Alice ara yüzünde beş temel bölüm söz konusudur. Bu bölümleri tek tek inceleyiniz. Kodları düzenleme alanı, 3B dünyanın yer aldığı alan, metotlar (fonksiyon + prosedür) yer aldığı alan, kontrol araçları ve son olarak nesne ağacına erişebileceğiniz bir kısım yer alır.

2



Kod Düzenleme Alanı

Scene: Kendi fonksiyonlarınızı, prosedürlerinizi gibi tanımlamaları buradan gerçekleştirirsiniz.

Initialize Event Listeners: Tanımlanmış olaylar (events) burada yer alır. Bir olay eklemek istediğinizde bu alan kullanılır.

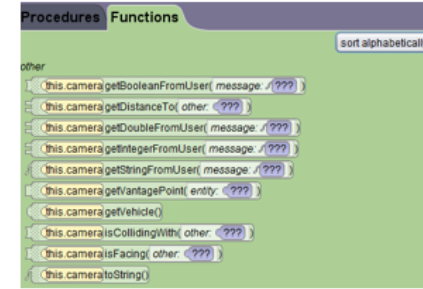
MyFirstMethod: Kod bloklarını oluşturduğunuz/düzenlediğiniz alandır.



Metotlar: Prosedürler

Prosedürler bölümünde yer alan kodlardan birkaç tanesini kod düzenleme alanına basılı sürükleyiniz. En az bir tane konum (position) ve oriyantasyon (orientation) kodunu kullandığınızdan emin olunuz.

3



Metotlar: Fonksiyonlar

Prosedürlerden farklı olarak genellikle bir sonuç döndürürler ve genelde etkileşim/hesaplama gereken durumlarda başvurulurlar.

Sizde eklediğiniz prosedürlere uygun fonksiyonlar var ise kullanınız ve sonuçlarını not ediniz. Eğer fonksiyonlardan hiçbirini kullanamıyorsanız farklı prosedürler eklemeyi deneyiniz.

“Sort alphabetically” yazan açılır pencereyi kullanarak alt seçeneklerini deneyerek inceleyiniz.

Don't be shy! Show them how fabulous you are.

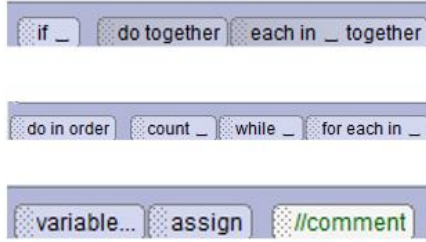
İletişim Bilgileri

Uludağ Üniversitesi
Eğitim Fakültesi
Bilgisayar ve Öğretim Teknolojileri
Eğitimi Bölümü

+90 224 29 42 230
emrahkayabasi@outlook.com

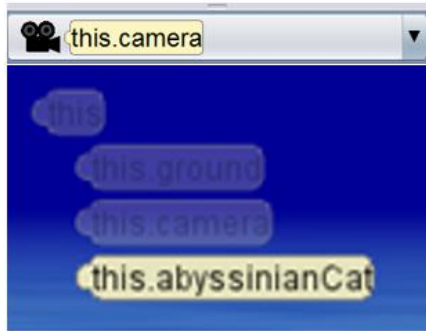
Web Adresi:
emrahkayabasi.net

4



Kontrol Araçları (Tiles)

Yukarıda listelenen kontrol araçları yardımıyla şart/eş zamanlı işlem/ veri girdileri / sayma işlemleri / hazırlanan kodlar için kendinize yorumlar / değişken tanımlamalar / değışkene bir değeri atama gibi işlemleri gerçekleştirebilirsiniz. Do together'ı kullanarak kod bloklarını düzenleyiniz.



Nesne Ağacı

Yukarıda gösterilen bölümler kullanılarak eklediğiniz nesnelere veya 3B dünyanın ait kamera/arka plan gibi nesnelere doğrudan erişim olanağı sağlar.

5



Sahne (Kod Düzenleme Görünümü)

Run: 3B dünyanın ve oluşturduğunuz kod dizilerinin çalıştırılması için kullanılır.

Setup Scene: Alice ara yüzünü değıştiren nadir bölümlerdendir. Ara yüzü Kod Düzenleme (Code Editor) görünümünden Sahne Düzenleme (Scene Editor) görünümüne geçiş için kullanılır.

“Setup Scene” ve yeni ara yüzdeki “Edit Code” butonlarıyla gerçekleşen Ara yüzdeki değışiklikleri gözlemleyiniz.



Sahne Tasarım Ara yüzü

Bu ara yüzde üç temel alan söz konusudur. Bunlar sahne alanı, nesne düzenleme araçları ve galeri bölümüdür.

6



Sahne (Sahne Düzenleme Görünümü)

Edit Code: Sahne Düzenleme ara yüzünden, Kod Düzenleme ara yüzüne geçiş sağlar.

Ok Tuşları: 3B dünyada nesnelere yerini değıştirirken ihtiyaç duyabileceğiniz kamera açılarını ayarlamak için kullanılır.

Starting Camera View: Kamera açılarını önden, yukarıdan vs ayarlamak için kullanılır.

Run: Hazırladığımız projelerin çalıştırılması için kullanılır.

Görev 1: Alice 3.1'de yaptıklarımızı File menüsünden Save seçeneğini kullanarak kaydedin ve programdan tamamıyla çıkın. Alice'i yeniden başlatarak kaydettiğiniz yeri kullanarak bu dosyayı açınız. Bunun için File / Open seçeneğini kullanabilirsiniz.

Görev 2: File / Save As ve File/New seçeneklerini kullanarak ne işe yaradığını not ediniz.

7

Ek Görevler

Görev 2: Sizde eklediğiniz nesnelere herhangi birinin ne kadar bir zamanda zıplamasını kullanıcı insiyatifine bırakmak için, kullanıcıdan veri girişi yöntemi ile ediniz. Örneğin, kullanıcı Ejderha nesnesinin 3 saniyede zıplamasını istiyor ise, açılışta kullanıcıya bir pencere gelsin ve Saniye bilgisi kullanıcıdan alınsın.

Görev 3: Kontrol Araçları bölümünde yer alan Comment'i kullanınız ve ne işe yaradığını not ediniz.

Görev 4: Kontrol Araçları bölümünde yer alan Do Together'ı kullanınız ve ne işe yaradığını not ediniz.

Görev 5: MyFirstMethod isimli metot ile ön tanımlı olarak gelen Do in order ne işe yarar. Açıklayınız.

8

Araş. Gör. Emrah Kayabaşı



Lisans:
Orta Doğu Teknik
Üniversitesi
Bilgisayar ve Öğretim
Teknolojileri Eğitimi
Eylül 2007- Haziran
2012

Yüksek Lisans:
Uludağ Üniversitesi
Bilgisayar ve Öğretim
Teknolojileri Eğitimi
Mart 2013 - ...

İletişim Bilgileri:
E-mail: emrahkayabasi@outlook.com
Telefon: +90 507 134 46 71
İnternet Sitesi: emrahkayabasi.net

Uludağ Üniversitesi

Eğitim Fakültesi, Bilgisayar ve Öğretim
Teknolojileri Eğitimi Bölümü, BURSA



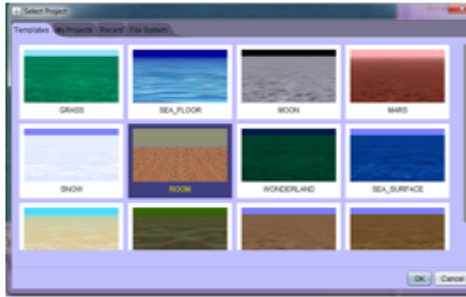
9

Metodlar

Etkinlik 2:

- İlkel (primitive) Metodların Kullanımı
- Kendi Metodunuzun Oluşturulması
- Birden Fazla Parametrelili Metodların Kullanımı

1



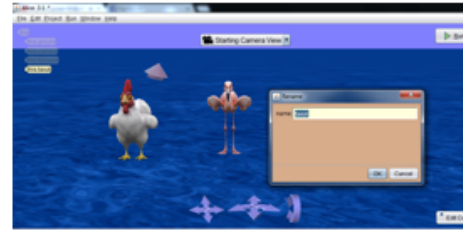
Tema Seçimi

Alice ilk çalıştırıldığında karşınıza gelen ekranda ne tarz bir 3B dünya oluşturmak istiyorsanız ona göre verilen temalardan birini seçersiniz. OK/TAMAM butonuna basıldığında Alice ara yüzüne erişmiş olursunuz. Amacınız doğrultusunda bir tema seçiniz.



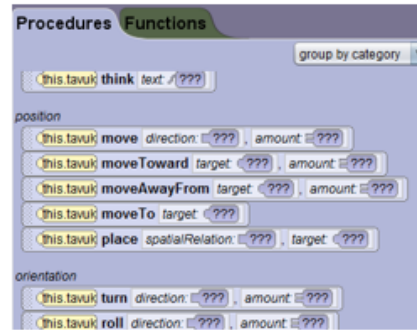
Nesne Ekleme

Sahne düzenleme görünümüne geçiş yaparak, Galerî bölümünden Flyer sınıfının altından Flamemengo ve Chicken nesnelarını sahnenize ekleyiniz. Bu nesneye dilerseniz kodlarınızı oluştururken kolaylık sağlaması açısından bir isim veriniz. Örneğin: tavuk, kuş, at gibi.



Yeniden Adlandırma

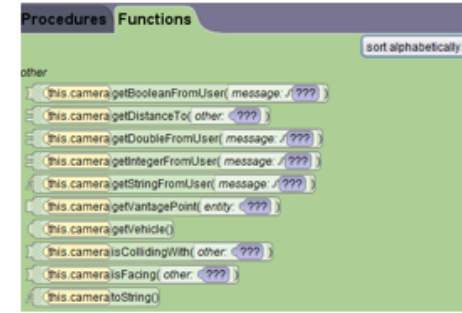
Bu işlemi gerçekleştirmek için nesnenin üstünde sağ tuşa tıklanır ve açılır pencereden RENAME seçilir. Daha sonra yeni isim girilerek OK butonuna basılır.



İlkel (primitive) metotların incelenmesi

Kod Düzenleme Görünümüne geçiş yaparak, ön tanımlı metotları (prosedür ve fonksiyonları) yukarıda gösterilen bölümden inceleyiniz. Burada değişik kategorilerde yer alan metotlardan en az 3 tanesini kullanarak ne işe yaradıklarını not ediniz.

Fark ettiğiniz gibi her metodun yanında "Edit" yazmamaktadır. Yani düzenlemeye izin vermemektedir. Bu tarz Alice ile hazır gelen metotlara ilkel/ön tanımlı metotlar denilir.



Kayıp Nesneleri Bulma

Kameranın o anda göstermediği ama 3B dünyanın içerisinde yer alan bazı nesneleri bulmada güçlük çekebilirsiniz. Bu nesneleri bulmak için:

- 1- Sahne düzenleme alanında yer alan okları kullanarak nesnenin yerini tespit edebilirsiniz ama bu kısa mesafelerde işe yarayacaktır.
- 2- Kameranın ön tanımlı prosedürlerinden "point at" i kullanarak nesnelerinizi hızlıca bulabilirsiniz.

Görev - 1

Sahne Düzenleme Görünümüne geçiş yaparak nesne ağacından kamerayı (this.camera) seçiniz. Object Marker özelliğini kullanarak Flamemengo Nesnenize ve Camera Marker özelliğini kullanarak oluşturduğunuz Marker'ı 3B dünyanın herhangi bir yerine götürünüz. Camera nesnesinin prosedürlerinden "pointAt" ve "moveTo"yu kullanarak ne işe yaradığını açıklayınız.

CamelCase İsimlendirme

Programlama yapılırken isimlerde karışıklık olmasını önlemek için CamelCase isimlendirmeye başvurulur. Örneğin, şirin bir lama nesnesine isim vermeniz gerektiğinde SirinLama şeklinde isimlendirmeyi yapmanız gerekir. Genel olarak nesne tabanlı programlama yapılan uygulamalarda başvuru olan yöntemdir. Metotlar incelendiğinde iki veya daha fazla kelimeleli metotlarda aynı durum söz konusudur.



Zıplama Metodu

Sahne düzenleme görünümüne geçiş yaparak, Dalmaçyalı, Tilki ve Ejderha isimlerinde üstteki şekilde görülen nesnelere yönleri size doğru olacak şekilde ekleyiniz.

Kod Düzenleme arız yüzüne Edit Code butonu ile geçiş yapınız ve üçlü zıplama için yapılması gereken adımları gösteren bir organizasyon şeması oluşturunuz.

Üçünün de zıplamalarını içeren yeni bir metot oluşturmak için, Kod Düzenleme alanından Scene sekmesine geçiş yaparak Add New Scene Procedure seçeneğini tıklayınız.

5

Zıplama Metodu (Devamı)

Karşınıza metodunuzu isimlendirmenizi gösteren bir pencere açılacak buraya Zıplama yazarak OK butonuna basınız. Zıplama metodunun içeriğini düzenleyebileceğiniz yeni bir sekme karşınıza çıkacak.

Dalmaçyalı, Tilki ve Ejderha nesnelerinin her birine yukarı veya aşağı hareket edebilmeleri için nesnelere seçtikten sonra move prosedürünü kod düzenleme alanına sürükleyiniz. Aşağı veya yukarı hareket için "direction" parametresini, Kaç metre zıplamalarını istediğinizi belirtmek içinse "amount" parametresini kullanınız.

Oluşturduğunuz metodu "x" butonuyla kapatınız ve MyFirstMethod sekmesine geçiş yapınız. Burada nesne ağacından this seçeneğini seçtiğinizden emin olun ve prosedür sekmesinde oluşturduğunuz Zıplama metodunu Kod Düzenleme alanına basılı sürükleyin.

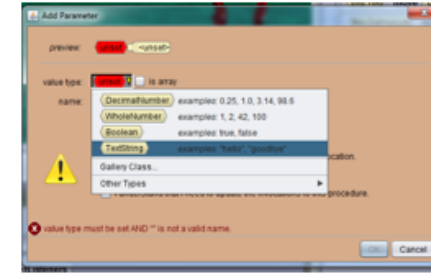
Sahnenin sağ üst köşesinde yer alan Run tuşuna basarak nesnelere hareketlerini gözlemleyin.



Parametrelili Metotlar Oluşturma

Zıplama2 isimli bir metodu Tilki nesnesi için oluşturunuz. Bunun için Scene sekmesine geçiş yaptıktan sonra type hierarchy bölümünden tilkiyi/Fox classını seçmeniz gerekmektedir. Seçtiğinizde sağ taraftan Add Parameter butonuna tıklayarak yeni bir parametre oluşturmaya başlıyoruz.

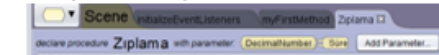
6



Parametrelerinizi İsimlendirme ve Ne Tür Değerler Alacağını Belirleme



Daha önce oluşturduğunuz parametreleri silmek veya yeniden adlandırmak için hangi nesneye tanımladıysanız onu seçtikten sonra Scene sekmesinin altından şekilde gösterildiği gibi aşağı ok butonuna basınız. Delete'yi silmek, Rename'yi yeniden adlandırmak için kullanabilirsiniz. Yeni oluşturacağınız parametrenin ne tür değerler alabileceğini belirlemek için value type bölümünü kullanacaksınız. WholeNumber'i doğal sayıları seçmek için kullanınız. Parametrenin ismini name bölümünden Süre olarak belirleyin ve OK tuşuna basınız.



Artık bundan sonra ne kadar sürede zıplamasını istiyorsanız bunu Süre parametresiyle halletmiş olacaksınız. Başka bir deyişle duration parametresi yerine Süreyi kullanmış oluyorsunuz.

7

Ek Görevler

Görev 3: Position/Orientation seçeneklerinden CollisionStartListener'ı ve Mouse seçeneklerinden addDefaultModelManipulation'ı bir arada kullanarak yeti, cadının üstüne geldiğinde bööö! Demesini sağlayın. Daha sonra bu her iki olayın ne işe yaradığını not ediniz.

Görev 4: Position/Orientation seçeneklerinden PointofChangeListener'ı kullanarak ne işe yaradığını not ediniz.

8

Araş. Gör. Emrah Kayabaşı



Lisans:
Orta Doğu Teknik
Üniversitesi
Bilgisayar ve Öğretim
Teknolojileri Eğitimi
Eylül 2007- Haziran
2012

Yüksek Lisans:
Uludağ üniversitesi
Bilgisayar ve Öğretim
Teknolojileri Eğitimi
Mart 2013 - ...

İletişim Bilgileri:
E-mail: emrahkayabasi@outlook.com
Telefon: +90 507 134 46 71
İnternet Sitesi: emrahkayabasi.net

Uludağ Üniversitesi

Eğitim Fakültesi, Bilgisayar ve Öğretim
Teknolojileri Eğitimi Bölümü, BURSA



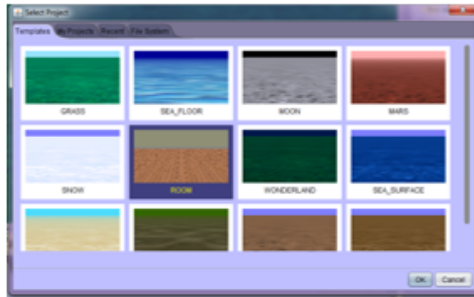
9

Olaylar

Etkinlik 3:

- Alice'de Olay Tanımlama
- Olay Türleri
 - Sahne başlangıcına ve geçen süreye tanımlı olaylar
 - Klavye'ye tanımlı olaylar
 - Fareye tanımlı olaylar
- Olayları Aktifleştirme /Silme/ Pasifleştirme
- Nesne konumlarına tanımlı olaylar

1



Tema Seçimi

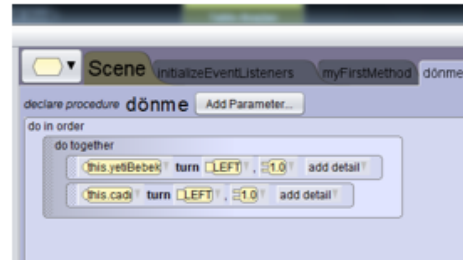
Alice ilk çalışıldığında karşınıza gelen ekranda ne tar bir 3B dünya oluşturmak istiyorsanız ona göre verilen temalardan birini seçersiniz. OK/TAMAM butonuna basıldığında Alice ara yüzüne erişmiş olursunuz. "Grass" temasını seçiniz.



Nesne Ekleme

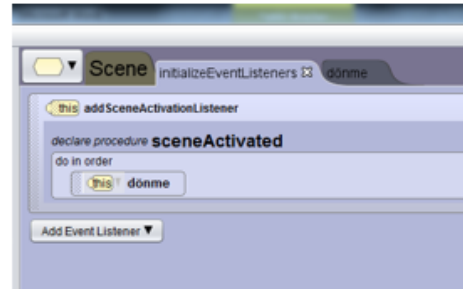
Sahne düzenleme görünümüne geçiş yaparak cadı ve yeti bebeği (witch, yetiBaby) ekleyiniz. Nesnelerin isimlerini çamelÇaşe isimlendirme kullanarak veriniz. Balerinlerin yüzlerini kameraya dönük olarak ayarlayın (rotate özelliği ile) ve Sahne görünümünden Kod Düzenleme Görünümüne "Edit Code" butonuna basarak geçiniz.

2



Metod Tanımlama

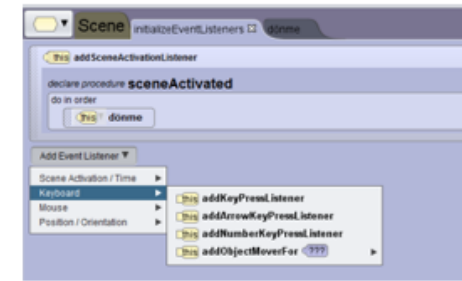
Scene sekmesini kullanarak Döndür isimli bir metod/prosedür oluşturunuz. Type hierarchy bölümünde Scene'in seçildiğinden emin olunuz. Bu metod içerisinde cadı ve yetinin, sola doğru 1 kez dönmelerini sağlayacak turn prosedürünü içerisinde barındırsın. Do together kontrolü ile de her ikisinin aynı anda dönsünler.



Hazırlanan metodu Olay(Event)'a Ekleme

initializeEventListeners sekmesine geçtikten sonra, Alice'de öntanımlı olarak gelen worldMyFirstMethod'u basılı sürükleyerek kaldırdınız. Daha sonra onun bulunduğu yere(do in order'in içine) dönme isimli prosedürünüzü basılı sürükleyiniz. Yaptığınız işlemleri "dönme" ismi ile kaydediniz.

3



Olay Türleri

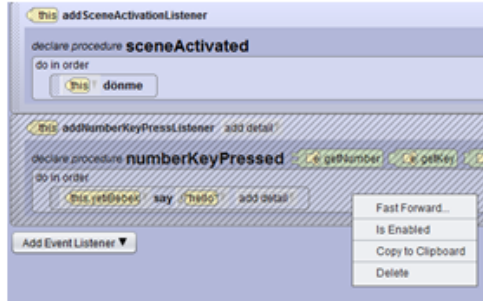
Olaylar dört farklı kategorilerde sınıflandırılmıştır. Bunlar şekilde de gösterildiği gibi

1. Scene Activation/Time: İlk sahne aktifleştirildiğinde veya herhangi bir zamana gelindiğinde gerçekleşmesi gereken olayları tanımlamak için kullanılır.
2. Keyboard: Bu olay türü ile klavye tuşlarını kullanarak nesnelere yönlendirme işlemini yapabilirsiniz.
3. Mouse: Fare işaretçileri ve tuşları kullanılarak bir olay tanımlamak için kullanılır.
4. Position / Orientation: Nesnelerin konumları veya bakış açılarından yola çıkarak olay tanımlamak için kullanılır. Bu nesnelerin çarpması, birbirlerine olan yakınlıkları, yüzlerinin dönük olması gibi olayları içerisinde barındırır.

Görev -1

addSceneActivationListener ne işe yaramaktadır. Cevabınızı not ediniz. Dönme2 isimli bir alice dünyası oluşturarak 5.saniyeden itibaren nesnelerin tekrardan dönmelerini sağlayan bir olay tanımlayınız.

4



Eklenen Olayları / Metotları Aktifleştirme + Pasifleştirme + Silme

Alice'de hazırladığımız metotları, eklediğiniz eylemleri aktifleştirme veya pasifleştirme işlemini yapmak için metotlarınızın veya olaylarınızın üstünde sağ tuşa tıklayarak "is enabled" seçeneğine onay vermeniz veya kaldırmanız gerekir. Pasifleştirdikleriniz işlem sıralamasında gerçekleştirilmeden atlanır ve üstü çizgili hale gelir.

Silme işlemini gerçekleştirmek içinse Delete seçeneğini kullanılır.

Denemeler

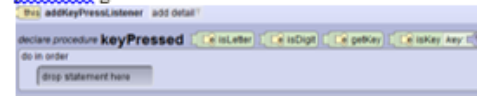
Öncelikle initializeEventListener sekmesine gidilir. Burada yer alan AddEventListener açılır penceresine tıklayarak Scene Activation/Time seçeneklerinin her birini deneyerek, oluşturduğunuz metotları Pasifleştirdikten sonra projenize Dönme3 ismini verip kaydediniz.

5



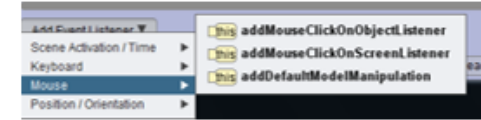
Klavye'deki bir tuşa olay tanımlama

Öncelikle initializeEventListener sekmesine gidilir. Burada yer alan AddEventListener açılır penceresine tıklayarak Keyboard seçeneğinin alt seçeneklerinden addKeyPressListener seçilir. Karşımıza şu şekilde bir prosedür gelir:



Bir if kontrol aracı, do in order içerisine basılı sürükledikten sonra yukarıdaki şekilde gösterilen prosedür parçası if döngüsünün içerisine şekildeki gibi dahil edilir. Daha sonra hangi tuşa basıldığında olay gerçekleştirilecekse o seçilir. Örnekte, ilk olarak herhangi bir tuşa basıldığında cadı "Hello" diyor ve daha sonra eğer o klavye tuşu L ise cadı ve yeti dönme işlemini gerçekleştiriyor. Sizde örnektekinden farklı, if kontrol aracının içine kullanıldığı metot oluşturarak projenizi Dönme4 olarak kaydediniz.

6



Fare işaretçisi ve tuşlarının kullanımı

Fare işaretçisine veya tuşlarına bir olay tanımlamada üç temel seçenek söz konusudur. Bunlar:

1. addMouseClickOnObjectListener: Bir nesnenin üzerine tıklandığında gerçekleştirilmesi gereken bir olay var ise bu seçenek kullanılır.
2. addMouseClickOnScreenListener: Sahnenin herhangi bir alanına tıklandığında gerçekleştirilmesini istediğiniz bir olay var ise bu seçenek kullanılır.
3. addDefaultModelManipulation: Bu seçenek sayesinde fare ile 3B dünyanızdaki tüm nesnelere hareket ettirme şansına sahip olursunuz. Bu hareketler projeniz RUN tuşuyla çalıştırdıktan sonra gerçekleştirilir.

Görev -2

Sizde fare tuşlarının olaylarda kullanımı ile ilgili yukarıda verilen bilgiler ışığında her birini deneyiniz. Bu noktada 3B dünyanıza yeni nesnelere eklemekte özgürsünüz. Metotların herbirinin ne işe yaradığını, kullandığınız olaylar ile gerçekleştirilen işlemleri not ediniz. Örneğin, "fareye tanımlı olaylardan birisini kullanarak hazırladığımız kod blokları sayesinde, cadı nesnesine tıklandığında sağa doğru bir metre hareket etmesini not etmeniz gerekmektedir.

7

Ek Görevler

Görev 3: freesound.org sitesine üye olduktan sonra projenizde ihtiyaç duyabileceğiniz sesleri indirmeye başlayınız. Alice'de kullanabileceğiniz sesler -wav uzantılarını destekleyemediğinden birde dönüştürücü indirmeyi unutmayınız. Kullandığınız dönüştürücünün ismini not ediniz. Bunun yanı sıra ses dosyalarınızı da ödev teslimatında sıkıştırıp göndermeyi unutmayınız.

Görev 4: Projenizde kullanabileceğiniz sesleri bulabileceğiniz, en az iki site bulunuz ve not ediniz.

BONUS: Alice'de 60 saniyelik bir hikâye kesidini oluşturunuz. Aranızdan seçilen beş proje sene sonu notunuza doğrudan etki edecek +5 puan kazanacak.

5

Araş. Gör. Emrah Kayabaşı



Lisans:
Orta Doğu Teknik
Üniversitesi
Bilgisayar ve Öğretim
Teknolojileri Eğitimi
Eylül 2007- Haziran
2012

Yüksek Lisans:
Uludağ Üniversitesi
Bilgisayar ve Öğretim
Teknolojileri Eğitimi
Mart 2013 - ...

İletişim Bilgileri:
E-mail: emrahkayabasi@outlook.com
Telefon: +90 507 134 46 71
İnternet Sitesi: emrahkayabasi.net

Uludağ Üniversitesi

Eğitim Fakültesi, Bilgisayar ve Öğretim
Teknolojileri Eğitimi Bölümü, BURSA



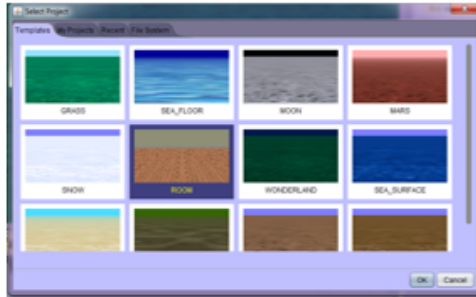
6

Algoritma

Etkinlik 4:

- Akış diyagramları oluşturma
- Kontrollü Döngüler
- Ses ekleme

1



Tema Seçimi

Alice ilk çalıştırıldığında karşınıza gelen ekranda ne tar bir 3B dünya oluşturmak istiyorsanız ona göre verilen temalardan birini seçersiniz. OK/TAMAM butonuna basıldığında Alice ara yüzüne erişmiş olursunuz.

"Snow" temasını seçiniz.

Akış Diyagramı Oluşturma

Algoritma Bilgisayara verilen bir görevin tamamlaması veya bir soruna çözüm getirmek amacıyla kullanılan işlem basamaklarıdır. Algoritmaların gösteriminde başvurulan temel yollardan birisi akış diyagramlarıdır.

İhtiyaç duyacağınız nesnelere:

1. Taksı nesnesi
2. İip nesnesi
3. Çim alan
4. Taksı ve İipin üstünde gideceği yol
5. Ses Dosyaları (Çarpışma anı ve araba hareketleri için)

Akış diyagramınızda taksı ve İip birbirine döndükten sonra hareket etmeleri ve çarpışmalarını gösteriniz. Çarpışma anında, çarpışmaya ait bir ses çıkacak, arabaların hareket esnasında çıkardıkları arkaplan müziği ise son bulacak. Son olarak, arabaların her ikisinden de Aaaa! diye düşünme balonları çıkacak. Akış diyagramlarınızı bir A4 kağıda not ediniz.

2



Nesne Ekleme

Sahne düzenleme görünümüne geçiş yaparak taksı ve İip (hatchback, humvee) isimli iki nesne ekleyiniz. Eklediğiniz nesnelere şekildaki gibi birbirine dönük olsun ve arkaplanına da "newblanket" nesnesini ekleyip düzenleyerek bir yol imajı veriniz. Bunu yaparken nesne düzenleme araçlarından Resize, Translation, Default ve Rotation'ı nerede ve nasıl kullandığınızı not ediniz. Yaptığınız düzenlemeleri çarpışma.a3w ismiyle kaydediniz.



Nesnelere birbirine döndürme ve hareket

Şimdi de yapmanız gereken taksı ve İipin yüzleri birbirine dönmelerini sağlayarak çarpışacak kadar birbirlerine yaklaşmalarını olacak. Bu hareket 5 saniye kadar sürecek. Bu sırada arabaların hareketleri ile ilgili bir ses çıkacak ve çarpıştıktan sonra bu ses son bulacak. Bu süreci, sesi eklediğiniz prosedürün içerisinde ayarlayabilirsiniz.

3



Çarpışmayı saptamak ve çarpışma sesi ekleme

Taksı ve İipin birbirlerine çarpıştıklarını anlamak için bir olay tanımlamak gereklidir. Bunun için collisionStartListener olayı tanımlanır ve oluşturulan arayların ilkinde taksı, diğerine İip eklenir.

Çarpışma anı gerçekleştiğinde ise bir çarpışma sesi eklenir ve taksı ile İipin eş zamanlı olarak Aaaa! diye bağurmaları gerekmektedir.

Tüm yaptıklarınızı Çarpışma2.a3w ismiyle kaydediniz

Görev -1

Oluşturduğumuz olayı ve yazdığımız metodları pasif hale getiriniz. Şimdi İip ve taksinin birbirine doğru hareketini, count döngüsünü kullanarak 5 kere 1m olacak şekilde ayarlayınız. Dosyanızı çarpışma3 ismiyle kaydediniz.

4

Ek Görevler

Görev 2: git isimli bir metot oluşturunuz ve `myFirstMethod`'u kaldırmız. Git metotunun içerisinde sandalalrda birisinin yüzünün tıklanan nesneye dönmesini sağlayın ve ileriye doğru 2 metre gitmesini sağlayınız.

Görev 3: Sandallarınızdan birini `mouse`'un tuşlarına basıldığında hedefteki nesneye götürmesiniz sağlayınız. Bunu bir olay tanımlayarak yapabilirsiniz.

Görev 4: Bu etkinlikte mantıksal sınımaları yaptığımız yerlerden iki örnek vererek ne işe yaradıklarını kendi cümlelerinizle açıklayınız.

5

Aras. Gör. Emrah Kayabaşı



Lisans:
Orta Doğu Teknik
Üniversitesi
Bilgisayar ve Öğretim
Teknolojileri Eğitimi
Eylül 2007- Haziran
2012

Yüksek Lisans:
Uludağ Üniversitesi
Bilgisayar ve Öğretim
Teknolojileri Eğitimi
Mart 2013 - ...

İletişim Bilgileri:
E-mail: emrahkayabasi@outlook.com
Telefon: +90 507 134 46 71
İnternet Sitesi: emrahkayabasi.net

Uludağ Üniversitesi

Eğitim Fakültesi, Bilgisayar ve Öğretim
Teknolojileri Eğitimi Bölümü, BURSA



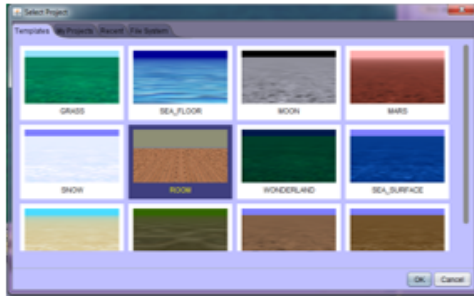
6

Mantıksal Sınıamalar

Etkinlik 4:

- Kızgın penguen
 - Nesnelerin parçalarıyla hareket
 - Olaylar (`Events`) tanımlama
 - Mantıksal Sınıamalar yapma
- Sandal ile açılmak

1



Tema Seçimi

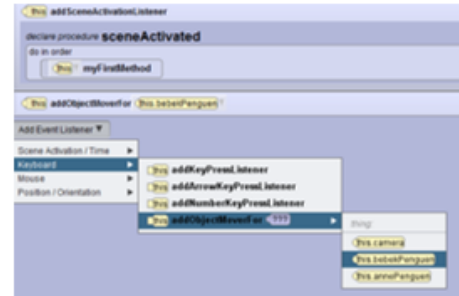
Alice ilk çalıştırıldığında karşınıza gelen ekranda ne tarz bir 3B dünya oluşturmak istiyorsanız ona göre verilen temalardan birini seçersiniz. OK/TAMAM butonuna basıldığında Alice ara yüzüne erişmiş olursunuz. “Snow” temasını seçiniz.



Nesne Ekleme

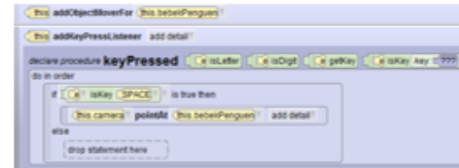
Sahne düzenleme görünümüne geçiş yaparak bebekPenguin ve annePenguin (penguin) isimli iki nesne ekleyiniz. Bu çalışmada olayları kullanarak bebek penguinin hareketini sağlayıp, annePenguin nesnesine yaklaşıncaya annenin kızıp zıplayarak kanat çırpmasını sağlayacaksınız. bebekPenguin isimli nesne klavye tuşlarını kullanarak hareket ettirilebilecek. Kamera ayarlarını da yine yapmanız gerekecek.

2



Olayları Tanımlama

Öncelikle klavye tuşlarıyla hareket edecek olan bebekPenguin nesnemize addObjectMoverFor olayını kullanarak bir olay tanımlıyoruz. Bu sayede A, S, D, W ve yön tuşlarıyla bu hareket işlemi gerçekleştirilebilir hale geliyor.

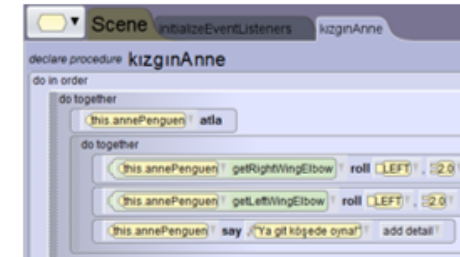


Bunun yanı sıra space tuşuna basıldığında bebekPenguin'e kameranın odaklanmasını da yukarıdaki olay tanımları ile ayarlıyoruz. Bu sayede kamera dışına (off-camera) çıktığı zaman bebek penguini takip edebileceğiz.



Son olarak bebek penguin, Anne penguinine yaklaştığında hazırlayacağımız kızgınAnne metodunun çalışması için addProximityEnterListener ekliyoruz. Bu sayede bebek penguin her yaklaştığında hazırlayacağımız kızgınAnne metodu çalışacak.

3



Anne penguin'i kızdırma

Anne penguin kızdığında zıplayıp kanatlarını çırpmasını istiyoruz. Bunun için öncelikle kızgınAnne isimli bir metod tanımlıyoruz ve içerisine aynı anda annenin iki defa zıplamasını ve kanatlarını çırpmasını sağlayan komutlarımızı oluşturuyoruz.

Görev -1

Deniz / water teması kullanılarak yeni bir proje oluşturunuz. Sahnenizde denizle ilgili en az beş tane nesne bulunsun (sandal1, sandal2, korsan gemisi, ada gibi). Çalışmanızı sandal adıyla kaydediniz.

4

Ek 6: Ders İeriđi ve Öğrenme Kazanımları

1	Dersin Adı:	Ü boyutlu (3D) programlama
2	Dersin Kodu:	BIL4109
3	Dersin Türü:	Semeli
4	Dersin Seviyesi:	Lisans
5	Dersin Verildiđi Yıl:	4.yıl
6	Dersin Verildiđi Yarıyıl:	7. yarıyıl
7	Dersin AKTS Kredisi:	5
8	Ders Saati (saat/hafta):	3 saat
9	Dersin Veriliş Şekli:	Yüz yüze
10	Dersin Koordinatörü:	Yrd. Do. Dr. Semiral Öncü
11	Dersin Öğrenme Kazanımları	
1		Ü boyutlu dünyaların ve ü boyutlu programlamanın ne olduđu kavrayabilme
2		Ü boyutlu nesnelere tasarlayabilme, düzenleme, animasyonlar oluşturabilme
3		Ü boyutlu nesnelere, belirlenen amaçlar doğrultusunda yerleştirme ve kullanabilme
4		Oluşturdukları programların sonuçlarını eşzamanlı görerek, animasyonlarda düzenleme ve düzeltmeler yapabilme
5		Grup ve bireysel çalışmaları etkin bir şekilde kullanarak bir ürün ortaya çıkarabilme
6		Komut dizileri oluşturabilme ve sırasıyla çalıştırabilme
7		Oluşturulan komut dizileri ile animasyonlarındaki nesnelere hareketleri arasındaki ilişkiyi kavrayabilme
8		3D dünyada prosedürleri ve fonksiyonları kullanabilme
9		Ses kayıtları oluşturma ve/veya var olan kayıtları kullanarak seslendirme yapabilme
10		3D nesnelere ile ses kayıtları arasında ilişkiler kurabilme
11		3D dünyada öykü, oyun ve/veya videolar oluşturarak yayımlayabilme

Özgeçmiş

Doğum Yeri ve Yılı	: Karabük - 1988		
Öğrenim Gördüğü Kurumlar	Başlama Yılı	Bitirme Yılı	Kurum Adı
Lisans	2007	2012	Orta Doğu Teknik Üniversitesi
Yüksek Lisans	2013	Uludağ Üniversitesi
Bildiği Yabancı Diller ve Düzeyi	: İngilizce - Çok iyi		
Çalıştığı Kurumlar	Başlama ve Ayrılma	Kurum Adı	
	1. 2010 – 2011	Nic.tr alan adları yönetimi	
	2. 2013 –	Uludağ Üniversitesi	
Yurt Dışı Görevleri	:		
Kullandığı Burslar	:		
Aldığı Ödüller	:		
Üye Olduğu Bilimsel ve Mesleki Topluluklar	:		
Editör veya Yayın Kurulu	:		
Üyeliği	:		
Yurt İçi ve Yurt Dışında Katıldığı Toplantılar	:		
Katıldığı Yurt İçi ve Yurt Dışı Bilimsel Toplantılar	:		

Yılmaz, N., Kayabaşı, E., & Fidan, A. (2014). A Case Study: Learning Experiences of Prospective Teachers in a Digital Storytelling Tool Called Toondoo. EDULEARN14 Proceedings, 1246-1252.

Kayabaşı, E., Öncü, S. (2015). Programlama Öğretiminde Üç Boyutluluk Algısı: Alice'le Programlama Deneyimi. ICITS2015 Abstract Proceedings, 74.

Kayabaşı, E., Öncü, S., Şentürk, A., & Şengel, E. (2015). Wonder their opinions?: Students eagerly justify why they need tablet computers. ITTES2015 Abstract Proceedings.

Yayımlanan Çalışmalar :

Diğer Profesyonel

Etkinlikler :

15/01/2016

Emrah KAYABAŞI

ULUDAĞ ÜNİVERSİTESİ

TEZ ÇOĞALTMA VE ELEKTRONİK YAYIMLAMA İZİN FORMU

Yazar Adı Soyadı	Emrah Kayabaşı
Tez Adı	Öğretmen Adaylarının Alice Deneyimi: 3B Ortamda Programlama
Enstitü	Eğitim Bilimleri Enstitüsü
Anabilim Dalı	Bilgisayar ve öğretim teknolojileri eğitimi
Bilim Dalı	Bilgisayar ve öğretim teknolojileri eğitimi
Tez Türü	Yüksek Lisans
Tez Danışman(lar)ı	Yrd. Doç. Dr. Semiral Öncü
Çoğaltma (Fotokopi Çekim) İzni	<input type="checkbox"/> Tezimden fotokopi çekilmesine izin veriyorum <input checked="" type="checkbox"/> Tezimin sadece içindekiler, özet, kaynakça ve içeriğinin %10 bölümünün fotokopi çekilmesine izin veriyorum <input type="checkbox"/> Tezimden fotokopi çekilmesine izin vermiyorum
Yayımlama İzni	<input type="checkbox"/> Tezimin elektronik ortamda yayımlanmasına izin veriyorum <input checked="" type="checkbox"/> Tezimin elektronik ortamda yayımlanmasının ertelenmesini istiyorum 1 yıl <input checked="" type="checkbox"/> 2 yıl <input type="checkbox"/> 3 yıl <input type="checkbox"/> <input type="checkbox"/> Tezimin elektronik ortamda yayımlanmasına izin vermiyorum

Hazırlamış olduğum tezin yukarıda belirttiğim hususlar dikkate alınarak, fikri mülkiyet haklarım saklı kalmak üzere Uludağ Üniversitesi Kütüphane ve Dokümantasyon Daire Başkanlığı tarafından hizmete sunulmasına izin verdiğimi beyan ederim.

Tarih : 03.02.2016

İmza : 