

**ROBOTİK KODLAMA ETKİNLİKLERİNİN ORTAOKUL
ÖĞRENCİLERİNİN KODLAMAYA İLİŞKİN ÖZ-YETERLİK
ALGILARINA ETKİSİ VE ETKİNLİKLERE İLİŞKİN
ÖĞRENCİ YAŞANTILARI**

**EFFECTS OF ROBOTIC CODING ACTIVITIES ON THE
EFFECTIVENESS OF SECONDARY SCHOOL STUDENTS
'SELF-EFFICACY AND STUDENT EXPERIENCE ABOUT
ACTIVITIES**

İBRAHİM KASALAK

Hacettepe Üniversitesi

Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı

Yüksek Lisans Tezi olarak hazırlanmıştır.

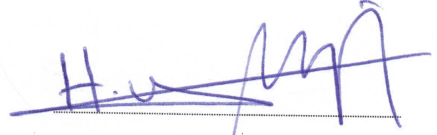
2017

KABUL ve ONAY

Eđitim Bilimleri Enstitüsü M¼d¼rl¼ę¼'ne,

İbrahim KASALAK'ın hazırladığı "Robotik Kodlama Etkinliklerinin Ortaokul Öğrencilerinin Kodlamaya İlişkin Öz-yeterlik Algılarına Etkisi ve Etkinliklere İlişkin Öğrenci Yaşantıları" başlıklı bu çalışma jürimiz tarafından **Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı Yüksek Lisans Tezi** olarak kabul edilmiştir.

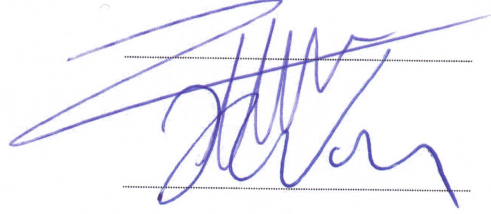
Başkan Prof. Dr. Halil YURDUGÜL



Üye (Danışman) Prof. Dr. Arif ALTUN



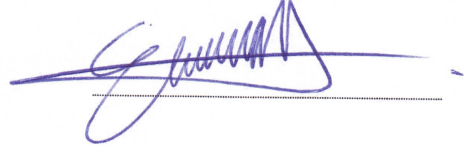
Üye Doç. Dr. Selçuk ÖZDEMİR



Üye Doç. Dr. Hakan TÜZÜN



Üye Doç. Dr. Erman YÜKSELTÜRK



ONAY

Bu tez Hacettepe Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliđi'nin ilgili maddeleri uyarınca yukarıdaki jüri üyeleri tarafından / / tarihinde uygun gör¼lm¼ş ve Enstitü Yönetim Kurulunca / / tarihinde kabul edilmiştir.

Prof. Dr. Ali Ekber ŞAHİN
Eđitim Bilimleri Enstitüsü M¼d¼r¼

YAYIMLAMA VE FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe Üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır. Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alınarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Tezimin/Raporumun tamamı dünya çapında erişime açılabilir ve bir kısmı veya tamamının fotokopisi alınabilir.

(Bu seçenekle teziniz arama motorlarında indekslenebilecek, daha sonra tezinizin erişim statüsünün değiştirilmesini talep etmeniz ve kütüphane bu talebinizi yerine getirse bile, teziniz arama motorlarının önbelleklerinde kalmaya devam edebilecektir)

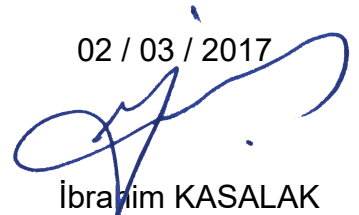
Tezimin / Raporumun 02 / 03 / 2018 tarihine kadar erişime açılmasını ve fotokopi alınmasını (İç Kapak, Özet, İçindekiler ve Kaynakça hariç) istemiyorum.

(Bu sürenin sonunda uzatma için başvuruda bulunmadığım takdirde, tezimin/raporumun tamamı her yerden erişime açılabilir, kaynak gösterilmek şartıyla bir kısmı veya tamamının fotokopisi alınabilir).

Tezimin/Raporumun tarihine kadar erişime açılmasını istemiyorum ancak kaynak gösterilmek şartıyla bir kısmı veya tamamının fotokopisinin alınmasını onaylıyorum.

Serbest Seçenek/Yazarın Seçimi:

02 / 03 / 2017



İbrahim KASALAK

ETİK BEYANNAMESİ

Hacettepe Üniversitesi Eğitim Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada,

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversitede veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.



İbrahim KASALAK

TEŞEKKÜR

Yüksek lisans eğitimine başlangıç tarihimin 2005 olduğu düşünülürken, 22 yaşında yeni mezun bir genç olarak başladığım yüksek lisans eğitimini 12 yıllık bir öğretmen olarak tamamlamak çok ayrı bir duygu oldu benim için.

Yıllar sonra tez yazmaya karar verdiğim günden, tamamladığım güne kadar beni her zaman yüreklendiren, desteğini ve bilgisini esirgemeyen sevgili hocam, tez danışmanım Prof. Dr. Arif Altun'a sonsuz teşekkürlerimi sunarım.

Tez yazma sürecimde fikirlerini paylaşan, etkinliklerin ve ölçek maddelerinin geliştirilmesi süreçlerinde katkı sunan Dr. Gökhan Akçapınar'a ve ölçek geliştirme sürecindeki istatistiksel analizlerde yardımcı olan Dr. Fatma Bayrak'a kendileri de yoğun olmalarına rağmen zaman ayırdıkları için çok teşekkür ederim.

Hacettepe BÖTE'de birçoğunu 2001'den beri tanıdığım hocalarıma, her bölüme geldiğimde gösterdikleri sıcak yaklaşımlarından dolayı teşekkürlerimi sunarım.

Tez jürimde yer alan Prof. Dr. Halil Yurdugül, Doç. Dr. Selçuk Özdemir, Doç. Dr. Hakan Tüzün ve Doç. Dr. Erman Yükseltürk'e araştırmama ilişkin yapıcı görüş ve önerileri için teşekkürlerimi sunarım.

Bilişim teknolojileri öğretmenlerine gönüllü olarak sunduğu eğitimler ve girişimci gençler yetiştirme gibi fikirleriyle bizleri heyecanlandıran Doç. Dr. Selçuk Özdemir'e teşekkürü borç bilirim. Mesleki gelişmelerimize katkı sağlayan etkinlikleri bizlere sunan Bilişim Teknolojileri Eğitimcileri Derneğine de ayrıca teşekkür ederim. Robotik kodlama eğitimine nasıl başlamam gerektiği konusunda verdiği fikirlerle beni yüreklendiren, robotik kodlama etkinlikleriyle ilişkili STEM eğitimine yönelik bilgi ve tecrübesini paylaşan Yrd. Doç. Dr. Bahadır Yıldız'a da teşekkürlerimi sunarım.

Sevgili hayat arkadaşım, Mürüvet'im, evde bazen benim ilgilenmem gereken konularla da ilgilenerek bu tezi yazmamdaki en büyük destekçim oldu. Oğlum Alper ve kızım Elif, onlarla yeterince ilgilenemesem bile, "Ama babamızın tezini yazması lazım, sessiz olmalıyız" diyerek beni her zaman anlayışla karşıladılar. Onlara da gösterdikleri anlayıştan, verdikleri karşılıksız sevgiden dolayı sonsuz teşekkürlerimi sunarım.

ROBOTİK KODLAMA ETKİNLİKLERİNİN ORTAOKUL ÖĞRENCİLERİNİN KODLAMAYA İLİŞKİN ÖZYETERLİK ALGILARINA ETKİSİ VE ETKİNLİKLERE İLİŞKİN ÖĞRENCİ YAŞANTILARI

İbrahim KASALAK

ÖZ

Bu çalışma kapsamında, ortaokul düzeyinde robotik kodlama etkinlikleri ile öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algıları arasında anlamlı bir ilişki olup olmadığı tespit edilmesi amaçlanmıştır. Bunun yanında robotik kodlama etkinliklerine ilişkin öğrenci yaşantıları araştırılmış, robotik kodlama etkinliklerinin düzenlenmesi sürecinde dikkat edilmesi gerekenlere ilişkin bulgulara yer verilmiştir.

Çalışma iki aşamadan oluşmuştur. Birinci aşamada “blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeği” geliştirilmiştir. Ayrıca geliştirilen ölçme aracı ile katılımcılardan elde edilen öz-yeterlik puanları çeşitli değişkenler açısından incelenmiştir. Ortaokul düzeyinde eğitim gören toplamda 329 öğrenciden toplanan verilerle yapılan geçerlik ve güvenirlik çalışması sonucunda 5 maddeden oluşan “basit blok temelli programlama görevleri” ve 7 maddeden oluşan “karmaşık blok temelli programlama görevleri” şeklinde isimlendirilen 2 faktörlü 5’li Likert tipinde bir “Blok Temelli Öz-yeterlik algısı Ölçeği” geliştirilmiştir ($\alpha=.893$). Açıklayıcı faktör analizi sonunda 5 maddelik birinci faktör toplam varyansın %11.462’ini, 7 maddelik ikinci faktör ise toplam varyansın %46.763’ünü açıklamış ve bu yapıya ilişkin model doğrulayıcı faktör analizi ile de doğrulanarak bireylerin blok temelli programlamaya ilişkin öz-yeterlik algısını ölçen geçerli ve güvenilir bir araç elde edilmiştir.

İkinci aşamada ise 5 haftalık robotik kodlama etkinlikleri planlanmış, ardından etkinlikler bir devlet okulundaki 58 öğrenciyle uygulanmıştır. Öğrencilerin ön test-son test sonuçlarına göre hem basit hem de karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarında grup içi pozitif yönde anlamlı değişim meydana gelmiştir (basit $t= -5.01$, $p=0.000$; karmaşık $t= -8.84$, $p=0.00$). Çeşitli değişkenler açısından blok temelli programlamaya ilişkin öz-yeterlik algılarındaki farklılaşmaya

bakıldığında ise, cinsiyete (basit $t = -0.58$, $p = 0.56$; karmaşık $t = 0.87$, $p = 0.39$), evde bilgisayar sahipliğine (basit $t = -1.23$, $p = 0.22$; karmaşık $t = -1.23$, $p = 0.22$), evde internet bağlantısı sahipliğine (basit $t = -0.37$, $p = 0.22$; karmaşık $t = -0.44$, $p = 0.66$) ve ders dışında Scratch programına çalışma olanağına göre (basit $t = -0.91$, $p = 0.37$; karmaşık $t = -0.91$, $p = 0.37$) anlamlı olarak farklılaşmadığı bulunmuştur.

Robotik kodlama etkinliklerine ilişkin öğrenci yaşantılarını tespit etmek amacıyla Alanyazında İngilizcesi bulunan “etkinlik algısı ölçeği” Türkçeye uyarlanmıştır. Her etkinliğin sonunda uygulanan etkinlik algısı ölçeğinden, gözlem ve görüşme notlarından elde edilen bulgular yorumlandığında, öğrencilerin etkinlikleri eğlenceli ve ilgi çekici buldukları, etkinliklere katılmaya istekli oldukları, etkinliklerin kişisel gelişimlerine olumlu katkı sağladığını düşündükleri yönünde bulgular elde edilmiştir.

Anahtar sözcükler: Blok temelli programlama, kodlama, robotik kodlama, öz-yeterlik algısı

Danışman: Prof. Dr. Arif ALTUN, Hacettepe Üniversitesi, Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı

EFFECTS OF ROBOTIC CODING ACTIVITIES ON THE EFFECTIVENESS OF SECONDARY SCHOOL STUDENTS 'SELF-EFFICACY AND STUDENT EXPERIENCE ABOUT ACTIVITIES

İbrahim KASALAK

ABSTRACT

In this study, it was aimed to determine whether there is a meaningful relationship between robotic coding activities at secondary school level and self-efficacy perceptions related to block-based programming. In addition, students' experiences in robotic coding activities have been researched and findings about what should be considered during the organization of robotic coding activities.

The study has two phases. In the first step, the "Block-Based Self-Efficacy Perception Scale" was developed. In addition, the self-efficacy scores obtained from participants with the developed measurement tool were examined in terms of various variables. As a result of the validity and reliability study performed on 329 students in the middle school level, A two-factor Likert-type "Block-Based Self-Efficacy Perception Scale" was defined as "Simple Block-Based Programming Tasks" consisting of 5 items and "Complex Block-Based Programming Tasks" consisting of 7 items ($\alpha = .893$). At the end of the exploratory factor analysis, the first factor of 5 items explained 11.462% of the total variance, the second factor of 7 items explained 46.763% of the total variance and validated by model confirmatory factor analysis of this structure, a reliable scale has been achieved.

In the second phase, 5-week robotic coding activities were planned, followed by 58 students in a public school. According to pre-test and post-test results of the students, there was a significant change in intra-group positive direction in the self-efficacy perception scores of both simple and complex block-based programming (simple $t = -5.01$, $p = 0.000$, complex $t = -8.84$, $p = 0.00$). In terms of various variables, when we look at the differentiation of self-efficacy perceptions regarding block-based programming, it is found that it does not differ significantly according to gender (simple $t = -0.58$, $p = 0.56$, complex $t = 0.87$, $p = 0.39$), computer ownership at home (simple $t = -1.23$, $p = 0.22$, complex $t = -1.23$, $p = 0.22$), Internet

connection ownership at home (simple $t = -0.37$, $p = 0.22$, complex $t = -0.44$, $p = 0.66$) and the possibility to study Scratch program out of course (simple $t = -0.91$, $p = 0.37$, complex $t = -0.91$, $p = 0.37$).

In order to determine the students' experiences regarding robotic coding activities, the "activity perception scale" in English was adapted to Turkish. When findings obtained from the "activity perception scale" and from observation and interviews were interpreted at the end of each activity, findings were obtained that students thought that they found the activities fun and interesting, were willing to participate in the activities, and contributed positively to the personal development of the activities.

Keywords: Blok based programming, coding, robotic coding, self-efficacy perception

Advisor: Prof. Dr. Arif ALTUN, Department of Computer Education and Instructional Technology

İÇİNDEKİLER

KABUL ve ONAY	ii
YAYIMLAMA VE FİKRİ MÜLKİYET HAKLARI BEYANI	iii
ETİK BEYANNAMESİ.....	iv
TEŞEKKÜR	v
ÖZ	vi
ABSTRACT	viii
İÇİNDEKİLER.....	x
TABLolar DİZİNİ	xiii
ŞEKİLLER DİZİNİ	xiv
SİMGELER VE KISALTMALAR DİZİNİ	xv
1. GİRİŞ	1
1.1. Araştırmanın Önemi.....	5
1.2. Araştırma Problemi	7
1.3. Araştırma Soruları.....	7
1.4. Sınırlılıklar.....	8
1.5. Tanımlar	8
2. İLGİLİ ALAN YAZIN	11
2.1. Kompütasyonel Düşünme.....	11
2.2. Blok Temelli Programlama (Kodlama) Eğitimi.....	13
2.3. Programlama Öğretim Programı Geliştirme Çalışmaları.....	17
2.4. Öz-yeterlik Algısı ve Programlama Performansı	20
2.5. Robotik Kodlama Eğitimi.....	22
2.6. Genel Değerlendirme.....	26
3. YÖNTEM.....	29
3.1. Araştırmacının Rolü ve Uygulamanın Yürütüldüğü Ortam	29
3.2. Araştırma Modeli.....	30
3.3. Çalışma Grubu.....	33
3.4. Öğretim Sürecinin Tasarımı	35
3.5. Veri Toplama Araçları	40
3.5.1. Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği.....	41
3.5.1.1. Ön Uygulama ve Madde Havuzunun Oluşturulması	42
3.5.1.2. Verilerin Toplanması	42

3.5.1.3. Verilerin Analizi	43
3.5.1.4. Madde Geçerliđi Analizleri.....	44
3.5.1.5. Yapı Geçerliđi Analizleri	46
3.5.1.6. Açımlayıcı Faktör Analizi Sonuçları	47
3.5.1.7. Doğrulamayı Faktör Analizleri	49
3.5.1.8. Güvenirlik Analizi.....	50
3.5.1.9. Ölçeđin Uygulanmasına İlişkin Bazı Deđişkenler Açısından Betimsel Analizler.....	51
3.5.2. Robotik Kodlama Etkinliklerine İlişkin Öğrenci Yaşantılarını Tespit Etmeye Yönelik Veri Toplama Araçları	53
3.5.2.1. Etkinlik Algısı Ölçeđi.....	54
3.5.2.2. Gözlemlere ve Görüşmelere Ait Konu Başlıkları	54
3.5.2.3. Verilerin analizi	54
4. BULGULAR.....	56
4.1. Ön Analizler	56
4.2. Robotik Kodlama Etkinlikleri ve Kodlamaya İlişkin Öz-yeterlik Algısı	58
4.3. Robotik Kodlama Etkinlikleri ve Çeşitli Deđişkenlere Göre Kodlamaya İlişkin Öz-yeterlik Algısı.....	59
4.3.1. Cinsiyet ve Kodlamaya İlişkin Öz-yeterlik Algısı	60
4.3.2. Evde Bilgisayar Sahipliđi ve Kodlamaya İlişkin Öz-yeterlik Algısı	61
4.3.3. Evde İnternet Bağlantısı Sahipliđi ve Kodlamaya İlişkin Öz-yeterlik Algısı.....	61
4.3.4. Ders Dışında Scratch Programına Çalışma Olanađı ve Kodlamaya İlişkin Öz-yeterlik Algısı.....	61
4.4. Öğrenci Yaşantıları	62
4.4.1. Robotik Kodlama Etkinliklerini Eğlenceli Bulmaya İlişkin Etkinlik Algısı	64
4.4.2. Robotik Kodlama Etkinliklerinin Kişisel Gelişime Etkisine İlişkin Etkinlik Algısı.....	66
4.4.3. Etkinliđi Yapmaya İlişkin İsteklilik ve Etkinliđi İlgi Çekici Bulma	68
5. SONUÇ, YORUM VE ÖNERİLER.....	71
5.1. Sonuç ve Yorumlar	71
5.1.1. Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Ölçeđi.....	72
5.1.2. Robotik Kodlama Etkinlikleri ve Kodlamaya İlişkin Öz-yeterlik Algısı...	73
5.1.3. Robotik Kodlama Etkinliklerine İlişkin Öğrenci Yaşantıları.....	75
5.1.3.1. Robotik Kodlama Etkinliklerini Eğlenceli Bulmaya İlişkin Etkinlik Algısı	77

5.1.3.2. Robotik Kodlama Etkinliklerinin Kişisel Gelişime Etkisine İlişkin Etkinlik Algısı	78
5.1.3.3. Etkinliği Yapmaya İlişkin İsteklilik ve Etkinliği İlgi Çekici Bulmaya İlişkin Etkinlik Algısı	79
5.1.3.4. Öğrenci Yaşantılarına İlişkin Genel Değerlendirme	80
5.1.4. Öneriler	81
5.1.4.1. Araştırmacılara Öneriler	81
5.1.4.2. Robotik Kodlama Etkinliği Düzenleyeceklere Öneriler	82
6. KAYNAKLAR	84
7. EKLER	88
Ek 1: Tez Çalışması Etik Komisyon İzin Muafiyeti Formu	88
Ek 2: Tez Çalışması Orijinallik Raporu	89
Ek 3: Örnek Robotik Kodlama Etkinlik Planı	91
Ek 4: Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği	94
Ek 5: Etkinlik Algısı Ölçeği	99
Ek 6: Etkinliği Eğlenceli Bulmaya İlişkin Yarı Yapılandırılmış Görüşme Soruları	101
Ek 7: Etkinliklerin Kişisel Gelişime Etkisine İlişkin Yarı Yapılandırılmış Görüşme Soruları	101
8. ÖZGEÇMİŞ	102

TABLolar DİZİNİ

Tablo 3.1: Ölçek Geliştirme Aşaması Araştırma Grubuna Ait Betimsel Veriler	33
Tablo 3.2: Robotik Kodlama Eğitimi Araştırma Grubuna Ait Betimsel Veriler	34
Tablo 3.3: Robotik Kodlama Etkinlikleri ile Kazandırılması Planlanan Kazanımlar	38
Tablo 3.4: KMO and Bartlett Test Sonuçları	44
Tablo 3.5: Madde Toplam Korelasyon Değerleri	45
Tablo 3.6: Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği Temel Bileşenler Analizi Sonuçları	48
Tablo 3.7: Birinci Düzey Doğrulayıcı Faktör Modeli Uyum İndeksleri	49
Tablo 3.8: Faktörlere Ait Güvenirlik Katsayıları ve Madde Toplam Korelasyonları	51
Tablo 3.9: Demografik Verilere İlişkin Ortalama ve Standart Sapmalar	52
Tablo 4.1: Çeşitli Değişkenler Açısından Blok Temelli Programlamaya İlişkin Öz- yeterlik Algısı Ön Test Puanları	57
Tablo 4.2: Robotik Kodlama Eğitimi Sonunda Kodlamaya İlişkin Öz-yeterlik Algıları Grup İçi Farklılaşma Tablosu.....	58
Tablo 4.3: Robotik Kodlama Eğitimi Sonunda Kodlamaya İlişkin Öz-yeterlik Algılarının Grup İçi Çeşitli Değişkenlere Göre Analiz Tablosu	60
Tablo 4.4: Robotik Kodlama Etkinliklerine İlişkin Bireylerin Etkinlik Algısı Puanları Ortalama ve Standart Sapma Değerleri Tablosu	63
Tablo 4.5: Etkinlik Algısı Ölçeğinin Etkinliği Eğlenceli Bulmaya İlişkin Maddelerine Ait Ortalama ve Standart Sapma Değerleri.....	64
Tablo 4.6: Etkinlik Algısı Ölçeğinin Etkinliğin Kişisel Gelişime Etkisine İlişkin Maddelerine Ait Ortalama ve Standart Sapma Değerleri	67
Tablo 4.7: Etkinlik Algısı Ölçeğinin Etkinliği Yapma İsteğine ve Etkinliği İlgi Çekici Bulmaya İlişkin Maddelerine Ait Ortalama ve Standart Sapma Değerleri	69

ŞEKİLLER DİZİNİ

Şekil 1: Scratch Programına Ait Ekran Görüntüleri	15
Şekil 2: Scratch For Arduino Programına Ait Ekran Görüntüleri	15
Şekil 3: Kodlamaya İlişkin Öz-yeterlik Algısı Ölçeği Geliştirme Süreci	30
Şekil 4: Robotik Kodlama Etkinlikleri Araştırma Modeli.....	32
Şekil 5: Etkinler İçin Hazırlanan Robotik Kodlama Eğitim Seti	32
Şekil 6: Robotik Kodlama Etkinliği Esnasında Öğrenciler	37
Şekil 7: Birinci Düzey Doğrulayıcı Faktör Analizi Bağlantı Diyagramı (standart katsayılar).....	50

SİMGELER VE KISALTMALAR DİZİNİ

- TTKB** : Talim Terbiye Kurulu Başkanlığı
- MEB** : Millî Eğitim Bakanlığı
- ISTE** : International Society for Technology in Education (Uluslararası Eğitimde Teknoloji Topluluğu)
- CSTA** : Computer Science Teachers Association (Bilgisayar Bilimi Eğitimcileri Topluluğu)
- A-12** : Anasından 12. Sınıfa Kadar Olan Eğitim Dönemi
- A-8** : Anasınıfından 8. Sınıfa Kadar Olan Eğitim Dönemi
- A-6** : Anasınıfından 6. Sınıfa Kadar Olan Eğitim Dönemi
- RMSEA** : Yaklaşık Hataların Ortalama Karekökü (Root Mean Square Error of Approximation)
- S-RMR** : Standardize Edilmiş Hataların Ortalama Karelerinin Karekökü (Standardized Root Mean Square Residual)
- CFI** : Karşılaştırmalı Uyum İndeksi (Comparative Fit Index)
- GFI** : İyilik Uyum İndeksi (Goodness of Fit Index)
- NNFI** : Normlaştırılmamış Uyum indeksi (Non-normed Fit Index)
- IFI** : Artan Uyum indeksi (Incremental Fit Index)
- f** : Frekans
- α** : Güvenirlik Katsayısı
- χ^2** : Kay Kare Katsayısı
- d** : Etki Büyüklüğü (Cohen's d)
- N** : Örneklem Büyüklüğü
- \bar{X}** : Ortalama
- SS** : Standart Sapma
- df** : Serbestlik Derecesi
- p** : Anlamlılık Düzeyi

1. GİRİŞ

Prensky (2001), yeni nesil bireyleri, dijital teknolojilere yatkınlıklarının açık şekilde görülmesi nedeniyle “dijital yerliler” olarak adlandırmaktadır. Resnick ve diğerleri’ne (2009) göre, bu yeni nesil gençler rahat bir şekilde mesaj gönderebiliyor, çevrimiçi oyunlar oynayabiliyor ve internette gezinebiliyorlar. Ancak, her zaman dijital medya ile bu kadar etkileşim içinde olan gençlerin ancak çok azı kendi oyunlarını, animasyonlarını ve simülasyonlarını yapabiliyorlar. Başka bir ifadeyle, “okuyabiliyor” ancak “yazamıyorlar”. Resnick ve diğerleri ’ne (2009) göre dijital yatkınlık yalnızca sohbet (bu günkü karşılığı sosyal medya kullanımı olabilir), internette araştırma yapabilme ve etkileşim değil, tasarım, oluşturma-yaratma ve icat etme yetkinliği gerektirmektedir.

Yeni nesillere yazabilme becerileri ya da daha geniş kapsamda tasarım, oluşturma-yaratma ve icat etme gibi yetkinlikleri kazandırabilmek amacı ile bazı girişimlerin oluşmaya başladığına tanıklık etmekteyiz. Örneğin, Alice, kodu game lab, code.org ve Scratch gibi görsel blok temelli programlama platformları, erken yaştaki öğrencilerin geleneksel programlama dillerinin karmaşık kod yapılarını öğrenmelerine gerek kalmadan, kendi programlarını, oyun ve animasyonlarını geliştirebilmelerine ortam sağlamaktadır (Resnick ve diğerleri, 2009).

Bu süreçte ön planda tartışılan ve önemine her ortamda atıflar yapılan becerilerden birisi de kodlama becerisidir. Kodlama, bireylerin matematiksel ve kompütasyonel becerilerini (değişken ve koşul ifadeleri gibi) geliştirmekle beraber, bunun yanında bireyler kodlama yaparken problem çözmeye ilişkin öğrenme stratejilerini, projelerini tasarlamayı ve fikirler arasında bağlantılar kurmayı öğrenmektedirler. Bu beceriler yalnızca bilgisayar uzmanlarının değil her yaştan, uğraştan ve meslekten herkes için gerekli becerilerdir (Wing, 2006; Resnick, 2013).

Akpınar ve Altun (2014), temel eğitimde kazandırılacak bilgisayar ve temel medya okuryazarlığı içeriği ve becerileriyle bireylerin bilgi teknolojileri konusunda tipik tüketiciler olma kültürünün geliştirildiğini, bunun ancak büyük ticari sektörün bir talebi olabileceğini, eğitim camiasının bu bakış açısıyla bakamayacağını ifade etmektedirler. Buna ilaveten, yazarlar, ilköğretim 5. sınıftan itibaren algoritma, programlama, veri tabanı gibi kavramların temel eğitim kapsamında kazandırılması gereken bilgi ve beceriler olarak kabul edilmesi gerektiğine vurgu yapmaktadırlar.

Prensky'nin (2001) "dijital yerliler" olarak isimlendirdiği yeni neslin, Resnick ve diğerlerinin (2009) isimlendirmesiyle "dijital yatkınlık" özellikleri yakın zamanda teknolojinin inanılmaz evrimi ile daha fazla önem kazanmış, ekonomistlerin geleceğe yönelik öngöruları ve sanayi sektörünün talepleri doğrultusunda eğitim sektörünün de daha fazla dikkatini çekmeye başlamıştır. Son yıllarda hızla gelişen teknolojinin beraberinde getirdiği ve her geçen gün kendini hissettirmeye başlayan Endüstri 4.0 ile birlikte 3 boyutlu ve 4 boyutlu yazıcı teknolojileri, akıllı cihazlar, nesnelerin interneti gibi gelişmelerin günlük hayatımızı derinden etkilemeye başladığı görülmektedir. Programlı işlevleri yerine getirmek için zamanla malzeme özelliklerine dayalı olarak değişebilen, kendi kendini inşa edebilen çıktılar veren yazıcı teknolojisine 4 boyutlu yazıcı denilmektedir (Stratasys - 3D Printing & Additive Manufacturing, 2016). Çok yeni bir teknoloji olan 4 boyutlu yazıcı teknolojisinin, üretim sektörünü bambaşka bir boyuta taşıyacağı düşünülmektedir. Teknolojideki bu gelişmeler neticesinde mevcut sektörlerde hizmet veren birçok çalışana yakın zamanda ihtiyaç kalmayacağı son zamanlarda sık sık dile getirilmektedir. BMW Grup Katmanlı Üretim Merkezi Başkanı Jens Ertel'in (2016) 3 boyutlu yazıcılarla parça üretimi konusunda "Bu süreçte ilk olarak prototipleme yapılabilir ama uzun vadede bunu kitle üretimine doğru genişletmek istiyoruz" şeklindeki açıklaması gelecekte iş hayatında bireylerin sahip olması gereken yeterlikler hakkında ipucu vermektedir. Nesnelerin interneti ile cihazlar arası veri transferi ve yorumlaması sayesinde insan odaklı sistemlerin büyük çoğunluğunun yerini akıllı cihazlara bırakacağı bir gerçeklik olarak tüm ekonomi sektörünü ve eğitim camiasını ilgilendirmektedir. Son 200 yıllık sanayi toplumunun eğitim sistemi dönüşüme uğramakta, 21. yüzyıl becerilerine sahip bireyler yetiştirmeye yönelik eğitim sistemi geliştirme girişimleri artmaktadır. Günlük yaşam problemlerine çözüm bulmaya yönelik disiplinler arası somut uygulamaya dönük etkinlikler içeren, Science (Fen), Technology (Teknoloji), Engineering (Mühendislik) ve Mathematics (Matematik) kelimelerinin baş harflerinden oluşan STEM eğitimi girişimleri (farklı disiplinler içerisinde çeşitli boyutlarda kodlama, robotik kodlama ve akıllı cihaz tasarımı etkinliklerini de barındırmaktadır), kodlama eğitime ve robotik kodlama eğitime yönelik girişimler, Finlandiya eğitim sistemindeki 2016 yılı sonunda yapılan köklü değişiklik bunlara örnek olarak gösterilebilir. 2016 Ekim ayı içerisinde Amerika Birleşik Devletleri'ne yönelik gerçekleştirilen ve on milyonlarca internet sitesini etkileyen siber saldırıda güvenlik kameralarından akıllı ev cihazlarına kadar

internet ağına bağlı akıllı cihazların siber saldırgan olarak kullanıldığı tespit edilmiş, akıllı cihazların ve nesnelerin internetinin bugünden hayatımıza ne düzeyde dahil olduğu bu örnek gelişmeyle dikkatleri çekmiştir (bkz. BBC, 2016). Gelecek dünya konjonktüründe söz sahibi olmak isteyen ülkeler Endüstri 4.0'ın beraberinde getirdiği büyük veri ve nesnelerin interneti, 3D-4D baskı teknolojileri ve akıllı cihaz teknolojileri gibi gelişmelere uyum sağlayabilecek şekilde üretim yetisini kazanmış insan kaynağına sahip olmak zorundadırlar. Aksi takdirde buna uyum sağlayamayan toplumlar günümüzün mevcut seri üretim yapan sistemlerinin âtil duruma düşmesi ile işlevsiz toplumlar olarak büyük zorluklarla karşılaşacaklardır. 2013 Nobel Ekonomi Ödülü Sahibi olan Robert J. Shiller (Yale Üniversitesi Ekonomi Profesörü), 2016 Davos Dünya Ekonomik Forumunda: “Yangın sigortası satın almak için bir evin yanıp kül olmasını bekleyemezsiniz. Dördüncü Sanayi Devrimi'ne hazırlanmak için toplumumuzun altüst olmasını bekleyemeyiz.” diyerek değişimin ne kadar ciddi sonuçlar doğuracak boyutlarda olduğuna dikkat çekmekte ve bu konuda adım atılması için uyarmaktadır (Shiller, 2016).

Endüstri 4.0'ın beraberinde getirmekte olduğu değişiklikler 21. yüzyıl toplumunda bireylerin sahip olması gereken becerileri de belirlemektedir. Hızla gelişen ve değişen geleceğin dünyasına yönelik öngörüler, bireylerde bulunması gerekli 21. yüzyıl becerilerine yönelik çalışmaların artmasına neden olmaktadır. Söz konusu yeterliklere yönelik yapılan çalışmalardan en dikkat çeken Uluslararası Eğitimde Teknoloji Topluluğu'nun (ISTE) 2007 ve 2016 yıllarında hazırladığı raporlardır. Bu raporlarda “kompütasyonel düşünme” ve “programlama becerisi” bireylerde bulunması gereken 21. yüzyıl becerileri arasında sayılmaktadır. Mantıksal akıl yürütmenin bir parçası olarak kodlama becerisi Avrupa Komisyonunun hazırladığı raporda da “21. yüzyıl becerileri” arasında sayılmaktadır (European Commission, 2016). Amerikan İşgücü İstatistik Bürosunun 2012'den 2022'ye istihdam raporuna göre gelecekteki iş gücünün en temel becerileri arasında da “kompütasyonel düşünme” yer alırken, hazırlanan çeşitli raporlarda gelecekte yeri doldurulamayacak boyutta BT Uzmanı açığı olacağı öngörülmektedir (bkz. United States Bureau of Labor Statistics, 2013; Bidwell, 2013).

Bu gelişmeler kompütasyonel düşünme kazandırmaya ve programlama eğitimi vermeye yönelik bilgisayar bilimi öğretim programı geliştirme çalışmalarını arttırmıştır (Angeli ve diğerleri, 2016). 2016 yılı itibariyle Avusturya, Arjantin,

Uruguay, Kanada, Hindistan, İtalya, İngiltere, Güney Kore, Suudi Arabistan, Malezya, Danimarka, Estonya, Finlandiya, Almanya, Polonya “Bilgisayar Bilimi” eğitimi kapsamında kodlama eğitimini öğretim programlarına dâhil etmişlerdir veya bu yönde çalışma sürdürmektedirler (Bkz.Code.org 2015 Annual Report, 2015).

Robotik kodlama uygulamaları, soyut bir işlem gerektiren yazılım süreçlerini somutlaştırmakta ve öğrencilere yazmış oldukları kodların derlendikten sonra bir donanım ile nasıl çalışabildiğini doğrudan gözleme olanağı sağlamaktadır. Bu nedenle, birçok eğitimci, programlama öğretimini bu tür donanımsal desteklerle çeşitlendirme yolunu tercih etmektedir. 21. yüzyıl becerileri arasında sayılan kompütasyonel düşünme becerisine katkı sağladığı bilinen kodlama eğitimine robotik donanımların dahil edilmesi ile düzenlenen robotik kodlama etkinlikleri sayesinde hem akıllı cihazların hem de nesnelerin internetinin çalışma mantığının somut uygulamalarla anlaşılabilirliği bir öğretim ortamı sunulmaktadır.

Allsop (2015), İngiltere ve diğer Avrupa ülkeleri ile kıyasladığında, Türkiye’nin yetişmiş Bilişim Teknolojileri Öğretmeni potansiyeli ile programlama eğitiminde bu ülkelere göre avantajlı durumda olduğunu belirtmektedir. Avrupa Kodlama Girişimi’nin (The European Coding Initiative, 2014) Avrupa Birliği üye ülkelerinde kodlama eğitimi verebilecek yeterlikte öğretmen açığı bulunması nedeniyle öğretmenlere yönelik kodlama eğitimleri düzenlemesi de bu saptamayı destekleyici örnek bir girişimdir. Allsop’un (2015) Türkiye’nin kodlama eğitimi konusunda diğer ülkelere göre avantajlı olduğunu belirtmesinin sebebi, Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümlerinden mezun olan bilişim teknolojileri öğretmenlerinin sayısının çokluğudur. Mezun öğretmenlerden ataması yapılmış olanların öğretim programlarında yer almamasından dolayı yıllarca programlama eğitimi vermemiş olması ve bu nedenle programlama yeterliklerinin yıllar içerisinde ne düzeyde gerilediğinin bilinmemesi, ülkemizin bu anlamda avantajlı konumda olduğu iddiasını tartışmalı hale getirmektedir. Ülkemizde kodlama eğitimi ve robotik kodlama eğitimi konusunda gerçekleştirilen çalışmaların ve projelerin azlığı, akademik araştırma kapsamında da kodlama eğitimi konusunda çok yetersiz bir konumda olduğumuzu ortaya koymaktadır (bkz. Akpınar & Altun, 2014; Yükseltürk & Altıok, 2016; Bilişim Garaj Akademisi, 2016).

Ülkemizde kodlama eğitimi kapsamında yürütülen çalışmalar neticesinde “Problem Analiz ve Çözme Yaklaşımları”, “Algoritma ve Strateji Geliştirme (algoritma

oluşturma mantığı, sözde kod, akış şemaları vb.)”, “Programlama”, “Yazılım Projesi Geliştirme, Uygulama ve Yaygınlaştırma” konu başlıkları altında kodlama eğitimi 2012 yılında Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programına dâhil edilmiştir (TTKB, 2012). Lise düzeyinde ise 2016 yılı itibariyle Bilgisayar Bilimi Dersi kapsamında “Problem Çözme ve Algoritmalar”, “Programlama”, “Web Tabanlı Programlama”, “Mobil Programlama” konu başlıklarının yer aldığı öğretim programı hazırlanmıştır (TTKB, 2016). 2017 yılı ocak ayı itibariyle yayınlanan bilişim teknolojileri ve yazılım dersi taslak öğretim programında ise 1. sınıftan 8. sınıfa kadar tüm sınıf düzeylerinde öğretim programlarının %44’ü “Hesaplamalı Düşünme” ünitesi altında “problem çözme kavramları ve yaklaşımları” ve “programlama” konu başlıklarını içermektedir. Ünite başlığında bulunan “computational thinking” teriminin Türkçeleştirilmiş hali olan “hesaplamalı düşünme” ifadesi çalışmamız kapsamında tercih edilen “kompütasyonel düşünme” ifadesine karşılık gelmektedir. 1. sınıftan 4. sınıfa kadar serbest etkinlik saatlerinde işlenip işlenmemesi öğretmen inisiyatifine bırakılan bilişim teknolojileri ve yazılım dersinin, 5. ve 6. sınıflarda haftada 2 saat zorunlu, 7. ve 8. sınıflarda ise haftada 2 saat seçmeli olarak yürütülmesi planlanmaktadır (TTKB, 2017).

1.1. Araştırmanın Önemi

Papert’in (1980) ilk çalışmalarından code.org, Scratch gibi günümüz çalışmalarına kadar olan süreçte kodlama eğitiminin ilköğretim düzeyinde verilebilecek şekilde düzenlenmesi yönünde girişimlerin başarılı sonuçlar vermeye başladığından ve birçok ülkede kodlama eğitiminin öğretim programlarına dâhil edildiğinden önceki bölümde değinilmişti. Ülkemizde de kodlama eğitimi ortaokullarda 2012 yılında Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programına dâhil edilmiştir (TTKB, 2012). Millî Eğitim Bakanlığı’nın programlama eğitimine yönelik içerik ve ders saati bağlamında çalışmalar yaptığı hakkında basın ve yayın organlarında açıklamalar mevcuttur. Bu kapsamda gerçekleştirilen çalışmalardan bir tanesi olan liseler için bilgisayar bilimi öğretim programı geliştirme çalışması 2016 yılı Ekim ayı itibariyle tamamlanmış, Talim Terbiye Kurulu Başkanlığı tarafından liseler için “Bilgisayar Bilimi Dersi Öğretim Programı” yayımlanmıştır. Öğretim programı incelendiğinde kodlama ve robotik kodlama kazanımlarının yer aldığı görülmektedir (TTKB, 2016). 2017 yılı ocak ayı itibariyle taslak hali yayınlanan 1. sınıftan 8. sınıfa kadar bilişim teknolojileri ve yazılım dersi taslak öğretim programında da tüm sınıf düzeylerinde

“hesaplamalı düşünme” ünitesi altında “problem çözme kavramları ve yaklaşımları” ve “programlama” konu başlıkları bulunmaktadır (TTKB, 2017). Ülkemizde A-12 düzeyinde programlama eğitimini konu alan çalışmalara yönelik yapılan alanyazın taraması neticesinde, ülkemizde bu anlamda yeterince çalışma yapılmadığı görülmektedir. MIT laboratuvarlarında geliştirilen ve ülkemizde ilköğretim düzeyinde programlama eğitiminde yaygın olarak kullanılan Scratch yazılımının öğrenciler üzerinde problem çözme becerisi, öz-yeterlik, motivasyon, ilgi artışı gibi etkilerine bakan çalışmalar yapıldığı, ancak bunun sınırlı sayıda olduğu görülmüştür (bkz. Genç & Karakuş, 2011; Gülbahar & Kalelioğlu, 2014; Kukul & Gökçearslan, 2014; Yükseltürk & Altıok, 2016).

Programlama kompütasyonel düşünme anlamına gelmemekle birlikte, birçok araştırmacı ve bu konuda çalışan bilim insanı, programlama eğitiminin bireylerdeki kompütasyonel düşünme becerilerine katkı yaptığı düşüncesindedir (bkz. Wing, 2006; Settle & Perkovic, 2010; Gülbahar & Kalelioğlu, 2014; Allsop, 2015; Angeli ve diğerleri, 2016). 21. yüzyıl becerileri arasında sayılan kompütasyonel düşünme becerisini artırdığı bilinen programlama eğitiminin verimliliğini artırmaya yönelik yapılan araştırmalar her geçen gün daha fazla önem kazanmaktadır.

21. yüzyıl becerileri arasında sayılan “kompütasyonel düşünme” ve “programlama” becerisine katkı sağladığı bilinen kodlama etkinliklerini içerisinde barındıran, nesnelerin interneti ve akıllı cihazların çalışma mantığının somut olarak görülebildiği Robotik kodlama etkinliklerinin öğrencilerimize sunulması oldukça önem taşımaktadır. Ülkemizde robotik kodlama etkinliklerine ilişkin uygulama örnekleri henüz çok yeni başlamıştır. Son birkaç yıldır ders dışı etkinlikler tarzında hafta sonu kursları ve maker kulüpleri gibi ortamlarda robotik kodlama etkinliklerinin gerçekleştirildiği görülmektedir. Ancak ülkemizde A-12 düzeyinde robotik kodlama etkinliklerine yönelik alanyazında yeterince araştırma bulgusuna rastlanılmamıştır. TTKB (2016) bilgisayar bilimi öğretim programında kodlama ve robotik kodlamaya ilişkin konu başlıkları olması, robotik kodlamaya ilişkin yapılacak akademik araştırmaların önemini artırmaktadır.

A-12 düzeyinde kodlama etkinliklerine ilişkin yapılan araştırmalarla kıyaslandığında, blok temelli programlama ile hazırlanmış robotik kodlama etkinliklerine yönelik yapılan çalışmalar daha da sınırlıdır. Alanyazın tarandığında;

- A-12 düzeyinde robotik kodlama etkinliklerinin bireylerin blok temelli programlamaya ilişkin öz-yeterlik algılarına etkisine yönelik
- Bireylerin robotik kodlama etkinliklerine ilişkin yaşantılarını tespit etmeye yönelik yeterince bilimsel veri bulunmadığı görülmüştür.

Dolayısı ile, bu çalışma kapsamında robotik kodlama etkinliklerinin bireylerin blok temelli programlamaya ilişkin öz-yeterlik algılarında farklılaşma meydana getirip getirmediği incelenmiştir. Robotik Kodlama Etkinliklerine ilişkin öğrenci yaşantıları tespit edilmiş, robotik kodlama etkinliklerinin düzenlenmesi sürecinde dikkat edilmesi gerekenlere ilişkin bulgulara yer verilmiştir. Yapılan bu çalışmanın, robotik kodlama etkinliklerine ilişkin çalışma yapacak diğer araştırmacılara kaynaklık etmesi amaçlanmıştır, ayrıca çalışma kapsamında geliştirilen ve alanyazında rastlanılmayan “Blok Temelli Programlamaya İlişkin Öz-yeterlik algısı Ölçeği” ile alan yazına katkı sunulması amaçlanmıştır. Sonuç olarak, çalışmadan elde edilen bulgularla A-12 düzeyinde kodlama eğitimi ve robotik kodlama eğitimi verilmesi yönünde harcanan zaman ve emeğin doğru yönlendirilmesine katkı sağlanması amaçlanmaktadır.

1.2. Araştırma Problemi

Ortaokul düzeyinde robotik kodlama eğitimi ile öğrencilerin kodlamaya ilişkin öz-yeterlik algıları değişmekte midir? Öğrencilerin Robotik kodlama etkinliklerine ilişkin yaşantıları nasıldır?

1.3. Araştırma Soruları

1. Ortaokul 5. sınıf düzeyinde uygulanan robotik kodlama eğitimi öğrencilerin kodlamaya ilişkin öz-yeterlik algılarında anlamlı bir değişim oluşturmaktadır mıdır?
2. Ortaokul 5. sınıf düzeyinde uygulanan robotik kodlama eğitimi öğrencilerin kodlamaya ilişkin öz-yeterlik algılarında çeşitli değişkenlere göre anlamlı bir değişim oluşturmaktadır mıdır?
 - a. Cinsiyete göre anlamlı bir değişim oluşturmaktadır mıdır?
 - b. Evde bilgisayar sahipliğine göre anlamlı bir değişim oluşturmaktadır mıdır?

- c. Evde internet bağlantısı sahipliğine göre anlamlı bir değişim oluşturmakta mıdır?
- d. Ders dışında Scratch programına çalışma olanağına göre anlamlı bir fark oluşturmakta mıdır?

3. Öğrencilerin robotik kodlama etkinliklerine ilişkin yaşantıları nasıldır?

1.4. Sınırlılıklar

Yapılan çalışma sosyoekonomik ve sosyokültürel olarak benzer ailelerden gelen öğrenciler ile yapılmıştır. Düşük gelir düzeyinden ve alt sosyokültürel çevreden ailelerin dezavantajlı sayılabilecek gruptaki çocuklarıyla uygulama yapılmıştır. Farklı sosyoekonomik ve sosyokültürel özellikler taşıyan bir örneklem grubunda sonuçlar farklılık gösterebilir. Örneklem grubu ortaokul 5. sınıf öğrencileri ile sınırlıdır. Robotik kodlama etkinlikleri fen bilimlerinin elektrik-elektronik konularını da barındırmakla beraber, öğrenciler bu konularda daha önce yeterince eğitim almamışlardır. Bu konulara yönelik kazanımlar fen bilimleri öğretmenlerinin bilgi ve kaynak desteği alınarak uygulamayı gerçekleştiren bilişim teknolojileri öğretmeni tarafından derse hazırlık sürecinde ön bilgi düzeyinde yüzeysel olarak anlatılmıştır. Disiplinler arası iş birliği halinde yürütülen etkinliklerle çok daha zengin bir öğrenme ortamı oluşturulabilir. Robotik kodlama etkinlikleri Arduino Uno R3 kart ve Arduino'ya bağlanan sensörler ve aktüatörler ile yürütülmüştür. Farklı robotik kitleler ile sonuçlar farklılık gösterebilir. Programlama platformu olarak Scratch for Arduino kullanılmıştır, farklı blok temelli robotik programlama platformlarında sonuçlar farklılık gösterebilir.

1.5. Tanımlar

Programlama: Bilgisayarda gerçekleştirilecek bir işlem veya bir problem çözümü için gerekli olan bilgiler, bilgisayar dilinde komutlara çevrilir. Bu komutların bir araya getirilip derlenmesi ve çalıştırılması sonucunda meydana gelen işlem algoritmasına, programlama denir (Kesici ve Kocabaş, 2007).

Programlama Performansı: Herhangi bir programlama dilini kullanarak verilen bir programlama görevini başarı ile yerine getirme, programı anlama, program hatalarını ayıklayabilme, yeniden bir program yazabilme, programın algoritma şemasını oluşturabilme becerisi olarak alınmıştır (Altun ve Mazman, 2013).

Blok Temelli Programlama (Kodlama): Komut bloklarını bir paletten komut dosyası bölmesine sürükleyerek ve yapboz parçaları gibi "blok yığınları" oluşturmak için bu komut bloklarını birleştirerek yapılan programlama türüdür (Maloney, Peppler, Kafai, Resnick, & Rusk, 2008). Blok Temelli Programlama ve Kodlama bu çalışma kapsamında eşanlı kullanılabilecektir.

Öz-yeterlik (Self-efficacy): Bireyin, belli bir performansı göstermek için gerekli etkinlikleri organize edip, başarılı olarak yapma kapasitesi hakkında kendine ilişkin yargısıdır (Bandura, 1994).

Öz-yeterlik Algısı Düzeyi: Bireylerin öz-yeterlik algısı ölçeğinden aldıkları puandır.

Programlama Öz Yeterliliği: Bireyin bir programlama görevini yerine getirmede kendi yeterliliklerine ilişkin yargısı, kendi başarısına ilişkin inancıdır.

Basit Blok Temelli Programlama: Blok temelli programlamaya ilişkin kavramlardan, programlama bilgisi olan bir danışman olmadan, bireyin kendi kendine kolayca keşfedebileceği, blok temelli programlamaya ilişkin performans değerlendirmelerinde katılımcıların büyük çoğunluğunun daha yüksek puan aldıkları blok temelli programlamaya ilişkin kavramlar olarak kabul edilmiştir (bkz. Maloney, Peppler, Kafai, Resnick, & Rusk, 2008; Altun ve Mazman, 2013).

Karmaşık Blok Temelli Programlama: Blok temelli programlamaya ilişkin kavramlardan, programlama bilgisi olan bir danışman olmadan, bireyin kendi kendine kolayca keşfedemediği, blok temelli programlamaya ilişkin performans değerlendirmelerinde katılımcıların büyük çoğunluğunun daha düşük puan aldıkları blok temelli programlamaya ilişkin kavramlar olarak kabul edilmiştir (bkz. Maloney, Peppler, Kafai, Resnick, & Rusk, 2008; Altun ve Mazman, 2013).

Basit Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Düzeyi: Blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeğinin basit blok temelli programlamaya ilişkin öz-yeterlik algısı alt faktöründen alınan puandır.

Karmaşık Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Düzeyi: Blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeğinin karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı alt faktöründen alınan puandır.

Kompütasyonel Düşünme (Computational thinking): Soyutlama, genelleme, parçalara ayırma, algoritmik düşünme ve hata ayıklama (hataların tespit edilmesi)

ve düzeltilmesi) unsurlarını kullanabilecek bir düşünce sürecidir (Angeli ve diğerleri, 2016).

2. İLGİLİ ALAN YAZIN

2.1. Kompütasyonel Düşünme

Alanyazın taraması sonucunda “computational thinking” teriminin farklı Türkçe kullanımlarına rastlanılmıştır. Hesaplamalı düşünme, bilişimsel düşünme, bilgi-işlemsel düşünme, işlemsel düşünme, bilgisayarca düşünme, kompütasyonel düşünme en çok karşımıza çıkan kullanımlardır. Bu çalışma kapsamında “computational thinking” teriminin kompütasyonel düşünme olarak ifade edilmesine karar verilmiştir.

Çağımızda kompütasyonel düşünme sadece bilgisayar mühendislerinin değil tüm bireylerin günlük yaşamda karşılaştıkları problemleri çözme yeteneklerini geliştirmelerinde etkin rol oynamaktadır. 21. yüzyıl becerileri arasında sayılan ve gelecekteki iş gücü becerileri arasında olması gerektiği belirtilen kompütasyonel düşünmenin A-12 eğitim düzeyindeki izleri Seymour Papert’in (1980) çalışmasına kadar dayanmaktadır. Papert’e (1980) göre, öğrenciler programlama yaparken, önce verilen probleme bir çözüm üretmek zorundadırlar. Daha sonra bu çözümü kullanarak bilgisayar ile iletişim kurmanın bir yolunu bulmak durumundadırlar.

Wing’in (2006) çalışması A-12 düzeyinde kompütasyonel düşünmeyi artırmaya yönelik ilginin yeniden artmasına neden olmuştur. Wing’e (2006) göre kompütasyonel düşünme öğretim programlarında kazandırılması gereken temel beceriler arasında kabul edilmelidir. Kompütasyonel düşünmeyi, bilgisayarları kullanarak problemleri formüle etme ve çözme amaçlı düşünme işlemi olarak tanımlamaktadır. Kompütasyonel düşünmenin bu tarihten sonra daha fazla gündeme gelmeye başladığı söylenebilir. Örneğin Uluslararası Eğitimde Teknoloji Topluluğu (ISTE) 2007 yılında hazırlanan “Öğrenci Standartları” raporunda, kompütasyonel düşünmenin alt maddeleri olan eleştirel düşünme, problem çözme, karar verme, yaratıcılık ve yenilikçilik gibi kavramlar ayrı ayrı yer almasına rağmen “kompütasyonel düşünme” kavram olarak yer almamaktadır. Hâlbuki ISTE 2011 yılında “Kompütasyonel Düşünmenin Operasyonel Tanımlaması” raporunu hazırlamıştır (bkz. ISTE, 2011).

2010 senesinde Amerikan Ulusal Araştırmalar Konseyi tarafından hazırlanan raporda, kompütasyonel düşünme bilgisayar alanyazınından, bilgisayar programlamadan ve diğer bilgisayar uygulamalarından (oyunlar gibi) ayrılmıştır.

Kompütasyonel düşünme tanımı, bilgisayar biliminin soyutlama, parçalara ayırma (çözümleme), örüntü üretme, görselleştirme, problem çözme ve algoritmik düşünme gibi temel kavramları ile genişletilmiştir.

Benzer bir tanımlama ile Furber (2012), kompütasyonel düşünmeyi “çevremizi saran dünyadaki bilgiyi işlemenin farklı yönlerini tanıma süreci, doğal ve yapay sistemlerin ve süreçlerin anlaşılması ve sebeplendirilmesi amacıyla bilgisayar biliminin araçlarının ve tekniklerinin kullanılması” olarak tanımlamaktadır.

Kompütasyonel düşünme hakkında yapılan çalışmalarda karşımıza çıkan en yaygın tanımlama ISTE'nin 2011 yılında hazırladığı raporda sunulan tanımlamadır. Bilgisayar Bilimi Eğitmcileri Derneği (CSTA) ve Uluslararası Eğitimde Teknoloji Topluluğu (ISTE), yükseköğretimden, endüstriden ve A-12 eğitim düzeyinde liderlerin katılımıyla, aşağıdaki maddeleri içeren ama bunlarla sınırlı olmayan bir problem çözme işlemi olarak işlevsel bir kompütasyonel düşünme tanımı geliştirmişlerdir (ISTE, 2011).

1. Problemleri bilgisayar veya başka bir araç kullanarak çözmeyi sağlayacak şekilde formüle etme
2. Veriyi mantıksal olarak organize etme ve analiz etme
3. Veriyi model veya simülasyonlar gibi soyutlaştırmalar yolu ile gösterme
4. Algoritmik düşünme yolu ile otomatik cevaplar üretme
5. Olası çözümleri, olası aşamaları ve kaynakları en etkili ve verimli şekilde birleştirerek tanımlama, analiz etme ve uygulama
6. Bu problem çözme sürecini diğer farklı problem durumlarına transfer etme ve genelleme

Özden (2015), bilgisayarca (kompütasyonel) düşünme terimini “bilgisayarları üretim amaçlı olarak hayat problemlerinin çözümünde kullanabilmek için gerekli olan bilgi, beceri ve tutumlara sahip olmak” diye tanımlamaktadır.

Angeli ve diğerleri (2016) çalışmaları kapsamında kompütasyonel düşünme konusunda yapılmış çalışmaları içeren bir alanyazın taraması yaparak kompütasyonel düşünmenin üzerinde hemfikir olunmuş tek bir tanımlaması olmadığını belirtmişlerdir. Çalışmaları kapsamında gözden geçirdikleri alanyazında kullanılan tanımlamalara dayanarak araştırmacıların, kompütasyonel düşünmenin

soyutlama, genelleme, parçalara ayırma, algoritmik düşünme ve hata ayıklama (hataların tespit edilmesi ve düzeltilmesi) unsurlarını kullanabilecek bir düşünce süreci olduğunu kabul etmeye başladıklarını belirtmişlerdir.

Amerikan İşgücü İstatistik Bürosunun 2012'den 2022'ye istihdam raporuna göre gelecekteki iş gücünün en temel becerileri arasında "kompütasyonel düşünme" ve "programlama becerisi" yer almaktadır (United States Bureau of Labor Statistics, 2013). Her yeni çağın getirdiği farklı problemlere çözüm teknikleri üretebilmek adına bireylerin yeni becerilere sahip olması beklenmektedir. Uluslararası Eğitimde Teknoloji Topluluğu (ISTE) tarafından 2016 yılında hazırlanan raporda, birbiriyle bağlantılı dijital bir dünyada öğrencilerin iyi düzeyde gelişmelerinin ve adaptasyonlarının sağlanması amacıyla öğrencilere kazandırılması gerekli beceri ve yeterlikler listelenmiştir. Bu listede, "kompütasyonel düşünme" öğrencilerde bulunması gereken 7 temel beceri ve yeterlikten birisi olarak belirtilmiştir (ISTE, 2016).

Programlama eğitimi ile kompütasyonel düşünme arasındaki bağlantı ise şu şekilde açıklanabilmektedir; programlama kompütasyonel düşünme anlamına gelmemekle birlikte, birçok araştırmacı ve bu konuda çalışan bilim insanı, programlama eğitiminin bireylerdeki kompütasyonel düşünme becerilerine katkı yaptığı düşüncesindedirler (Wing, 2006; Settle & Perkovic, 2010; ScratchEd Team, 2011; Gülbahar & Kalelioğlu, 2014; Allsop, 2015; Angeli ve diğerleri, 2016).

2.2. Blok Temelli Programlama (Kodlama) Eğitimi

Kodlama bireylerin matematiksel ve kompütasyonel becerilerini (değişken ve koşul ifadeleri gibi) geliştirmekle beraber, bunun yanında bireyler kodlama yaparken problem çözmeye ilişkin öğrenme stratejilerini, projelerini tasarlamayı ve fikirler arasında bağlantılar kurmayı öğrenmektedirler. Bu beceriler yalnızca bilgisayar uzmanlarının değil her yaştan, uğraştan ve meslekten herkes için gerekli becerilerdir (Wing, 2006; Resnick, 2013).

Balanskat ve Engelhardt'e (2014) göre, endüstri 4.0 ile birlikte yeni nesil teknolojilerin gelişiminin beraberinde getirdiği bir ihtiyaç olarak programlama becerisi birçok sektördeki çalışanlar için önemli bir yetenek haline gelmiştir. Bu nedenle bilgisayar programlama becerisi 21. yüzyıl becerileri arasında yer almaktadır.

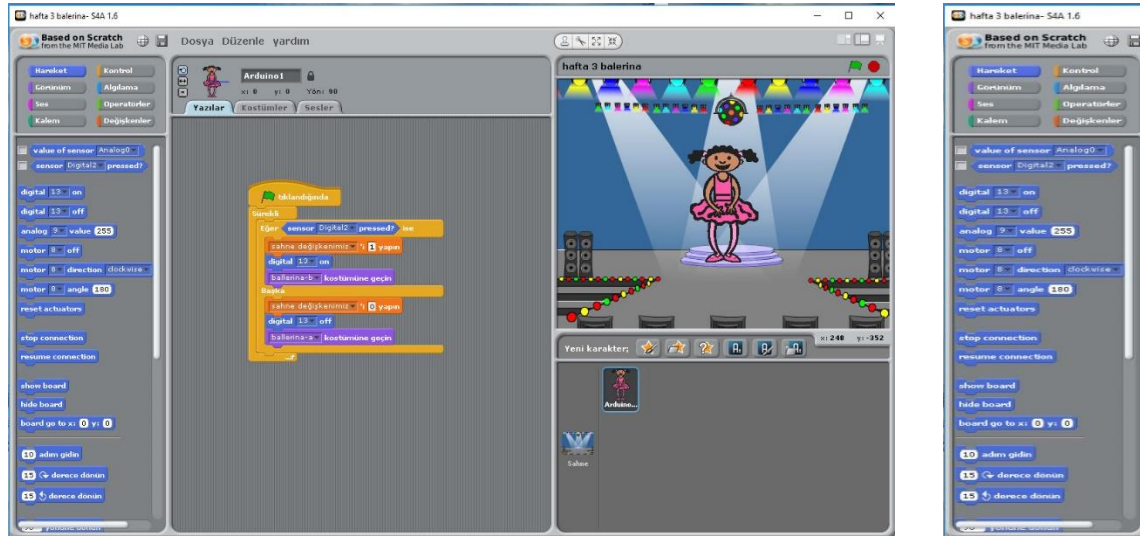
Altun ve Akpınar (2015), programlama eğitiminin bireylerin bilişsel becerilerine katkısına ilişkin aşağıdaki saptamalarda bulunmaktadır:

“Programlama bir probleme çözüm bulma etkinliği, analiz yetisi de geliştirebilmektedir, çünkü problemi alt parçalara ayırmak ve genellenebilir temel bir çözüm oluşturmak gerekmektedir (Saeli ve diğ., 2011). Dolayısıyla programlamayla değişik alanlardaki problem çözüme becerilerinin gelişimi de hedeftir ki programlamanın problem çözüme becerisi geliştirmeye dönük katkılarıyla ilgili bulgular kuvvetlidir (Feurzeig ve diğ.,1970; Papert, 1993; Mulder, 2002, Dasso ve diğ.,2005). Belli disiplinlerdeki kavramların daha kolay anlaşılabilmesi ve işevuruklaşabilmesi programlama yoluyla yapılabilmektedir. Örneğin Matematikteki değişken, fonksiyon, alt öğeler gibi kavramlar programlama yoluyla daha kolay anlaşılabilir. Ayrıca programlama yoluyla başka konu alanlarındaki (örneğin Fen Bilgisi) kavramların ve süreçlerin daha derinlemesine çalışılması daha mümkün olabilmektedir (bkz. Resnick ve Ocko;1990). Kavramsal bilgilerin prosedürler içinde çalışılması ve bir kavramın oluşum sürecinin incelenmesi de programlama etkinliğinin doğası gereği son derece kolayca programlama içinde yapılabilmektedir. Bu nedenle, ders tasarımcısı ve/veya öğretmeni ders etkinliklerini bilgi aktaran şekilde değil, bilgi oluşturmak şeklinde düzenlemesi yönünde yönlendirmek de kolay olabilmektedir” (s. 2-3).

Alice, kodu game lab, code.org ve Scratch gibi görsel programlama dilleri blok temelli görsel programlama yapıları, küçük yaştaki öğrencilerin geleneksel programlama dillerinin karmaşık kod yapılarını öğrenmelerine gerek kalmadan, kendi programlarını, oyun ve animasyonlarını geliştirebilmelerini sağlamaktadır (Resnick ve diğerleri, 2009). Scratch for Arduino programı, Scratch programına Arduino'ya bağlı sensörleri ve aktüatörleri yönetmek için yeni bloklar sağlayan bir Scratch modifikasyonudur (Scratch for Arduino, 2015). Bu nedenle blok temelli programlama yapısı olan Scratch'le ilgili alanyazın taramasında elde edilen bulgulara araştırmamız kapsamında yer verilmiştir. Şekil 1'de Scratch programına ait ekran görüntüsü ve hareket bloğu, Şekil 2'de ise Scratch for Arduino programına ait ekran görüntüsü ve hareket bloğu yer almaktadır. Şekillerden Scratch for Arduino programının, Scratch programının hareket bloğuna Arduino'ya bağlı sensörleri ve aktüatörleri yönetmek için yeni bloklar eklenmiş bir Scratch modifikasyonu olduğu görülmektedir.



Şekil 1: Scratch Programına Ait Ekran Görüntüleri



Şekil 2: Scratch For Arduino Programına Ait Ekran Görüntüleri

Maloney, Peppler, Kafai, Resnick ve Rusk'ın (2008) çalışmalarında Scratch, acemi programcılar için medya düzenlemeyi kolaylaştırıcı görsel, blok temelli bir programlama dili olarak tanımlanmıştır. Çalışma kapsamında 18 ay süresince 8-18 yaş arası gençlerin bir okul sonrası kulüp merkezinde herhangi bir eğitim veya deneyimli bir danışman olmadan geliştirdikleri 536 Scratch projesi analiz edilmiştir. Projelerin analizi sonucunda katılımcıların ortaya koydukları temel programlama

kavramlarının araştırılması amaç olarak belirtilmiştir. Süreç boyunca katılımcıların programlama komutlarını ve kavramlarını kullanma durumlarını analiz edilmesinin yanı sıra, kulüp üyelerinin programlama hakkındaki fikirleri ve Scratch'e yönelik algıları gözlemlenmiştir. Bu çalışmada herhangi bir formal eğitim programına bağlı olmadan informal bir ortamda acemi programcıların neler öğrenecekleri ve diğer başka yazılımlar varken gençleri programlamayı seçmeye hangi sebeplerin motive ettiği tartışılmıştır. Katılımcılara "bilgisayar programlaması senin için ne ifade ediyor?" diye sorulduğunda katılımcılar genellikle "Bilgisayar programlaması da nedir?" yanıtını vermişlerdir. Araştırmacıların ilk fark ettikleri şey, gençlerin Scratch ile programlama arasında bir bağlantı kurmadıkları olmuştur. Gençlerin Scratch'i programlama olarak görmemeleri, Scratch'i oyun, medya oluşturma aracı olarak görmelerinin, Scratch'in öğrenciler arasında rağbet görmesinde etkili olmuş olabileceği belirtilmiştir. Değişkenlerin ve rastgele numara üretme bloklarının çok kullanışlı olmasına rağmen projelerde daha az yaygın kullanıldığı, fakat zamanla artış gösterdiği de bulgular arasında sayılmıştır. Bunun sebebi olarak da çalışma kapsamında katılımcılara formal bir eğitimin verilmemiş olması, danışmanlarında daha önceden programlama deneyimlerinin olmaması gösterilmektedir. Bu kavramların programlama bilgisi olan bir danışman olmadan kolayca keşfedilemeyeceği de araştırma kapsamında belirtilmiştir.

Çatlak, Tekdal ve Baz (2015), Scratch yazılımının programlama öğretiminde kullanımına yönelik yapılan mevcut çalışmaları araştırmak üzere doküman inceleme yöntemi kullanılarak bir alanyazın taraması yapmışlardır. Bu kapsamda ulaşılan 53 makalenin 20 tanesi Korece ve 1 tanesi de İspanyolca olması nedeniyle çalışmaya dâhil edilmemiş, 32 makale ile çalışma sürdürülmüştür. Araştırmanın bulgularına göre, Scratch yazılımının programlama öğretiminde etkili olduğu ve programlama öğrenimini zevkli ve daha anlaşılır hale getirdiği belirtilmiştir. Ayrıca programlama derslerine, özünde oyun teması barındıran Scratch yazılımı ile başlamanın katılımcıların derse olan ilgi ve motivasyon değişkenleri üzerinde olumlu etkilerine dair bulgulara ulaşıldığı da belirtilmiştir. Bu durumun gerekçeleri olarak, Scratch programlama ortamının kodlama bilgisi gerektirmeyen kullanıcı dostu arayüze sahip olması, kodlama yerine kod bloklarını sürükle - bırak yöntemi ile basit seviyede kullanım sunması, kullanıcıların kendilerine özgü materyaller geliştirebilmelerine olanak sağlaması ve programlama dillerini öğrenmek isteyen

her seviyede kullanıcıya kullanım imkânı sağlamasının gösterilebileceği de bulgular arasında sıralanmıştır.

Yükseltürk ve Altıok (2016) araştırmaları kapsamında 2013-2014 öğretim yılında Kırıkkale Üniversitesi Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümünde bir seçmeli ders kapsamında 169 Bilişim Teknolojileri Öğretmen adayından özgürce bir eğitsel oyun konusu seçmeleri, seçtikleri eğitsel oyun konusunu bireysel olarak “Scratch” görsel programlama aracını kullanarak hazırlamalarını istemişlerdir. Sonuç olarak oyunlar hedef kitlesi, konusu ve oyunun sahip olması gereken özelliklere göre belirlenen değerlendirme puanına göre değerlendirildiğinde oyunların kalitesinin “Orta veya İyi” olduğu görülmüştür. Scratch görsel programlama aracının seçilmesinde sebep olarak BT öğretmenlerinin güncel programlama öğretimindeki yöntemler konusunda bilgi sahibi olmaları gerektiği ilk neden olarak gösterilmiştir. Diğer bir neden olarak ise; bir oyun tasarlamak için gerekli olan programlama becerisinin zor ve karmaşık olması, Scratch programının başlangıç düzeyindeki kullanıcılara daha ilgi çekici, kolay, heyecan verici ve eğlenceli ortam sunan bir görsel programlama dili olarak kolaylaştırıcı etkisi gösterilmiştir. Ürünlerin incelenmesi ve öğretmen adayları ile yapılan görüşmeler neticesinde, öğretmen adayları kendilerini başarılı bulmuş, bunun sebebi olarak ise “Tamamlayabilme ve kendine ait bir ürün ortaya koymanın çok güzel bir duygu olması” gösterilmiştir.

2.3. Programlama Öğretim Programı Geliştirme Çalışmaları

Programlama eğitimi, öğrencilere “kompütasyonel düşünme” becerisini kazandırmanın yanı sıra, birçok sektörde önemli bir yetenek haline gelerek 21. yüzyıl becerileri arasında sayılması ile son yıllarda öğretim programı geliştirme çalışmalarında önemle üzerinde durulan bir konu olarak karşımıza çıkmaktadır. Bunun yanında, programlamanın temele alındığı robotik tarzı etkinliklerle fen, teknoloji, mühendislik, matematik (STEM) alanlarında düzenlenen eğitimin kalitesinin artırılması amacıyla, küçük yaşlarda programlama eğitime başlanması ve ilerleyen yaşlarda da bu eğitimin sürekliliğinin sağlanması fikri son yıllarda sıklıkla gündeme gelmektedir. Amerika Birleşik Devletleri Başkanı Barack Obama'nın “Herkes kodlama öğrenmeli” çağrısı da bu düşünceye örnek gösterilebilir. Programlama eğitime İngiltere’de de son yıllarda büyük önem verilmiş, programlama eğitimi 5 yaşından 16 yaşına kadar bilgisayar bilimi dersi

kapsamında öğretim programlarına dâhil edilmiştir (Department for Education, 2013). 2013 yılından buyana devam eden İngiltere merkezli Avrupa Kodlama Haftası (Codeweek) etkinliklerine Avrupa'nın tüm ülkelerinden ve Türkiye, Makedonya, Tunus, Ukrayna, İzlanda, Norveç, İsveç gibi diğer ülkelerden binlerce okul ve on binlerce öğrenci katılmaktadır (bkz. <http://codeweek.eu/about/>, 20.11.2016). Dünyaca ünlü teknoloji devlerinin bulunduğu Güney Kore'de ise ilkokullardan liseye kadar tüm eğitim öğretim programlarında programlama ve yazılım dersleri saatleri artırılıp zorunlu hale getirilmiştir. 2016 yılı itibariyle Avusturya, Arjantin, Uruguay, Kanada, Hindistan, İtalya, İngiltere, Güney Kore, Suudi Arabistan, Malezya, Danimarka, Estonya, Finlandiya, Almanya, Polonya "Bilgisayar Bilimi" eğitimi kapsamında kodlama eğitimini öğretim programlarına dâhil etmişlerdir veya çalışma sürdürmektedirler (code.org, 2015). Ayrıca, Avrupa Birliği Avrupa Kodlama Girişimi tarafından, Bulgaristan, Güney Kıbrıs, Çek Cumhuriyeti, Yunanistan, Letonya ve Portekiz'de öğrencilere kodlama eğitimi verebilecek öğretmen yetiştirme çalışması devam etmektedir (The European Coding Initiative, 2014).

Lise düzeyinde öğrencilere kompütasyonel düşünme becerisi kazandırma amaçlı kodlama öğretim programı geliştirme çabaları son 20 yıldan beri devam etmekle birlikte, bu tür öğretim programı geliştirme çalışmaları A-8 düzeyinde aynı seviyeye ulaşamamıştır (Angeli ve diğerleri, 2016). Oysaki bilgisayarların ilk yaygınlaştığı 1970'ler ve 1980'lerde "Her çocuğa programlamayı nasıl öğretebiliriz?" sorusu oldukça ilgi çekmekteydi (Resnick ve diğerleri, 2009). MIT Media laboratuvarlarında Yaşam Boyu Okul Öncesi Grubu (Lifelong Kindergarten Group) tarafından yapılan araştırmalarda, çocuklarda ve gençlerde programlama eğitimine yönelik ilginin daha sonraki yıllarda azalmasının sebeplerine yönelik bazı tespitlerde bulunulmuştur. Papert (1980) tarafından tanıtılan Logo yazılımı ve diğer girişimlerin ilk başlardaki ilgiyi daha sonra sürdürmemelerinin sebepleri Resnick ve diğerleri (2009) tarafından şu şekilde açıklanmaktadır:

- İlk programlama dillerinin kullanımı zordu ve çocuklar söz dizilimine hâkim olamadılar.
- Programlama genellikle sayıların listelenmesi, basit çizgiler çizilmesi gibi gençlerin ilgisini çekmeyen etkinlikler olarak sunulmakta idi.

- Programlama, genellikle program yazarken bir şeyler ters gittiği zaman kimsenin yardım sağlayamayacağı şekilde veya her şey doğru gittiği zaman kişiyi daha derin araştırmalara teşvik edemeyecek şekilde sunulmakta idi.

A-6 düzeyinde yapılan bazı çalışmalarda, öğrencilerin soyutlama becerilerinin yeterli olmaması nedeniyle bu yaşlarda programlama eğitiminin verilemeyeceği yönünde sonuç ortaya konulmaktadır (bkz. Armony, 2012). Armony (2012) soyutlamayı bir problemin çözümü için düşünme süreçleri ve çözümün otomatikleşmesi amacıyla gerekli bilgisayar biliminin en temel bileşenlerinden birisi olarak göstermektedir. Armony'ye (2012) göre, Piaget'sel bir bakış açısıyla, çocukların 7-8 yaşından önce soyut kavramları anlaması beklenemez ve bunun yanında çocuk, 7 ve 8 yaşında ise somut nesnelere üzerinde problem çözme becerileri gösterebilmesine rağmen soyut kavram ve olguların çözümünde bu problem çözme becerilerini uygulayamaz. Diğer yandan, Gibson (2012) bilgisayar biliminin lise düzeyinde ilk defa gösterilmesinin çok geç olduğunu belirtmektedir. Gibson (2012) araştırmasında çocukların somut referans sistemleriyle düşünmeleri sağlandığı zaman soyut düşünebildiklerini söylemektedir.

MIT laboratuvarlarında Yaşam Boyu Okul Öncesi Grubu, çocuklarda kompütasyonel düşünmeyi artırmak, 21. yüzyıl becerilerini kazandırıcı bir programlama ortamı yaratmak için Papert'ın (1980) "Programlama dilleri "low floor-zemini düşük" (başlamak için kolay seviyede), "high ceiling-tavanı yüksek" (zamanla karmaşık projeler oluşturacak şekilde) olmalıdır." görüşünü temel alarak çalışma yürütmüşlerdir. Alice, Squeak Etoys gibi mevcut programlama dilleri incelendiğinde, kompütasyonel düşünmenin gelişimi için Papert'ın olabildiğince düşük zeminli - yüksek tavanlı ilkesine tam uygun bir programlama dili bulamamışlardır. Üç temel prensip ile programlama ortamlarını oluşturmaya başladıklarını belirtmektedirler: 1- Daha Düşündürücü 2- Daha Anlamlı 3 – Daha Sosyal. Onlarca yıllık Lego Mindstorms ve robotik setlerin geliştirilmesi çalışmalarında Lego Firması ile yakın çalışmaları neticesinde elde ettikleri deneyimlerle Scratch'i sanki Lego bloklarıyla yaratıcı şekilde oynayan çocukların mantığıyla sürükle-bırak şeklinde blok temelli olarak geliştirmişlerdir. Lego bloklarını eline alan çocukların hemen düşünmeye başladığına, yaratıcı ürünler tasarladığına yönelik gözlemleri Scratch'in geliştirilmesinde ilham kaynağı olmuştur. Scratch'in kodlama bölümü, kod bloklarının bulunduğu bölüm, sahne gibi

tüm alanlar, sanki gerçek bir masaüstünde çalışan çocuğun yaratıcılığını sergilediği ve sonucunu hemen gördüğü ortama benzetilmiştir. Çizim yapabildikleri, ses kaydı yapabildikleri bölümler, öğrencilerin yaratıcılık becerilerinin gelişimlerine katkı sağlayacak şekilde yaş gurubunun ilgi alanlarına hitap edecek şekilde tasarlanmıştır. Yine kişilerin ancak kendileri için anlamlı olan projelerde daha fazla eğlendikleri ve daha fazla öğrendikleri düşünülerek, kişilerin kendi ilgi alanlarına hitap edecek şekilde çeşitliliğe (hikâyeler, oyunlar, simülasyonlar, animasyonlar) ve kişiselleştirmeye (fotoğraf-müzik klipleri yükleyebilme, kendi grafiklerini oluşturma ya da seslerini kaydetme) önem verilmiştir. Projelerini paylaşabilecekleri, başkalarının projelerine katkı sunabilecekleri, birbirlerini destekleyebilecekleri, tartışabilecekleri bir sosyal paylaşım yapılabilen bir web sitesi oluşturulmuştur. Bu web sitesinde sosyal paylaşım sitelerinde olduğu gibi “yorum yap”, “beğen”, “bilgisayarıma indir” gibi özellikler bulunmaktadır (bkz. Resnick ve diğerleri, 2009).

Papert'dan (1980) günümüze kadar birçok girişim analiz edilerek çocuklar dâhil her yaş grubuna hitap edecek şekilde geliştirilen blok temelli programlama dili olan Scratch, günümüzde 150'den fazla ülkede 40'tan fazla dilde kullanılmaktadır (bkz. <https://scratch.mit.edu/about>, 3.8.2016). Bu kapsamda yürütülen araştırmalar MIT Yaşam Boyu Okul Öncesi Grubunun Scratch'le programlamaya yönelik öngörülerini desteklemektedir. Çatlak, Tekdal ve Baz (2015) tarafından 32 makale ile gerçekleştirilen doküman inceleme çalışmasına göre, Scratch yazılımının programlama öğretiminde etkili olduğu, programlamayı daha zevkli ve anlaşılır hale getirdiği, ayrıca programlama eğitiminde ilgi ve motivasyonu artırdığı bulgularına ulaşılmıştır. Ulaşılan makalelerin tüm eğitim kademelerine yönelik olduğu, ağırlıklı olarak ilkokul ve ortaokul seviyesinde olduğu görülmektedir. Araştırmadan çıkan bu sonuçlar göstermektedir ki, programlama eğitiminde birer engel olarak karşımıza çıkan; çocuklarda soyut kavramların öğretilemeyeceği, programlamanın gençlerin ilgi alanlarına hitap etmemesi gibi engellerin, MIT Yaşam Boyu Okul Öncesi Grubu tarafından hazırlanan Scratch yazılımı ile aşılabildiği sonucunu çıkartabiliriz.

2.4. Öz-yeterlik Algısı ve Programlama Performansı

Bandura'ya (1977) göre bireylerin belirli görevleri yerine getirmedeki kendi başarısına ilişkin yargısı, yani öz-yeterlik algısı, o görevi başarılı bir şekilde yerine getirmedeki performansı ve çabası ile doğrudan ilişkilidir. Stajkovic ve Luthans (1998) öz-yeterlik algısı ile performans arasındaki ilişkiyi ele alan 114 çalışmayı

inceleyerek yaptıkları meta analiz çalışması sonunda, öz-yeterlik algısı ile incelenen ilgili görev performansı arasında anlamlı ve pozitif ilişki olduğunu ifade etmişlerdir. Bu nedenle bireylerin herhangi bir görev bağlamındaki öz-yeterlik algısının yüksek olması doğrudan performansı ile ilişkili olacağından önemlidir. Alanyazın taraması sonucunda elde edilen bulgular, öğrencilerin programlamaya ilişkin öz-yeterlik algısı puanlarının, programlama performanslarını etkilediğini göstermektedir (Ramalingam, LaBelle, & Wiedenbeck, 2004; Aşkar & Davenport, 2009; Davidson, Larzon & Ljunggren, 2010; Altun ve Mazman, 2013).

Aşkar ve Davenport (2009) mühendislik 1. sınıf öğrencileriyle yaptıkları çalışmada java programlamaya ilişkin öz-yeterlik algısının cinsiyete, bölüme, ön bilgisayar bilgisine, bilgisayar kullanım sıklığına ve ailenin bilgisayar kullanımına göre farklılık gösterip göstermediğini incelemişlerdir. Çalışmaya toplam 326 üniversite öğrencisi katılmıştır. Ramalingam ve Wiedenbeck'in (1998) C++ programlama dili için geliştirdiği, güvenilirliği oldukça yüksek olan ölçme aracı Türk örnekleme üzerinde İngilizce dilinde eğitim alan bir üniversitede İngilizce olarak kullanılmıştır. Çalışma sonunda erkeklerin programlamaya ilişkin öz-yeterlik algısı kadınlara göre daha yüksek bulunurken, bilgisayar mühendisliği bölümü öğrencilerinin programlamaya ilişkin öz-yeterlik algısı da diğer bölümlere göre daha yüksek olarak bulunmuştur. Ayrıca bilgisayar kullanımına ilişkin deneyim yılının, bilgisayar kullanım sıklığının ve aile üyelerinin bilgisayar kullanımının programlama öz-yeterliğini anlamlı şekilde yordadığını bulmuşlardır.

Altun ve Mazman (2013) tarafından Hacettepe Üniversitesi Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümünde 2.sınıfa devam eden 64 öğrenci ile yapılan çalışmada veriler Altun ve Mazman (2012) tarafından Türkçeye uyarlanan "Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği" ile toplanmıştır. Ölçek öğrencilere üniversite öğretimlerinde ilk kez programlama dersinin verildiği ikinci sınıfın güz yarıyılında başında bir kez ve dersi aldıktan sonraki ikinci yarıyıl başında bir kez uygulanmıştır. Veriler tekrarlı ölçümler varyans analizi ile analiz edilmiştir. Çalışma sonunda programlama dersi hem ön deneyimi olan grup hem de olmayan grupta programlamaya ilişkin öz-yeterlik algısını anlamlı bir şekilde arttırmış ve bu artışın ön deneyimi olmayan grup için ön deneyimi olan gruba göre daha fazla olduğu bulunmuştur. Ayrıca ön deneyime sahip olan ve olmayan gruplar arasındaki öz-yeterlik algıları arasındaki fark programla dersi sonunda azalmıştır. Ön deneyimi

olmayan öğrencilerin öz-yeterlik algılarının düşük olmasından dolayı, içeriğin en baştan yıldıracı olmayacak şekilde basitten karmaşığa giderek tasarlanmasının, öz-yeterlik algılarının şekillenmesinde olumlu etki edebileceği belirtilmiştir.

Yükseltürk ve Altıok (2016) tarafından Scratch'le programlamanın Bilişim Teknolojileri Öğretmen Adaylarının programlamaya ilişkin tutumlarına olan etkilerini analiz etmeye yönelik bir çalışma yapılmıştır. 2013-2014 akademik yılında Scratch ünitesi bulunan bir seçmeli programlama dersinde 151 BT öğretmen adayına veri toplama aracı olarak 1 adet çevrimiçi Kişisel Bilgiler Formu, Altun ve Mazman (2012) tarafından Türkçeye uyarlanan Programlama Öz-yeterlik algısı Ölçeği ve 1 adet Bilgisayar Programlamaya Yönelik Tutum Ölçeği kullanılmıştır. Sonuç olarak, katılımcıların programlama öz-yeterlik algılarında tüm “karmaşık programlama görevleri” yönünde anlamlı bir artış olmuştur. Ayrıca katılımcıların programlamaya yönelik olumsuz tutumlarında da anlamlı bir düşüş yaşanmıştır ve Scratch'le programlamanın ölçekteki bazı maddeler yönünden olumlu etkileri olduğu görülmüştür. Sonuç olarak Scratch platformunun bilgisayar öğretmen adaylarının programlamaya ilişkin öz-yeterlik algılarına ve tutumlarına anlamlı düzeyde olumlu etkisinin görüldüğü belirtilmiştir. Elde ettikleri bulgulara dayanarak başlangıç düzeyindeki programcıların kompütasyonel becerilerinin ve yaratıcı düşünme becerilerinin artırabilmek için programlama eğitimlerinin Scratch kullanılarak yeniden tasarlanması tavsiye edilmiştir.

Çalışmalar incelendiğinde, ortaokul öğrencilerinin programlama eğitimlerinde blok temelli programlamanın yaygın olarak kullanıldığı ancak blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeği ile yapılmış bir çalışma olmadığı gözlemlenmiştir. Bu nedenle, ortaokul düzeyinde programlama eğitiminde en yaygın kullanılan Scratch programı temel alınarak, blok temelli programlamaya ilişkin bir öz-yeterlik algısı ölçeğinin geçerlik ve güvenirlik çalışmaları yapılarak bir “blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeği” geliştirilmesi amaçlanmıştır. Ayrıca geliştirilen ölçme aracı ile katılımcılardan elde edilen öz-yeterlik puanlarının çeşitli değişkenler açısından incelenmesi hedeflenmiştir.

2.5. Robotik Kodlama Eğitimi

Scratch'le kodlama eğitimi ile ilgili alanyazın taraması bulgularından bahsedilirken, A-6 düzeyinde yapılan bazı çalışmalarda, öğrencilerin soyutlama becerilerinin yeterli olmaması nedeniyle bu yaşlarda programlama eğitiminin verilemeyeceği

yönünde ortaya konulan sonuçlara önceki bölümlerde yer verilmiştir (bkz. Armony, 2012). Programlama eğitiminde birer engel olarak karşımıza çıkan; çocuklarda soyut kavramların öğretilmeyeceği, programlamanın gençlerin ilgi alanlarına hitap etmemesi gibi engellerin, MIT Yaşam Boyu Okul Öncesi Grubu tarafından hazırlanan Scratch yazılımı ile aşılabildiğine ilişkin araştırma sonuçlarında örnekler verilmiştir (bkz. Maloney, Peppler, Kafai, Resnick, & Rusk, 2008; Resnick ve diğerleri, 2009; Genç ve Karakuş; 2011, Kalelioğlu ve Gülbahar, 2014; Kukul ve Gökçearslan, 2014; Yükseltürk ve Altıok, 2016). Scratch for Arduino programı, Scratch programına Arduino'ya bağlı sensörleri ve aktüatörleri yönetmek için yeni bloklar sağlayan bir Scratch modifikasyonudur (Scratch for Arduino, 2015). Blok temelli programlama yapılarından olan Scratch kullanılarak düzenlenen kodlama etkinlikleri ile ilgili elde edilen bulguların Scratch'in bir modifikasyonu olan Scratch for Arduino ile düzenlenen kodlama etkinliklerinde de benzer şekilde elde edilmesi beklenmektedir.

Rusk, Resnick, Berg, ve Pezalla-Granlund (2008) çalışmalarında, çocuklara ve ailelerine yönelik düzenledikleri Workshoplar, okul sonrası merkezlerinde ve bir robotik kursunda elde ettikleri deneyimlere dayanarak robotik kodlama etkinlik örnekleri hazırlamışlardır. Etkinlikler, bir tema çevresinde oluşturulan, sanatsal etkinliklerle mühendisliğin harmanlandığı, ortaya çıkarılan ürünün hikayesinin oluşturulduğu ve son olarak seçilen bir yerde ürünlerin sergilendiği ve ürünlerin hikayelerinin anlatıldığı etkinlikler olarak tasarlanmıştır. Özetle etkinlikler şu dört temel strateji takip edilerek tasarlanmıştır; 1) Temalara odaklanma; 2) Sanat ve mühendisliği birleştirme 3) Hikâye anlatımı teşvik etme 4) Yarışmalardan ziyade sergiler düzenlemek. Sonuç olarak, robotik etkinliklerinin zengin eğitim fırsatları sunduğu, ancak bu etkinliklerin kullanılan robotik materyalin tasarlanış amacıyla sınırlı olabildiğine dikkat çekilmiştir. Otomobil, araç ve mobil robotların bazı gençlerin ilgisini çekerken, bazı gençlerin ilgisini daha çok sanat, müzik ve hikâye anlatımına yönelik etkinliklerin çektiği belirtilmiştir. Öğrencilerin gruplar halinde çalışmasını sağlarken, benzer ilgi alanına sahip olanların kendilerine uygun bir tema etrafında aynı grupta yer almalarının sağlanmasının önemli olabileceği belirtilmiştir.

Beug (2012) Scratch'le kodlama etkinliklerinin ve Arduino ile programlama etkinliklerinin öğrencilerin programlama performanslarıyla ilişkisini araştırmak

amacıyla paralel bir öğretim programı hazırlamıştır. Her iki platform için temel programlama becerilerini içerecek şekilde tasarlanan (değişkenler, koşullar, fonksiyonlar, döngüler) ve 5 oturum süren etkinlikler, 5 farklı gruba ayrılan 119 lise öğrencisi ile yürütülmüştür. Öğrencilerin demografik verilerini, temel programlama bilgilerini, programlamaya ilişkin tutumlarını ve etkinliklerin tümü tamamlandıktan sonra uygulanan Scratch ve Arduino etkinliklerine ilişkin yaşantılarını tespit etmek amacıyla 4 bölümden oluşan bir ölçek hazırlanmıştır. Ölçeğe yönelik bir geçerlik ve güvenilirlik analizi raporlanmamıştır. Ön-test son-test yöntemiyle toplanan veriler analiz edildiğinde Scratch grubundaki öğrencilerin programlama bilgisine ilişkin maddelere verdikleri puanlarda artış yaşanırken, Arduino grubundaki öğrencilerin maddelere verdikleri puanlarda artış yaşanmadığı, hatta bazı maddelere verilen puanların azaldığı da bulgular arasında yer almıştır. Katılımcıların daha önce programlama deneyimi olmamasına rağmen ön-testte bazı maddelere yüksek puanların verilmiş olmasının sebebi olarak, “değişkenler”, “koşullar” gibi kavramların matematik kavramlarıyla çağrışım yapmış olmasından dolayı bu maddelere ilişkin yanlış anlaşılma yaşanmış olabileceği belirtilmiştir. Arduino grubunda olan öğrencilerden az bir kısmı etkinlikler esnasında çok sıkıldığını belirtmiştir. Araştırmacı bunun sebebinin son iki etkinlikte etkinliğin 20-30 dakikasında Arduino kartın bilgisayara tanıtılmaya çabalanmasından dolayı kaynaklanmış olabileceğini belirtmiştir. Uygulama yapılan liselerdeki öğretmenler, Arduino'nun başlangıç seviyesinde programlama eğitimi için çok karmaşık olduğunu belirtmişler ve etkinliklerin Scratch'le başlamasının uygun olduğunu söylemişlerdir. Araştırmada dikkati çeken ayrıntı ise, karşılaştırma yapılan etkinliklerden birincisinde blok temelli programlama platformu olan Scratch seçilirken, diğerinde ise metin tabanlı programlama dili olan arduino programlama dilinin seçilmiş olmasıdır. Halbuki robotik kodlama eğitiminin Scratch for Arduino gibi blok temelli programlama platformu ile düzenlenmesi halinde daha geçerli araştırma bulguları elde edilmesi mümkün olabilirdi. Ayrıca robotik kodlama etkinliklerinin düzenleneceği ortamın tasarlanması ve düzenlenmesi sürecinin ne kadar önemli olduğu bu araştırma kapsamındaki etkinlikler esnasında Arduino kartın bilgisayara tanıtılmakla zaman kaybedilmesi şeklinde yaşanan güçlüklerden de anlaşılmaktadır.

Przybylla ve Romeike (2014) tarafından gerçekleştirilen araştırma kapsamında öğrencilerin kompütasyonel sistemler hakkındaki bilişim sınıflarına ilişkin algularına yönelik araştırma bulguları sunulmaktadır. Araştırma kapsamında Berlin ve Budapeşte’de bulunan 62’si erkek 51 kız 113 ortaokul öğrencisinden 6’lı Likert tipinde ölçekle veri toplanmıştır. Araştırma bulgularında, öğrencilerin ancak %19’nun robotik etkinliği yaptıklarını belirttikleri tespit edilmiş, bunun sebebi olarak ise robotik etkinliklerinin ders saatlerinde yapılmaması, ancak hobi olarak boş saatlerde ve hafta sonu kulüplerinde öğrencilerin bu olanağa ulaşabilmeleri gösterilmiştir. Öğrencilerin ancak %41’i bilişim sınıflarında yaratıcı olduklarını düşündüklerini belirtmişlerdir. Araştırmacılar bunun sebebi olarak, modern araçların kullanılmasına rağmen, öğretmen merkezli yaklaşımlar, eski moda yöntemler ve çözülmesi gereken sorunları bulmada öğrenci katılımının sağlanmaması gibi sınıf ziyaretlerinde elde ettikleri izlenimleri göstermişlerdir. Araştırmacılar, öğrencilerin sanal nesnelere çok gerçek dünyadaki nesnelere içeren etkinliklere daha fazla ilgi duydukları, fiziksel kodlama ile öğrencilerin eğlendikleri, etkinliklere devam edebilmek için daha uzun süre kalmak istedikleri yönünde elde ettikleri gözlem bulgularını raporlamışlardır. Öğrencilerin robotik parçalarını nereden satın alabileceklerini sordukları, kullandıkları parçaları daha sonra değişim yapma imkanlarının olup olmadığını sordukları da öğrencilerin robotik etkinliklerine olan istek ve ilgilerini gösteren bulgular arasında sunulmuştur.

Ülkemizde programlama eğitiminin robotik kodlama etkinlikleri şeklinde sürdürülmesine yönelik araştırma yapıldığında dikkati çeken bir proje örneği Gazi Üniversitesi Öğretim Üyesi Doç. Dr. Selçuk Özdemir’in kurucu başkanlığını yürüttüğü yüz yüze eğitimlerin yanı sıra çevrimiçi öğrenme ortamı da sunan Bilişim Garaj Akademisi’dir (bkz. <https://www.bilisimgarajakademisi.com>, 17.12.2016). Platformda öğrencilere kodlama ve robotik kodlama eğitiminin yanı sıra 3D tasarım eğitimi de verilmektedir. Okullarda ise robotik kodlama etkinliklerine ilişkin uygulama örnekleri henüz çok yeni başlamıştır. Son birkaç yıldır ders dışı etkinlikler tarzında hafta sonu kursları ve maker kulüpleri gibi ortamlarda robotik kodlama etkinliklerinin gerçekleştirildiği görülmektedir. Ancak ülkemizde A-12 düzeyinde robotik kodlama etkinliklerine yönelik alanyazında araştırma bulgusuna rastlanılmamıştır.

2.6. Genel Deęerlendirme

Programlama eęitimi kapsamında yapılan alıřmalar incelendięinde, nceki yıllarda daha ok programlama eęitiminin yazılım uzmanlıęı olarak ele alındıęı alıřmalara ulařılmaktadır. A-6 dzeyinde yapılan bazı alıřmalarda, ęrencilerin soyutlama becerilerinin yeterli olmaması nedeniyle bu yařlarda programlama eęitiminin verilemeyeceęi ynnde sonuların ortaya konulduęu alıřmalara rastlanılmaktadır (bkz. Armony, 2012). Ancak, programlamanın blok temelli rneklerinin ortaya ıkmasıyla beraber zellikle son on - on beř yılda yapılan alıřmalarda blok temelli programlamanın A-12, A-8 ve A-6 dzeyinde de olumlu sonular verebildięi, ęrencilerin etkinliklere istekli katıldıkları, ilgi duydukları, programlama performanslarında, problem özme ve komptasyonel dřnme becerilerinde olumlu artıř elde edildięi ynnde bulgulara rastlanılmaktadır (bkz Resnick ve dięerleri, 2009; Kalelioęlu ve Glbahar, 2014; Kukul ve Gkearsan, 2014; Ykseltrk ve Altıok, 2016).

Komptasyonel dřnme ve programlama becerileri 21. yzyıl toplumunda bireylerin sahip olması gereken beceriler arasında sayılmaktadır (ISTE, 2016; Angeli ve dięerleri, 2016). Programlama komptasyonel dřnme anlamına gelmemekle birlikte, birok arařtırmacı ve bu konuda alıřan bilim insanı, programlama eęitiminin bireylerdeki komptasyonel dřnme becerilerine katkı yaptıęı dřncesindedir (bkz. Wing, 2006; Settle & Perkovic, 2010; Glbahar & Kalelioęlu, 2014; Allsop, 2015; Angeli ve dięerleri, 2016). Komptasyonel dřnme becerisine olumlu katkısı ile programlama eęitimi, programlama uzmanlık eęitiminin dıřında temel beceri eęitimi kapsamında ele alınmaya bařlamıřtır. Programlamanın blok temelli platformda yapılması ile ortaokul dzeyinde dahi eęitiminin verilebilir hale gelmesi birok lkenin A-8 ve A-12 dzeyinde eęitim programları arasında kodlama eęitiminin yer almasına sebep olmuřtur (Code.org 2015 Annual Report, 2015). Dnya apında komptasyonel dřnme ve programlama eęitim kapsamında yapılan alıřmalara paralel olarak lkemizde de programlama eęitimi 2012 yılında Biliřim Teknolojileri ve Yazılım Dersi ęretim Programına dhil edilmiřtir (TTKB, 2012). Bu kapsamda gerekleřtirilen alıřmalardan bir tanesi olan liseler iin bilgisayar bilimi ęretim programı geliřtirme alıřması da 2016 yılı itibariyle tamamlanmıř, Talim Terbiye Kurulu Bařkanlıęı tarafından liseler iin "Bilgisayar Bilimi Dersi ęretim Programı" yayınlanmıřtır

(TTKB, 2016). 2017 yılı ocak ayı itibariyle taslak hali yayınlanan 1. sınıftan 8. sınıfa kadar bilişim teknolojileri ve yazılım dersi öğretim programında da tüm sınıf düzeylerinde “hesaplamalı düşünme” ünitesi altında “problem çözme kavramları ve yaklaşımları” ve “programlama” konu başlıkları bulunmaktadır (TTKB, 2017).

Kodlama eğitiminde en yaygın kullanılan platformlar code.org ve Scratch platformlarıdır. Code.org daha çok başlangıç düzeyinde belirli görevleri yerine getirmek üzere tasarlanmış bir ortamken, Scratch daha özgür tasarımlar yapma ortamı sunmaktadır. Scratch, günümüzde 150’den fazla ülkede 40’tan fazla dilde kullanılmaktadır (bkz. <https://scratch.mit.edu/about>, 3.8.2016). Scratch for Arduino programı ise, Scratch programına Arduino’ya bağlı sensörleri ve aktüatörleri yönetmek için yeni bloklar sağlayan bir Scratch modifikasyonudur (Scratch for Arduino, 2015). Alanyazında robotik kodlama etkinliklerine fiziksel kodlama olarak söz edildiği de bulgular arasındadır. Araştırmacılar, öğrencilerin sanal nesnelere çok gerçek dünyadaki nesnelere içeren etkinliklere daha fazla ilgi duydukları, fiziksel kodlama ile öğrencilerin eğlendikleri, etkinliklere devam edebilmek için daha uzun süre kalmak istedikleri yönünde elde ettikleri bulgularını raporlamışlardır (Rusk, Resnick, Berg, & Pezalla-Granlund, 2008; Przybylla & Romeike, 2014). Ülkemizde A-12 düzeyinde robotik kodlama etkinliklerine ilişkin uygulama örnekleri henüz çok yeni başlamıştır. Okullarda son birkaç yıldır ders dışı etkinlikler tarzında hafta sonu kursları ve maker kulüpleri gibi ortamlarda robotik kodlama etkinliklerinin gerçekleştirildiği görülmektedir. Ancak ülkemizde A-12 düzeyinde robotik kodlama etkinliklerine yönelik alanyazında araştırma bulgusuna rastlanılmamıştır. Araştırmamız kapsamında robotik kodlama etkinliklerinin öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algılarında farklılaşma meydana getirip getirmediğine bakılacaktır. Ayrıca, öğrencilerin robotik kodlama etkinliklerine ilişkin yaşantılarını tespit etmek amacıyla gözlem, görüşme ve etkinlik algısı ölçeğinden elde edilecek bulgular yorumlanacak, alanyazında elde edilen bulgularla örtüşüp örtüşmediği tespit edilecektir.

Bandura’ya (1977) göre bireylerin belirli görevleri yerine getirmedeki kendi başarısına ilişkin yargısı, yani öz-yeterlik algısı, o görevi başarılı bir şekilde yerine getirmedeki performansı ve çabası ile doğrudan ilişkilidir. Programlama eğitiminin bireylerin programlamaya ilişkin performanslarına katkısını tespit etmek amacıyla yapılan bazı çalışmalarda performans göstergesi olarak programlama öz-yeterlik

algısı ölçeklerinin kullanıldığı görülmektedir (Aşkar & Davenport, 2009; Altun ve Mazman, 2013; Yükseltürk & Altıok, 2016). Bu çalışma kapsamında gerçekleştirilen robotik kodlama etkinliklerinin öğrencilerin programlama performanslarına katkısını tespit etmek amacıyla blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeği kullanılabileceği sonucu çıkarılmıştır.

3. YÖNTEM

Bu bölümde araştırmacının rolü ve uygulamanın yürütüldüğü ortam, araştırma modeli, çalışma grubu, araştırma modeli, veri toplama araçları ve verilerin analizi ile ilgili bilgiler yer almaktadır.

3.1. Araştırmacının Rolü ve Uygulamanın Yürütüldüğü Ortam

Bu bölümde araştırmacının rolüne ve uygulama sürecine ilişkin bilgiler yer almaktadır. Araştırmanın yürütüldüğü ortamın betimlenmesine ilişkin bilgiler, araştırmanın yapıldığı 2015-2016 eğitim öğretim yılına ait verilere dayalı olarak sunulmaktadır.

Araştırmacı Ankara'nın Altındağ ilçesinde bulunan bir devlet okulunda bilişim teknolojileri öğretmeni olarak görev yapmaktadır. 5. ve 6. sınıflarda haftada 2 ders saati olarak zorunlu dersler arasında yer alan bilişim teknolojileri ve yazılım dersini yürütmektedir. Sınıflar ortalama 35'şer kişilik olup, okulda 20 bilgisayar bulunan tek bilişim teknolojileri sınıfında dersler kalabalık bir ortamda yürütülmektedir. Bunun yanında okul yönetimi tarafından 5. sınıflarda destekleme ve yetiştirme kursu kapsamında açılması uygun görülen bilişim teknolojileri ve yazılım dersini cumartesi günleri ikişer saatten 4 grupta yürütmektedir. Gruplar destekleme ve yetiştirme kursları yönergesine uygun olarak en fazla 20'şer kişi olacak şekilde oluşturulmuştur.

Hafta sonu yürütülen kurs kapsamında oluşturulan sınıfların en fazla 20'şer kişilik olması nedeniyle araştırmaya elverişli olması, kursa kaydolun öğrencilerin daha önceden düzenlenecek çalışma konusunda bilgilendirilmesi ve araştırmaya katılma konusunda gönüllü olmaları nedeniyle araştırmanın cumartesi günleri düzenlenen kurs gruplarıyla yürütülmesine karar verilmiştir. Araştırmacı, etkinliklerin düzenleyicisi, ölçeklerin katılımcı öğrenciler tarafından doldurulmasını sağlayıcı, gözlem ve görüşmelerin gerçekleştirilmesi süreçlerinin yürütücüsü olarak "katılımcı gözlemci" rolündedir.

Araştırma kapsamında ihtiyaç duyulan "blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeği" geliştirme sürecinde soruların oluşturulması aşamasında soruları oluşturan 4 kişilik ekipte yer almış, ölçeğin çevrimiçi hale getirilmesi, ölçeğin

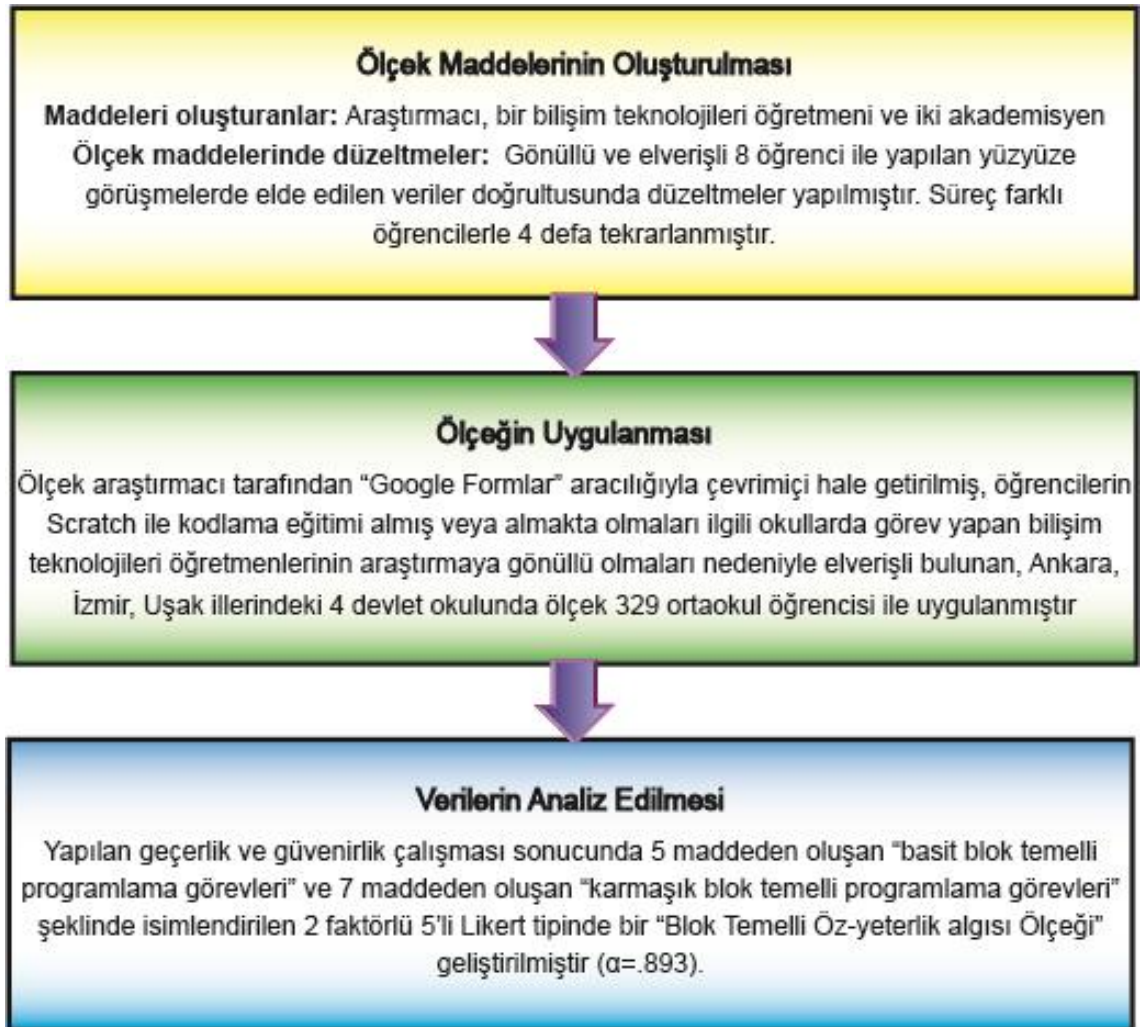
uygulanacağı okulların tespit edilmesi ve ilgili okullardaki öğretmenlerle iletişim kurarak öğretmenler tarafından ölçeğin uygulattırılması süreçlerini yürütmüştür.

Öğrenci yaşantılarını tespit edilmesi amacıyla ihtiyaç duyulan “etkinlik algısı ölçeğinin” Türkçeye uyarlanması çalışması da araştırmacı ve iki akademisyen tarafından yürütülmüştür.

3.2. Araştırma Modeli

Araştırma kapsamındaki tüm çalışmalar 2015-2016 eğitim öğretim yılında gerçekleştirilmiştir.

Çalışmalar iki aşamada yürütülmüştür. Birinci aşamada “blok temelli programlamaya ilişkin özyeterlik algısı” ölçeğinin alanyazında bulunmadığı tespit edilmiş, “blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeği” geliştirme çalışması yürütülmüştür. Ölçeğin geliştirme süreci şekil 3’de görülmektedir:



Şekil 3: Kodlamaya İlişkin Öz-yeterlik Algısı Ölçeği Geliştirme Süreci

Ölçekten elde edilen verilere ve ölçeğin uygulanmasına ilişkin çeşitli değişkenler açısından betimsel analiz verilerine ilerleyen bölümlerde ayrıca yer verilmiştir.

“Blok temelli programlamaya ilişkin özyeterlik algısı ölçeği” geliştirme süreci devam ederken araştırmacı ve alanında uzman iki akademisyen tarafından robotik kodlama etkinliklerine ilişkin öğrencilerin etkinlik algılarını tespit etmeye yönelik alanyazında İngilizcesi bulunan “etkinlik algısı ölçeğinin” Türkçeye uyarlama çalışması yürütülmüştür. Her etkinliğin sonrasında uygulanmak üzere öğrenci yaşantılarını tespit etmeye yönelik “etkinlik algısı ölçeğinin” yanısıra yarı yapılandırılmış görüşme soruları hazırlanmıştır.

İkinci aşamada ise Robotik kodlama etkinliklerinin ortaokul öğrencilerinin kodlamaya ilişkin özyeterlik algılarına etkisini araştırmak için ön test – son test kontrol grupsuz deneysel araştırma modeli uygulanmıştır.

Robotik kodlama etkinlikleri daha önce Scratch’le kodlama eğitimi alan öğrencilerle gerçekleştirilerek robotik kodlama etkinliklerinin öğrencilerin kodlamaya ilişkin özyeterlik algılarına etkisi araştırılacağı için, aynı zamanda kurs öğretmeni olan araştırmacı tarafından 5 haftalık Scratch’le kodlama eğitimi düzenlenmiş, katılımcı gözlemci olarak notlar alınmıştır. Ancak düzenlenen bu ön eğitimin araştırmanın amaçları arasında olmaması nedeniyle, yalnızca sonraki araştırmalara öneri niteliğinde gözlemler gerçekleştirilmiştir.

Haftada 2’şer ders saatinden 5 haftalık robotik kodlama etkinlik planlaması yapılmıştır. Her etkinliğin sonrasında ilgili etkinliğe ilişkin gözlem ve görüşmelerden elde edilen veriler doğrultusunda öğrencilerin ek bilgiye ihtiyaç duydukları konular ve öğrenme süreçlerinde zorlandıkları noktalar tespit edilmiştir. Elde edilen bulgular doğrultusunda sonraki etkinlik planında düzeltme ve düzenlemeler gerçekleştirilmiştir.

Robotik kodlama etkinlikleri öncesinde etkinliğe ilişkin veri toplama araçlarının ve etkinlik planlarının hazırlanmasının yanısıra bir taraftan da etkinlik esnasında ihtiyaç duyulacak etkinlik araç ve gereçlerinin temin süreci tamamlanmıştır. Etkinlik esnasında öğrencilerin düzenli bir şekilde etkinlikleri gerçekleştirebilmesi amacıyla robotik parçaları bir kutu içerisine yerleştirilerek eğitim seti haline getirilmiştir. Etkinlikler destekleme ve yetiştirme kursu kapsamında açılan bilişim teknolojileri ve yazılım dersinde gerçekleştirileceği için yönetmelik gereği sınıflar en fazla 20’şer

Robotik kodlama etkinliklerinin 5 hafta olacak şekilde planlanmasının sebebi ise zaman sınırı olarak özetlenebilir. Blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeğinin geliştirme sürecinin mart ayına kadar sürmesi en önemli etken olarak gösterilebilir. Etkinlikler 6 hafta olacak şekilde planlanma yapılırken, kursların kapanış tarihi okul yönetimi tarafından değiştirilmiş ve bu nedenle planlama 5 haftaya göre düzenlenmek zorunda kalınmıştır. Etkinliklerin daha fazla ders saatine yayılması durumunda daha fazla tekrara yer verilecek şekilde planlama yapılabilirdi.

3.3. Çalışma Grubu

Araştırmanın çalışma grubu veri toplama süreci ile ilişkili olarak iki farklı gruba yürütülmüştür. Çalışma kapsamında ilk aşama araştırmada kullanılacak ölçme araçlarının geliştirilmesi hedeflenmiş ve buna özgü bir çalışma grubu belirlenmiştir. Bu süreçte, “Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği” geliştirme çalışması kapsamında hazırlanan ölçek, söz konusu okullarda eğitim gören öğrencilerin Scratch’le kodlama eğitimi almış veya almakta olduğunun tespit edilmesi ve ilgili okullarda görev yapan bilişim teknolojileri öğretmenlerinin araştırmaya gönüllü olmaları nedeniyle elverişli bulunan, Ankara, İzmir, Uşak illerindeki 4 devlet okulunda ortaokul düzeyinde eğitim gören toplamda 329 öğrenciyle uygulanmıştır. Bu öğrencilerin belirlenmesinde, öğrencilerin robotik kodlama etkinliklerine katılacak öğrencilerden olmamasına dikkat edilmiş, bu nedenle robotik kodlama etkinliklerinin yürütüleceği, araştırmacı olarak kendi görev yaptığım okuldan veri toplanmamıştır. Bu öğrencilere ait demografik verilerden elde edilen betimsel istatistikler Tablo 3.1’de verilmiştir.

Tablo 3.1: Ölçek Geliştirme Aşaması Araştırma Grubuna Ait Betimsel Veriler

		N	%
Cinsiyet	Kız	163	%49,5
	Erkek	166	%50,5
Sınıf Düzeyi	5. Sınıf	233	%70,8
	6. Sınıf	61	%18,5
	7. Sınıf	25	%7,6
	8. Sınıf	8	%3,0

Daha Önce Scratch'le Programlama Dersi Alma	Evet	203	%61,7
	Hayır	126	%38,3
Halen Scratch'le Programlama Dersi Alma	Evet	261	%78,6
	Hayır	68	%20,7
Daha Önce Scratch'le Program Yazma	Evet	209	%63,5
	Hayır	120	%36,5
Halen Scratch'le Program Yazma	Evet	250	%76,0
	Hayır	78	%23,7

Çalışmanın ikinci aşamasında ise robotik kodlama eğitimi sonunda öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algılarında bir değişiklik meydana gelip gelmediğinin tespit edilmesi ve sonuçların farklı değişkenler açısından incelenmesine yönelik uygulama yürütülmüştür. Bu aşamadaki çalışma grubunu hafta sonu destekleme ve yetiştirme kursuna katılan 66 öğrenci oluşturmuştur. Başlangıçta kursa kayıt olan öğrenci sayısı 74 olmasına rağmen, 66 öğrenci kursa katılmıştır. Ancak, bu öğrencilerden 8 tanesi sağlık sorunları, okul değişikliği gibi nedenlerle eğitimi yarıda bırakmış ve bu nedenle çalışma grubu 58 kişi olarak tamamlanmıştır. Araştırma grubuna ait demografik verilerden elde edilen betimsel istatistikler Tablo 3.2'de verilmiştir.

Tablo 3.2: Robotik Kodlama Eğitimi Araştırma Grubuna Ait Betimsel Veriler

		N	%
Cinsiyet	Kız	27	46,6
	Erkek	31	53,4
Sınıf Düzeyi	5. Sınıf	58	100
Evde Bilgisayar Sahipliği	Evet	34	58,6
	Hayır	24	41,4
Evde İnternet Bağlantısı Sahipliği	Evet	24	41,4
	Hayır	34	58,6
Daha Önce Scratch Programı Dersi Alma	Evet	58	100
	Hayır	0	0

Halen Scratch Programı Dersi Alma	Evet	0	0
	Hayır	58	100
Daha Önce Scratch For Arduino Robotik Dersi Alma	Evet	0	0
	Hayır	58	100
Ders Dışında Scratch Programına Çalışma Olanığı	Var	20	34,5
	Yok	38	65,5
Ders Dışında Scratch Programına Çalışma Sıklığı	Haftada 1 gün	6	10,3
	Haftada 1-2 gün	5	8,6
	Haftada 3-4 gün	4	6,9
	Her gün 1 saatten fazla	1	1,7
	Her gün 1 saatten az	4	6,9
Ders Dışında Scratch For Arduino ile Robotik Çalışma Olanığı	Var	0	0
	Yok	58	100

3.4. Öğretim Sürecinin Tasarımı

Uygulama sürecindeki araştırma grubundaki öğrencilere araştırma süreci öncesinde 5 haftalık Scratch'le kodlama eğitimi verilerek öğrencilerin kodlamaya ilişkin ön bilgi sahibi olmaları sağlanmıştır. Scratch'le kodlama etkinliklerinin tamamlanmasının ardından robotik kodlama etkinlikleri düzenlenmiştir.

Robotik kodlama etkinliklerinin tasarlanması ve etkinlik planlarının hazırlanması süreci iki boyutta ele alınmıştır. Çalışmanın birinci boyutu etkinliklerin amaçlarının belirlenmesi ve bu amaçları gerçekleştirebilmek için gerekli olan etkinliğin tasarlanması sürecidir. Bu kapsamda Scratch yazılımını geliştiren MIT Yaşam Boyu Okul Öncesi Grubu tarafından hazırlanan bir makalede (Resnick ve diğerleri, 2009) belirtilen Scratch platformunda programlamaya yönelik özellikler dikkate alınarak, Hacettepe Üniversitesi Eğitim Fakültesi Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümünde görevli iki akademisyen, araştırmacı ve diğer bir bilişim teknolojileri öğretmeni tarafından Scratch for Arduino programını kullanarak öğrencilerin devre kurarken üstesinden gelebilecekleri düzeyde kazanımlar belirlenmiş ve etkinlikler bu kazanımlar çerçevesinde oluşturulmuştur.

Çalışmanın İkinci boyutu ise öğrencilerin bu etkinlik için gerekli olan ön bilgi ve becerilerin tespiti ve bu ön bilgi ve becerilerin karşılanmasına yönelik yapılması gereken ön çalışmaları kapsamaktadır. Çalışmanın bu iki boyutu iki ayrı çalışma gibi görünse de robotik kodlama etkinlikleri tasarlanırken kodlamaya yönelik kazanım ve etkinlik süreci, öğrencilerin elektronik devrelerini hangi karmaşıklık düzeyinde tasarlayabilecekleri hesaba katılmadan tasarlanamamaktadır. Robotik kodlama etkinlikleri fen bilgisi dersiyile doğrudan bağlantılı etkinliklerdir. Robotik kodlama etkinliklerinde yer alan görevler incelendiğinde, bu görevlerin öğrenciler tarafından yerine getirilebilmesi için elektrik ve elektronik konularıyla ilgili bazı bilgi ve becerilerin öğrenciler tarafından daha önceden kazanılmış olması veya etkinlikler esnasında disiplinler arası çalışmayla bu bilgi ve becerilerin öğrencilere kazandırılması gerektiği öngörülmektedir. Hedef gruptaki ortaokul 5. sınıf öğrencilerinin bu kazanımlara ne düzeyde sahip olduğu ile ilgili iki fen bilgisi öğretmeni ile ön çalışma yapılmıştır. Yapılan disiplinler arası çalışma neticesinde bu kazanımların 5. sınıf fen bilgisi kazanımlarını da içermekle beraber 6., 7., 8. sınıf konularına da ait olduğu görülmüştür. Mevcut eğitim sistemimizde öğrencilerin etkinlikleri gerçekleştirebilecekleri düzeyde elektrik-elektronik bilgisine ilişkin bu kazanımları disiplinler arası iş birliği şeklinde farklı derslerde kazandırma olanağı olmaması nedeniyle, robotik kodlama etkinliklerinde öğretmen olarak görev yapan bilişim teknolojileri öğretmeni tarafından ön bilgi - derse giriş niteliğinde öğretim süreci yürütülmüştür. Ancak etkinliklerin fen bilgisi ve teknolojiye yönelik kazanımları çalışmamızın amaçları arasında olmadığı için bu kazanımlara yönelik raporlama yapılmamıştır.



Şekil 6: Robotik Kodlama Etkinliği Esnasında Öğrenciler

Çalışmalar sonucunda 5 haftalık etkinlik planı hazırlanmış, etkinlikler esnasında elde edilen gözlem bulguları dikkate alınarak bir sonraki etkinliğe ait planlarda düzeltmeler, değişiklikler yapılmıştır. Örnek etkinlik planı ve etkinliğe ait gözlem notları EK 3'te sunulmuştur. Aşağıdaki tabloda öğrencilere kazandırılması planlanan robotik kodlamaya yönelik kazanımlar yer almaktadır.

Tablo 3.3: Robotik Kodlama Etkinlikleri ile Kazandırılması Planlanan Kazanımlar

KAZANIMLAR	
Robotik Kodlama Etkinliği 1	<p>Bu etkinlik sonunda öğrencilerin;</p> <ul style="list-style-type: none">• Arduino Uno R3 Kart, Breadboard (ekmek tahtası), LED ampul ve iletken kabloyu tanımaları• Giriş ve çıkış (input - output) kavramlarını bilmeleri• Dijital veri ve analog veri kavramlarının ne olduğunu bilmeleri• Robotik sistemlerin çalışma prensibinin giriş, işlem ve sonuç şeklinde gerçekleştiğini kavramaları• Scratch for Arduino ara yüzündeki Arduino ile bağlantılı kod bloklarını tanımaları• LED'i açıp kapatan Scratch kodlarını yazmaları.• Scratch programında "tıklandığında", "sürekli", "1 sn bekle" kontrollerini uygulamaları beklenmektedir.
Robotik Kodlama Etkinliği 2	<p>Bu etkinlik sonunda öğrencilerin;</p> <ul style="list-style-type: none">• Arduino uno R3 kartın dijital çıkış (output) pinlerinin birden fazlasını eşzamanlı olarak kontrol edebilmesi• Breadboard (ekmek tahtası) paralel bağlantılarını tanıyarak tam olarak kullanabilmesi,• Birden fazla dijital çıkış (output) pinlerini kullanarak<ul style="list-style-type: none">○ Scratch programında paralellik mantığıyla,○ Sürekli döngü içinde LED'leri farklı zaman aralıklarında yanıp söndürecek algoritmayı (kodlamayı) gerçekleştirebilmeleri• Değişken tanımlayabilmeleri• Değişken değerlerini artırabilmeleri• Karşılaştıkları hataları çözebilmeleri• Scratch programında paralellik mantığını, "tıklandığında", "sürekli", "1 sn bekle" kontrollerini uygulamaları beklenmektedir.
Robotik Kodlama Etkinliği 3	<p>Bu etkinlik sonunda öğrencilerin;</p> <ul style="list-style-type: none">• Arduino uno R3 kartın dijital giriş (input) pinlerini kullanabilmeleri• Dijital giriş cihazlarından olan düğme (buton) bağlantılarını yapabilmeleri• Düğme (buton) örneğinde dijital giriş (input) değerini (true-false) kavramaları, bu tür değerlerin önceden tanımlı değişken olduklarını kavramaları

-
- Arduino bađlantılı karakter ekleyebilmeleri
 - **Buton** kullanarak, butona basılıp basılmama durumuna göre;
 - “Eđer ise-bařka (öyle deđilse)” **kořul ifadesini**,
 - **dijital giriř (input)** pinlerinden okunan deđerler (true-false) ile **“sensor dijital 2-3 pressed”** kod bloklarını ve
 - **dijital ıkıř (output)** pinlerini yöneten **“digital on” – “digital off”** kod bloklarını kullanarak Led ampul yaktırabilecek kod bloklarını düzenleyebilmeleri
 - Deđiřken tanımlayabilmeleri
 - **Deđiřken deđerlerini duruma bađlı olarak** “eđer ise – bařka” kořul ifadesi kontrol bloklarından yararlanarak **deđiřtirebilmeleri**
 - “Eđer ise – bařka” kořul ifadesi kontrol bloklarından yararlanarak **deđiřken deđerlerine bađlı olarak sahneye efekt uygulayabilmeleri**
 - Karřılařtıkları hataları, problemleri özebilmeleri beklenmektedir.
-

Bu etkinlik sonunda öđrencilerin;

- Arduino uno R3 kartın analog giriř (input) pinlerini kullanabilmeleri
 - Analog giriř sensörlerinden olan ıřık sensörünün bađlantısını yapabilmesi
 - Iřık sensörü örnekliđinde analog giriř (**input**) pininden okunan deđerin belirli bir aralıktaki olduđunu (**0 – 1024 arasında deđer üretir**) anlamaları,
 - Analog pinlerini okunan deđerleri alarak istenildiđinde kullanımımıza sunan sensör okuma bloklarının scratch for arduino programı iinde **önceden hazır tanımlı deđiřken olduklarını kavramaları**
 - **Arduino bađlantılı karakter** eklemeyebilmeleri
 - Buton kullanarak;
 - “Eđer ise-bařka (öyle deđilse)” **kořul ifadesini**,
 - **analog giriř (input)** pinlerinin deđerlerden okunan deđerleri (0 ile 1024 arasında) ile **“value of sensor analog5”** kod blođunu kullanarak,
 - **dijital ıkıř (output)** pinlerini yöneten **“digital on” – “digital off”** kod bloklarını kullanarak Led ampul yakabilecek kod bloklarını düzenleyebilmeleri,
 - **karakterin kostümünü deđiřtirebilmeleri**
 - Deđiřken tanımlayabilmeleri
 - **Deđiřken deđerlerini duruma bađlı olarak** “eđer ise – bařka” kořul ifadesi kontrol bloklarından yararlanarak **deđiřtirebilmeleri**
-

-
- “Eğer ise – başka” **koşul ifadesi kontrol bloklarından yararlanarak değişken değerlerine bağlı olarak sahneyi değiştirebilmeleri**
 - **Büyüktür (>), küçüktür (<) ve eşittir (=) operatörlerini** kullanabilmeleri
 - Karşılaştıkları hataları, problemleri çözebilmeleri beklenmektedir.
-

Bu etkinlik sonunda öğrencilerin;

- Arduino uno R3 kartın dijital giriş (input) pinlerini kullanabilmeleri
 - Dijital giriş sensörlerinden olan **dijital ışık sensörünün bağlantısını yapabilmeleri**
 - Dijital ışık sensörü örneğinde **dijital giriş (input) pininden okunan değer belirlenmiş bir aralıkta olduğunu (0 – 1024 arasında değer üretir) anlamaları,**
 - Dijital pinlerinden okunan true – false değerlerini alarak istenildiğinde kullanımımıza sunan sensör okuma bloklarının scratch for arduino programı içinde önceden hazır tanımlı değişken blokları olduklarını kavramaları
 - Arduino bağlantılı karakter ekleyebilmeleri
 - Buton kullanarak;
 - “Eğer ise-başka (öyle değilse)” koşullu ifadesini,
 - dijital giriş (input) pinlerinden okunan değerleri (true -false) “digital2 pressed” kod bloğunu kullanarak,
 - dijital çıkış (output) pinlerini yöneten “digital on” – “digital off” kod bloklarını kullanarak Led ampul yakabilecek kod bloklarını düzenleyebilmeleri
 - karakterin kostümünü değiştirebilmeleri
 - Değişken oluşturabilmeleri
 - Değişken değerlerini duruma bağlı olarak “eğer ise – başka” koşullu ifadesi kontrol bloklarından yararlanarak değiştirebilmeleri
 - “Eğer ise – başka” koşullu ifadesi kontrol bloklarından yararlanarak değişken değerlerine bağlı olarak sahneyi değiştirebilmeleri
 - **Analog sensör ile dijital sensörden gelen değişkenlerin farkını kavrayabilmeleri**
 - Karşılaştıkları hataları, problemleri çözebilmeleri beklenmektedir.
-

Robotik Kodlama Etkinliği 5

3.5. Veri Toplama Araçları

Araştırma sorularının yanıtlarına ulaşmak amacıyla kullanılacak veri toplama araçlarının tespitinde öncelikle alanyazın taraması yapılmış, alan yazında blok

temelli programlara ilişkin öz-yeterlik algısı ölçeği ve etkinlik algısı ölçeği olup olmadığı araştırılmıştır. Blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeğine ilgili alanyazında rastlanılmamış, Altun ve Mazman (2012) tarafından Türkçeye uyarlaması yapılmış olan Programlamaya İlişkin Öz-yeterlik Algısı Ölçeğinin çalışmamız kapsamında kullanılabilirliği araştırılmıştır. Blok temelli programlamanın kendine özgü yapısı nedeniyle Programlamaya İlişkin Öz-yeterlik Algısı Ölçeğinin çalışmamız kapsamında kullanılmayacağı tespit edilmiştir. Bu nedenle Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği geliştirmeye yönelik bir çalışma yürütülmüştür. Çalışma sonunda elde edilen ölçek EK 4'te sunulmuştur.

Robotik kodlama etkinliklerine ilişkin öğrenci yaşantılarını tespit etmek amacıyla ihtiyaç duyulan Etkinlik Algısı Ölçeğinin Türkçesine rastlanılmamış, bu nedenle Deci, Eghrari, Patrick, ve Leone (1994) tarafından ilgi çekici olmayan (sıkıcı) bilgisayar görevleri için geliştirilen etkinlik algısı ölçeğinin Türkçeye uyarlama çalışması yapılmıştır. Yapılan çalışma neticesinde elde edilen Etkinlik Algısı Ölçeğinin Türkçe uyarlaması EK 5'de sunulmuştur. Robotik kodlama etkinliklerine ilişkin öğrenci yaşantılarını tespit etmek amacıyla öğrencilerle yapılacak görüşmelerde öğrencilere yöneltilecek görüşme sorularını geliştirme çalışması yürütülmüş, geliştirilen yarı yapılandırılmış görüşme soruları EK 6'da ve EK 7'de sunulmuştur.

3.5.1. Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği

Veriler iki bölümden oluşan bir ölçme aracı ile toplanmıştır (EK 4). Ölçme aracının birinci bölümünü cinsiyet, sınıf düzeyi, daha önce Scratch'le programlama dersi alıp almadığı, halen Scratch'le programlama dersi alıp almadığı, daha önce Scratch for Arduino robotik dersi alıp almadığı, ders dışında Scratch programına çalışma olanağı olup olmadığı, ders dışında Scratch programına çalışma sıklığı, ders dışında Scratch for Arduino ile robotik çalışma olanağına ilişkin soruların yer aldığı demografik bilgi kısmı oluşturmuştur. İkinci bölümde ise Blok Temelli Programlamaya İlişkin 12 maddeden oluşan sorular yer almaktadır. Ölçekte toplamda 12 madde yer almakta olup sorular 5'li Likert tipinde hazırlanmış ve 1- Hiç Güvenmiyorum, 2- Biraz Güveniyorum, 3- %50 / %50, 4- Oldukça Güveniyorum, 5- Tamamen Güveniyorum belirtecek şekilde puanlanmıştır.

3.5.1.1. Ön Uygulama ve Madde Havuzunun Oluşturulması

Hacettepe Üniversitesi Eğitim Fakültesi Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümünde görevli iki akademisyen, araştırmacı ve diğer bir bilişim teknolojileri öğretmeni tarafından blok temelli programlamaya ilişkin öz-yeterlik algısı ifadelerini tespit etme çalışması yapılmıştır. Altun ve Mazman (2012) tarafından Türkçeye uyarlaması yapılmış olan Programlamaya İlişkin Öz-yeterlik Algısı Ölçeğini (Aşkar ve Davenport, 2009) ortaokul düzeyinde Scratch programına yönelik kullanılabilir şekilde uyarlama çalışmasının, MIT laboratuvarlarında geliştirilen Scratch'in programlama eğitiminin genç insanlara verilmesinin önündeki engelleri kaldırmaya yönelik özel tasarımı nedeniyle mümkün olmadığı görülmüştür. Maddelerin geliştirilmesi sürecinde, Scratch yazılımını geliştiren MIT Yaşam Boyu Okul Öncesi Grubu tarafından hazırlanan bir makalede (Resnick ve diğerleri, 2009) belirtilen, Scratch'te yer alan programlamaya ilişkin özellikler referans alınmıştır. Bunun yanı sıra Altun ve Mazman (2012) tarafından Türkçeye uyarlaması yapılmış olan Programlamaya İlişkin Öz-yeterlik Algısı Ölçeğinde bulunan ve uyarlaması yapılabilen maddeler de hazırlanan ölçeğe dâhil edilmiştir. Hazırlanan bu öz-yeterlik algısı ölçeği bir devlet okulunda eğitim gören, blok temelli programlama becerisinin iyi düzeyde olduğu gözlemlenen 2 erkek - 2 kız öğrenciye ve blok temelli programlama becerisinin düşük düzeyde olduğu gözlemlenen 2 erkek - 2 kız öğrenciye olmak üzere toplam 8 öğrenciye uygulanmıştır. Uygulamanın ardından öğrencilerle birebir görüşme tekniği ile soruları anlayıp anlamadıklarını tespit etmeye yönelik görüşmeler gerçekleştirilmiştir. Anlaşılmayan, net olmayan ya da eksik ifade edilen maddelere yönelik yapılan görüşmelerle elde edilen veriler neticesinde ölçekte düzeltmelere gidilmiştir. Yeniden düzenlenen sorularla aynı çalışma farklı 8 öğrenci ile tekrarlanmıştır. Alınan dönütler neticesinde tekrar düzeltmeler yapılmıştır. Bu süreç 4 defa tekrarlanmış ve ön çalışma neticesinde ölçeğe 14 maddelik son şekli verilmiştir.

3.5.1.2. Verilerin Toplanması

Ölçek maddeleri oluşturulduktan sonra, ölçek çevrimiçi erişim için hazır hale getirilmiştir. Öğrencilerin Scratch'le kodlama eğitimi almış veya almakta olduğunun tespit edilmesi ve ilgili okullarda görev yapan bilişim teknolojileri öğretmenlerinin araştırmaya gönüllü olmaları nedeniyle elverişli bulunan Ankara, İzmir, Uşak illerindeki 4 devlet okulunda eğitim gören 329 ortaokul öğrencisi ile ölçek

uygulanmıştır. Geliştirilen ölçek araştırma kapsamında düzenlenecek olan robotik kodlama etkinliklerinin öğrencilerin blok temelli kodlama ilişkin özyeterlik algularına etkisini tespit etmek amacıyla kullanılacağı için, ölçek araştırmacı olarak kendi görev yaptığım okulda uygulanmamıştır. “Google Formlar” ile hazırlanan çevrimiçi ölçek kullanılarak toplanan veriler SPSS 24 ve Lisrel 8.7 programlarında analiz edilmek üzere Microsoft Excel (.xls) formatında yerel bilgisayarın sabit diskine kaydedilmiştir.

3.5.1.3. Verilerin Analizi

Verilerin analizine geçilmeden önce kayıp verilen tespit çalışması yapılmıştır. Öğrencilerin ölçeklere verdikleri yanıtlarda kayıp verilerin ilgili maddelere verilen toplam yanıtlar içerisindeki oranı %5'ten küçük olduğu tespit edilmiştir. Bu doğrultuda, tespit edilen eksik veriler o maddeye ilişkin ortalamayı değiştirmemesi nedeniyle ilişkili maddenin ortalamasıyla doldurulmuştur.

Maddelere ilişkin normallik varsayımını kontrol etmek için basıklık ve çarpıklık katsayıları hesaplanmıştır. 4. maddenin çarpıklık katsayı değerinin -3,36 ve basıklık katsayı değerinin 12,001 olduğu, 5. maddenin çarpıklık değerinin -3,131 ve basıklık katsayı değerinin 10.424 olduğu görülmüştür. Bu değerlerin kabul edilebilir değerler olan $10 < \text{basıklık} < 10$ ve $-3 < \text{çarpıklık} < 3$ aralığı dışında olduğu (bkz. Kline, 2005) görülmüş ve aşağıdaki söz konusu maddeler ölçekten çıkartılmıştır:

4. Madde: Scratch programında projeme yeni karakter ekleyebilirim.

5. Madde: Scratch programında projeme yeni sahne ekleyebilirim.

Verilerin faktör analizi için uygun olup olmadığının tespit edilmesi amacıyla Kaiser-Meyer-Olkin (KMO) ve Barlett Küresellik testleri yapılmıştır. KMO örneklem yeterlilik indeksi, %60'dan yüksek olması ve Barlett küresellik testinin anlamlı çıkması (< 0.05) verilerin faktör analizi için uygun olduğunu göstermektedir (Büyüköztürk, 2007).

Bu çalışmadaki verilerin KMO ve Bartlett küresellik testi sonuçları Tablo 3.4'te gösterilmektedir.

Tablo 3.4: KMO and Bartlett Test Sonuçları

Kaiser-Meyer-Olkin Örneklem Yeterliği Ölçümü		,887
ωBartlett Küresellik Testi	Yaklaşık Ki-Kare	1835,038
	df	66
	p	,000

Tablo 3.4'den görüldüğü gibi KMO değeri örneklem yeterliliğin iyi düzeyde olduğunu gösteren bir değer olan .887 bulunmuştur. Verilerin faktör analizi için uygunluğunu gösteren Bartlett küresellik testi sonucu da ($\chi^2 = 1835,038$ $p < 0.05$) anlamlı bulunmuştur.

Ölçeğin yapı geçerliliği için açımlayıcı ve doğrulayıcı faktör analizleri yapılmıştır. Doğrulayıcı faktör analizinde kurulan modelin iyilik uyumunun değerlendirilmesinde iyilik uyum indekslerinden Ki Kare (χ^2), AGFI, GFI, NFI, CFI, RMSEA ve S-RMR değerleri esas alınmıştır.

Güvenirlik çalışması için ölçeğin iç tutarlılık katsayısı (Cronbach alpha) hesaplanmıştır. Bunun yanı sıra her bir maddenin ölçeğin tümü ile tutarlılığını gösteren madde toplam korelasyonları hesaplanmıştır.

Ölçekten elde edilen puanlar üzerinden grup içi karşılaştırmalar için Eşleştirilmiş-Örneklem t Testi (Paired-Samples t Test) analizleri yapılmıştır. Grupların bazı değişkenler açısından bağımlı değişken verileri üzerinden karşılaştırıldığı analizler için bağımsız-örneklem t testi yapılmıştır.

3.5.1.4. Madde Geçerliği Analizleri

Maddelerinin ayırt edicilik gücünün belirlenmesi, madde toplam test korelasyon katsayıları ile yapılmıştır. Bu yöntem ile, testin her bir maddesinden alınan puan ile testin tamamından alınan ilişki gösterilmektedir. Madde toplam korelasyonlarının 0.3'ten küçük olması o maddenin diğer maddelerle aynı yapıyı ölçmediğini, yani maddenin ölçeğin geri kalanı korelasyonunun zayıf olduğunu gösterir (Field, 2005). Tablo 3.5'te ölçeğe ait madde toplam korelasyonları yer almaktadır.

Tablo 3.5: Madde Toplam Korelasyon Değerleri

Faktör	Madde	Madde Top. Korelasyonu
Basit Blok Temelli Programlama Görevleri	M3. Bir karaktere herhangi bir hareket vermek istediğimde, Scratch'te bunu nereden yapabileceğimi bilirim.	0.491
	M6. Sahnedeki karakteri istediğim hızda hareket ettirebilirim.	0.536
	M7. Sahnedeki karakteri sürekli hareket ettirebilirim.	0.578
	M8. Scratch'te bir karakterin görünümünü (kostüm, renk, boyut, konuşma gibi) bir koşula bağlı olarak (örneğin: eğer ise) değiştirebilirim.	0.597
	M9. Scratch'te bir karakterin hareketini (hızı, yönü, konumu gibi) bir koşula bağlı olarak (örneğin: eğer ise) değiştirebilirim.	0.640
Karmaşık Blok Temelli Programlama Görevleri	M1. Scratch'te yazılmış bir program (yazılar) gördüğümde, çalıştırıldığında neler olacağını söyleyebilirim.	0.629
	M2. Başkası tarafından hazırlanan bir programı (yazılarını) okuyup anlayabilirim.	0.638
	M10. Bir oyunda kullanıcının elde ettiği puan değerinin tutulacağı bir değişken oluşturabilirim.	0.585
	M11. Bir oyunda istenilenler başarıldıkça "Puan" veya "Skor" değerinin arttığı veya azaldığı bir program hazırlayabilirim.	0.656
	M12. İstenilenler açıkça tanımlandığında oldukça karmaşık ve uzun kodlardan (yazılardan) oluşan bir oyun hazırlayabilirim.	0.702
	M13. Scratch'te hazırlanan bir programdaki hataları bulabilirim.	0.589
	M14. Scratch'te hazırlanan bir programdaki hataları düzelterip çalışabilir hale getirebilirim.	0.652

Tablo 3.5'te görüldüğü üzere ölçeğe ilişkin hiçbir maddenin madde toplam korelasyonu 0.3'ün altında olmayıp, toplamda on iki maddenin madde toplam korelasyonları 0.491 - 0.702 arasında değişmektedir.

3.5.1.5. Yapı Geçerliđi Analizleri

İlk olarak ölçeđin tümüne toplamda 12 maddenin faktör yüklerini ve faktör yapılarını belirlemek amacıyla açımlyıcı faktör analizi yapılmıřtır. Analiz sonucunda faktörler altındaki yük deđerleri incelenmiř, ortaya çıkan faktörler yüklerine göre 12 maddenin öz deđeri 1'den büyük olan iki faktör altında toplandıđı dođrulanmıřtır. 3., 6., 7., 8. ve 9. maddeler birinci faktörde toplanırken 1., 2., 10., 11., 12., 13. ve 14. maddeler ikinci faktörde toplanmıřtır.

Ramalingam ve Wiedenbeck (1998) tarafından geliřtirilen programlamaya iliřkin öz-yeterlik algısı ölçeđi “özdüzenleme”, “bađımsızlık ve kararlılık”, “basit programlama görevleri” ve “karmařık programlama görevleri” olarak dört faktörde toplanmıřtır. Bu ölçeđin Türkçeye uyarlama çalıřmasında ise ölçek iki faktör altında toplanmıř ve faktörler “basit programlama görevleri” ve “karmařık programlama görevleri” olarak isimlendirilmiřtir (Altun ve Mazman, 2012).

Bu çalıřma kapsamında geliřtirilen blok temelli programlamaya iliřkin öz-yeterlik algısı ölçeđinde, ölçek maddelerinin altında toplandıđı iki faktörün isimlendirilmesinde, Resnick ve diđerleri'nin (2009) çalıřmasında blok temelli programlamaya yönelik kavramların isimlendirmesi ve Altun ve Mazman (2012) tarafından yapılan çalıřmada kullanılan faktör isimlendirmesi referans alınmıřtır. Açımlyıcı faktör analizinden elde edilen 1. faktör altında toplanan maddeler Resnick ve diđerleri'nin (2009) makalesinde “temel programlama kavramlarına” yönelik görevlere karřılık gelmektedir. 2. faktör altında toplanan maddelerden deđiřken oluřturma ve kullanmaya yönelik görevleri ieren 10. ve 11. maddeler “daha fazla uzman desteđi ile öđrenilebilecek kavramlara” yönelik görevlere karřılık gelmekte iken 1., 2., 12., 13. ve 14. maddeler ise Altun ve Mazman (2012) tarafından Türkçeye uyarlanan Programlamaya iliřkin öz-yeterlik algısı ölçeđindeki “karmařık programlama görevlerine” iliřkin faktör altında toplanan maddelerin blok temelli programlamaya yönelik küçük yař grubunun anlayabileceđi ifadelere sahip uyarlaması řeklinedir.

Yapılan deđerlendirme sonucunda, blok temelli programlama iliřkin öz-yeterlik algısı ölçeđine ait iki faktörün, 5 maddeden oluřan (3., 6., 7., 8. ve 9.) birinci faktörü “basit blok temelli programlama görevleri” olarak, 7 maddeden oluřan (1., 2., 10., 11., 12., 13. ve 14.) ikinci faktörü ise “karmařık blok temelli programlama görevleri” olarak isimlendirilmiřtir. Ölçek 5'li Likert tipi olduđundan basit blok temelli

programlamaya ilişkin öz-yeterlik algısı için alınabilecek maksimum puan 25, minimum puan ise 5, karmaşık blok temelli programlamaya yönelik öz-yeterlik algısı için alınabilecek maksimum puan 35, minimum puan ise 5 puan olarak belirlenmiştir.

3.5.1.6. Açımlayıcı Faktör Analizi Sonuçları

Blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeğinin, son hali verilen 12 maddelik formuna temel bileşenler analizi tekniği kullanılarak açımlayıcı faktör analizi yapılmıştır. Analiz sonucunda faktörler altındaki yük değerleri incelenmiş, ortaya çıkan faktörler yüklerine göre öz değeri (eigenvalue) 1'den büyük olan (5.612, 1.375) iki faktör altında toplandığı bulunmuştur. Bu iki faktörden 5 maddelik birinci faktör toplam varyansın %11.462'ini, 7 maddelik ikinci faktör ise toplam varyansın %46.763'ünü açıklamıştır.

Programlamaya ilişkin öz-yeterlik algısı ölçeğinin temel bileşenler analizi sonuçları Tablo 3.6'da verilmektedir.

Tablo 3.6: Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği Temel Bileşenler Analizi Sonuçları

Faktör	Madde	Faktör Yükleri	
		1	2
Basit Blok Temelli Programlama Görevleri	M3. Bir karaktere herhangi bir hareket vermek istediğimde, Scratch'te bunu nereden yapabileceğimi bilirim.	0.636	-
	M6. Sahnedeki karakteri istediğim hızda hareket ettirebilirim.	0.753	-
	M7. Sahnedeki karakteri sürekli hareket ettirebilirim.	0.789	-
	M8. Scratch'te bir karakterin görünümünü (kostüm, renk, boyut, konuşma gibi) bir koşula bağlı olarak (örneğin: eğer ise) değiştirebilirim.	0.769	-
	M9. Scratch'te bir karakterin hareketini (hızı, yönü, konumu gibi) bir koşula bağlı olarak (örneğin: eğer ise) değiştirebilirim.	0.690	-
Karmaşık Blok Temelli Programlama Görevleri	M1. Scratch'te yazılmış bir program (yazılar) gördüğümde, çalıştırıldığında neler olacağını söyleyebilirim.	-	0.535
	M2. Başkası tarafından hazırlanan bir programı (yazılarını) okuyup anlayabilirim.	-	0.578
	M10. Bir oyunda kullanıcının elde ettiği puan değerinin tutulacağı bir değişken oluşturabilirim.	-	0.725
	M11. Bir oyunda istenilenler başarılıyken “Puan” veya “Skor” değerinin arttığı veya azaldığı bir program hazırlayabilirim.	-	0.824
	M12. İstenilenler açıkça tanımlandığında oldukça karmaşık ve uzun kodlardan (yazılardan) oluşan bir oyun hazırlayabilirim.	-	0.767
	M13. Scratch'te hazırlanan bir programdaki hataları bulabilirim.	-	0.726
	M14. Scratch'te hazırlanan bir programdaki hataları düzelterek çalışabilir hale getirebilirim.	-	0.670

Tablo 3.6'da görüldüğü üzere faktör analizi sonuçlarına göre; “Basit Blok Temelli Programlama Görevleri” faktörü altında 5 madde yer almakta ve maddelerin yük değerleri 0.789 ile 0.636 arasında değişmektedir. Bu faktör değişkenliğinin %11.462'ini açıklamaktadır. “Karmaşık programlama görevleri” faktörü altında ise

toplamda 7 madde yer almakta, maddelerin yük değerleri 0.824 ile 0.578 arasında değişmektedir. Bu faktör değişkenliğinin %46.763'ünü açıklamaktadır.

3.5.1.7. Doğrulayıcı Faktör Analizleri

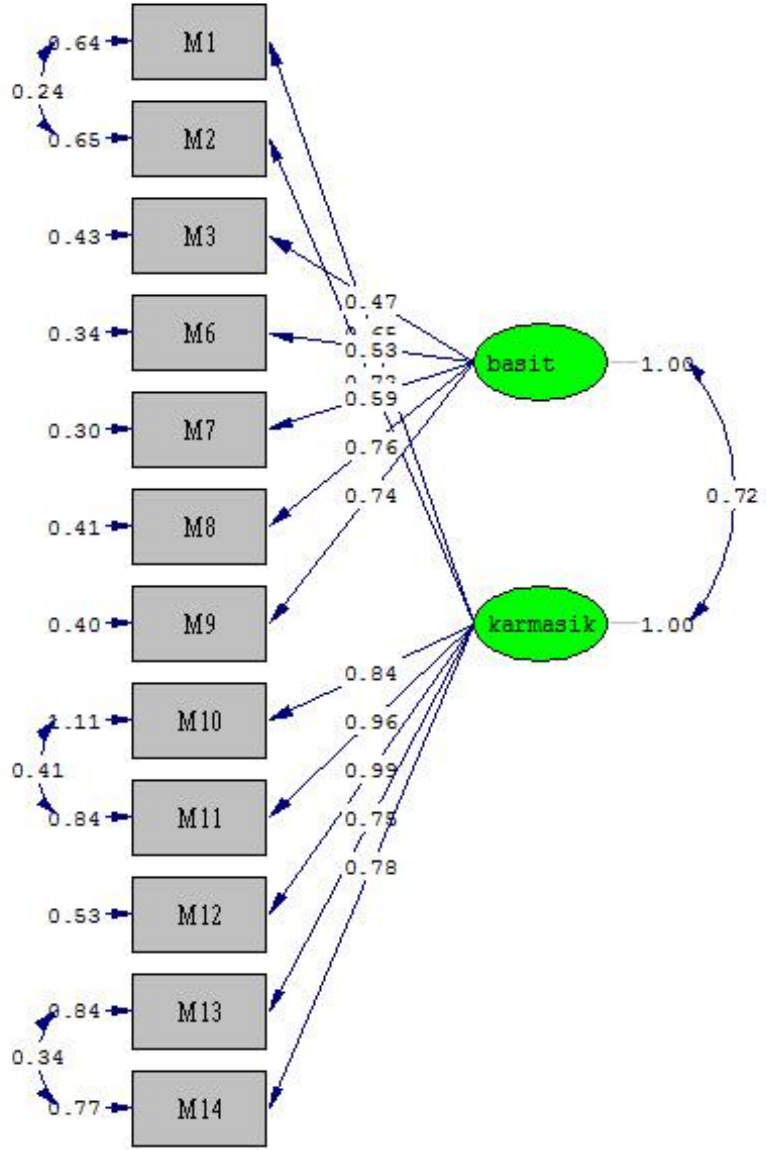
Açımlayıcı faktör analizi ile iki faktör ("Basit Blok Temelli Programlama Görevleri ve "Karmaşık Blok Temelli Programlama Görevleri") ve 12 maddeden (gösterge değişken) elde edilen yapının doğrulanması amacıyla Lisrel 8.70 programı kullanılarak birinci düzey doğrulayıcı faktör analizi yapılmıştır. İlk olarak iki gizil değişken ("Basit Blok Temelli Programlama Görevleri ve "Karmaşık Blok Temelli Programlama Görevleri") ve 12 gösterge değişken (M1, M2.....M12) ile oluşturan model doğrulayıcı faktör analizine tabi tutularak elde edilen uyum indeksleri incelenmiştir. Elde edilen iyilik uyum indeksleri uygun değer aralığında bulunmadığından, öneriler modifikasyon indeksleri incelenmiş ve aynı faktörün açıkladığı maddeler olan "M1 ve M2", "M10 ve M11" ve "M13 ve M14" arasındaki hata kovaryansları serbest bırakılmıştır. Modifikasyonlar yapıldıktan sonra program tekrar çalıştırılarak analiz edilmiştir. Analiz sonucunda elde edilen değerler Tablo 3.7'de gösterilmiştir.

Tablo 3.7: Birinci Düzey Doğrulayıcı Faktör Modeli Uyum İndeksleri

Uyum Ölçütleri	Mükemmel Uyum	Kabul Edilebilir Uyum	Model Değeri
RMSEA	$0 < RMSEA < 0.05$	$0.05 < RMSEA < 0.08$	0.061
S-RMR	$0 \leq S-RMR \leq 0.05$	$0.05 < S-RMR < 0.1$	0.047
NNFI	$0.97 \leq NNFI \leq 1$	$0.95 < NNFI < 0.97$	0.98
NFI	$0.97 \leq NFI \leq 1$	$0.95 < NFI < 0.97$	0.97
CFI	$0.97 \leq CFI \leq 1$	$0.95 < CFI < 0.97$	0.98
GFI	$0.95 \leq GFI \leq 1$	$0.90 < GFI < 0.95$	0.95
AGFI	$0.90 \leq AGFI \leq 1$	$0.85 < AGFI < 0.90$	0.92
IFI	$0.95 \leq IFI \leq 1$	$0.90 < IFI < 0.95$	0.98

Tablo 3.7'de görüldüğü üzere modelin tüm değerlerinin oldukça iyi uyum gösterdiği bulunmuştur (bkz. Schermelleh-Engel & Moosbrugger, 2003; Sümer, 2000). Birinci

Düzyey dođrulatoryı Faktör analizinin bađlantı diyagramı (path diagram) Őekil 7'de yer almaktadır.



Chi-Square=110.57, df=50, P-value=0.00000, RMSEA=0.061

Őekil 7: Birinci Düzey Doğrulatoryı Faktör Analizi Bađlantı Diyagramı (standart katsayılar)

3.5.1.8. Güvenirlik Analizi

Ölçeđin tümünün iç tutarlık katsayısı (Cronbach alpha) .893 olarak bulunmuştur. Her bir faktör için hesaplanan güvenirlik katsayıları Tablo 3.8'de gösterilmektedir.

Tablo 3.8: Faktörlere Ait Güvenirlik Katsayıları ve Madde Toplam Korelasyonları

Faktör	Madde	α
Basit Blok Temelli Programlama Görevleri	M3	0.828
	M6	
	M7	
	M8	
	M9	
Karmaşık Blok Temelli Programlama Görevleri	M1	0.868
	M2	
	M10	
	M11	
	M12	
	M13	
	M14	

Tablo 3.8’de görüldüğü gibi faktörlere ait güvenirlilik katsayılarının tümü (cronbach alpha) 0.8’in üzerinde olup bu sonuç ölçme aracının güvenirliliğinin oldukça iyi olduğunu göstermektedir (George ve Mallery, 2003).

3.5.1.9. Ölçeğin Uygulanmasına İlişkin Bazı Değişkenler Açısından Betimsel Analizler

Bireylerin programlamaya ilişkin öz-yeterlik algısı puanlarını etkileyen faktörleri belirlemek amacıyla çeşitli değişkenler (cinsiyet, daha önce Scratch’le programlama dersi alma, daha önce Scratch’le program yazma, halen Scratch’le programlama dersi alma, halen Scratch’le program yazma) açısından öz-yeterlik algısı puanları incelenmiştir.

Bireylerin cinsiyet, daha önce Scratch'le programlama dersi alma, daha önce Scratch'le program yazma, halen Scratch'le programlama dersi alma, halen Scratch'le program yazmaya ilişkin ölçek puanlarının ortalamaları ve standart sapmaları Tablo 3.9'da verilmiştir.

Tablo 3.9: Demografik Verilere İlişkin Ortalama ve Standart Sapmalar

		Basit Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı (max. 25)			Karmaşık Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı (max. 35)	
		N	Ortalama	Standart Sapma	Ortalama	Standart Sapma
Cinsiyet	Kız	163	22,349	3,630	26,276	6,139
	Erkek	166	22,410	3,131	26,405	6,408
Daha Önce Scratch'le Programlama Dersi Aldım	Evet	203	22,393	3,449	26,955	6,301
	Hayır	126	22,359	3,287	25,353	6,107
Daha Önce Scratch'le Program Yazdım	Evet	209	22,548	3,288	27,193	6,071
	Hayır	120	22,088	3,536	24,858	6,351
Halén Scratch'le Programlama Dersi Alıyorum	Evet	261	22,621	2,968	26,983	5,778
	Hayır	68	21,46	4,55	23,88	7,42
Halén Scratch'le Program Yazıyorum	Evet	250	22,811	2,788	27,340	5,790
	Hayır	79	21,02	4,56	23,18	6,69

İlk olarak bireylerin basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının cinsiyete göre farklılaşıp farklılaşmadığını ortaya koymak amacıyla bağımsız örneklem t testi yapılmıştır. Bağımsız örneklem t testi sonuçlarına göre bireylerin basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ($\bar{X}_{kız} = 22,349$; $\bar{X}_{erkek} = 22,410$; $t = 0.164$, $p = 0.870$) ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ($\bar{X}_{kız} = 26,276$; $\bar{X}_{erkek} = 26,405$; $t = 0.187$, $p = 0.852$) cinsiyete göre farklılaşmadığı bulunmuştur.

Daha önce Scratch'le programlama dersi alan ve almayan öğrenciler arasında basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının anlamlı derecede farklılaşmadığı bulunmuştur ($\bar{X}_{dersalan}=22,393$; $\bar{X}_{dersalmayan}=22,359$; $t=0.87$, $p=0.931$). Daha önce Scratch'le programlama dersi alan öğrencilerin karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanları daha önce Scratch'le programlama dersi almayan öğrencilere göre anlamlı derecede yüksek olarak bulunmuştur ($\bar{X}_{dersalan}= 26,955$; $\bar{X}_{dersalmayan}=25,353$; $t=2.268$, $p=0.024$).

Daha önce Scratch'le program yazan ve yazmayan öğrenciler arasında basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının anlamlı derecede farklılaşmadığı bulunmuştur ($\bar{X}_{prgyazan}=22,548$; $\bar{X}_{prgyazmayan}=22,088$; $t=1.188$, $p=0.236$). Daha önce Scratch'le program yazan öğrencilerin karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanları daha önce Scratch'le program yazmayan öğrencilere göre anlamlı derecede yüksek olarak bulunmuştur ($\bar{X}_{prgyazan}= 27,193$; $\bar{X}_{prgyazmayan}=24,858$; $t=3.301$, $p=0.01$).

Halen Scratch'le programlama dersi alan öğrencilerin hem basit hem de karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanları halen Scratch'le programlama dersi almayan öğrencilere göre anlamlı derecede yüksek olarak bulunmuştur (basit: $\bar{X}_{halendersalan}= 22,621$; $\bar{X}_{halendersalmayan}=21,46$; $t=2.551$, $p=0.011$; karmaşık: $\bar{X}_{halendersalan}= 26,983$; $\bar{X}_{halendersalmayan}=23,88$; $t=3.705$, $p=0.00$).

Halen Scratch'le program yazan öğrencilerin hem basit hem de karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanları halen Scratch'le program yazmayan öğrencilere göre anlamlı derecede yüksek olarak bulunmuştur (basit: $\bar{X}_{halenprgyazan}=22,811$; $\bar{X}_{halenprgyazmayan}=21,02$; $t=4,218$, $p=0.00$; karmaşık: $\bar{X}_{halenprgyazan}= 27,340$; $\bar{X}_{halenprgyazmayan}=23,18$; $t=5.355$, $p=0.00$).

3.5.2. Robotik Kodlama Etkinliklerine İlişkin Öğrenci Yaşantılarını Tespit Etmeye Yönelik Veri Toplama Araçları

Robotik kodlama etkinliklerinin öğrencilerin kodlamaya ilişkin öz-yeterlik algılarında anlamlı bir değişim meydana getirip getirmediğine yönelik bulguları etkileyen çeşitli değişkenler olabileceği varsayımıyla etkinlikler esnasında ve her etkinliğin hemen bitiminde öğrenci yaşantılarına yönelik nitel ve nicel veriler toplanmıştır. Verilerin toplanması sürecinde etkinlikleri düzenleyen öğretmen tarafından katılımcı gözlemci olarak notlar alınmıştır.

Bu verilerin toplanmasında nitel gözlem ve görüşme tekniklerinden yararlanılmış, ayrıca her etkinliğin sonunda tüm öğrencilere etkinlik algısı ölçeği uygulanmıştır. Gözlem ve görüşme notları yazılı olarak kayıt altına alınırken, etkinlik algısı ölçeği verileri ise “Google Formlar” aracılığıyla her etkinliğin sonunda çevrimiçi olarak toplanmış, MS. Excel (.xls) formatında bilgisayarın yerel diskine kaydedilmiştir.

3.5.2.1. Etkinlik Algısı Ölçeği

Robotik Kodlama etkinliklerine ilişkin öğrenci yaşantılarını tespit etmek amacıyla kullanılacak Etkinlik Algısı Ölçeğinin Türkçesine alanyazında rastlanılmaması nedeniyle Deci, Eghrari, Patrick, ve Leone (1994) tarafından ilgi çekici olmayan (sıkıcı) bilgisayar görevleri için geliştirilen etkinlik algısı ölçeğinin Türkçeye uyarlama çalışması yapılmıştır. Türkçeye uyarlaması yapılan maddeler iki akademisyen ve iki öğretmen tarafından incelenmiştir. Yapılan değerlendirmeler sonucunda kültürler ve eğitim sistemlerindeki farklılıklar nedeniyle 25 maddelik ölçeğin ancak 11 maddesinin çevirisinin ülkemiz 5. sınıf öğrencileri tarafından doğru anlaşılacağı öngörülmüş ve ölçeğe son hali verilmiştir (EK 5).

3.5.2.2. Gözlemlere ve Görüşmelere Ait Konu Başlıkları

Etkinlikler esnasında yapılan gözlemler; etkinliğe ilgi duyma, etkinliği eğlenceli bulma, etkinliği yapmaya istekli olma, etkinliği gerçekleştirirken zorlanma durumlarına yönelik öğrencileri olumlu ve olumsuz yönde etkileyen durumları içermektedir. Bu durumlar hem düzenlenen etkinliğin tasarımından kaynaklı hem de robotik kodlama etkinliklerinin doğasından kaynaklı olabileceğinden dolayı sorular ve gözlemler her iki yönlü düzenlenmiştir. Bu değişkenlere yönelik gözlem notları her etkinlik için ayrı ayrı raporlanmıştır. Gözlem notlarından elde edilen verilere ilave olarak öğrencilerle birebir görüşmeler yapılmış, öğrencilere EK 6’da ve EK 7’de sunulmuş olan görüşme soruları sözlü olarak yöneltilmiştir.

3.5.2.3. Verilerin analizi

Etkinlik Algısı Ölçeğinden elde edilen nicel verilere ilişkin betimsel analizler yapılmış ve analiz sonuçları raporlanmıştır. Verilerin analizinde SPSS 24 programından yararlanılmıştır. Uygulanan robotik kodlama etkinlikleri sonunda öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının grup içi değişimlerini analiz etmek amacıyla Eşleştirilmiş-Örneklem t Testi (Paired-Samples t Test) yapılmıştır. Öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algısı

puanlarının çeşitli deęişkenlere göre gruplar içi deęişimlerini analiz etmek amacıyla son-test puanlarından ön-test puanlarının çıkarılması ile elde edilen fark deęerleri üzerinde Baęımsız-Örneklem t Testi (Independent-Samples t Test) yapılmıştır. Nitel verilerin analizinde ise içerik analizi yönteminden yararlanılmış, yapılan yüz yüze görüşme ve gözlemlerden elde edilen veriler kategoriler altında toplanmıştır.

4. BULGULAR

Bu bölümde araştırmada incelenen problemlere ilişkin toplanan verilerin istatistiksel analizi yapılarak elde edilen bulgulara yer verilmektedir. Yapılan analizler sonucunda elde edilen bulgu ve sonuçlar her alt problem ışığında aşağıdaki bölümde yer almaktadır.

4.1. Ön Analizler

Ortaokul 5. sınıf düzeyinde uygulanan robotik blok temelli programlama eğitimi sonunda bireylerin blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarında meydana gelen değişimi etkileyen faktörleri belirlemek amacıyla çeşitli değişkenler (cinsiyet, evde bilgisayar sahip olma, evde internet bağlantısına sahip olma, ders dışında Scratch programına çalışma olanağı) açısından öz-yeterlik algısı puanlarında meydana gelen farklılaşmalar incelenmiştir. Bunun öncesinde blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarında çeşitli değişkenler açısından grup içi anlamlı düzeyde bir farklılık bulunup bulunmadığına yönelik bireylerin hazırbulunuşluk düzeylerine ilişkin analizlere ihtiyaç duyulmuştur.

Etkinlikler öncesinde çalışma grubunun çeşitli değişkenler (cinsiyet, evde bilgisayar sahip olma, evde internet bağlantısına sahip olma, ders dışında Scratch programına çalışma olanağı) açısından basit ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı ön-test puanlarının farklılaşp farklılaşmadığını ortaya koymak amacıyla bağımsız örneklem t testi yapılmıştır. Tablo 4.1'de basit blok temelli programlamaya ilişkin öz-yeterlik algısı ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısına ilişkin grup içi çeşitli değişkenler açısından blok temelli programlamaya ilişkin öz-yeterlik algısı ön test puanları yer almaktadır.

Tablo 4.1: Çeşitli Değişkenler Açısından Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Ön Test Puanları

				N	\bar{X}	SS	t	df	p	Ort. Farkı	Alt Değ.	Üst Değ.
				%95 Güven Aralığı Değ.								
Cinsiyet	Kız	Basit	27	21.89	3.07	0.17	56	0.86	0.14	-1.49	1.77	
			31	21.75	3.12							
	Erkek	Karmaşık	27	25.87	4.95	-0.75	56	0.46	-0.95	-3.50	1.60	
			31	26.82	4.73							
Evde Bilgisayar Sahipliği	Var	Basit	34	22.27	2.85	1.36	56	0.18	1.10	-0.52	2.73	
			24	21.17	3.31							
	Yok	Karmaşık	34	26.89	4.78	0.95	56	0.35	1.22	-1.36	3.79	
			24	25.67	4.88							
Evde İnternet Bağlantısı Sahipliği	Var	Basit	24	22.00	2.83	0.39	56	0.70	0.32	-1.33	1.97	
			34	21.68	2.26							
	Yok	Karmaşık	24	26.14	4.42	-0.32	56	0.75	-0.41	-3.00	2.18	
			34	26.55	5.14							
Ders Dışında Scratch Programına Çalışma Olanlığı	Var	Basit	20	22,48	1,98	1.21	56	0.23	1.02	-0.67	2.71	
			38	21,46	3,48							
	Yok	Karmaşık	20	26,70	5,09	0.37	56	0.72	0.49	-2.20	3.18	
			38	26.21	4.73							

Bağımsız örneklem t testi sonuçlarına göre bireylerin hem basit hem de karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı ön-test puanlarının cinsiyete (basit $t=0.17$, $p=0.86$, karmaşık $t=-0.75$, $p=0.46$), evde bilgisayar sahipliğine (basit

t=1.36, p=0.18, karmaşık t=0.95, p=0.35), evde internet bağlantısı sahipliğine (basit t=0.39, p=0.70, karmaşık t=-0.32, p=0.75) ve ders dışında Scratch programına çalışma olanağına göre (basit t=1.21, p=0.23, karmaşık t=0.37, p=0.72) anlamlı olarak farklılaşmadığı bulunmuştur.

4.2. Robotik Kodlama Etkinlikleri ve Kodlamaya İlişkin Öz-yeterlik Algısı

Uygulanan robotik kodlama etkinlikleri sonunda öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının grup içi değişimlerini analiz etmek amacıyla Eşleştirilmiş-Örneklem t Testi (Paired-Samples t Test) yapılmıştır. Tablo 4.2’de basit blok temelli programlamaya ilişkin öz-yeterlik algısı ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısına ilişkin grup içi analiz bulguları yer almaktadır.

Tablo 4.2: Robotik Kodlama Eğitimi Sonunda Kodlamaya İlişkin Öz-yeterlik Algıları Grup İçi Farklılaşma Tablosu

			\bar{X}	SS	t	df	p
Grup İçi Değişim	Basit	Ön-Test	21.81	3.07	-5.01	57	0.00
		Son-Test	24.10	1.62			
	Karmaşık	Ön-Test	26.38	4.82	-8.84	57	0.00
		Son-Test	32.26	3.23			

Öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının grup içi değişimleri tablo 4.2’de görülmektedir. Eşleştirilmiş-Örneklem t Testi (Paired-Samples t Test) sonuçlarına göre hem basit hem de karmaşık blok temelli

programlamaya ilişkin öz-yeterlik algısı puanlarında grup içi pozitif yönde anlamlı değişim meydana gelmiştir (basit $t = -5.01$, $p = 0.00$; karmaşık $t = -8.84$, $p = 0.00$).

4.3. Robotik Kodlama Etkinlikleri ve Çeşitli Değişkenlere Göre Kodlamaya İlişkin Öz-yeterlik Algısı

Uygulanan eğitim sonunda öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının çeşitli değişkenlere göre gruplar içi değişimlerini analiz etmek amacıyla son-test puanlarından ön-test puanlarının çıkarılması ile elde edilen fark değerleri üzerinde bağımsız-örneklem t testi (Independent-Samples t Test) yapılmıştır. Tablo 4.3'de basit blok temelli programlamaya ilişkin öz-yeterlik algısı ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısına yönelik grup içi çeşitli değişkenlere göre analiz bulguları yer almaktadır.

Tablo 4.3: Robotik Kodlama Eğitimi Sonunda Kodlamaya İlişkin Öz-yeterlik Algılarının Grup İçi Çeşitli Değişkenlere Göre Analiz Tablosu

			N	\bar{X}	SS	t	df	p	Ort. Farkı	Alt Değ.	Üst Değ.
Cinsiyet	Kız	Basit	27	2.00	3.45	-0.58	56.00	0.56	-0.54	-2.38	1.31
			31	2.54	3.55						
	Erkek	Basit	27	6.50	5.54						
			31	5.34	4.63						
Evde Bilgisayar Sahipliği	Var	Basit	34	1.82	3.07	-1.23	56.00	0.22	-1.13	-2.98	0.72
			24	2.95	3.96						
	Yok	Basit	34	5.23	4.74						
			24	6.79	5.45						
Evde İnternet Bağlantısı Sahipliği	Var	Basit	24	2.08	3.33	-0.37	56.00	0.71	-0.35	-2.22	1.52
			34	2.43	3.62						
	Yok	Basit	24	5.53	4.66						
			34	6.13	5.38						
Ders Dışında Scratch Programına Çalışma Olanğı	Var	Basit	20	1.72	2.42	-0.91	56.00	0.37	-0.87	-2.80	1.06
			38	2.59	3.92						
	Yok	Basit	20	6.05	5.06						
			38	5.79	5.12						

4.3.1. Cinsiyet ve Kodlamaya İlişkin Öz-yeterlik Algısı

İlk olarak robotik kodlama eğitimi sonunda öğrencilerin basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının cinsiyete göre farklılaşp farklılaşmadığını ortaya koymak amacıyla bağımsız örneklem t testi yapılmıştır.

Bağımsız örneklem t testi sonuçlarına göre bireylerin basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ($t= -0.58$, $p=0.56$) ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ($t= 0.87$, $p=0.39$) cinsiyete göre farklılaşmadığı bulunmuştur.

4.3.2. Evde Bilgisayar Sahipliği ve Kodlamaya İlişkin Öz-yeterlik Algısı

Robotik blok temelli eğitim sonunda öğrencilerin basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının evde bilgisayara sahip olup olmamaya göre farklılaşıp farklılaşmadığını ortaya koymak amacıyla yapılan bağımsız örneklem t testi sonuçlarına göre bireylerin basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ($t= -1.23$, $p=0.22$) ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ($t= -1.16$, $p=0.25$) evde bilgisayar sahip olup olmamaya göre farklılaşmadığı bulunmuştur.

4.3.3. Evde İnternet Bağlantısı Sahipliği ve Kodlamaya İlişkin Öz-yeterlik Algısı

Robotik blok temelli eğitim sonunda öğrencilerin basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının evde internet bağlantısına sahip olup olmamaya göre farklılaşıp farklılaşmadığını ortaya koymak amacıyla yapılan bağımsız örneklem t testi sonuçlarına göre bireylerin basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ($t= -0.37$, $p=0.71$) ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ($t= -0.44$, $p=0.66$) evde internet bağlantısına sahip olup olmamaya göre farklılaşmadığı bulunmuştur.

4.3.4. Ders Dışında Scratch Programına Çalışma Olanağı ve Kodlamaya İlişkin Öz-yeterlik Algısı

Robotik blok temelli eğitim sonunda öğrencilerin basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ders dışında Scratch programına çalışma olanağına göre farklılaşıp farklılaşmadığını ortaya koymak amacıyla yapılan bağımsız örneklem t testi sonuçlarına göre bireylerin basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ($t= -0.91$, $p=0.37$) ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının ($t= 0.19$, $p=0.85$) ders dışında Scratch programına çalışma olanağına göre farklılaşmadığı bulunmuştur.

4.4. Öğrenci Yaşantıları

Robotik kodlama etkinliklerine ilişkin öğrenci yaşantılarını tespit etmek amacıyla etkinlikler esnasında elde edilen gözlem ve öğrencilerle yapılan görüşmeler neticesinde elde edilen verilere ilişkin bulgulara bu bölümde yer verilecektir. Ayrıca her bir robotik kodlama etkinliği sonunda öğrencilere uygulanan etkinlik algısı ölçeğinden elde edilen puanların ortalama ve standart sapma değerlerine yer verilmiştir. Öğrenci yaşantıları, etkinlikleri eğlenceli bulma, etkinliklerin kişisel gelişime katkısı, etkinlikleri yapmaya ilişkin isteklilik ve etkinlikleri ilgi çekici bulma başlıkları altında gruplanmış ve alt başlıklar altında bulgulara yer verilmiştir. Öğrencilerin etkinlik algısı ölçeğinden aldıkları puanların ortalama ve standart sapma değerleri Tablo 4.4'de gösterilmektedir.

Tablo 4.4: Robotik Kodlama Etkinliklerine İlişkin Bireylerin Etkinlik Algısı Puanları Ortalama ve Standart Sapma Değerleri Tablosu

	1-) Bu etkinliği yapmak eğlenceliydi.		2-) Bu etkinliğin benim gelişimim için önemli olduğuna inanıyorum.		3-) Bu etkinliği yaparken çok eğlendim.		4-) Bence bu gerçekten önemli bir etkinlikti.		5-) Bu etkinliği yapmak istediğim için yaptım.		6-) Bence bu çok sıkıcı bir etkinlikti.		7-) Bu etkinliği faydalı olduğunu düşündüğüm için tekrar yapmak isterim.		8-) Bu etkinliği yapmanın benim için faydalı olabileceğine inanıyorum.		9-) Bu etkinliğin okulda daha iyi olmama yardımcı olabileceğine inanıyorum.		10-) Bunun çok ilgi çekici bir etkinlik olduğunu düşündüm.		11-) Bu etkinliğin bana kattığı bazı şeyler olduğu için tekrar yapmak isterim.	
	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS
Etkinlik 1	4.95	.29	4.91	.28	4.88	.42	4.80	.51	4.63	.90	1.18	.69	4.70	.65	4.89	.31	4.68	.66	4.64	.69	4.73	.79
Etkinlik 2	4.83	.67	4.84	.36	4.85	.50	4.75	.56	4.67	.76	1.19	.65	4.75	.62	4.80	.49	4.51	.88	4.63	.79	4.75	.65
Etkinlik 3	4.94	.24	4.92	.27	4.94	.24	4.90	.30	4.79	.58	1.41	1.14	4.86	.41	4.88	.39	4.79	.45	4.81	.53	4.94	.24
Etkinlik 4	4.90	.36	4.94	.24	4.86	.40	4.90	.30	4.80	.73	1.31	.97	4.86	.41	4.88	.43	4.76	.51	4.88	.38	4.88	.32
Etkinlik 5	5.00	.00	4.96	.19	4.94	.23	4.92	.33	4.75	.80	1.27	.95	4.91	.29	4.96	.19	4.87	.34	4.89	.37	4.92	.26

4.4.1. Robotik Kodlama Etkinliklerini Eğlenceli Bulmaya İlişkin Etkinlik Algısı

Öğrencilerin etkinliği eğlenceli bulmaya ilişkin algılarına yönelik hazırlanan maddelere verdikleri puanların ortalama ve standart sapma değerleri Tablo 4.5'te gösterilmiştir.

Tablo 4.5: Etkinlik Algısı Ölçeğinin Etkinliği Eğlenceli Bulmaya İlişkin Maddelerine Ait Ortalama ve Standart Sapma Değerleri

	1-) Bu etkinliği yapmak eğlenceliydi.		3-) Bu etkinliği yaparken çok eğlendim.		6-) Bence bu çok sıkıcı bir etkinlikti.	
	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS
Etkinlik 1	4.95	.29	4.88	.42	1.18	.69
Etkinlik 2	4.83	.67	4.85	.50	1.19	.65
Etkinlik 3	4.94	.24	4.94	.24	1.41	1.15
Etkinlik 4	4.90	.36	4.86	.40	1.31	.97
Etkinlik 5	5.00	.00	4.94	.23	1.27	.95

Tablo 4.5 incelendiğinde 5 etkinliğe yönelik “Bu etkinliği yapmak çok eğlenceliydi” maddesine verilen ortalama puanların en düşük 4.83, en yüksek 5.00 olduğu görülmektedir. Yine bu maddeyle aynı içerikteki bir diğer madde olan “Bu etkinliği

yaparken çok eğlendim” maddesinden alınan en düşük ortalama puanın 4.85, en yüksek ortalama puanın ise 4.94 olduğu görülmektedir. “Bence bu çok sıkıcı bir etkinlikti.” maddesine verilen ortalama puanlar ise 1.18 ile 1.41 arasında değişmektedir. Bu değerler öğrencilerin robotik kodlama etkinliklerini eğlenceli bulma algı düzeylerinin oldukça yüksek, sıkıcı bulma algı düzeylerinin ise oldukça düşük olduğunu göstermektedir.

Robotik kodlama etkinliklerinin öğrencilerin kodlamaya ilişkin öz-yeterlik algılarında değişim meydana getirip getirmediğini kontrol edebilmek için aynı zamanda araştırma yapılan okulunun bilişim teknolojileri öğretmeni olan araştırmacı tarafından ön bilgi niteliğinde Scratch’le kodlama etkinlikleri düzenlendiğine yöntem kısmında değinilmişti. Sonraki araştırmalara kaynaklık etmesi amacıyla düzenlenen 5 haftalık Scratch’le kodlama etkinliklerine ilişkin yapılan gözlemlerde, öğrencilerin 4. etkinlikten sonra sıkılmaya başladıkları, saate daha fazla baktıkları, etkinlikleri yapma konusunda daha isteksiz davrandıkları, daha fazla teşvik edilmeye ihtiyaç duydukları gözlemlenmiştir. Robotik kodlama etkinliklerinde ise öğrencilerin oldukça eğlendikleri, etkinlikle ilgili bilgi verirken hemen yapmak için sabırsızlandıkları, arkadaşlarıyla büyük bir istekle etkinlik hakkında bilgi alışverişinde buldukları, derslere katılım zorunlu olmamasına rağmen büyük bir istekle derslere katıldıkları gözlemlenmiştir. 1. etkinlikten itibaren öğrencilerin derse katılım, etkinlik esnasında verilen ipuçlarını uygulama konusunda fazla bir teşvike ihtiyaç duymadan harekete geçme, verilen ipuçlarının ötesinde daha fazla ileri uygulama yapma konusunda istekli ve sorgulayıcı olma konularında oldukça istekli oldukları gözlemlenmiştir. Öğrencilerden zamanın nasıl geçtiğini anlamadığını belirtenler olmuş, derslerin daha fazla saat olması yönünde istekte bulunanlar olmuştur. Bu bulgular öğrencilerin etkinlikler esnasında oldukça yüksek oranda eğlendikleri yönünde elde edilen etkinlik algısı ölçeği bulgularını destekler niteliktedir.

Yapılan gözlemlere ilave olarak etkinliklerin sonrasında yapılan görüşmelerde öğrencilere etkinliği eğlenceli bulma algılarına ilişkin yarı yapılandırılmış görüşme soruları yöneltilmiştir (EK 6). Veriler yanıtla bakıldığında öğrencilerin Scratch’te kodlama yaparken eğlendiklerini belirttikleri ama 4. etkinlikten sonra etkinlikleri önceki kadar eğlenceli bulmadıklarını söyledikleri, sıkılmaya başladıklarını belirttikleri tespit edilmiştir. Ancak robotik kodlama etkinlikleri esnasında eğlenip eğlenmedikleriyle ilgili yöneltilen sorulara verdikleri yanıtlar incelendiğinde, ilk

etkinlikten itibaren çok eğlendiklerini belirttikleri görülmektedir. İlk etkinliğin sonunda öğrencilerin “bilgisayarda komut vererek ışık yaktırmanın çok eğlenceli olduğu, bunun kendilerini çok heyecanlandığı” yönünde yanıtlar verdikleri görülmüştür. Öğrenciler etkinliği eğlenceli bulmaya ilişkin yöneltilen sorulara tüm etkinliklerde benzer yanıtlar vermişler, etkinlikleri çok eğlenceli bulduklarını belirtmişlerdir. Buton, sensör gibi donanımsal parçaları yönetebilmenin çok eğlenceli, heyecan verici olduğunu belirtmişlerdir. Ancak robotik kodlama etkinliklerinin sonucusu olan 5 etkinliğin sonunda, öğrenciler henüz hayallerindeki gibi istedikleri şekilde hareket eden robot, araba gibi akıllı cihazlar tasarlayamadıkları için etkinliklerin yeterli olmadığı, daha fazla ders almak istedikleri yönünde isteklerini iletmışlerdir. Öğrencilerin “etkinlikler esnasında sıkıldığı bölümler oldu mu?” sorusuna yönelik verdikleri yanıtlara bakıldığında öğrencilerin etkinliklerde çok eğlenmelerine rağmen elektrikle ilgili devre kurma süreçlerinde düşük düzeyde de olsa sıkıldığını belirten öğrenciler olduğu görülmektedir. Öğrencilerin etkinlikler esnasında az da olsa sıkılmalarının nedenine ilişkin yöneltilen soruya verdikleri yanıtlar incelendiğinde, elektrik konusunun fen bilgisi dersinde henüz yeterince işlenilmemiş olmasının öğrencilerin zorlanmasına sebep olduğu, bundan dolayı az da olsa sıkıldıklarını belirttikleri görülmektedir.

4.4.2. Robotik Kodlama Etkinliklerinin Kişisel Gelişime Etkisine İlişkin Etkinlik Algısı

Katılımcıların, etkinliğin kişisel gelişimlerine etkisine ilişkin algılarına yönelik ölçek maddelerine verdikleri puanların ortalama ve standart sapma değerleri tablo 4.6’da gösterilmiştir.

Tablo 4.6: Etkinlik Algısı Ölçeğinin Etkinliğin Kişisel Gelişime Etkisine İlişkin Maddelerine Ait Ortalama ve Standart Sapma Değerleri

	2-) Bu etkinliğin benim gelişimim için önemli olduğuna inanıyorum.		4-) Bence bu gerçekten önemli bir etkinlikti.		7-) Bu etkinliği faydalı olduğunu düşündüğüm için tekrar yapmak isterim.		8-) Bu etkinliği yapmanın benim için faydalı olabileceğine inanıyorum.		9-) Bu etkinliğin okulda daha iyi olmama yardımcı olabileceğine inanıyorum.		11-) Bu etkinliğin bana kattığı bazı şeyler olduğu için tekrar yapmak isterim.	
	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS	\bar{X}	SS
Etkinlik 1	4.91	.28	4.80	.51	4.70	.65	4.89	.31	4.68	.66	4.73	.79
Etkinlik 2	4.84	.36	4.75	.56	4.75	.62	4.80	.49	4.51	.88	4.75	.65
Etkinlik 3	4.92	.27	4.90	.30	4.86	.41	4.88	.39	4.79	.45	4.94	.24
Etkinlik 4	4.94	.24	4.90	.30	4.86	.41	4.88	.43	4.76	.51	4.88	.32
Etkinlik 5	4.96	.19	4.92	.33	4.91	.29	4.96	.19	4.87	.34	4.92	.26

Tablo 4.6 incelendiğinde robotik kodlama etkinliklerinin öğrencilerin kişisel gelişimlerine ilişkin etkinlik algılarına yönelik maddelere verdikleri puan ortalamalarının 4.51 ile 4.91 aralığında olduğu görülmektedir. Bu sonuçlara göre öğrencilerin robotik kodlama etkinliklerinin kişisel gelişimlerine ilişkin etkinlik algılarının pozitif yönde oldukça yüksek olduğu söylenebilir. “Bu etkinliğin kişisel gelişimim için önemli olduğuna inanıyorum” maddesine verilen ortalama puanlar 4.91 ile 4.96 arasında değişmekte iken “Bu etkinliğin okulda daha iyi olmama yardımcı olacağına inanıyorum” maddesine verilen ortalama puanlar 4.51 ile 4.87 arasında değişmektedir. Öğrencilerin etkinliğin kişisel gelişimlerine yönelik etkisine

ilişkin algılarının etkinliğin okulda daha iyi olmalarına yardımcı olabileceğine ilişkin algılarına göre az da olsa pozitif yönde ayrıştığı dikkat çekmektedir.

Öğrencilerin robotik kodlama etkinliklerinin kişisel gelişimlerine etkisine ilişkin etkinlik algılarını daha iyi analiz edebilmek amacıyla 5. etkinlik sonunda öğrencilerle görüşmeler gerçekleştirilmiş, etkinliklerin kişisel gelişimlerine etkisine ilişkin etkinlik algılarını tespit etmeye yönelik yarı yapılandırılmış görüşme soruları yöneltilmiştir (EK 7). Öğrencilerin verdikleri yanıtlar incelendiğinde ölçekten elde edilen bulgulara paralel bulgular elde edilmiştir. Öğrenciler etkinliklerin çok faydalı olduğunu düşündüklerini, akıllı cihazların nasıl çalıştığını anladıklarını, kendilerinin de ilerde akıllı cihaz geliştirebilecek becerilere sahip olacaklarına inandıklarını belirtmişlerdir. Okuldaki başarılarına ve konumlarına sağlayacağı katkıya yönelik verdikleri yanıtlar incelendiğinde, öğrencilerin etkinliklerin kendilerine faydalı olduğuna inandıklarını belirtmelerine rağmen bu etkinliklerin okulda nasıl bir fayda sağlayacağına yönelik net bir öngörüde bulunamadıkları görülmektedir. Öğrencilerin, etkinliklerin okulda daha iyi olmalarına yardımcı olabileceği yönünde soruya yorum yaparken zorlanmalarının sebebinin, başarı değerlendirme ölçütü olarak derslerden aldıkları sınav notlarını algıladıkları, etkinliklerin hangi dersin sınav notuna faydalı olacağı konusunda bir çıkarımda bulunamadıkları için yorum yapmakta zorlandıkları görülmektedir.

4.4.3. Etkinliği Yapmaya İlişkin İsteklilik ve Etkinliği İlgil Çekici Bulma

Katılımcıların etkinlik algısı ölçeğinin kişinin etkinliği yapmaya ilişkin istekli-gönüllü olmaya ilişkin algısına ve etkinliği ilgi çekici bulmaya ilişkin algısına yönelik maddelerine verdikleri puanların ortalama ve standart sapma değerleri Tablo 4.7'de gösterilmiştir.

Tablo 4.7: Etkinlik Algısı Ölçeğinin Etkinliği Yapma İsteğine ve Etkinliği İlgi Çekici Bulmaya İlişkin Maddelerine Ait Ortalama ve Standart Sapma Değerleri

	5-) Bu etkinliği yapmak istediğim için yaptım.		10-) Bunun çok ilgi çekici bir etkinlik olduğunu düşündüm.	
	\bar{X}	SS	\bar{X}	SS
Etkinlik 1	4.63	.90	4.64	.69
Etkinlik 2	4.67	.76	4.63	.79
Etkinlik 3	4.79	.58	4.81	.53
Etkinlik 4	4.80	.73	4.88	.38
Etkinlik 5	4.75	.80	4.89	.37

Tablo 4.7 incelendiğinde öğrencilerin etkinlikleri isteyerek yapmaya ilişkin etkinlik algılarını tespit etmeye yönelik hazırlanan maddeye verdikleri puan ortalamalarının 4.63 ile 4.80 arasında değiştiği görülmektedir. Buradan öğrencilerin etkinlikleri yapmak istedikleri için yaptıkları, istemeseler dahi etkinliği yapmak zorunda oldukları yönünde bir algıya sahip olmadıkları anlaşılmaktadır. “Bunun çok ilgi çekici bir etkinlik olduğunu düşünüyorum” maddesine verilen puanların ortalamaları incelendiğinde 4.63 ile 4.89 arasında değiştiği görülmektedir. Oldukça yüksek olan bu ortalama puanlar öğrencilerin etkinliği ilgi çekici bulduklarını göstermektedir. Yapılan gözlemler neticesinde, öğrencilerin gönüllülük esasına dayalı olan etkinliklere yoğun bir katılım ve devamlılık gösterdikleri, öğrencilerin etkinlik gününe kadar sık sık gelerek etkinlikte neler yapacaklarını sordukları - her bir etkinliği merakla bekledikleri, etkinliği bir an önce yapmak konusunda sabırsızlandıkları, etkinliğin başlangıcında o etkinliğin hedeften haberdar etme kısmında öğrencilerin öğretmeni çok büyük bir ilgi ve dikkatle takip ettikleri, mutluluklarının ve öğrenmeye karşı heyecanlarının vücut dillerinden ve hareketlerinden gözlemlendiği yönünde

bulgular tespit edilmiştir. Öğrencilerin 5 haftalık etkinlikler sona erdikten sonra etkinliklerin devam etmesi yönündeki talepleri de öğrencilerin etkinlikleri oldukça ilgi çekici, eğlenceli ve faydalı gördükleri yönünde tespitleri destekler niteliktedir.

5. SONUÇ, YORUM VE ÖNERİLER

5.1. Sonuç ve Yorumlar

Bu çalışma kapsamında kompütasyonel düşünme ve programlama becerisinin ilköğretim düzeyinden itibaren öğrencilere kazandırılmasının önemine değinilmiş, bu konuda hazırlanan raporlar ve bu yönde yapılan girişimlerden örnekler verilmiştir. Dünya çapında kompütasyonel düşünme ve programlama eğitim kapsamında yapılan çalışmalara eşzamanlı olarak ülkemizde de programlama eğitimi 2012 yılında Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programına dâhil edilmiştir (TTKB, 2012). Bu kapsamda gerçekleştirilen çalışmalardan bir tanesi olan liseler için bilgisayar bilimi öğretim programı geliştirme çalışması da 2016 yılı itibariyle tamamlanmış, Talim Terbiye Kurulu Başkanlığı tarafından liseler için “Bilgisayar Bilimi Dersi Öğretim Programı” yayınlanmıştır (TTKB, 2016). Ancak her iki öğretim programında da kazanımlar öğretmenlerin dersleri kendileri tasarlayabileceği şekilde açık uçlu bırakılmıştır. 2017 yılı ocak ayı itibariyle yayınlanan bilişim teknolojileri ve yazılım dersi taslak öğretim programında 1. sınıftan 8. sınıfa kadar tüm sınıf düzeylerinde öğretim programlarının %44’ü “Hesaplamalı Düşünme” ünitesi altında “problem çözme kavramları ve yaklaşımları” ve “programlama” konu başlıklarını içermektedir. 1. sınıftan 4. sınıfa kadar serbest etkinlik saatlerinde işlenip işlenmemesi öğretmen inisiyatifine bırakılan bilişim teknolojileri ve yazılım dersinin, 5. ve 6. sınıflarda haftada 2 saat zorunlu, 7. ve 8. sınıflarda ise haftada 2 saat seçmeli olarak yürütülmesi planlanmaktadır (TTKB, 2017). Yapılan çalışmalar incelendiğinde, öğretim programının nasıl daha verimli olabileceğine yönelik bulgu ve örnek uygulamaların sınırlı olduğu gözlemlenmektedir. Bu çalışma kapsamında elde edilen bulgular öğretim programlarında yer alan kodlama ve robotik kodlama eğitimlerine yönelik etkinlikler neticesinde öğrencilerin kodlamaya ilişkin öz-yeterlik algılarında meydana gelen farklılaşmaları ortaya koymakta, etkinliklerin nasıl daha verimli olabileceği yönünde bulgularla gelecek çalışmalara yol göstermesi beklenmektedir. Kodlamaya ilişkin bir performans göstergesi olarak geliştirilen “Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği” bu alanda yapılacak çalışmalarda kullanılabilir. Çalışma kapsamında elde edilen bulgulara yönelik sonuç ve yorumlar alt başlıklar halinde bu bölümde ele alınmıştır.

5.1.1. Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği

Çalışma kapsamında, robotik kodlama eğitiminin öğrencilerin blok temelli programlamaya ilişkin performanslarına etkisini ölçmek amacıyla bir performans göstergesine ihtiyaç duyulmuştur. Alanyazında programlama performansını ölçmek amacıyla yapılan çalışmalarda performans göstergesi olarak programlama öz-yeterlik algısı ölçeklerinin kullanıldığı görülmüştür. Ancak programlamaya ilişkin öz-yeterlik algısı ölçeklerinin blok temelli programlamaya ilişkin öz-yeterlik algısını ölçmek için uygun bir araç olmadığı görülmesi nedeniyle blok temelli programlamaya ilişkin öz-yeterlik algısı geliştirme çalışması yapılmıştır.

Ayrıca geliştirilen ölçme aracı ile çeşitli illerde bulunan 4 okuldan elde edilen blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarının çeşitli değişkenler açısından (cinsiyet, daha önce Scratch'le programlama dersi alma, daha önce Scratch'le program yazma, halen Scratch'le programlama dersi alma, halen Scratch'le program yazma) farklılaşıp farklılaşmadığı ortaya konulmuştur.

Bu çalışma kapsamında geliştirilen blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeğinde, ölçek maddelerinin altında toplandığı iki faktörün isimlendirilmesinde, Resnick ve diğerlerinin (2009) çalışmasında blok temelli programlamaya ilişkin kavramların isimlendirmesi ve Altun ve Mazman (2012) tarafından yapılan çalışmada kullanılan faktör isimlendirmesi referans alınmıştır. Yapılan değerlendirme sonucunda, blok temelli programlama ilişkin öz-yeterlik algısı ölçeğine ait iki faktörün, 5 maddeden oluşan (3., 6., 7., 8. ve 9.) birinci faktörü "basit blok temelli programlamaya ilişkin öz-yeterlik algısı" olarak, 7 maddeden oluşan (1., 2., 10., 11., 12., 13. ve 14.) ikinci faktörü ise "karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı" olarak isimlendirilmiştir. Sonuç olarak, elde edilen 12 madde ve 2 faktörlü yapının iç tutarlılık katsayısı (cronbach alpha) .893 bulunmuş, bu 12 madde toplam varyansın %58.225'ini açıklamış ve bu yapıya ilişkin model doğrulayıcı faktör analizi ile de doğrulanarak blok temelli programlamaya ilişkin öz-yeterlik algısını ölçen geçerli ve güvenilir bir araç elde edilmiştir. Geliştirilen bu blok temelli öz-yeterlik algısı ölçeği daha sonraki çalışmalarda bireylerin blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarını ölçmek üzere farklı çalışmalarda da kullanılabileceği öngörülmektedir.

Basit blok temelli programlamaya ilişkin öz-yeterlik algısı puanları ve karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanları çeşitli değişkenler açısından

incelendiğinde, ilk olarak cinsiyet bakımından istatistiksel olarak anlamlı bir fark olmadığı ortaya çıkmıştır. Bu bulgu ilgili alan yazındaki birçok araştırma ile benzerlik göstermektedir (bkz. Ramalingam & Wiedenbeck, 1998; Byrne & Lyons, 2001; Pillay & Jugoo, 2005; Altun ve Mazman, 2013).

Daha önce Scratch'le programlama dersi alma ve daha önce Scratch'le program yazma durumlarına göre öğrencilerin basit blok temelli programlamaya ilişkin öz-yeterlik algılarında anlamlı bir fark bulunmamıştır. Basit blok temelli programlama görevlerine ilişkin maddelerde yer alan kavramlara ilişkin bu bulgular, Resnick ve diğerlerinin (2009) çalışmasında bu kavramların özel bir eğitim veya desteğe gerek kalmadan kolayca öğrenilebileceği yönündeki bulguları ile paralellik göstermektedir. Daha önce Scratch'le programlama dersi alan öğrencilerin karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanları daha önce Scratch'le programlama dersi almayan öğrencilere göre, daha önce Scratch'le program yazan öğrencilerin karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanları daha önce Scratch'le program yazmayan öğrencilere göre anlamlı derecede yüksek olarak bulunmuştur. Bu bulgular, Altun ve Mazman'ın (2013) öz-yeterlik algısı puanlarına yönelik yaptıkları çalışmada, programlama deneyiminin artışı ile öz-yeterlik puanının anlamlı olarak arttığı yönünde elde ettikleri bulgularla örtüşmektedir. Ayrıca bu bulgular, karmaşık blok temelli programlama görevleri arasında yer alan 10. ve 11. maddelerin, Resnick ve diğerleri (2009) tarafından daha fazla uzman desteği ile öğrenilebilecek nitelikte olduğu yönündeki tespitlerini de desteklemektedir.

Halen Scratch'le programlama dersi alan öğrencilerin halen Scratch'le programlama dersi almayan öğrencilere göre, halen Scratch'le program yazan öğrencilerin halen Scratch'le program yazmayan öğrencilere göre hem basit hem de karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanları anlamlı derecede yüksek olarak bulunmuştur. Bu bulgular, programlama dersi alma ile doğru orantılı olarak bireylerin programlamaya ilişkin öz-yeterlik algılarının da arttığı yönündeki bulguları (bkz. Ramalingam ve Wiedenbeck, 1998; Jegede, 2009; Altun ve Mazman, 2013) destekler niteliktedir.

5.1.2. Robotik Kodlama Etkinlikleri ve Kodlamaya İlişkin Öz-yeterlik Algısı

Robotik kodlama etkinlikleri sonrasında öğrencilerde blok temelli programlamaya ilişkin öz-yeterlik algılarında meydana gelen değişimleri tespit etmek amacıyla

yapılan Eşleştirilmiş-Örneklem t Testi (Paired-Samples t Test) sonuçlarına göre hem basit hem de karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarında grup içi pozitif yönde anlamlı değişim meydana gelmiştir (basit $t = -5.01$, $p = 0.000$; karmaşık $t = -8.84$, $p = 0.00$). Bu bulgu Rusk, Resnick, Berg, ve Pezalla-Granlund (2008) tarafından yapılan çalışma bulgularını desteklemekte iken Beug, (2012) tarafından yürütülen çalışma ile örtüşmemektedir. Çalışmamız kapsamında, Rusk, Resnick, Berg, ve Pezalla-Granlund'un (2008) çalışmalarında belirttikleri, "etkinlikler tasarlanırken donanımsal sorunlar yaşanmamalı, öğrencilerin ilgi alanlarına hitap edecek şekilde etkinlikler düzenlenmeli" ilkelerine uygun etkinlik planlamaları düzenlenmesine dikkat edilmesi, iki çalışmanın sonuçlarının birbirini desteklemesinin sebebi olarak gösterilebilir. Beug'un (2012) çalışmasında ise blok temelli programlama platformu ile metin tabanlı programlama dili olan Arduino karşılaştırması yapılmış olması ve etkinlikler esnasında ortamın hazırlanmaması nedeniyle Arduino kartı bilgisayarlara tanıtma gibi sorunlar yaşanması, çalışmalarında Robotik etkinliklerinin öğrenci performansına olumsuz etki etmiş olabileceği söylenebilir.

Bu bulguların geçerliği ve güvenilirliğine yönelik yürütülen çalışma kapsamında, araştırma grubunun etkinlik öncesi çeşitli değişkenlere göre grup içi hazırbulunuşluk düzeyleri arasında anlamlı bir farklılık olup olmadığı ve etkinlikler esnasında evde bilgisayar sahipliği gibi değişkenlerin öğrenciler arasında öz-yeterlik algılarında anlamlı bir değişim meydana getirip getirmediğine yönelik analizlere yer verilmiştir. Robotik kodlama etkinlikleri öncesinde öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algılarında çeşitli değişkenlere göre grup içi farklılıklar olup olmadığını tespit etmek amacıyla uygulanan ön-test verileri üzerinde yapılan analiz bulgularına göre bireylerin hem basit hem de karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı ön-test puanlarının, evde bilgisayar sahipliğine, evde internet bağlantısı ve ders dışında Scratch programına çalışma olanağına göre anlamlı olarak farklılaşmadığı bulunmuştur. Elde edilen bulgular, araştırma grubundaki öğrencilerin çeşitli değişkenlere göre ön-test puanlarında anlamlı bir farklılık olmadığı yönündedir. Bu bulgulara göre, araştırmaya katılan grubun robotik kodlama etkinlikleri sonunda çeşitli değişkenlere göre blok temelli programlamaya ilişkin öz-yeterlik algılarındaki değişimlerinin analizinde hazırbulunuşluk düzeylerinden kaynaklanan değerlendirme hatalarının bulunmadığı söylenebilir.

Robotik kodlama etkinlikleri sonunda öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algılarında çeşitli değişkenlere göre grup içi değişimlerini analiz etmek amacıyla son-test ön-test fark değerleri üzerinde bağımsız-örneklem t testi (Independent-Samples t Test) yapılmıştır. Elde edilen bulgular, robotik kodlama etkinlikleri sonunda öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algılarında çeşitli değişkenlere göre anlamlı farklılaşma olmadığını göstermektedir. Robotik kodlama etkinlikleri tasarlanırken Rusk, Resnick, Berg, ve Pezalla-Granlund'un (2008) belirttikleri "etkinlikler tasarlanırken öğrencilerin aynı ilgi alanlarından olacak şekilde gruplar oluşturulmasına özen gösterilmeli, ilgi alanlarına hitap edecek şekilde etkinlikler düzenlenmeli" tespitine uygun şekilde etkinlik ortamı oluşturulmasına gayret edilmiştir. Rusk, Resnick, Berg, ve Pezalla-Granlund'un (2008) belirttikleri etkinlik tasarım ilkelerine uygun bir şekilde etkinliklerin düzenlenmesi neticesinde, öğrenci yaşantılarının tespit edilmesine ilişkin bulgularda da görüleceği gibi hem kız hem erkek öğrenciler etkinlikleri oldukça yüksek oranda eğlenceli, ilgi çekici bulmuşlar ve etkinliklere katılmak konusunda istekli olmuşlardır. Etkinlik tasarım sürecinden kaynaklı bu durumun, etkinliklerden her iki cinsiyetin de daha fazla verim almalarında ve öz-yeterlik algılarındaki değişimde cinsiyete göre farklılık oluşmamasında etkili olduğu söylenebilir. Sonraki çalışmalarda bu yoruma ilişkin araştırma yapılarak, yorumun doğruluğuna yönelik tespitlerde bulunulabilir. Öğrencilerin hiçbirisinin okul dışında robotik setlere erişim imkanına sahip olmaması, evde bilgisayar sahipliği ve evde internet bağlantısı sahipliği olsa dahi okul dışında robotik kodlama uygulamalarına çalışma imkanına ulaşamamalarına ve bu değişkenlerin öz-yeterlik algılarında anlamlı bir değişim meydana getirmemesine sebep olarak gösterilebilir. Robotik setlere okul dışında erişim imkanlarının olması durumunda sonuçlarda farklılaşma meydana gelebilir.

Yapılan analizler, robotik kodlama etkinlikleri ile öğrencilerin hem basit hem de karmaşık blok temelli programlamaya ilişkin öz-yeterlik algısı puanlarında grup içi pozitif yönde anlamlı değişim meydana geldiğini (basit $t = -5.01$, $p = 0.00$; karmaşık $t = -8.84$, $p = 0.00$), meydana gelen değişimde çeşitli değişkenlerin anlamlı düzeyde etkisinin olmadığını göstermektedir.

5.1.3. Robotik Kodlama Etkinliklerine İlişkin Öğrenci Yaşantıları

Robotik kodlama etkinliklerine ilişkin öğrenci yaşantılarını tespit etmek amacıyla Deci, Eghrari, Patrick, ve Leone (1994) tarafından ilgi çekici olmayan (sıkıcı)

bilgisayar görevleri için geliştirilen etkinlik algısı ölçeğinin Türkçeye uyarlama çalışması yapılmıştır. Çalışma sonunda elde edilen etkinlik algısı ölçeği çevrimiçi hale getirilmiş, her etkinliğin hemen ardından öğrencilere uygulanmıştır. Bunun yanında etkinlikler esnasında katılımcı gözlemci olarak notlar tutulmuş, etkinliklerin ardından öğrencilerle görüşmeler yapılarak notlar alınmıştır.

Öğrencilerin robotik kodlama etkinliklerine yönelik öğrenci yaşantılarına ilişkin elde edilen veriler hakkında yorum yapabilmek için, robotik kodlama etkinlikleri esnasında kullanılan Scratch for Arduino programının MIT yaşam boyu öğrenme grubu tarafından geliştirilen Scratch programının bir modifikasyonu olduğunu göz önünde bulundurmak gerekmektedir. Scratch programının kodlama eğitiminde yaygın olarak kullanıldığı bilinmektedir. Henüz somut işlem dönemindeki ortaokul 5. sınıf öğrencilerinin robotik kodlama etkinliklerine hazırlık sürecinde gerçekleştirilen Scratch'le programlama etkinliklerine yönelik ilgi ve etkinlikleri gerçekleştirmeye ilişkin isteklilik düzeylerinin yüksek olduğu yapılan gözlemlerle tespit edilmiştir. Bu tespit Resnick ve diğerlerinin (2009) bir bilgisayar kulübünde Scratch'le programlamaya yönelik yaptıkları çalışmalarında programlama eğitime yönelik Papert (1980)'den günümüze meydana gelen gelişmeleri analiz ederek yaptıkları "programlama dillerinde somutlaştırılma düzeyiyle doğru orantılı olarak bireylerin programlama dilini anlamaları kolaylaşmakta, bireylerde ilgi ve motivasyon artışı yaşanmaktadır" tespitlerini destekler niteliktedir. Scratch programı, klasik programlama mantığını blok temelli programlama ve somutlaştırmaya yönelik özel arayüz tasarımı ile çok daha somut hale getirmiş, öğrencilerin daha kolay öğrenmelerine ve motive olmalarına olanak tanımıştır. Tüm bu bulgulara rağmen, robotik kodlama etkinliklerine hazırlık sürecinde düzenlenen Scratch'le kodlama etkinliklerinin 4. haftası sonunda öğrencilerin saate daha fazla baktıkları, başka konularla meşgul olmaya başladıkları, derse karşı ilgilerinin dağılmaya başladığı gözlemlenmiştir. Scratch for Arduino programı kullanılarak gerçekleştirilen robotik kodlama etkinlikleri ise tamamen somut etkinliklerle, oldukça fazla sensör ve aktüatör aracılığıyla hayal dünyalarındaki birçok hayali gerçekleştirme imkânı sunması, öğrencilerin çok daha fazla eğlenmelerine, etkinlikleri gerçekleştirme yönünde istekli olmalarına, etkinlikleri kişisel gelişimleri açısından faydalı bulmalarına neden olmaktadır. Bu tespit yine Resnick ve Diğerlerinin (2009) somutlaştırma ve hayal gücünü ortaya çıkartabilecekleri imkanların fazlalığı ile

öğrenmenin kolaylaşması, ilgi ve motivasyon artışı yönündeki bulgularını destekler niteliktedir. Papert (1980) çalışmasında, “Programlama dilleri “low floor-zemini düşük” (başlamak için kolay seviyede), “high ceiling-tavanı yüksek” (zamanla karmaşık projeler oluşturacak şekilde) olmalıdır” şeklinde bir tespitte bulunmuştur. Robotik kodlama etkinlikleri bu anlamda blok temelli yapısı, en düşük düzeyde etkinliklere imkân tanınması ve akıllı cihazlardan nesnelere internetine kadar uzanan bir yelpazede hayal dünyalarındaki birçok hayali gerçekleştirme imkânı sunması ile Papert’ın 1980’de belirttiği ilkeye uygun bir öğrenme ortamı sunmaktadır. Çalışma kapsamında gerçekleştirilen robotik kodlama etkinliklerine ilişkin öğrenci yaşantılarını tespit etmeye yönelik uygulanan etkinlik algısı ölçeğinden, yapılan gözlem ve görüşmelerden elde edilen bulgular, Papert’dan (1980) MIT Yaşam Boyu Eğitim Grubunun Scratch’le ve Scratch for Arduino ile gerçekleştirdikleri çalışmalara kadar elde edilen bulguları destekler niteliktedir. Robotik kodlama etkinlikleri esnasında öğrencilerin etkinlikler esnasında oldukça eğlendikleri, robotik kodlama etkinliklerinin kişisel gelişimleri açısından önemli olduğunu düşündükleri, etkinlikleri kendi istekleriyle yaptıkları ve etkinlikleri ilgi çekici buldukları yönünde bulgular elde edilmiştir.

Öğrenci yaşantıları, “etkinlikleri eğlenceli bulma”, “etkinliklerin kişisel gelişime katkısı”, “etkinlikleri yapmaya ilişkin isteklilik ve etkinlikleri ilgi çekici bulma” başlıkları altında gruplandırılmış ve yorumlanmıştır.

5.1.3.1. Robotik Kodlama Etkinliklerini Eğlenceli Bulmaya İlişkin Etkinlik Algısı

Etkinliği eğlenceli bulmaya ilişkin etkinlik algısı ölçeği maddelerine verilen ortalama puanlar incelendiğinde öğrencilerin robotik kodlama etkinliklerini eğlenceli bulma algı düzeylerinin oldukça yüksek, sıkıcı bulma algı düzeylerinin ise oldukça düşük olduğunu göstermektedir. Gözlem ve görüşme notlarından elde edilen bulgularda etkinlik algısı ölçeğinden elde edilen bulguları destekler niteliktedir. Öğrencilerin oldukça eğlendikleri gözlemlenmiş, görüşmelerde de öğrenciler tüm etkinliklerde çok eğlendiklerini belirtmişlerdir. Elde edilen bulgular MIT Yaşam Boyu Öğrenme Grubunun MIT Media Laboratuvarında elde ettikleri bulguları destekler niteliktedir (bkz. Rusk, Resnick, Berg, & Pezalla-Granlund, 2008). Gözlem ve görüşme notlarından elde edilen bulgular öğrencilerin robotik kodlama etkinliklerinin hemen öncesinde gerçekleştirilen Scratch’le kodlama etkinlikleri esnasında da oldukça

eğlendikleri yönündedir. Ancak 4. Haftadan sonra sıkılmaya başlayan öğrencilerin robotik kodlama etkinliklerinde 1. etkinlikten itibaren oldukça eğlendikleri ve etkinlikleri gerçekleştirme konusunda oldukça istekli oldukları görülmüştür.

Öğrencilerin “etkinlikler esnasında sıkıldığı bölümler oldu mu?” sorusuna yönelik verdikleri yanıtlara bakıldığında öğrencilerin etkinliklerde çok eğlenmelerine rağmen elektrik konusunun fen bilgisi dersinde henüz yeterince işlenilmemiş olmasından dolayı öğrencilerin elektrik devresi kurmalarını gerektiren etkinliklerde zorlandıkları, öğrencilerin bundan dolayı devre kurma ile ilgili görevleri yerine getirirken düşük düzeyde de olsa sıkıldıklarını belirttikleri görülmektedir. Elektrik konusu ve elektrik devresi kurma uygulamasının ön hazırlık eğitiminin yalnızca etkinliği gerçekleştirebilecek düzeyde kısa tutulmasının öğrencilerin devre kurarken zorlanmalarına sebep olduğu görülmekte, devre kurma eğitiminin daha uzun zamana yayılması gerektiği anlaşılmaktadır. Bunun yanında bu etkinliklerinin disiplinler arası iş birliği halinde gerçekleştirilebildiği eğitim ortamlarında öğrencilerin farklı disiplinlerde daha fazla kazanım elde edebileceği de öngörülmektedir. Bu öngörüye destekler nitelikte, son yıllarda STEM (Fen, Teknoloji, Mühendislik, Matematik) eğitimi adı altında gerçekleştirilen etkinlikler dikkat çekicidir. Öğrencilerin "Derste sıkıldığı bölümler, zamanlar oldu mu? Sıkıldıysan daha çok neden sıkıldın?" şeklinde yöneltilen soruya verdikleri yanıtlar incelendiğinde, elektrik devresini oluşturmakta zorlandıkları zamanlarla ilgili “zorlandığım için sıkıldım” ya da “çok eğlendim ama yapamayınca sıkıldığım durumlar oldu” şeklinde yanıtlar verdikleri görülmüştür.

5.1.3.2. Robotik Kodlama Etkinliklerinin Kişisel Gelişime Etkisine İlişkin Etkinlik Algısı

Öğrencilerin etkinlik algısı ölçeğine verdikleri puan ortalamaları analiz edildiğinde robotik kodlama etkinliklerinin kişisel gelişimlerine ilişkin etkinlik algılarının pozitif yönde oldukça yüksek olduğu söylenebilir. Öğrencilerin etkinliğin kişisel gelişimlerine yönelik etkisine ilişkin algılarının etkinliğin okulda daha iyi olmalarına yardımcı olabileceğine ilişkin algılarına göre az da olsa pozitif yönde ayrıştığı dikkat çekmektedir. Robotik kodlama etkinliklerinin öğrencilerin kişisel gelişimlerine etkisine ilişkin etkinlik algılarını daha iyi analiz edebilmek amacıyla yöneltilen sorulara verdikleri yanıtlar incelendiğinde ölçekten elde edilen bulgulara paralel bulgular elde edilmiştir. Öğrenciler etkinliklerin kişisel gelişimlerine çok faydalı

olduğunu düşündüklerini, akıllı cihazların nasıl çalıştığını anladıklarını, kendilerinin de ilerde akıllı cihaz geliştirebilecek becerilere sahip olacaklarına inandıklarını belirtmişlerdir. Robotik kodlama etkinliklerinin okuldaki başarılarına yardımcı olabileceğine ilişkin algılarına yönelik sorulara verdikleri yanıtlar incelendiğinde öğrencilerin etkinliklerin okulda daha iyi olmalarına yardımcı olacağına inandıklarını belirtmelerine rağmen okulda daha iyi olmalarına nasıl yardımda bulunacağına ilişkin net bir öngörüde bulunamadıkları görülmektedir. Bunun sebebi ise yapılan gözlemlerde ve görüşmelerde şu şekilde tespit edilmiştir, öğrenciler fen bilimleri, matematik, bilişim teknolojileri gibi birbiriyle ilişkili disiplinleri barındıran derslerde birbiriyle bağlantısız olarak eğitim görmektedirler. Bu yüzden farklı disiplinlere katkı sağlayacak şekilde düzenlenen robotik kodlama etkinliklerinin öğrencilerin okul başarısına ve okuldaki konumlarına (öğretmenlerin ilgisi, arkadaşlarının takdiri) yönelik sağlayacağı katkı konusunda etkinlik algıları yüksek olsa dahi kişisel gelişime ilişkin etkinlik algıları kadar yüksek olmadığı, daha doğrusu okul başarısına ve okuldaki konumlarına ilişkin tam bir görüş sahibi olmadıkları görülmektedir. Çalışma kapsamında düzenlenen etkinliklerde her ne kadar fen bilimleri öğretmeni ile çalışma gerçekleştirilmiş olsa dahi, farklı derslerde birbirini destekler nitelikte bir planlama ve uygulama imkânı mümkün olmamıştır. Bu şekilde disiplinler arası iş birliğini içerecek şekilde düzenlenmiş bir eğitim-öğretim ortamında öğrencilerin okul başarılarına yönelik etkinlik algılarında daha pozitif yönde bir etkinlik algısı artışı öngörülse de ancak daha sonraki çalışmalarda bu öngörünün bulguları ortaya konulabilecektir.

5.1.3.3. Etkinliği Yapmaya İlişkin İsteklilik ve Etkinliği İlgi Çekici Bulmaya İlişkin Etkinlik Algısı

Öğrencilerin etkinlikleri isteyerek yapmaya ilişkin etkinlik algılarını tespit etmeye yönelik hazırlanan maddeye verdikleri puan ortalamalarına bakıldığında öğrencilerin etkinlikleri yapmak istedikleri için yaptıkları, istemeseler dahi etkinliği yapmak zorunda oldukları yönünde bir algıya sahip olmadıkları, “Bunun çok ilgi çekici bir etkinlik olduğunu düşünüyorum” maddesine verilen puanların ortalamaları incelendiğinde öğrencilerin etkinliği yüksek bir oranla ilgi çekici buldukları görülmektedir. Yapılan gözlemler neticesinde, öğrencilerin gönüllülük esasına dayalı olan etkinliklere yoğun bir katılım ve devamlılık gösterdikleri, öğrencilerin her bir etkinliği merakla bekledikleri, etkinliğin başlangıcında o etkinliğin hedeften

haberdar etme kısmında oldukça heyecanlandıkları yönünde bulgular tespit edilmiştir. Öğrencilerin 5 haftalık etkinlikler sona erdikten sonra etkinliklerin devam etmesi yönündeki talepleri de öğrencilerin etkinlikleri oldukça ilgi çekici, eğlenceli ve faydalı gördükleri yönünde tespitleri destekler niteliktedir. Etkinliklerin somut işlem dönemindeki öğrencilerin ilgisini çekecek şekilde görme, işitme, dokunma gibi birden fazla duyu organına hitap edecek şekilde somut etkinlikler içermesi öğrencilerin etkinlikleri isteyerek yapmalarındaki önemli bir faktör olduğu düşünülebilir. Bunun yanında günümüzün hızla gelişen teknolojileri olan akıllı cihaz teknolojilerinin çalışma mantığını anlamaya dönük kazanımlar içermesi ve öğretmenin bu örnekleri göstererek ve bunların çalışma mantığıyla ilgili sorular sorarak öğrencilerde merak duygusu uyandırılması da öğrencilerin etkinlikleri ilgi çekici bulmalarına ve etkinlikleri isteyerek yapmalarına yol açmış olabilir. 1975 yılında Kanadalı araştırmacı Rubin'in "Kimler çok iyi dil öğrenir?" sorusunun yanıtını bulmak için yaptığı araştırmada bulduğu ilginç sonuç araştırmamız kapsamında elde ettiğimiz bulguları destekler niteliktedir. Rubin (1975) araştırmasında, en iyi öğrenenlerin öğrenmeye en çok ilgi duyanlar olduğunu bulmuştur. Rubin (1975), etrafımızdaki insanların yabancı dil öğrendikleri takdirde iş bulacakları, terfi alacakları halde yine de o dili öğrenemediklerini, en iyi öğrenenlerin o dilin kültürüne ilgi duyanlar olduğunu belirtmektedir.

5.1.3.4. Öğrenci Yaşantılarına İlişkin Genel Değerlendirme

Öğrencilerin robotik kodlama etkinliklerine katılım konusunda bu düzeyde istekli olmaları ve etkinlikleri ilgi çekici, eğlenceli ve kişisel gelişimleri açısından faydalı bulmalarının sebepleri olarak çalışma kapsamında elde edilen bulgular şu şekilde yorumlanabilir:

Bu yaş grubundaki öğrenciler soyut bilgilerden ziyade somut uygulamalara ilgi duymaktadırlar. Kodlama her ne kadar bilgisayar ekranında görme, işitme gibi farklı duyu organlarına hitap eden çıktılar verse de somut temas edebildikleri nesnelere çalışmak, onları yönetebilmek öğrencilere daha ilgi çekici gelmektedir.

Bu yaş grubundaki ve özellikle daha küçük yaş grubundaki bireylerde çevrelerindeki dünyayı anlama güdüsü oldukça yüksektir. Günümüz teknolojisinin geldiği noktada akıllı cihazların çalışma mantığına yönelik yapılan robotik kodlama etkinlikleri öğrencilerin oldukça ilgilerini çekmekte, etkinliklere istekli olarak katılmalarına ve etkinlikler süresince eğlenmelerine neden olmaktadır. Günümüz dünyasının

teknolojilerini anlamayı kolaylaştıran robotik kodlama etkinlikleri, öğrencilerin etkinlikleri kişisel gelişimleri açısından faydalı bulmalarının sebepleri arasında da sayılabilir.

Etkinliklerin tasarımında öğrencilerin hedeften haberdar edilmesi ve güdülenmesi de büyük önem taşımaktadır. Robotik kodlama etkinlikleri öncesinde öğretmen tarafından görsellerle, videolarla güncel akıllı cihaz örnekleri, uygulamaları gösterilmiş, öğrencilerin güdülenmesi sağlanmıştır. Etkinlik tasarımında öğrencilerin etkinliklere hazır duruma getirilmesi aşaması etkinliklerin başarıya ulaşmasında büyük önem taşımaktadır.

Etkinlikler esnasında etkinlikleri gerçekleştirme konusunda zorlandığı gözlemlenen öğrenciler, etkinlikleri daha rahat gerçekleştirebildiği gözlemlenen öğrencilerle ikiye bölünmüş grup oluşturacak şekilde oturtulmuştur. Bu şekilde öğrencilerin birbirlerinden öğrenmeleri sağlanmış (akran öğrenmesi), etkinliğin daha verimli yürütülmesi sağlanmıştır.

Robotik kodlama etkinlikleri esnasında Arduino Uno R3 kart, sensörler ve aktüatörler kullanılmıştır. Bu şekilde öğrencilerde sınırsız bir hayal gücü meydana getirilmesi amaçlanmış, öğrencilerin hayallerini gerçekleştirebilecek bilgi ve deneyime ulaşmak için daha fazla etkinlik yapma istekleri canlı tutulmuştur. Alanyazında rastlanılan birbirine geçmeli parçalardan oluşan hazır setlerin kullanımının çalışmamız kapsamında gerçekleştirilen robotik kodlama etkinliklerine daha fazla heyecan getirip getirmeyeceği bilinmemekle birlikte, öğrencilerin kapasitelerini ve hayal güçlerini sınırlandırabileceği riski göz önünde bulundurulmalıdır. Ayrıca hazır setlerle yapılan robotik kodlama etkinliklerinin maliyetinin, bu çalışma kapsamında kullanılan malzemelerle kıyaslandığında oldukça yüksek olduğu görülmektedir.

5.1.4. Öneriler

5.1.4.1. Araştırmacılara Öneriler

Öğrencilerin kodlamaya (blok temelli programlama) ilişkin öz-yeterlik algılarını tespit etmek amacıyla hazırlanan ölçeğin robotik kodlamaya ilişkin öz-yeterlik algılarını tespit edebilecek şekilde maddeler eklenerek robotik kodlamaya ilişkin öz-yeterlik algısı ölçeği uyarlama çalışması yapılabilir.

Robotik kodlama etkinlikleri daha önce Scratch'le kodlama eğitimi almış öğrencilerle düzenlenmiştir. Robotik kodlama etkinlikleri daha önce hiç kodlama eğitimi almamış öğrencilerle düzenlenerek robotik kodlama etkinliklerinin hiç kodlama eğitimi almamış öğrencilerin kodlamaya ilişkin öz-yeterlik algılarında farklılaşma meydana getirip getirmediğine ve öğrencilerin robotik kodlama etkinliklerine ilişkin yaşantılarına bakılabilir.

Kodlama etkinlikleri ve robotik kodlama etkinliklerinin öğrencilerin öz-yeterlik algılarına etkisini ve öğrenci yaşantılarındaki farklılaşmayı karşılaştırabilmek amacıyla deney ve kontrol gruplu deneysel araştırmalar yapılabilir.

Robotik kodlama etkinliklerinde öğrencilerin elektrik devresi kurarken zorlandıkları yönünde bulgular elde edilmiştir. Etkinlikler esnasında elektrik konusunda eğitimler daha uzun tutularak etkinlikler düzenlenebilir. Fen bilgisi zümre öğretmenleri ile disiplinler arası etkinlikler düzenlenebilir, fen bilgisi dersinde gördükleri konularla, robotik kodlama etkinliklerinde oluşturdukları elektrik devreleri için gerekli bilgilerin örtüştüğü etkinlik uygulamaları düzenlenebilir.

Robotik kodlama etkinliklerinin yalnızca kodlama kazanımlarına yönelik değil, elektrik konusu kazanımlarına yönelik de araştırmalar yapılabilir. Bunun yanında etkinlikler farklı disiplinlerle de ilişkilendirilerek hedeflenen kazanımlara yönelik araştırmalar da yapılabilir.

5.1.4.2. Robotik Kodlama Etkinliği Düzenleyecekler Öneriler

Katılımcıların robotik kodlama etkinlikleri öncesinde akıllı cihazlar, 3 boyutlu yazıcı teknolojileri, endüstri 4.0 gibi güncel konular hakkında bol görsel kullanılarak bilgilendirilmeleri, öğrencilerin etkinliklere istekli ve gönüllü katılmaları, etkinlikleri gerçekleştirmeye ilişkin ilgi ve isteklerinin yüksek olması yönünde katkı sağlayabilir.

Etkinlikler hazır setler ile düzenlenmeyecekse, kullanılacak kart, sensörler ve aktüatörlerin ortamda karmaşaya meydan vermeyecek biçimde kutulanmaları katılımcıların dikkatlerinin dağılmasının ve kendilerini elektronik parçaların dağınıklığı içerisinde rahatsız hissetmelerinin önüne geçebilir.

Alanyazında elde edilen bulgular ve bu çalışma kapsamında elde edilen bulgular sonucunda robotik kodlama etkinliğinin düzenleneceği bilişim teknolojileri sınıfının düzenli olmasının, bilgisayara etkinlikler esnasında kullanılacak kartın (örneğin Arduino uno R3) sürücüsünün ve gerekli yazılımların sağlıklı çalışır şekilde

yüklenmiş olmasının etkinliğin amacına ulaşmasında, katılımcıların süreçten en yüksek düzeyde fayda sağlamasında ve etkinlikleri eğlenceli bulmasında önem taşıyacağı söylenebilir.

Robotik kodlama etkinlikleri katılımcılarının (bu araştırma kapsamında gözlemlenen grup 5. sınıf öğrencileri ile sınırlıdır) aynı ilgi alanına sahip bireylerden ikişerli grup oluşturacak şekilde oturtulması, katılımcıların etkinlikler esnasında daha fazla eğlenmelerini sağlayabilir. Etkinlikleri gerçekleştirmekte zorlandığı gözlenen öğrencilerin etkinliği daha kolay gerçekleştirdiği gözlenen öğrencilerle oturtulması, etkinlikleri gerçekleştirmekte zorlandığı gözlenen öğrencilerin akran öğrenmesi yoluyla süreçten daha fazla fayda sağlamalarına yardımcı olabilir.

Robotik kodlama etkinlikler esnasında elektrik devrelerinin nasıl yapılacağına ilişkin oldukça fazla görsel içeren bilgilendirici kaynakların öğrencilere sunulması, etkinliklerin daha rahat başarılmasında yardımcı olabilir.

6. KAYNAKLAR

- Akpınar, Y., & Altun, A. (2014). Bilgi toplumu okullarında programlama eğitimi gereksinimi. *Elementary Education Online*, 13(1), 1-4.
- Allsop, Y. (2015). İct'den kodlamaya: İngiltere'de teknoloji eğitimi. *Eğitim Teknolojileri Zirvesi*, (s. 303-308). Ankara.
- Altun, A., & Mazman, S. G. (2012). Programlamaya ilişkin öz yeterlilik algısı ölçeğinin Türkçe formunun geçerlilik ve güvenilirlik çalışması. *Eğitimde ve Psikolojide Ölçme ve Değerlendirme Dergisi*, 3(2), 297-308.
- Altun, A., & Mazman, S. G. (2013). Programlama-1 dersinin BÖTE bölümü öğrencilerinin programlamaya ilişkin öz yeterlilik algıları üzerine etkisi. *Journal of Instructional Technologies & Teacher Education*, 2(3), 24-29.
- Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K-6 computational thinking. *Educational Technology & Society*, 19(3), 47-57.
- Armoni, M. (2012). Teaching CS in kindergarten: How early can the pipeline begin? *ACM Inroads*, 3(4), 18-19.
- Aşkar, P., & Davenport, D. (2009). An investigation of factors related to self-efficacy for java programming. *The Turkish Online Journal of Educational Technology – TOJET January*, 8(1).
- Balanskat, A., & Engelhardt, K. (2004). *Computing our future, computer programming and coding - Priorities, school curricula and initiatives across Europe*. Aralık 21, 2016 tarihinde <https://goo.gl/Pbz7IW> adresinden alındı
- Bandura, A. (1977). Self-efficacy: Toward a unifying theory of behavioral change. *Psychological Review*, 84(2), 191-215.
- Bandura, A. (1994). Self-efficacy. In V. S. Ramachaudran (Ed.). *Encyclopedia of Human Behavior*, 4, 71-81.
- Beug, A. (2012). *Teaching introductory programming concepts: A comparison of Scratch and Arduino*. (Yayımlanmamış Yüksek Lisans Tezi). The Faculty of California Polytechnic State University, Obispo, San Luis.
- Bidwell, A. (2013). *Tech companies work to combat computer science education gap*. 22 Aralık, 2016 tarihinde <https://goo.gl/xTXdG3> adresinden alındı
- Bilişim Garaj Akademisi. (2016). Aralık 20, 2016 tarihinde <https://goo.gl/fcrfXs> adresinden alındı
- Büyüköztürk, Ş. (2007). *Sosyal bilimler için veri analizi el kitabı*. Ankara: Pegem A.
- Byrne, P., & Lyons, G. (2001). The effect of student attributes on success in programming. *Proceedings of ITICSE*. 49-52.
- Code.org 2015 Annual Report. (2015). *Global computer science education*. Aralık 20, 2016 tarihinde <https://goo.gl/bHifH0> adresinden alındı

- 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), 13-25.
- Davidson, K. L., & Ljunggren, K. (2010). *Self-Efficacy in programming among STS students*. Technical Reports from Computer Science Education course of Uppsala University. Ağustos 28, 2016 tarihinde <https://goo.gl/52RbkD> adresinden alındı
- Deci, E. L., Eghrari, H., Patrick, B. C., & Leone, D. (1994). Facilitating internalization: The self-determination theory perspective. *Journal of Personality*, 62, 119-142.
- Department for Education. (2013). *The national curriculum in England*. Cheshire, UK: Crown.
- Ertel, J. (2016). *BMW to use HP's new 3D printing system for customized parts*. 12 19, 2016 tarihinde BMW BLOG: <https://goo.gl/S4nRlm> adresinden alındı
- European Commission. (2016). *Coding - the 21st century skill*. Aralık 21, 2016 tarihinde <https://goo.gl/z8L012> adresinden alındı
- Field, A. (2005). *Discovering statistics using SPSS (2 b.)*. London: Sage.
- Furber, S. (2012). *Shut down or restart? The way forward for computing in UK schools*. Londra: The Royal Society.
- Genç, Z., & Karakuş, S. (2011). Tasarımla öğrenme: Eğitsel bilgisayar oyunları tasarımında Scratch kullanımı. *5th International Computer & Instructional Technologies Symposium (ICITS)*. Elazığ.
- George, D., & Mallery, P. (2003). *SPSS for Windows step by step: A simple guide and reference. 11.0 update (4 b.)*. Boston: Allyn & Bacon.
- Gibson, J. P. (2012). Teaching graph algorithms to children of all ages. In *Proceedings of the 17th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education (ITiCSE'12)* (s. 34-39). New York, NY: ACM.
- Gülbahar, Y., & Kalelioğlu, F. (2014). The effects of teaching programming. *Informatics in Education-An International Journal*, 13(1), 33-50.
- ISTE. (2007). *ISTE Standarts for students 2007*. Aralık 19, 2016 tarihinde <https://goo.gl/mQ8xfz> adresinden alındı
- ISTE. (2011). *Operational definition of computational thinking for K-12 education*. Aralık 20, 2016 tarihinde <https://goo.gl/JBYhfj> adresinden alındı
- ISTE. (2016). *ISTE Standarts for students 2016*. Aralık 19, 2016 tarihinde <https://goo.gl/Z1E5bf> adresinden alındı
- 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).
- Kesici, T., & Kocabaş, Z. (2007). *Bilgisayar-1 Liseler için*. Ankara: MEB.
- Kukul, V., & Gökçearslan, Ş. (2014). Scratch ile programlama eğitimi alan öğrencilerin problem çözme becerilerinin incelenmesi. 8. *Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu*, (s. 58-63). Edirne.

- Maloney, J., Peppler, K., Kafai, Y. B., Resnick, M., & Rusk, N. (2008). Programming by choice: Urban youth learning programming with Scratch. (s. 367-371). Portland, Oregon, USA: SIGCSE.
- Mazman, S. G. (2013). *Programlama performansını etkileyen faktörlerin bilişsel tabanlı bireysel farklılıklar temelinde modellenmesi*. Ankara: Hacettepe Üniversitesi.
- National Research Council. (2010). *Committee for the workshops on computational thinking: Report of a workshop on the scope and nature of computational thinking*. Washington, DC: National Academy Press.
- Özden, Y. (2015). *Computational Thinking = Bilgisayarca Düşünme becerileri?* Aralık 20, 2016 tarihinde M. Yaşar Özden - Blok: <https://goo.gl/m98xNg> adresinden alındı
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Inc., Publishers.
- Pillay, N., & Jugoo, V. (2005). An investigation into student characteristics affecting novice programming performance. *ACM SIGCSE Bulletin*, 37(4). doi:10.1145/1113847.1113888
- Prensky, M. (2001, 9). Digital natives, digital immigrants. *On The Horizon*, 9(5), 1-6.
- Przybylla, M., & Romeike, R. (2014). *Overcoming issues with students' perceptions of informatics in everyday life and education with physical computing - suggestions for the enrichment of computer science classes* (s. 6-20). Local Proceedings of the 7th International Conference on Informatics in; Situation, Evolution and Perspectives ISSEP 2014.
- 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.
- 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.
- Ramalingam, V., LaBelle, D., & Wiedenbeck, S. (2004). Self-efficacy and mental models in learning to program. *Proceedings of the 9th annual SIGCSE conference on Innovation and technology in computer science*. Leeds, United Kingdom.
- Resnick, M. (2013). *Learn to code - code to learn*. Aralık 21, 2016 tarihinde <https://goo.gl/K5EN0v> adresinden alındı
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., . . . Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM*, 52(11).
- Rubin, J. (1975). What the "good language learner" can teach us. *TESOL Quarterly*, 9(1), 41-51.
- Rusk, N., Resnick, M., Berg, R., & Pezalla-Granlund, M. (2008). New pathways into robotics: Strategies for broadening participation. *Journal of Science Education and Technology*, 17, 59-69.
- Schermelleh-Engel, K., & Moosbrugger, H. (2003). Evaluating the fit of structural equation models: Test of significance and descriptive goodness-of-fit measures. *Methods of Psychological Research Online*, 8(2), 23-74.
- Scratch for Arduino. (2015). *About S4A*. Aralık 22, 2016 tarihinde <http://s4a.cat> adresinden alındı

- ScratchEd Team. (2011). *Creative Computing - a design-based introduction to computational thinking*. MIT Media Lab.
- Settle, A., & Perkovic, L. (2010). *Computational thinking across the curriculum: A conceptual framework*. Technical Reports. Aralık 23, 2016 tarihinde <http://via.library.depaul.edu/tr/13> adresinden alındı
- Shiller, R. J. (2016). *Four Nobel economists on the biggest challenges for 2016*. Davos: World Economic Forum. Aralık 19, 2016 tarihinde <https://goo.gl/21hKqE> adresinden alındı
- Siber saldırıda 'akıllı' ev cihazları kullanıldı*. (2016, Ekim 24). Aralık 19, 2016 tarihinde BBC TÜRKÇE: <https://goo.gl/8Jw0Yr> adresinden alındı
- Stajkovic, A. D., & Luthans, F. (1998). Self-efficacy and work-related performance: A metaanalysis. *Psychological bulletin*, 124(2).
- Stratasys - 3D Printing & Additive Manufacturing. (2016). *4D Printing-Revolutionizing material form and control*. Şubat 11, 2017 tarihinde <https://goo.gl/8Mlp9H> adresinden alındı
- Sümer, N. (2000). Yapısal eşitlik modelleri: Temel kavramlar ve örnek uygulamalar, 3(6). 49-74. Aralık. *Türk Psikoloji Yazıları*., 3(6), 49-74.
- The European Coding Initiative. (2014). *The European Coding Initiative*. Aralık 20, 2016 tarihinde <https://goo.gl/QwajRH> adresinden alındı
- TTKB. (2012). *Bilişim teknolojileri ve yazılım dersi (5, 6, 7 ve 8. sınıflar) öğretim programı*. Ankara: MEB.
- TTKB. (2016). *Ortaöğretim bilgisayar bilimi dersi (kur 1, kur 2) öğretim programı*. Ankara: MEB.
- TTKB. (2017). *Bilişim teknolojileri ve yazılım dersi öğretim programı*. Ankara: MEB.
- United States Bureau of Labor Statistics. (2013). *Occupational employment projections to 2022*. Aralık 20, 2016 tarihinde <https://goo.gl/wWJqwy> adresinden alındı
- Wing, J. M. (2006). Computational thinking. *Communications of The Acm*, 49(3), 33-35.
- Yükseltürk, E., & Altıok, S. (2016). An investigation of the effects of programming with scratch. *British Journal of Educational Technology*. doi:10.1111/bjet.12453
- Yükseltürk, E., & Altıok, S. (2016). Investigation of pre-service information technology teachers' game projects prepared with Scratch. *SDU International Journal of Educational Studies*, 3(1), 59-66.

7. EKLER

Ek 1: Tez Çalışması Etik Komisyon İzin Muafiyeti Formu

01 / 09 / 2015

Hacettepe Üniversitesi
Eğitim Bilimleri Enstitüsü
Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı Başkanlığı'na

Tez Başlığı / Konusu:	ROBOTİK KODLAMA ETKİNLİKLERİNİN ORTAOKUL ÖĞRENCİLERİNİN KODLAMAYA İLİŞKİN ÖZ-YETERLİK ALGILARINA ETKİSİ VE ETKİNLİKLERE İLİŞKİN ÖĞRENCİ YAŞANTILARI
------------------------------	---

Yukarıda başlığı/konusu gösterilen tez çalışmam:

1. İnsan ve hayvan üzerinde deney niteliği taşımamaktadır,
2. Biyolojik materyal (kan, idrar vb. biyolojik sıvılar ve numuneler) kullanılmasını gerektirmemektedir.
3. Beden bütünlüğüne müdahale içermemektedir.
4. Gözlemsel ve betimsel araştırma (anket, ölçek/skala çalışmaları, dosya taramaları, veri kaynakları taraması, sistem-model geliştirme çalışmaları) niteliğinde değildir.

Hacettepe Üniversitesi Etik Kurullar ve Komisyonlarının Yönergelerini inceledim ve bunlara göre tez çalışmamın yürütülebilmesi için herhangi bir Etik Komisyondan/Kuruldan izin alınmasına gerek olmadığını; aksi durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini saygılarımla arz ederim.



İbrahim KASALAK
(Öğrencinin Adı Soyadı, İmzası)

Öğrenci Bilgileri

Adı Soyadı	İbrahim KASALAK
Öğrenci No	N10226069
Anabilim Dalı	Bilgisayar ve Öğretim Teknolojileri Eğitimi ABD
Programı	Bilgisayar ve Öğretim Teknolojileri Eğitimi
Statüsü	<input checked="" type="checkbox"/> Yüksek Lisans <input type="checkbox"/> Doktora <input type="checkbox"/> Bütünleşik Dr.

Danışman Görüşü ve Onayı

Tezin yürütülmesi için Etik Komisyondan izin alınmasına gerek olmadığını, tezin yürütülmesinin etik açıdan uygun olduğunu onaylarım.


Prof. Dr. Arif ALTUN

Ek 2: Tez Çalışması Orijinallik Raporu



HACETTEPE ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS/DOKTORA TEZ ÇALIŞMASI ORJİNALLİK RAPORU

HACETTEPE ÜNİVERSİTESİ
EĞİTİM BİLİMLER ENSTİTÜSÜ

BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ANA BİLİM DALI BAŞKANLIĞI'NA

Tarih: 04/03/2017

Tez Başlığı: Robotik Kodlama Etkinliklerinin Ortaokul Öğrencilerinin Kodlamaya İlişkin Öz-yeterlik Algılarına Etkisi Ve Etkinliklere İlişkin Öğrenci Yaşantıları

Yukarıda başlığı verilen tez çalışmamın tamamı (kapak sayfası, özetler, ana bölümler, kaynakça) aşağıdaki filtreler kullanılarak **Turnitin** adlı intihal programı aracılığı ile kontrol edilmiştir. Kontrol sonucunda aşağıdaki veriler elde edilmiştir.

Rapor Tarihi	Sayfa Sayısı	Karakter Sayısı	Savunma Tarihi	Benzerlik Endeksi	Gönderim Numarası
04/03/2017	85	141193	07/02/2017	%11	761116401

Uygulanan filtreler:

- 1- Kaynakça hariç
- 2- Alıntılar dâhil
- 3- 5 kelimedenden daha az örtüşme içeren metin kısımları hariç

Hacettepe Üniversitesi Eğitim Bilimleri Enstitüsü Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Uygulama Esasları'nı inceledim ve çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini saygularıyla arz ederim.

Adı Soyadı: İbrahim KASALAK
Öğrenci No: N10226069
Anabilim Dalı: Bilgisayar ve Öğretim Teknolojileri Eğitimi ABD
Programı: Bilgisayar ve Öğretim Teknolojileri Eğitimi
Statüsü: Y.Lisans Doktora Bütünleşik Dr.

04/03/2017

DANIŞMAN ONAYI

UYGUNDUR.

Prof. Dr. Arif ALTUN



HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF EDUCATIONAL SCIENCES
THESIS/DISSERTATION ORIGINALITY REPORT

HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF EDUCATIONAL SCIENCES
TO THE DEPARTMENT OF COMPUTER EDUCATION AND INSTRUCTIONAL TECHNOLOGIES

Date: 04/03/2017

Thesis Title: Effects of Robotic Coding Activities on the Effectiveness of Secondary School Students 'Self-Efficacy and Student Experience About Activities

The whole thesis that includes the *title page, introduction, main chapters, conclusions and bibliography section* is checked by using **Turnitin** plagiarism detection software take into the consideration requested filtering options. According to the originality report obtained data are as below.

Time Submitted	Page Count	Character Count	Date of Thesis Defence	Similarity Index	Submission ID
04/03/2017	85	141193	07/02/2017	%11	761116401

Filtering options applied:

1. Bibliography excluded
2. Quotes excluded
3. Match size up to 5 words excluded

I declare that I have carefully read Hacettepe University Graduate School of Educational Sciences Guidelines for Obtaining and Using Thesis Originality Reports; that according to the maximum similarity index values specified in the Guidelines, my thesis does not include any form of plagiarism; that in any future detection of possible infringement of the regulations I accept all legal responsibility; and that all the information I have provided is correct to the best of my knowledge.

I respectfully submit this for approval.

Name Surname: İbrahim KASALAK

Student No: N10226069

Department: The Department of Computer Education and Instructional Technologies

Program: Computer Education and Instructional Technologies

Status: Masters Ph.D. Integrated Ph.D.

04/03/2017

ADVISOR APPROVAL

APPROVED.

Prof. Dr. Arif ALTUN

Ek 3: Örnek Robotik Kodlama Etkinlik Planı

Etkinlik Adı : Robotik Kodlama Etkinliği 3

Ders Süresi : 2 ders saati

Gerekli Malzemeler:

- Bilgisayar
 - Arduino driver'ı, Arduino programı, S4AFirmware yüklenmiş, Programın COM ayarı yapılmış, Scratch for Arduino programı yüklenmiş, ortam hazırlanmış halde olmalı
- USB bağlantı kablosu
- Arduino Uno R3 Kart
- Breadboard (ekmek tahtası)
- Farklı renklerde LED ampuller
- Dijital düğme (buton)
- Bağlantı kabloları
- 10 KV direnç

KAZANIMLAR

Bu etkinlik sonunda öğrencilerin

- Arduino uno R3 kartın dijital giriş (input) pinlerini kullanabilmeleri
- Dijital giriş cihazlarından olan düğme (buton) bağlantılarını yapabilmeleri
- Düğme (buton) örneğinde dijital giriş (input) değerini (true-false) kavramaları, bu tür değerlerin önceden tanımlı değişken olduklarını kavramaları
- Arduino bağlantılı karakter ekleyebilmeleri
- Buton kullanarak, butona basılıp basılmama durumuna göre;
 - “Eğer ise-başka (öyle değilse)” koşul ifadesini,
 - dijital giriş (input) pinlerinden okunan değerler (true-false) ile “sensor dijital 2-3 pressed” kod bloklarını ve
 - dijital çıkış (output) pinlerini yöneten “digital on” – “digital off” kod bloklarını kullanarak Led ampul yaktırabilecek kod bloklarını düzenleyebilmeleri
- Değişken tanımlayabilmeleri
- Değişken değerlerini duruma bağlı olarak “eğer ise – başka” koşul ifadesi kontrol bloklarından yararlanarak değiştirebilmeleri
- “Eğer ise – başka” koşul ifadesi kontrol bloklarından yararlanarak değişken değerlerine bağlı olarak sahneye efekt uygulayabilmeleri
- Karşılaştıkları hataları, problemleri çözebilmeleri beklenmektedir.

ÖN BİLGİ GEREKSİNİMLERİ

- Elektrik akımı, kısa devre, direnç, iletken kablo,
- Scratch programında paralellik mantığını, “tıklandığında”, “sürekli”, “1 sn bekle” kontrollerini bilmeleri beklenmektedir.
- Değişken oluşturma konusunda ön bilgi düzeylerinin düşük olduğu varsayılmaktadır.

DERSE HAZIRLIK

- Öğrencilere bu etkinlikte yapacağımız devre ve program konusunda bilgi verilerek dikkatleri çekilir, güdülenmeleri sağlanır.

- Öğrencilerin her ne kadar önceki etkinliklerde elektrik konusunda bilgi gereksinimleri karşılanmaya çalışılsa da öğrencilerin 5. Sınıf olmaları nedeniyle bilgi eksikliklerinin pekiştirilmeye ihtiyacı vardır. Dersin başında 3-4 dakika kadar şekiller aracılığıyla direnç, kısa devre, anahtar konularında ön bilgilerini hatırlatmaya ihtiyaç duyulacaktır.

ETKİNLİĞİN İŞLENİŞ SÜRECİ

- Ön bilgilerin verilmesinden sonra düğmeye basıldığında LED ampulün yanacağını, kodlamaya başlamadan önce ekrana ekleyeceğimiz insan karakterinin kostümünü değiştirmek suretiyle oynama görüntüsü oluşturacağımızı, aynı zamanda sahnenin de ışıklı bir sahne ile değiştirilmesi ile hem Scratch ekranında hem breadboardda LED yaktırmak suretiyle güzel bir uygulama geliştireceğimizi açıklarız.
- Bu çalışmaya başlamadan önce ön bilgileri hatırlamak amacıyla tek bir LED ampulün breadboard üzerinden yaktırılması istenir. Breadboard, şekil üzerinden tekrar hatırlatılır.
- Dersin başında anlatılan elektrik devresi şemasına dönülür. Burada kısa devre olmaması için direnç takılması gerektiği tekrar hatırlatılır. Ardından buton bağlantı şeması adım adım çizilir. Ardından görsel, animasyon gibi yardımcı materyallerle devre şeması kavratılır.
- Öğrencilerden buton bağlantısını yapmaları istenir. Daha sonra bu butona basıldığında Scratch for Arduino sahnesindeki “sensor board” üzerinde “digital 2” göstergesinin “true”, basılmadığında “false” olduğunu fark etmeleri sağlanır. Aslında bu “true” ve “false” değerlerinin de butondan okunan önceden (otomatik) tanımlı değişken olduğu öğrencilere anlatılır. Çalışmamızda Arduino uno karakterine ihtiyacımız olmadığı için bu karakterini silmeleri istenir. Karakter silince Arduino uno ile bağlantının koptuğuna dikkat çekilir.
- Bir scratch karakteri eklenerek bu karakterin arduino ile bağlantı kurmadığı, Arduino kod bloklarının bu karakterde görünmediği gösterilir.
- Bunun üzerine Arduino bağlantılı karakter ekleme düğmesi gösterilir ve geliştireceğimiz uygulamada kullanacağımız “balerina-a” karakterini ekleriz, butona bastığımızda karakterin kostüm değiştirmesi için karaktere “balerina-b” kostümünü ekleriz. Öğrencilerden “eğer ise – öyle değilse” koşul ifadelerini kullanarak buton basıldığında karakterin kostümünü değiştirecek kod bloklarını oluşturmaları istenir. S4A’da “öyle değilse” ifadesinin “başka” olarak Türkçeye çevrildiği gösterilir. Çevirileri farklı olsa da aslında aynı blok oldukları belirtilir.
- “Dijital 2”ye bağlantısı yapılmış olan butona basıldığında “digital 13”e bağlı LED ampulün yanmasını sağlayacak kod bloklarını oluşturmaları istenir.
- Öğrencilerin çalışmalarını bitirmesi beklenir. Ardından “spotlight-stage” isimli sahneyi ekleyerek “renk-bloğunu 25 ile değiştirin” görünüm kod bloğu ile sahnede meydana gelen efekt gösterilir. Bu efektin meydana getirdiği etkiyi ortadan kaldıracak “renk değerini 0 yapın” kod bloğu gösterilir. Bu kod bloklarının çalışmasının butona basılıp basılmamasını “eğer ise-başka” kod bloklarıyla kontrol edeceğimizi söyleriz. Ancak sahne seçili iken Arduino ile ilgili hiçbir kod bloğu olmadığı gösterilir. Bunu başarmak için ne yapabileceğimiz tartışılır. Değişkenle başarabileceğimiz sonucuna varıldıktan sonra butona basılması ile değişkenin değerini “eğer ise-başka” koşul ifadesi kontrol bloklarıyla 1 veya 0 yapacağımızı, bu 1 ve 0 değerlerini kullanarak yine “eğer ise – başka” koşul ifadesi kontrol bloklarıyla sahneye efekt uygulayabilmemizi sağlayacak kod bloklarını oluşturmaları istenir.
- Görevi başaramayanların da başarabilenlerden yardım istemeleri söylenir. Akran öğrenmeleri yoluyla işbirlikçi-sosyal öğrenme ortamı oluşturulur.
- Öğrencilerin zorlandıklarının fark edildiği yerlerde öğrencilere yardımcı olunur, tekrar yapılır.

- Çalışma tamamlanırsa devreyi bozarak tekrar oluşturmaları istenir. Böylece yeni öğrenilen kazanımlar tekrar edilmiş olur.

ETKİNLİK SONU DEĞERLENDİRME

- Öğrencilerdeki görevi yerine getirebilme düzeyleri ve zorlandıkları noktalar gözlemlenerek tekrar ihtiyacı duyulan konunun olup olmadığına karar verilir.

YANSIMALAR ve TARTIŞMA

- Öğrencilerin oldukça eğlendiklerini gözlemlediğimiz güzel bir etkinlik oldu. Özellikle karakter ve sahne seçimi, renk çeşitliliği ve görselliği güzel hazırlanmış, fiziksel devreler ve buton gibi somutlaştırmayı artırıcı etmenler öğrencilerin ilgi ve etkinlikleri gerçekleştirmeye yönelik isteklilik düzeyini oldukça artırmakta.
- Öğrencilerin bu etkinlikte artık koşul kontrol ifade blokları olan “eğer ise- başka” kod bloklarını kullanarak zorlanmadan algoritma oluşturabildikleri söylenebilir.
- Değişken oluşturma ve değişken değerini değiştirme, bu değerleri kullanarak belirli işlemleri yerine getirme konusunda oldukça ilerleme kaydettiklerini gözlemledim. Birkaç etkinlik daha yapıldıktan sonra tamamen öğrenecekleri kanaatindeyim.
- Elektrik devreleri her ne kadar çizimlerle ve başka görsellerle desteklense de öğrenciler buton bağlantı devresini oluştururken zorlandılar. Bu nedenle daha önceki etkinliklerde yaptığım gibi devreyi kendilerinin, görsele bakmadan oluşturacakları bir etkinliği sürdürmedim. Öğrencilerin büyük çoğunluğu başaramayınca devre şeması açık halde iken öğrencilerin bakarak yapmalarına izin vermek zorunda kaldım. Elektrik konusunun daha fazla işlenmesi gerekli eğer öğrencilerden bu yönde bir kazanım da bekleniyorsa. Değilse şekle bakarak devreyi kurdular. Birkaç defa tekrarlanınca daha kalıcı olacaktır.
- Scratch programıyla çalışırken 3-4 etkinlik sonunda sıkılan öğrenciler, somut işlemler gerçekleştirilen “Scratch for Arduino” ile Robotik kodlama etkinliklerinde çok daha fazla uzun süre ilgilerini ve etkinliğe devam etme yönünde isteklerini korudular. Etkinlikleri gerçekleştirmek için sabırsızlandılar ve bir sonraki etkinlikle ilgili sorular sordular. Öğrencilerde istenmedik davranışlar gözlemlenmedi.

Ek 4: Blok Temelli Programlamaya İlişkin Öz-yeterlik Algısı Ölçeği

Merhaba, bu ölçek sizin scratch programına yönelik kendinize ilişkin öz yeterlik algınızı belirlemek üzere hazırlanmıştır. Araştırmaya yönelik katkınızdan dolayı teşekkür ederiz.

İbrahim KASALAK

Prof. Dr. Arif ALTUN

Uygun olan kutucukları fareyle tıklayarak işaretleyiniz.

* Gerekli

Adınız Soyadınız *

Yanıtınız

Cinsiyetiniz *

- Erkek
 Kız

Evinizde bilgisayar var mı? *

- Evet
 Hayır
 Var, ama kullanmama izin verilmiyor veya bilgisayar arızalı olduğu için çalışmıyorum

Evinizde İnternet bağlantısı var mı? *

- Evet
 Hayır
 Evet var, ama kullanmama izin verilmiyor veya bilgisayar arızalı olduğu için İnternete bağlanamıyorum.

Ders dışında Scratch programına çalışma olanağınız var mı? *

- Çalışma olanağım yok
 Çalışma olanağım var

Ders dışında Scratch programına çalışma olanađınız varsa hangi sıklıkta çalışmaktasınız?

Yoksa boş bırakınız, varsa ařađıdaki açılan kutudan seçiniz.

Seçin ▼

Ders dışında Arduino çalışma olanađınız var mı? (Arduino setiniz var mı?) *

- Çalışma olanađım yok
- Çalışma olanađım var

Ders dışında Arduino çalışma olanađınız varsa hangi sıklıkta çalışmaktasınız?

Yoksa boş bırakınız, varsa ařađıdaki açılan kutudan seçiniz.

Seçin ▼

Ders dışında sizi Scratch programına çalıştıran var mı? *

- Evet
- Hayır

Ders dışında sizi Arduino çalıştıran var mı? *

- Evet
- Hayır

SONRAKI

Aşağıdaki Scratch programına ilişkin verilen görevleri yaparken kendinize olan güveninizi 1 ile 5 arasında derecelendirerek belirtiniz. Anlamadığınız soru olursa boş bırakınız.

- 1- Hiç Güvenmiyorum
- 2- Biraz Güveniyorum
- 3- %50 / %50
- 4- Oldukça Güveniyorum
- 5- Tamamen Güveniyorum

Scratch programını kullandığınız zamanları düşünerek, aşağıdaki ifadeleri okuyunuz. Scratch programını, yazılar bloğunu hatırlamanız gayesiyle ekran görünümü sunulmuştur.

Scratch Ekran Görünümü - Yazılar Bloğu Orta Bölümde Açık



1-) Scratch'te yazılmış bir program (yazılar) gördüğümde, çalıştırıldığında neler olacağını söyleyebilirim.

	1	2	3	4	5	
Hiç Güvenmiyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

2-) Başkası tarafından hazırlanan bir programı (yazılarını) okuyup anlayabilirim.

	1	2	3	4	5	
Hiç Güvenmiyoru m	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

3-) Bir karaktere herhangi bir hareket vermek istediğimde, scratch'te bunu nereden yapabileceğimi bilirim.

	1	2	3	4	5	
Hiç Güvenmiyoru m	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

4-) Sahnedeki karakteri istediğim hızda hareket ettirebilirim.

	1	2	3	4	5	
Hiç Güvenmiyoru m	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

5-) Sahnedeki karakteri sürekli hareket ettirebilirim.

	1	2	3	4	5	
Hiç Güvenmiyoru m	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

6-) Scratch'te bir karakterin görünümünü (kostüm, renk, boyut, konuşma gibi) bir koşula bağlı olarak (örneğin: eğer ise) değiştirebilirim.

	1	2	3	4	5	
Hiç Güvenmiyoru m	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

7-) Scratch'te bir karakterin hareketini (hızı, yönü, konumu gibi) bir koşula bağlı olarak (örneğin: eğer ise) değiştirebilirim.

	1	2	3	4	5	
Hiç Güvenmiyoru m	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

8-) Bir oyunda kullanıcının elde ettiği puan değerinin tutulacağı bir değişken oluşturabilirim.

	1	2	3	4	5	
Hiç Güvenmiyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

9-) Bir oyunda istenilenler başarıldıkça "Puan" veya "Skor" değerinin arttığı veya azaldığı bir program hazırlayabilirim.

	1	2	3	4	5	
Hiç Güvenmiyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

10-) İstenilenler açıkça tanımlandığında oldukça karmaşık ve uzun kodlardan (yazılardan) oluşan bir oyun hazırlayabilirim.


	1	2	3	4	5	
Hiç Güvenmiyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

11-) Scratch'te hazırlanan bir programdaki hataları bulabilirim.

	1	2	3	4	5	
Hiç Güvenmiyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

12-) Scratch'te hazırlanan bir programdaki hataları düzeltip çalışabilir hale getirebilirim.

	1	2	3	4	5	
Hiç Güvenmiyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Güveniyorum

Çalışmaya katkınızdan dolayı teşekkür ederiz. 

Ek 5: Etkinlik Algısı Ölçeği

Aşağıdaki katıldığınız etkinliğe yönelik algınıza yönelik maddeleri 1 ile 5 arasında derecelendirerek belirtiniz. Anlamadığınız soru olursa boş bırakınız.

1- Hiç Katılmıyorum

2- Biraz Katılıyorum

3 - %50 / %50

4- Oldukça Katılıyorum

5- Tamamen Katılıyorum

1-) Bu etkinliği yapmak eğlenceliydi.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

2-) Bu etkinliğin benim gelişimim için önemli olduğuna inanıyorum.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

3-) Bu etkinliği yaparken çok eğlendim.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

4-) Bence bu gerçekten önemli bir etkinlikti.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

5-) Bu etkinliği yapmak istediğim için yaptım.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

6-) Bence bu çok sıkıcı bir etkinlikti.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

7-) Bu etkinliđi faydalı olduđunu dűşündűđüm için tekrar yapmak isterim.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

8-) Bu etkinliđi yapmanın benim için faydalı olabileceđine inanıyorum.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

9-) Bu etkinliđin okulda daha iyi olmama yardımcı olabileceđine inanıyorum.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

10-) Bunun çok ilgi çekici bir etkinlik olduđunu dűşündüm.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

11-) Bu etkinliđin bana kattıđı bazı şeyler olduđu için tekrar yapmak isterim.

	1	2	3	4	5	
Hiç Katılmıyorum	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Tamamen Katılıyorum

Ek 6: Etkinliđi Eđlenceli Bulmaya İlişkin Yarı Yapılandırılmış Görüşme Soruları

1. Etkinlik sonunda öğrencilere yöneltilen sorular:

- Scratch programıyla çalışmayı eđlenceli bulmuş muydun?
- Sıkıldığını hissettiğin oldu mu? Olduysa hangi etkinlikte veya kaçınıc haftadan sonra sıkıldığını hissetmiştin?
- Robotik kodlama etkinliklerine başladıktan sonra yine sıkıldın mı (Öğrenci Scratch programıyla çalışırken sıkıldığını belirtmişse)?
- Scratch'te çalışırken sıkılmaya başladığını söylemiştin. Robotik Kodlama Etkinliğinde ise sıkılmadığını söyledin. Sence bunun sebebi ne olabilir?

Her etkinliđin sonunda öğrencilere yöneltilen sorular:

- Sence etkinlik-ders eđlenceli miydi yoksa sıkıcı mıydı?
- Ders esnasında seni eđlendiren şeyler nelerdi? Ne yaparken eđlendin (Eđlendiğini belirttiyse)?
- Eđlenceli bulduysan derste sıkıldığın bölümler, zamanlar oldu mu? Sıkıldıysan daha çok neden sıkıldın?

Ek 7: Etkinliklerin Kişisel Gelişime Etkisine İlişkin Yarı Yapılandırılmış Görüşme Soruları

- Gerçekleştirdiğimiz Robotik kodlama etkinliklerinin kişisel gelişimine katkısı olduğunu düşünüyor musun?
- Robotik kodlama etkinliklerinin kişisel gelişimine nasıl katkı sağladığını düşünüyorsun?
- Robotik kodlama etkinliklerinin okuldaki başarısına, okulda daha iyi olmana katkı sağlayacağını düşünüyor musun? Nasıl?

8. ÖZGEÇMİŞ

Kişisel Bilgiler

Adı Soyadı	İbrahim KASALAK
Doğum Yeri	Şuhut / Afyonkarahisar
Doğum Tarihi	23.03.1983

Eğitim Bilgileri

Lisans	Hacettepe Üniversitesi Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü	2005
Lise	Afyonkarahisar Gazi Anadolu Teknik Lisesi (Bilgisayar Bölümü)	2001
Yabancı Dil	İngilizce: Okuma (Çok iyi), Yazma (İyi), Konuşma (Orta)	

Meslekte Deneyim

Beşikkaya Ortaokulu / Ankara	Bilişim Teknolojileri Öğretmeni	2015-
Uşak Anadolu Öğretmen Lisesi	Bilişim Teknolojileri Öğretmeni	2010-2015
Kırıkkale İl Millî Eğitim Müdürlüğü	MEBBİS Yöneticisi	2008-2010
Kırıkkale İl Millî Eğitim Müdürlüğü	İl Bilişim Teknolojileri Koordinatörü	2008-2010
Mehmet Ali Eren İlköğretim Okulu / Kırıkkale	Bilişim Teknolojileri Öğretmeni	2005-2008

Öğrencilerle Yürütülen Çalışmalar

Kodlama Etkinlikleri, Robotik Kodlama Etkinlikleri, Grafik Tasarım Çalışmaları, On Parmak Hızlı Yazma (Intersteno) Çalışmaları

Öğretmenlere Verilen Eğitimler / Sorumlu Olunan Projeler

Uşak Şehit Abdulkadir Kılavuz Anadolu Öğretmen Lisesi 149391 No'lu TÜBİTAK 4006 Bilim Fuarı Projesi, Proje Yürütücüsü	2015
Fatih Projesi Eğitimde Teknoloji Kullanım Kursu Eğitim Görevlisi	2012 -
Fatih Projesi Bilişim Teknolojileri Rehber Öğretmeni Yetiştirme Kursu Eğitim Görevlisi	2012 -
Web Tabanlı İçerik Geliştirme Kursu Eğitim Görevlisi	2009-2010
Web Tabanlı İçerik Geliştirme Projesi İl Koordinatörü (Kırıkkale)	2009-2010
İntel Öğretmen Programı Kursu Eğitim Görevlisi	2008-2010
İntel Öğretmen Programı İl Koordinatörü (Kırıkkale)	2008-2010
Flash CS4 (AS 3.0) ile Animasyon Hazırlama Kursu Eğitim Görevlisi	2008-2010
Adobe Photoshop CS4 Programı / Grafik Tasarım Kursu Eğitim Görevlisi	2008-2010
Web Tasarım Kursu Eğitim Görevlisi	2008-2009

Millî Eğitim Bakanlığı ve Diğer Kurumlardan Alınan Eğitimler

Robotik Kodlama Eğitimi	26/11/2015-27/11/2015
Android Programlama Kursu	07/09/2015-11/09/2015
Fatih Projesi Eğitimci Eğitimi Güncelleme Kursu	23/03/2015-05/04/2015
Eğitimde Fatih Projesi (Pardus Kullanımı) Kursu	30/09/2013-04/10/2013
Fatih Projesi Ağ Altyapısı Semineri	23/09/2013-25/09/2013
Eğitimde Fatih Projesi (Teknoloji Kullanımı) Kursu	10/12/2012-14/12/2012
İçerik Yönetim Sistemi (E-Öğrenme Portalı - Moodle) Kursu	18/06/2012-29/06/2012
Web Uygulamaları Güvenliği Kursu	28/01/2012-31/01/2012
Fatih Projesi Eğitimde Teknoloji Kullanım Kursu	16/01/2012-20/01/2012
Bilişim Teknolojileri İle Proje Hazırlama Kursu II. Kademe	14/11/2011-18/11/2011
Bilişim Teknolojileri İle Proje Hazırlama Kursu I. Kademe	04/10/2011-28/10/2011
Web Programcılığı Kursu (Vb.Net I.Kademe)	21/03/2011-01/04/2011
Web Tabanlı İçerik Geliştirme Kursu (II. Kademe)	12/10/2009-23/10/2009
Web Tabanlı İçerik Geliştirme Kursu (I. Kademe)	27/07/2009-07/08/2009
İntel Öğretmen Programı Semineri	19/01/2009-23/01/2009
İntel Öğretmen Programı Eğitimi Kursu	02/06/2008-13/06/2008
Bilişim Teknolojisi Uygulamaları Tanıtım Semineri	20/10/2008-24/10/2008
Uygulamalı Strateji Planlama Semineri	03/05/2008-17/05/2008
Flash Programı Kullanım Kursu	31/03/2008-25/04/2008
Photoshop (Fotoğraf ve Resim İşleme) Program Kursu	22/10/2007-19/11/2007
Temel Hazırlayıcı Kursu	11/11/2005-25/02/2006

Akademik Çalışmalar

<p>Kasalak, İ. (2008). okullardaki bilişim teknolojileri sınıflarının organizasyon yapısı, kullanım amaçları, yaşanan sorunlar ve çözüm önerileri: Kırıkkale örneği. 2. <i>Uluslararası Bilgisayar ve Öğretim Teknolojileri Eğitimi Sempozyumunda sunulmuştur</i>. 16-18 Nisan 2008, Kuşadası, Türkiye.</p> <p>Usluel, Y. K., Doğan, T., Atasayar, A., Kasalak, İ., & Mısırlı, Ö. (2007). The Teachers' stages in the innovation decision process and their satisfaction and importance level about provided support in the ICT adoption process. <i>Society for Information Technology and Teacher Education International Conference</i>, (s. 26-30). San Antonio, Texas.</p>

İletişim

E-Posta Adresi	i.kasalak@gmail.com
Jüri Tarihi	07 Şubat 2017