

Üniversitede Programlama Dersi Alan Öğrencilerin Eğitsel Yazılım Geliştirmeye Yönelik Öz-yeterlik Algıları

Mustafa AKSOĞAN¹

Fatih KALEMKUŞ²

Oğuzhan ÖZDEMİR³

Öz: Bu araştırmanın amacı Bilgisayar ve Öğretimi Teknolojileri Eğitimi dışında kalan ve ağırlıklı olarak bilgisayar programlama dersleri alan bölüm öğrencilerinin eğitsel yazılım geliştirme öz-yeterlilik algılarını ölçmektir. Bu amaç doğrultusunda 2018-2019 eğitim-öğretim yılında üç farklı devlet üniversitesinde öğrenim gören 372 öğrenciye ulaşılmıştır. Verilerin toplanması için Aşkar ve Dönmez'in (2004) geliştirdiği "Eğitim Yazılımı Geliştirme Öz-Yeterlilik Algısı Ölçeği" kullanılmıştır. Araştırma bulgularına göre, öğrencilerin eğitsel yazılım geliştirme aşamalarında proje yönetimi ve öğretim tasarımı aşaması için kendilerini yeterli gördükleri buna karşı eğitsel yazılım tasarlamada animasyon tasarımı, ses ve video birimi aşamasında kendilerini yetersiz gördükleri belirlenmiştir. Bunların yanı sıra eğitsel yazılım geliştirme öz-yeterlilik algısının cinsiyet değişkenine göre erkek öğrenciler, öğrenim gördüğü bölüme göre yazılım mühendisliği öğrencileri, öğrenim gördüğü sınıf düzeyine göre 3. sınıf öğrencileri lehine anlamlı farklılık gösterdiği belirlenmiştir. Ayrıca daha önce eğitsel yazılım geliştirme aşamasında görev alan öğrencilerin görev almayan öğrencilere göre ölçeğin tüm aşamalarında kendilerini daha yeterli gördükleri belirlenmiştir. Araştırma sonuçları doğrultusunda programlama ağırlıklı eğitim veren bölümlerde, öğrencilerin yazılım geliştirme süreçlerinde aktif bir şekilde yer almalarını sağlayacak faaliyetlerin yapılması önerilebilir.

Anahtar kelimeler:
Eğitsel yazılım, Öz-yeterlilik algısı, Programlama, Yazılım geliştirme

Self-Efficacy Perceptions of Students Who Taking Programming Course at University for Developing Educational Software

Abstract: The aim of this study is to assess the perceptions of self-efficacy in educational software development in a group of students who mainly take computer programming courses and students' enrolled for Computer and Instructional Technologies program were excluded from the study. For this purpose, 372 students studying at three different state universities were reached in the 2018-2019 academic year. "Educational Software Development Self-Efficacy Scale" developed by Aşkar and Dönmez (2004) was used to collect the data. According to the findings of the research, it was determined that the students considered themselves sufficient for project management and instructional design phase, which were the stages of educational software development. However, they found themselves inadequate in the animation design, sound and video unit stage. What is more, it was determined that the perception of educational software development self-efficacy differed significantly in favor of male students, software engineering students and 3rd grade students. In addition, it was found that the students who took part in the educational software development phase before, considered themselves more competent at all stages of the scale than the students who did not take part. In line with the results of the research, it may be suggested to carry out activities that will enable students to actively participate in software development processes in departments that provide programming-based education.

Key Words:
Educational software, Self-efficacy perception, Programming, Software development

Geliş Tarihi : 03.09.2020
Kabul Tarihi : 02.12.2020
Yayın Tarihi : 30.12.2020

¹ Öğr. Gör., Malatya Turgut Özal Üniversitesi, Akçadağ MYO, Bilgisayar Prog. mustafa.aksogan@ozal.edu.tr, ORCID: 0000-0002-6826-3902

² Milli Eğitim Bakanlığı, kalemkus@gmail.com, ORCID: 0000-0001-7218-955X

³ Doç. Dr., Fırat Üniversitesi, Eğitim Fakültesi, Bilgisayar ve Öğretim Teknolojileri Eğt., oguzhan@firat.edu.tr, ORCID: 0000-0002-5310-6605

GİRİŞ

İnsanoğlu varoluşundan beri bilgiyi sürekli olarak öğrenmenin ya da öğrenilen bilgiyi sonraki kuşaklara aktarmanın daha kolay ve daha hızlı yollarını bulmak için çeşitli araştırmalar yapmış ve yeni teknolojik ürünler geliştirmiştir (Çepni, 2005). Bu araştırmalar sonucunda ortaya çıkarak baş döndürücü hızla gelişmekte olan teknoloji, eğitim – öğretim süreçlerinde farklı medya araçlarının birlikte kullanılabilmesine ve yeni araçlar geliştirilmesine imkân tanımaktadır (Altun, 2010). Çünkü teknolojinin gelişmesi ile birlikte teknolojiyi kullanarak büyüyen ve teknoloji ile sürekli etkileşim halinde olan dijital yerliler olarak tanımladığımız günümüz kuşaklarının önceki kuşaklara göre öğrenme açısından farklılıklar göstermesi (Prensky, 2001), öğrenme – öğretme ortamlarının yeniden düzenlenmesi gerekliliğini ortaya koymaktadır. Bu vesileyle, öğrenme – öğretme ortamlarında kullanılan araç – gereçlerinde öğrenci ilgi ve ihtiyaçları doğrultusunda geliştirilmesi ihtiyacı daima zorunlu hale gelmiştir (Güven, 2006). Sürekli olarak teknolojiyi kullanan ve sanal ortamlarda vakit geçiren öğrenciler daha çok görsel – işitsel araçlarda, oyunlarda ve etkileşimli ortamlarda yaparak – yaşayarak deneyim kazanmaktadırlar (Öner ve Öztürk, 2019). Nitekim Mutiara ve ark. (2007) bireysel olarak farklı öğrenme düzeylerine sahip öğrencilerin, görsel-işitsel olarak zenginleştirilmiş etkileşimli öğretim materyallerini kullanmalarının öğrenmeyi kolaylaştırabileceğini belirtmiştir. Bunun yanı sıra bilgisayarlar öğrencilerin güdülenmesini artırması, hayat boyu öğrenmelerini kolaylaştırması ve öğretim programlarını ilgi ve ihtiyaçları doğrultusunda esnek hale getirmesiyle eğitim ortamlarının vazgeçilmez araçları haline gelmiştir (Alkan, 1998; Külahçı ve Gürol, 1991).

Erişen ve Çeliköz (2010) bilgisayar destekli öğretimi (BDÖ), öğrencinin öğrenme – öğretme ortamlarında bilgisayarla etkileşim halinde olması ve bu süreçte öğretim aracı olarak bilgisayardan yararlanması olarak tanımlarken, Uşun (2000) eğitim sürecini ve öğrenci konsantrasyonunu destekleyen, öğrencilerin bireysel öğrenme hızlarına göre faydalanabilecekleri, bireysel öğrenme türlerinin bilgisayar ile entegrasyonu sonucu ortaya çıkmış bir yöntem olarak tanımlamaktadır. Dolayısıyla eğitim sürecinde etkili öğrenmeyi kolaylaştıran yöntemlerden biri olarak kullanılan bilgisayar destekli öğretim (Yenice ve ark., 2003), öğretilecek içeriklerin ve eğitimsel etkinliklerin öğrencilere aktarılmasında etkili olarak kullanılabilir (Hannafin ve Peck, 1988). Bunun yanı sıra bilişim teknolojilerinin yaygın olarak kullanıldığı günümüzde, hızla gelişen eğitsel web teknolojileri bireylerin zaman, mekân yer ve yaş açısından oldukça önemli fırsatlar sunan eğitim araçlarını tercih etmektedirler (Pala ve Doğan, 2009). Bireylerin yoğun olarak e-öğrenme ortamlarına ilgisi gün geçtikçe mobil öğrenme ortamlarına yönelmeye başlamıştır (Wu ve ark., 2013). Wi-fi gibi kablosuz iletişim imkânı sunan teknolojilerin gelişmesiyle öğrenme ortamlarında kablosuz uygulamalar önemli yer tutmaya başlamış ve kablosuz aygıtlar göz önünde bulundurularak eğitsel yazılımlar geliştirilmiştir (Hwang ve ark., 2008). Dolayısıyla bireyler günümüzde cep telefonları, taşınabilir oyun araçları, cep bilgisayarları, tablet bilgisayarlar ve dijital ses kayıt cihazları gibi mobil araçları kullanarak etkileşimli oyunlar, simülasyonlar, e-içerikler ve sosyal paylaşım ortamlarında kolektif ya da bireysel öğrenme deneyimi yaşayabilmektedirler (Keskin, 2010).

Eğitim yazılımı, bilgisayar destekli öğretimi gerçekleştirmek ve bilişim teknolojilerini eğitim-öğretim süreçlerinde etkili bir şekilde kullanabilmek için kullanılır ve öğretim yazılımları olarak da bilinmektedir (Akkoyunlu, 2008). Eğitim yazılımları, herhangi bir içerik ya da problem durumunu öğretmede hızlı ve etkili öğrenmeyi sağlamak için bilgisayar yazılımlarından faydalanarak öğretilecek içeriğin, öğrenci ilgi ve ihtiyaçları doğrultusunda görsel ve işitsel olarak oldukça zenginleştirilmesi ve yine bilgisayar yardımıyla öğrencilere sunulmasında kullanılan yazılımlar olarak tanımlanmaktadır (Kazu ve Yavuzalp, 2008). Yine bir başka ifadeyle Keser (1991:13) öğrencilere öğretilecek içeriklerin programlama dilleri ya da sistemlerden faydalanarak bilişim araçlarına uygulanması sonucunda ortaya çıkarılan ve eğitici öğeleri de içeren ders yazılımları olarak tanımlamaktadır. Programlama dilleri kullanılarak geliştirilen eğitim yazılımları, ders programlarının etkili öğretim için bilgisayara uygun hale getirilmesiyle geliştirilen materyaller öğrenme – öğretme ortamlarını ve diğer öğrenme kaynaklarını desteklemektedir (Alkan ve ark., 1995; Kazu ve Yavuzalp, 2008). Kaya (2005)' e göre bilgisayar destekli öğretimi uygulamak bire-bir eğitim yazılımları, tekrar – alıştırmaya yazılımları, eğitsel oyun yazılımları, model oluşturma yazılımları, benzetim yazılımları ve problem çözme yazılımlarından faydalanılmaktadır. Öğrenme ortamlarında en çok kullanılan eğitim yazılımı türleri aşağıda verilmektedir.

Web tabanlı eğitim yazılımları: E-öğrenme ya da çevrimiçi eğitim olarak da adlandırılan web tabanlı eğitim sayesinde, öğrenciler bilişim teknolojileri araçlarıyla doğrudan ya da dolaylı olarak iletişim kurabilir, istenilen sayıda öğretmen ve öğrencinin katılabildiği eş zamanlı veya eş zamansız öğrenmenin gerçekleştiği, sosyal statülerin olmadığı ve işitsel ya da görsel olarak birçok avantaja sahip fırsatların sunulduğu ortamlardır (Engin ve ark., 2010).

Bire-bir eğitim yazılımları: İçeriğin gerçek bir öğretmen gibi öğrenciye aktarılmasında kullanılan bire-bir eğitim yazılımları, içeriğin aktarılmasının yanı sıra öğrencilere içerik ile ilgili alıştırmaya, dönüt ve değerlendirme imkânı sunmaktadır. Öğrencilerin kendi hızına göre çalışmalarına olanak tanır (Demir, 2009).

Tekrar – alıştırmaya yazılımları: Özel becerilerin tekrar edilmesini ve daha önce edinilen bilgilerin gözden geçirilmesini içeren bir yöntemdir. Öğrencileri çalıştıkları konu hakkında bireysel çalışma ve performansa bağlı geri dönütler almalarını sağlarlar (Kalaycı ve Bakır, 2018).

Eğitsel oyun yazılımları: Oyun oynamaya istekli olan öğrencilerin bu durumlarından faydalanılarak öğretilecek içeriklerin oyun oynama atmosferinde öğretilmesini ve problem çözme becerilerinin geliştirilmesini amaçlamaktadır (Seferoğlu, 2006).

Model oluşturma yazılımları: Öğrenme öğretme ortamlarında internet aracılığıyla öğretim yaparken, herhangi bir sistemin başka bir sistemi ya da süreci temsil etmede kullanılmasıdır (Kaya ve Önder, 2002).

Benzetim yazılımları: Günlük hayatta gerçekleştirilmesi güç olan olay ya da durumun kontrollü olarak bilgisayar ortamında canlandırılmasıdır. Öğrenciler için oldukça verimli öğrenme ortamları sunmaktadır. Günümüzde birçok benzetim yazılımı problem çözme ya da eğitsel oyun yazılımları ile beraber hazırlanmaktadır.

Problem çözme yazılımları: Öğrencilerin günlük hayatlarında karşılaştıkları problemleri çözmeleri için öğrencilerin problem çözme yeteneklerini geliştiren ve istenildiği kadar pratik yapma imkânı sunan yazılımlardır (Akkoyunlu, 2008).

Öğrenme öğretme ortamlarında etkili öğrenme sürecinin sağlanabilmesi için yararlanılan bilgisayar destekli eğitim parçası olarak bilinen eğitsel ara yüzler kullanılmaktadır. Eğitsel ara yüzler, öğrenme- öğretme ortamlarında kullanılan öğretim yazılımlarında öğrenme yaklaşımları doğrultusunda tasarlanmış ve öğrencilere yol gösteren parçalardır (Chan, 1995; Clark ve Mayer, 2003; Valetsianos ve Russell, 2014).

Eğitim yazılımlarının tasarlanmasında öğrenci ilgi ve ihtiyaçları, öğrenme ortamları ve öğretilecek hedef içeriklerin göz önünde bulundurulması gerekmektedir. Erensayın ve Güler (2017)'e göre eğitim yazılımlarının aşağıdaki özelliklere sahip olması gerekmektedir.

1. Eğitim yazılımları öğrenme –öğretme ilkelerine uygun olmalı, kullanımı kolay olmalı, yazılımda ipucu, düzeltme, dönüt ve pekiştirme yer almalıdır.
2. Eğitim yazılımı sunulan içerikte animasyon, ses ve resim gibi görsel-işitsel özellikleri birlikte barındırmalıdır.
3. Eğitim yazılımları öğretim programlarına ve öğrencilerin ilgi ve ihtiyaçlarına uygun olmalıdır.
4. Eğitim yazılımları kolay kullanım ara yüzüne sahip olmalı ve içerikler açık, anlaşılır olmalıdır.
5. Eğitim yazılımı kullanımı kolay, anlaşılabilir ara yüz tasarımı, kullanım kılavuzunun yanı sıra içeriğin aktarılmasında hedef, öğretme süreci, içerik ve değerlendirme gibi ölçütler göz önünde bulundurulmalıdır.
6. Eğitim yazılımları içeriğin aktarılmasında öğrenenlere ara ara sorular yönelterek gelen dönütler doğrultusunda öğrenmeyi yönlendirmelidir.

Öz-yeterlilik, kişinin belirli hedefleri veya görevleri yerine getirebilmek için yapması gereken davranışları yerine getirerek, başarılı olma ihtimali hakkında kendisiyle ilgili yargısıdır. Algılanan öz-yeterlilik, insanların yaşamlarını etkileyen olayları etkileme yeteneklerine olan inançlarıyla ilgilidir (Bandura, 1997). Eğitim araştırmaları öz-yeterliliği yüksek öğrencilerin daha başarılı, zorluklarla başa çıkma konusunda daha becerikli olduklarını ve başaramama duygusuna karşı daha az kaygılı olduklarını göstermektedir (Arseven, 2016). Bu çalışmalarda yola çıkarak öz-yeterlilik düzeyi yüksek olan öğrencilerin eğitsel yazılım geliştirme sürecinde daha başarılı olacağı düşünülebilir. Nitekim bireyin öz-yeterlilik algısının yüksek olması bir davranışı benimseme, davranışta bulunmaya başlama ve bunu sürdürmede önemli bir yol oynadığı bilinmektedir (Schwarzer ve Fuchs, 1996).

Literatür incelendiğinde eğitim yazılımı geliştirme üzerine yapılan aşağıdaki çalışmalara rastlanmaktadır:

Hinostroza ve ark. (2000) eğitim yazılımı geliştirme de profesyonel araç perspektifi sunmak amacıyla yapılan bir çalışmada eğitimsel yazılımın tasarımı, geliştirilmesi ve değerlendirilmesi ile ilgili eğitimsel yenilik sürecindeki önceki çalışmalar incelenmiştir. Yapılan incelemeler neticesinde kullanıcıların eğitsel yazılım içeriklerini

kullanmayı bilmelerinin, ihtiyaçları doğrultusunda eğitsel yazılım seçimlerinin ve eğitsel yazılım kullanım rollerinin eğitim yazılımı geliştirmede oldukça etkili olduğu belirtilmiştir.

Hinostroza ve Mellar (2001) tarafından bir eğitsel yazılım tasarım sürecine katılan iki öğretmenin eğitim pedagojilerinin incelendiği durum çalışmasından farklı bir eğitsel yazılım modeli sunmak amacıyla yapılan çalışmada, sunulan model ile öğretmenlerin pedagoji çalışmaları ile eğitimde bilgi teknolojileri çalışmaları arasında önemli bağlantılar olduğu ortaya konulmuştur.

Johnson ve Schleyer (2003) tarafından yüksek kaliteli eğitim yazılımı geliştirmek amacıyla yapılan çalışmada iyi bir eğitsel yazılım geliştirmenin analiz, tasarım, geliştirme, uygulama ve değerlendirme aşamalarından oluştuğunu bu aşamaların takım çalışması ile daha başarılı bir şekilde sürdürülebileceğini belirtilmiştir. Bunun yanı sıra dikkatle planlanmış bir eğitsel yazılım uygulaması geliştirme sürecinin başarılı bir şekilde sonuçlanmasının muhtemel olduğu ortaya konulmuştur.

Özkılıç ve İmer (2009) tarafından Mersin Üniversitesi Bilgisayar ve Öğretim Teknolojileri Eğitiminde öğrenim gören 296 öğrencinin katılımıyla yapılan bir çalışmada öğrencilerin eğitim yazılımı geliştirme algılarının belirlenmesi amaçlanmıştır. Yapılan çalışma sonucunda eğitim yazılımı geliştirme programlarının proje yönetimi, öğretim tasarımı, grafik tasarım ve eğitim boyutları, öğretmen adaylarının cinsiyete göre anlamlılık göstermediği gözlenirken, animasyon ve ses-video tasarımı boyutunda erkek öğretmen adayları lehine anlamlı bir fark bulunmuştur. Bilgisayarı olan öğretmen adaylarının tüm boyutlarda öz yeterlik algılarının daha yüksek olduğu görülmüştür.

Uzun ve ark. (2013) tarafından Bilgisayar ve Öğretim Teknolojileri Eğitimi öğrencilerinin eğitim yazılımı geliştirmelerine ilişkin öz yeterlik algılarının farklı değişkenler açısından incelemek amacıyla yapılan çalışmaya Uludağ üniversitesinden 60 BÖTE öğrencisi katılmıştır. Çalışma sonucunda öğrencilerin akademik başarıları ile eğitim yazılımı geliştirme öz yeterlik algıları arasında zayıf ilişki olduğu ve öz yeterlik algısının cinsiyete göre değişmediği ortaya çıkmıştır. Son sınıf öğrencileri lehine anlamlı bir farklılık olduğu belirlenmiştir. Ölçeğin alt boyutlarındaki öz yeterlik algıları ile derslerindeki başarıları arasında pozitif ilişki olduğu belirlenmiştir.

Eğitim yazılımı geliştirme konusunda kaliteli eğitsel yazılım geliştirmeye yönelik çalışmaların yanı sıra çoğunlukla Bilgisayar ve Öğretim Teknolojileri Eğitim bölümü öğrencilerinin öz yeterlik algılarının incelendiği anlaşılmaktadır. BÖTE bölümlerinde okuyan öğrenci sayısının her geçen yıl hızla azalma olduğu görülmektedir. Bu azalmayla birlikte eğitim yazılımı geliştirme noktasında oluşacak açığın giderilmesi ihtiyacı ortaya çıkacaktır. Bu durumda yazılım mühendisliği, bilgisayar mühendisliği ve bilgisayar programcılığı gibi yazılım geliştirme eğitimi alan öğrencilerin eğitim yazılımı geliştirmesi beklenebilir. Bu çalışma ile bu tür bölümlerde okuyan öğrencilerin eğitim yazılımı geliştirme öz yeterlik algılarının cinsiyet, üniversite, bölüm ve sınıf gibi değişkenler açısından incelenerek bu alanda literatüre katkıda bulunulacaktır.

Amaç

Bu araştırmanın amacı BÖTE dışında kalan ve ağırlıklı olarak bilgisayar programlama dersleri alan bölüm öğrencilerinin eğitsel yazılım geliştirme öz-yeterlilik algılarını ölçmektir. Bu amaç doğrultusunda aşağıdaki araştırma sorularına yanıt aranmıştır:

1. Bilgisayar Mühendisliği, Yazılım Mühendisliği ve Bilgisayar Programcılığı öğrencilerinin eğitsel yazılım geliştirme öz-yeterlilik algıları ne düzeydedir?
2. Cinsiyet değişkenine göre eğitsel yazılım öz-yeterlilik algısı arasında herhangi anlamlı bir farklılık var mıdır?
3. Öğrencilerin öğrenim aldıkları bölüme göre eğitsel yazılım öz-yeterlilik puanları arasında herhangi bir farklılık var mıdır?
4. Sınıf düzeyine göre eğitsel yazılım öz-yeterlilik puanları arasında herhangi bir farklılık var mıdır?
5. Daha önce herhangi bir eğitsel tasarım aşamasında görev alan ve almayan öğrencilere göre eğitsel yazılım öz-yeterlilik puanları arasında herhangi bir farklılık var mıdır?

YÖNTEM

Araştırmanın Deseni

Bu nicel araştırma ilişkisel tarama modeli yoluyla yürütülmüştür. İlişkisel araştırma modeli, iki ya da daha fazla değişkene ilişkin verilerin toplandığı ve değişkenler arasında bir ilişki olup olmadığının belirlendiği modeldir (Subaşı ve Demir, 2009).

Katılımcı Grubu

Araştırmanın örneklem grubunu üç farklı devlet üniversitesinde, farklı üç bölümde (Bilgisayar Programcılığı, Bilgisayar Mühendisliği, Yazılım Mühendisliği) öğrenim gören 372 öğrenci oluşturmaktadır. Araştırmanın örneklemine oluşturan öğrencilerin demografik özellikleri Tablo 1’de verilmiştir.

Tablo 1.

Demografik Bilgiler

Değişkenler	n	%	Değişkenler	n	%
<i>Cinsiyet</i>	120	32.3	<i>Daha önce eğitsel yazılım tasarımında görev alma</i>		
Kadın	252	67.7	Evet	104	28.0
Erkek	372	100.0	Hayır	268	72.0
Toplam	120	32.3	Toplam	372	100.0
<i>Sınıf</i>			<i>Bölüm</i>		
1. Sınıf	75	20.2	Yazılım Müh.	159	42.7
2. Sınıf	114	30.6	Bilgisayar Prog.	138	37.1
3. Sınıf	137	36.8	Bilgisayar Müh.	75	20.2
4. Sınıf	46	12.4	Toplam	372	100.0
Toplam	372	100.0			

Tablo 1’e göre örneklem grubunun %67,7’sini (n=252) erkek, %32,3’ünü (n=120) kadın öğrenciler oluşturmaktadır. Yapılan gözlemlerde bu üç bölümü genellikle erkek öğrencilerin tercih ettiği görülmüştür. Öğrencilerin daha önce eğitsel yazılım tasarım aşamasında görev alıp almama durumları incelendiğinde, öğrencilerin %28’i (n=104) daha önce eğitim yazılım tasarım aşamasında görev aldıklarını belirtirken, %72’si (n=268) daha önce herhangi bir eğitim yazılımının tasarım aşamasında görev almadıklarını ifade etmişlerdir. Bölüme göre dağılım incelendiğinde, örneklem grubunun %42,7’sini (n=159) Yazılım Mühendisliği, %37,1’ini (n=138) Bilgisayar Programcılığı ve %20,2’sini (n=75) Bilgisayar Mühendisliği bölümlerinde öğrenim gören öğrenciler oluşturmaktadır. Sınıf seviyelerine göre dağılım incelendiğinde, örneklem grubunun %20,2’sini (n=75) 1. sınıf, %30,6’sını (n=114) 2. sınıf, %36,8’sini (n=137) 3. sınıf ve %12,4’ünü (n=46) Bilgisayar bölümlerinde öğrenim gören 4. sınıf öğrencileri oluşturmaktadır.

İlgili bölüm müfredatları incelendiğinde her üç bölüme ait öğrencilerin aldıkları programlama derslerinde birbirlerine paralellik gösteren programlama dillerini kullandıkları (C, Java, Python, C++ vb.) görülmüştür. Derslere ait ders bilgi paketlerinde ise bu programlama derslerinde eğitsel yazılım geliştirmeyle alakalı herhangi bir konuya yer verilmediği belirlenmiştir.

Veri Toplama Aracı

Çalışmada “Eğitim Yazılımı Geliştirme Öz-Yeterlilik Algısı Ölçeği” kullanılmıştır. Ölçek öğrencilerin sorulara göre “Çok Güvenirim” (100), “Hiç Güvenmem” (0) veya “Fikrim Yok” arasında değişen 100’lü derecelendirme ölçeğine göre puanlama yaptıkları 22 maddeden meydana gelmektedir. Aşkar ve Dönmez (2004) tarafından geliştirilen bu ölçek, proje yönetimi ve öğretim tasarımı, animasyon ve ses-video tasarımı, grafik tasarımı ve programlama olmak üzere dört faktöre ayrılmıştır. Faktör 1 10 maddeden, Faktör 2 altı maddeden, Faktör 3 ve Faktör 4 ise üçer maddeden meydana gelmektedir. Her faktörden alınabilecek ve en üst düzeyde öz-yeterlilik algısını gösteren en yüksek ortalama puan 100, en alt düzeyde öz-yeterlilik algısını gösteren en düşük ortalama puan ise 0’dır. Orijinal ölçeğin iç tutarlılık katsayıları Faktör 1 için .92, Faktör 2 için .90, Faktör 3 için .80, Faktör 4 için .85 ve ölçeğin tüm faktörleri için .92 olarak hesaplanmıştır. Aynı ölçeği kullanan başka bir çalışmada iç tutarlılık katsayıları Faktör 1 için .91, Faktör 2 için .70, Faktör 3 için .84, Faktör 4 için .88 ve ölçeğin tüm faktörleri için .92 olarak bulunmuştur (Tat ve Şahinkayası, 2019). Bu çalışmaya ait ölçeğin iç tutarlılık katsayıları (Cronbach’s alfa) ise Tablo 2’de verilmiştir.

Tablo 2.*Ölçeğin Faktörlerine Ait Güvenirlilik Katsayıları*

Faktörler	n
Faktör 1 - Proje Yönetimi ve Öğretim Tasarımı	.957
Faktör 2 - Animasyon ve Ses-Video Tasarımı	.873
Faktör 3 - Grafik Tasarımı	.801
Faktör 4 - Programlama	.833
Tüm Ölçek	.954

Yıldız ve Uzunsakal (2018), 0.80 ile 1.00 arasındaki Cronbach alfa değerinin yüksek güvenilirlikte olduğunu belirtmişlerdir. Buna göre ölçeğin tüm faktörlerinin iç tutarlılık katsayılarının yüksek güvenilirlikte olduğu söylenebilir.

Verilerin Toplanması ve Analizi

Araştırmamızda farklı üç devlet üniversitesinin Bilgisayar Mühendisliği, Yazılım Mühendisliği ve Bilgisayar Programcılığı öğrencilerinin eğitsel yazılım geliştirme öz-yeterlilik algılarını ortaya koymak amacıyla, yüz yüze anket yöntemi kullanılarak verilere birincil kaynaktan ulaşılması hedeflenmiştir. Verilerin analizinde istatistiksel betimlemeler, Levene testi, t-testi, ANOVA testi, Scheffe testi ve Pearson Korelasyon kullanılmıştır.

BULGULAR ve YORUM

Bu bölümde araştırma bulguları ile ilgili bilgilere yer verilmiştir.

Birinci Araştırma Sorusuna İlişkin Bulgular

Örneklem grubunun eğitsel yazılım öz-yeterlilik algılarını ölçmek ve hangi faktörlerde kendilerini ne kadar yeterli hissettiklerini belirlemek amacıyla faktörlere ait istatistiksel analizler yapılarak sonuçlar Tablo 3'te verilmiştir.

Tablo 3.*Ölçek Faktörlerinin Ortalama Puanları ve Standart Sapmaları*

Faktörler	n	\bar{X}	Ss
Faktör 1 - Proje Yönetimi ve Öğretim Tasarımı	372	57,15	23,894
Faktör 2 - Animasyon ve Ses-Video Tasarımı	372	45,21	26,052
Faktör 3 - Grafik Tasarımı	372	57,14	25,120
Faktör 4 - Programlama	372	51,20	27,056
Tüm Ölçek	372	53,07	21,056

Tablo 3'e göre öğrencilerin bu faktör dağılımları içerisinde eğitsel yazılım olarak kendilerini en çok yeterli gördükleri faktörler grafik tasarımı ile proje yönetimi ve öğretim tasarımı olmuştur. Bu sonuca göre öğrenciler; ekran ara yüzünü tasarlayabilme, yönlendirebilme, yan menü oluşturma, proje planlama, tasarlama ve geliştirme, projenin amaçlarını belirleme, bu amaçlar doğrultusunda hareket etme, materyalleri toplama, test etme ve uygulama konusunda kendilerine en yüksek puanları vererek bu konularda kendilerini yeterli gördüklerini belirtmiştir. Bu faktörler içerisinde öğrencilerin kendilerini en yetersiz hissettikleri faktör ise animasyon tasarımı, ses ve video birimidir. Buna göre öğrenciler bir animasyon programı kullanarak bir animasyonu canlandırabilme, ses ekleyip çıkartabilme, kurgulama, efekt ekleme ve çeşitli video biçimlerine dönüştürme konusunda kendilerini daha az yeterli gördükleri belirlenmiştir. Bunun nedeni araştırmacılara göre öğrencilerin üniversite ders müfredatlarında aldıkları çok boyutlu programlama derslerinin yetersiz olmasından kaynaklandığı düşünülmektedir. Ölçeğin tamamına bakıldığında örneklem grubunda yer alan öğrencilerin genel olarak eğitsel yazılım geliştirme öz-yeterlilik algılarının orta düzeyde olduğu ve bu konuda kendilerini geliştirmelerine imkân veren çalışmalar yapmaya ihtiyaç duydukları söylenebilir.

İkinci Araştırma Sorusuna İlişkin Bulgular

Araştırmaya katılan öğrencilerin cinsiyete göre eğitsel yazılım öz-yeterlilik puanları arasında herhangi bir farklılık olup olmadığına bakmak için yapılan t-testi sonuçları ve grup ortalamaları farkına göre Cohen'in d formülüyle hesaplanan etki büyüklüğü Tablo 4'te verilmiştir.

Tablo 4.

Eğitsel Yazılım Öz-yeterlilik Algısının Cinsiyete Göre Yapılan t-testi ve Etki Büyüklüğü Sonuçları

Faktörler	Cinsiyet	n	\bar{X}	Ss	sd	t	p	d
Faktör 1	Kadın	120	51,25	26,130	370	-3,316	.001*	0,36
	Erkek	252	59,92	22,267				
Faktör 2	Kadın	120	39,99	24,251	370	-2,687	.008*	0,30
	Erkek	252	47,69	26,555				
Faktör 3	Kadın	120	53,08	25,278	370	-2,158	.032**	0,24
	Erkek	252	59,07	24,863				
Faktör 4	Kadın	120	43,68	26,531	370	-3,766	.000*	0,42
	Erkek	252	54,78	26,616				
Tüm Ölçek	Kadın	120	47,40	26,130	370	-3,549	.000*	0,34
	Erkek	252	55,77	22,267				

* p< .01; ** p< .05

Tablo 4'e göre faktör 1 (proje yönetimi/öğretim tasarımı) için erkek öğrencilerin ortalama puanlarının ($\bar{x}=59,92$), kadın öğrencilerin ortalama puanlarından ($\bar{x}=51,25$) daha yüksek olduğu görülmektedir. Yapılan t-testi sonucuna göre aradaki farkın anlamlı olduğu görülmüştür (p< .01).

Faktör 2 (animasyon ve ses-video) için verilen puanların ortalaması erkek öğrenciler için $\bar{x}=47,69$ ve kadın öğrenciler için $\bar{x}=45,32$ 'dir. Faktör 2 için erkek öğrencilerin aldıkları ortalama puan kız öğrencilerin aldıkları ortalama puandan daha yüksektir. Bu farkın anlamlı olup olmadığına bakılmak için yapılan t-testi sonucunda farkın anlamlı olduğu görülmüştür (p<.01).

Üçüncü faktör olan grafik tasarımı için puanların ortalamasına bakıldığında erkek öğrencilerin puanlarının ($\bar{x}=59,07$), kadın öğrencilerin puanlarından ($\bar{x}=53,08$) daha yüksek olduğu görülmektedir. Yapılan t-testi sonucuna göre aradaki farkın anlamlı olduğu görülmüştür (p<.05).

Dördüncü faktör (programlama) için alınan puanların ortalaması erkek öğrenciler için $\bar{x}=54,78$ ve kadın öğrenciler için $\bar{x}=43,68$ 'dir. Faktör 4 için erkek öğrencilerin aldıkları ortalama puan kız öğrencilerin aldıkları ortalama puandan daha yüksektir. Bu farkın anlamlı olup olmadığına bakılmak için yapılan t-testi sonucunda farkın anlamlı olduğu görülmüştür (p<.01).

Görüldüğü gibi anketin tüm faktörlerinde ve ölçeğin genelinde erkek öğrencilerin aldıkları ortalama puanlar, kadın öğrencilerin aldıkları ortalama puanlardan daha yüksektir. Bu sonuçlara göre, eğitsel bir yazılım tasarımı ve programlama konusunda, erkek öğrencilerin kadın öğrencilere göre kendilerini daha fazla yeterli gördükleri ve eğitsel yazılım geliştirme öz-yeterlilik algılarının daha yüksek olduğu söylenebilir. Bunun nedeninin erkeklerin teknoloji, bilgisayar oyunları ve mühendislik alanına daha meraklı olmasından kaynaklandığı düşünülmektedir.

Cohen (1988), etki büyüklüğünün 0.20 den küçük olması durumunda düşük, 0.50 olması durumunda orta, 0.80'den büyük olması durumunda yüksek etki büyüklüğü olduğunu söylemiştir. Buna göre programlama faktöründe etki büyüklüğünün orta, ölçeğin diğer boyutlarında ve ölçeğin tamamında etki büyüklüğünün düşüğe yakın olduğu söylenebilir.

Üçüncü Araştırma Sorusuna İlişkin Bulgular

Araştırmaya katılan öğrencilerin öğrenim aldıkları bölümlere göre eğitsel yazılım öz-yeterlilik algılarına dair istatistikler Tablo 5'te gösterilmiştir.

Tablo 5.

Bölmelere Göre Eğitsel Yazılım Öz-yeterlilik Algısına İlişkin Bilgiler

Faktörler	Bölüm	n	\bar{X}	Ss
Faktör 1	Bilgisayar Programcılığı	138	45,50	23,565
	Bilgisayar Mühendisliği	75	61,97	22,497
	Yazılım Mühendisliği	159	64,93	20,776
Faktör 2	Bilgisayar Programcılığı	138	42,63	25,431
	Bilgisayar Mühendisliği	75	39,66	26,138
	Yazılım Mühendisliği	159	50,07	25,860
Faktör 3	Bilgisayar Programcılığı	138	48,80	24,804
	Bilgisayar Mühendisliği	75	57,93	24,950
	Yazılım Mühendisliği	159	64,00	23,415
Faktör 4	Bilgisayar Programcılığı	138	38,31	23,909
	Bilgisayar Mühendisliği	75	54,53	26,851
	Yazılım Mühendisliği	159	60,81	26,341
Tüm Ölçek	Bilgisayar Programcılığı	138	44,19	21,270
	Bilgisayar Mühendisliği	75	54,32	21,526
	Yazılım Mühendisliği	159	60,19	19,097

Tablo 5'te görüldüğü gibi hem ölçek faktörlerinde hem de tüm ölçekte alınan ortalama puanlar arasında bölümlere göre farklılıklar vardır. Bu farkın anlamlı olup olmadığına bakmak için yapılan ANOVA testinin sonuçları Tablo 6'da verilmiştir.

Tablo 6.

Eğitsel Yazılım Öz-Yeterlilik Algısının Bölüme Göre Yapılan ANOVA Testi ve Etki Büyüklüğü Sonuçları

Faktörler	Varyansın Kaynağı	Karelerin Toplamı	sd	Karelerin Ortalaması	F	P	Fark	Etki Büyüklüğü
Faktör 1	Gruplar arası	30,087	2	15,044	30,546	.000*	Yazılım ve Bilgisayar Müh. > Bilgisayar Prog.	0,14
	Gruplar içi	181,730	369	,492				
	Toplam	211,817	371					
Faktör 2	Gruplar arası	6,983	2	3,491	5,262	.006*	Yazılım ve Bilgisayar Müh. > Bilgisayar Prog.	0,03
	Gruplar içi	244,817	369	,663				
	Toplam	251,800	371					
Faktör 3	Gruplar arası	17,124	2	8,562	14,561	.000*	Yazılım ve Bilgisayar Müh. > Bilgisayar Prog.	0,07
	Gruplar içi	216,985	369	,588				
	Toplam	234,109	371					
Faktör 4	Gruplar arası	38,440	2	19,220	30,422	.000*	Yazılım ve Bilgisayar Müh. > Bilgisayar Prog.	0,14
	Gruplar içi	233,132	369	,631				
	Toplam	271,572	371					
Tüm Ölçek	Gruplar arası	19,062	2	9,531	22,855	.000*	Yazılım ve Bilgisayar Müh. > Bilgisayar Prog.	0,11
	Gruplar içi	153,885	369	,417				
	Toplam	172,947	371					

* p< .01

Tablo 6'da bölümlere göre eğitsel yazılım öz-yeterlilik algısı ortalama puanları arasındaki farkın hem ölçeğin tüm boyutlarında hem de ölçeğin tamamında anlamlı olduğu görülmektedir (p<.01). Öğrenim gördükleri bölümlere göre aradaki farkın hangi bölümler arasında anlamlı olduğunu belirlemek için yapılan Scheffe analizinin sonuçlarına göre, Yazılım Mühendisliği ve Bilgisayar Mühendisliği bölümünde öğrenim gören öğrencilerin eğitsel yazılım geliştirme algılarının Bilgisayar Programcılığı bölümünde eğitim gören öğrencilerden daha yüksek olduğu görülmüştür. Bu farkın mühendislik öğrencilerinin eğitim sürelerinin ön lisans programına göre daha uzun olması ve donanımdan ziyade yazılım geliştirme amaçlı ders sayılarının bilgisayar programcılığı bölümüne göre daha fazla olmasından kaynaklandığı söylenebilir.

Cohen (1988), etki büyüklüğünün 0.10'dan az olması durumunda küçük, 0.25 olması durumunda orta, 0.40'dan büyük olması durumunda geniş etki büyüklüğünü olduğunu söylemiştir. Buna göre ölçeğin tüm faktörleri ve tamamında etki büyüklüğünün küçük olduğu söylenebilir.

Dördüncü Araştırma Sorusuna İlişkin Bulgular

Araştırmaya katılan öğrencilerin sınıf düzeyine göre eğitsel yazılım öz-yeterlilik algılarına dair istatistikler Tablo 7’te sunulmuştur.

Tablo 7.

Sınıf Düzeyine Göre Eğitsel Yazılım Öz-Yeterlilik Algısına İlişkin Bilgiler

Faktörler	Sınıf Düzeyi	n	\bar{X}	Ss
Faktör 1	1. Sınıf	75	44,26	26,222
	2. Sınıf	114	54,90	23,420
	3. Sınıf	137	64,40	23,299
	4. Sınıf	46	62,03	24,070
Faktör 2	1. Sınıf	75	31,27	22,801
	2. Sınıf	114	44,46	24,923
	3. Sınıf	137	64,36	22,126
	4. Sınıf	46	61,20	27,098
Faktör 3	1. Sınıf	75	35,66	26,311
	2. Sınıf	114	44,57	24,883
	3. Sınıf	137	53,13	23,985
	4. Sınıf	46	38,79	27,890
Faktör 4	1. Sınıf	75	38,92	22,397
	2. Sınıf	114	53,80	22,565
	3. Sınıf	137	68,92	18,334
	4. Sınıf	46	59,98	23,992
Tüm Ölçek	1. Sınıf	75	37,72	21,508
	2. Sınıf	114	50,15	19,706
	3. Sınıf	137	63,37	17,204
	4. Sınıf	46	54,64	21,744

Tablo 7’de görüldüğü gibi hem ölçek faktörlerinde hem de tüm ölçekte alınan ortalama puanlar arasında sınıf düzeyine göre farklılıklar vardır. Bu farkın anlamlı olup olmadığına bakmak için yapılan ANOVA testinin ve varyansa göre hesaplanan etki büyüklüğünün sonuçları Tablo 8’de verilmiştir .

Tablo 8.

Eğitsel Yazılım Öz-Yeterlilik Algısının Bölüme Göre Yapılan ANOVA Testi ve Etki Büyüklüğü Sonuçları

Faktörler	Varyansın Kaynağı	Karelerin Toplamı	sd	Karelerin Ortalaması	F	p	Fark	Etki Büyüklüğü
Faktör 1	Gruplar arası	45,547	3	15,182	33,602	.000*	2. ve 3. Sınıf > 1. ve 4. sınıf	0,22
	Gruplar içi	166,271	368	,452				
	Toplam	211,818	371					
Faktör 2	Gruplar arası	17,365	3	5,788	9,086	.000*	2. ve 3. Sınıf > 1. ve 4. sınıf	0,07
	Gruplar içi	234,434	368	,637				
	Toplam	251,799	371					
Faktör 3	Gruplar arası	21,346	3	7,115	12,307	.000*	2. ve 3. Sınıf > 1. ve 4. sınıf	0,09
	Gruplar içi	212,764	368	,578				
	Toplam	234,110	371					
Faktör 4	Gruplar arası	63,286	3	21,095	37,271	.000*	2. ve 3. Sınıf > 1. ve 4. sınıf	0,23
	Gruplar içi	208,287	368	,566				
	Toplam	271,573	371					
Tüm Ölçek	Gruplar arası	33,306	3	11,102	29,257	.000*	2. ve 3. Sınıf > 1. ve 4. sınıf	0,19
	Gruplar içi	139,641	368	,379				
	Toplam	172,947	371					

* p< .01

Tablo 8’de bölümlere göre eğitsel yazılım öz-yeterlilik algısı ortalama puanları arasındaki farkın hem ölçeğin tüm boyutlarında hem de ölçeğin tamamında anlamlı olduğu görülmektedir (p<.01). Sınıf düzeyine göre aradaki farkın hangi sınıf düzeyinde arasında anlamlı olduğunu belirlemek için yapılan Scheffe analizinin sonuçlarına göre, 2. sınıf ve 3. sınıf öğrencilerinin eğitsel yazılım geliştirme algılarının 1. sınıf ve 4. sınıf öğrencilerine göre

daha yüksek olduğu görülmüştür. Bunun nedeni programlama ile ilgili mesleki derslerin çoğunlukla 2. ve 3. sınıflarda verilmesi nedeniyle bu eğitim düzeyindeki öğrencilerin yazılım konusundaki bilgilerinin tazeliğini koruması olabilir. Eğitsel yazılım geliştirme öz-yeterlilik algıları en zayıf olanlar ise 1. sınıf öğrencileridir. Bunun nedeni ise üniversitede ders müfredatında 1. Sınıf düzeyinde yeterince mesleki dersin okutulmaması olabilir.

Beşinci Araştırma Sorusuna İlişkin Bulgular

Araştırmaya katılan öğrencilerin daha önce herhangi bir eğitsel yazılım tasarım aşamasında görev alma durumuna göre eğitsel yazılım öz-yeterlilik puanları arasında herhangi bir farklılık olup olmadığına bakmak için yapılan t-testi ve grup ortalamaları farkına göre Cohen'in d formülüyle hesaplanan etki büyüklüğü sonuçları Tablo 9'da verilmiştir.

Tablo 9

Eğitsel Yazılım Öz-Yeterlilik Algısının Daha Önce Eğitsel Yazılım Tasarım Aşamasında Görev Alma Durumuna Göre Yapılan t-testi ve Etki Büyüklüğü Sonuçları

Faktörler	Grup	n	\bar{X}	Ss	sd	t	p	d
Faktör 1	Hayır	268	51,63	23,482	370	-7,661	.001*	0.93
	Evet	104	71,30	18,590				
Faktör 2	Hayır	268	41,83	25,708	370	-4,106	.008*	0.48
	Evet	104	53,93	24,999				
Faktör 3	Hayır	268	53,17	24,680	370	-5,042	.032**	0.59
	Evet	104	67,35	23,406				
Faktör 4	Hayır	268	45,90	25,762	370	-6,381	.000*	0.74
	Evet	104	64,85	25,589				
Tüm Ölçek	Hayır	268	48,38	20,959	370	-7,160	.000*	0.85
	Evet	104	65,14	18,332				

* p < .01

Tablo 9'da görüldüğü gibi anketin tüm faktörlerinde ve ölçeğin genelinde daha önceden eğitsel yazılım tasarım aşamasında görev alan öğrencilerin aldıkları ortalama puanlar, görev almayan öğrencilerin aldıkları ortalama puanlardan daha yüksektir. Uygulanan t-testi sonuçlarına göre farkın anlamlı olduğu görülmüştür (p<.01). Bunun nedeni eğitsel yazılım geliştirme aşamasında görev alan öğrencilerin daha öncesinde sürecin nasıl olduğuna dair bilgi sahibi olmaları, kendilerine güvenmeleri ve eğitsel yazılım geliştirmelerinden kaynaklanabilir. Bilgisayar programlamaya yönelik bölümlerde ders müfredatına bu konuya yönelik derslerin eklenmesi öğrencilerin eğitsel yazılım geliştirme konusunda kendilerine daha fazla güvenmelerine yardımcı olabilir.

Ayrıca yapılan etki büyüklüğü hesaplaması sonucunda animasyon ve ses-video tasarımı ile grafik tasarımı faktörlerinde orta düzeyde, programlama faktöründe yüksek düzeye yakın ve proje yönetimi ve öğretim tasarımı ile ölçeğin tamamında yüksek düzeyde etki büyüklüğünün olduğu görülmüştür (d>0.80).

SONUÇ, TARTIŞMA ve ÖNERİLER

Bu araştırma çeşitli programlama dilleri üzerine eğitim alan öğrencilerin, eğitsel yazılım geliştirme öz-yeterlilik algılarını çeşitli değişkenler açısından incelemek amacıyla yapılmıştır. Öğrencilerin eğitsel yazılım geliştirme aşamalarında kendilerini en yeterli gördüklerini düşündükleri faktörün proje yönetimi ve öğretim tasarımı aşamasının olduğu görülmektedir. Aşkar ve Dönmez (2004) eğitsel yazılım geliştirme projelerinde görev alan proje yöneticilerinin öğretim tasarımı konusunda bilgi ve beceri sahibi olmalarının oldukça önemli olduğunu belirtmişlerdir. Bu açıdan bakıldığında araştırmaya katılan öğrencilerin öğretim tasarımı ve proje yönetimi konusunda kendilerini yeterli gördüklerini söyleyebiliriz. Buna ilaveten öğrencilerin ekran ara yüzünü tasarlayabilme, yönlendirebilme, yan menü oluşturma ve çizibilme konusunda kendilerini yeterli gördükleri belirlenmiştir. Ancak öğrencilerin eğitsel yazılım tasarlamada animasyon tasarımı, ses ve video birimi aşamasında kendilerini yetersiz gördükleri belirlenmiştir. Bu durum öğrencilerin üniversite ders müfredatlarında aldıkları çok boyutlu programlama derslerinin yetersiz olmasından kaynaklandığı düşünülebilir.

Araştırmaya katılan öğrencilerin eğitsel yazılım geliştirme öz-yeterlilik algıları cinsiyetlerine göre incelendiğinde, Bilgisayar Programcılığı, Bilgisayar Mühendisliği ve Yazılım Mühendisliği bölümlerindeki erkek öğrencilerin kız öğrencilere göre eğitsel yazılım geliştirme öz-yeterlilik algılarının ölçeğin tüm faktörlerinde daha yüksek olduğu belirlenmiştir. Özkılıç ve İmer (2009) eğitim yazılım geliştirme programlarının proje yönetimi, öğretim tasarımı, grafik tasarım ve eğitim boyutları, öğretmen adaylarının cinsiyete göre anlamlılık göstermediği gözlenirken,

animasyon ve ses-video tasarımı boyutunda erkek öğretmen adayları lehine anlamlı bir fark bulunmuştur. Bunun yanı sıra Uzun ve ark. (2013) tarafından yapılan çalışmada da BÖTE öğrenim gören öğrencilerin eğitsel yazılım geliştirme öz-yeterlik algılarının cinsiyetlerine göre farklılaşmadığı belirlenmiştir. Araştırma bulgularının bu sonuçlar ile çelişmesi ilgili yazarlar tarafından yapılan çalışmanın BÖTE öğrencilerini kapsamından kaynaklandığı düşünülebilir. Özkılıç ve İmer (2009) tarafından belirlenen animasyon ve ses-video tasarımı boyutunda erkek öğretmen adayları lehine anlamlı bir farklılık araştırma bulgularını desteklemektedir. Araştırma bulgularına göre erkek öğrenciler lehine anlamlı farklılığın sebebi olarak, erkeklerin teknoloji, bilgisayar oyunları ve mühendislik alanına daha meraklı olmasından kaynaklandığı düşünülebilir.

Araştırmaya katılan öğrencilerin eğitsel yazılım geliştirme öz-yeterlik algıları bölümlerine göre incelendiğinde, Yazılım Mühendisliğinde öğrenim gören öğrencilerin diğer bölümlerde öğrenim gören öğrencilere göre bilgisayar yazılımı konusunda kendilerini daha yeterli gördükleri belirlenmiştir. Bunun nedeni eğitim müfredatlarında donanımdan ziyade yazılım geliştirme amaçlı ders sayılarının diğer bölümlere göre daha fazla olması söylenebilir. Yine araştırmaya göre, bilgisayar yazılımı konusunda kendilerini en yetersiz gören öğrenciler ise Bilgisayar Programcılığı öğrencileri olduğu görülmektedir. Bu durumun sebebi olarak ise Bilgisayar Programcılığının bir ön lisans programı olması ve buna bağlı olarak 2 yıllık süre içerisinde aldıkları ders kredilerinin lisans öğrencilerine göre daha az olmasından kaynaklandığı söylenebilir.

Araştırmaya katılan öğrencilerin eğitsel yazılım geliştirme öz-yeterlik algıları sınıf seviyelerine göre incelendiğinde, 3. sınıf öğrencilerinin diğer sınıflarda öğrenim gören öğrencilere oranla kendilerine yazılım konusunda daha fazla güvendikleri belirlenmiştir. Özkılıç ve İmer (2009) tarafından yapılan çalışmada ise 4.sınıf öğrencileri lehine anlamlı farklılık olduğu belirlenmiştir. Araştırma bulgularının diğer araştırma bulguları ile çelişmesi diğer araştırmaların BÖTE öğrencileri ile yapılmış olmasından kaynaklanabilir. Çünkü BÖTE öğrencileri eğitsel yazılım geliştirme üzerine eğitim almaktadır. Araştırma bulgularına göre 3. Sınıf öğrencileri lehine anlamlı farklılığın çıkma sebebi olarak BÖTE dışındaki diğer yazılım geliştirme ile ilgili bölümlerde programlama ile ilgili mesleki derslerin çoğunlukla 2. ve 3. sınıflarda verilmesi nedeniyle 3. sınıf öğrencilerinin bu konudaki bilgilerinin tazeliğini koruması olduğu düşünülebilir. Kendilerini bu konuda daha az yeterli gördüklerini belirten öğrenciler ise 1. sınıf öğrencileridir. Bunun nedeni ise üniversitede 1. sınıflarda mesleki derslerin az olması söylenebilir.

Araştırmaya katılan öğrencilerin eğitsel yazılım geliştirme öz-yeterlik algıları herhangi bir eğitsel yazılım geliştirme tasarım aşamasında görev alma durumlarına göre incelendiğinde, daha önce eğitsel yazılım tasarım aşamasında görev alan öğrencilerin görev almayan öğrencilere göre yazılım konusunda kendilerini çok daha fazla yeterli gördükleri belirlenmiştir. Bandura (1997), öz-yeterlilik inançlarını besleyen birincil etmenin aktif deneyimler olduğunu söylemektedir. Buna göre bu öğrencilerin daha öncesinde sürecin nasıl olduğuna dair bilgi sahibi olmaları, deneyim yaşamaları ve eğitsel yazılım geliştirmelerinin kendilerini daha iyi hissetmelerine sebep olduğunu söylenebilir.

Araştırma Türkiye’de Doğu Anadolu Bölgesinde bulunan üç devlet üniversitesinde programlama üzerinde eğitim alan öğrenciler ile sınırlıdır. İleride Türkiye’nin farklı bölgelerindeki üniversitelerde eğitim alan daha geniş katılımcı gruplarıyla çalışma yapılması bu alanda araştırma yapmak isteyen araştırmacılara önerilebilir.

Öneriler

Araştırma bulgularına bağlı olarak aşağıdaki önerilerde bulunulabilir;

1. Programlama ağırlıklı eğitim veren bölümlerin müfredatlarında çok boyutlu programlama derslerine daha fazla yer verilebilir.
2. Programlama ağırlıklı eğitim veren bölümlerde kız öğrencilerin yazılım geliştirme sürecinin tüm aşamalarına etkin bir şekilde katılmaları için tedbirler alınabilir.
3. Programlama ağırlıklı eğitim veren bölümlerde tüm sınıf seviyelerinin ders müfredatlarında mesleki derslere ağırlıklı olarak yer vermelidirler.
4. Programlama ağırlıklı eğitim veren bölümlerde öğrencilerin yazılım geliştirme süreçlerinde aktif bir şekilde yer almaları sağlanmalıdır.
5. Programlama ağırlıklı eğitim veren bölümlerin müfredatlarında eğitsel yazılım geliştirmenin tüm aşamaları hakkında bilgilere yer verilmelidir.

KAYNAKLAR

- Akkoyunlu, B. (2008). *Öğretim Yazılımları*. B. Özer (Ed.), *Çağdaş Eğitimde Yeni Teknolojiler* (s. 49-57) içinde. Eskişehir: Anadolu Üniversitesi Yayınları.
- Alkan, C. (1998). *Eğitim Teknolojisi*. Ankara: Anı Yayıncılık.
- Alkan, C., Deryakulu, D. & Şimşek, N. (1995). *Eğitim Teknolojisine Giriş*. Ankara: Öner Matbaacılık.
- Altun, A. (2010). Kanada'daki medya okuryazarlığı eğitimi üzerine bir değerlendirme. *Abant İzzet Baysal Üniversitesi Eğitim Fakültesi Dergisi*, 10(2), 41-57.
- Arseven, A. (1986). *Bilgisayar Destekli Öğretim*. TED Birinci Bilgisayar Eğitimi Toplantısı, Ankara, 63-69.
- Aşkar, P., & Dönmez, O. (2004). Eğitim yazılımı geliştirme öz-yeterlik algısı ölçeği. *Eğitim Bilimleri ve Uygulama*, 3(6), 259-274.
- Bandura, A. (1997). *Self-Efficacy (The Exercise of Control)*, New York: W. H. Freeman and Company.
- Chan, T. W. (1995). Artificial agents in distance learning. *International Journal of Educational Telecommunications*, 1(2/3), 263-282.
- Clark, R., & Mayer, R. E. (2003). *E-learning and the Science of Instruction*. New Jersey: John Wiley & Sons, Inc., Hoboken.
- Cohen J. (1988). *The Analysis of Variance*. In *Statistical Power Analysis for the Behavioral Sciences* (second ed.). Lawrence Erlbaum Associates.
- Çepni, S. (2005). *Bilim, Fen, Teknoloji ve Eğitim Programlarına Yansımaları*. S. Çepni (Ed.), *Kuramdan Uygulamaya Fen ve Teknoloji Öğretimi* (s. 2-12) içinde. Ankara: Pegem A Yayıncılık.
- Demir, Ü. (2009). *Üniversite öğrencilerinin eğitsel yazılım ekran tasarımı seçiminde kişiliğin etkisi* (Doktora Tezi). Yükseköğretim Kurulu Tez Merkezi veri tabanından erişildi (Erişim No. 239374).
- Engin, A. O., Tösten, R., & Kaya, M. D. (2010). Bilgisayar destekli eğitim. *Kafkas Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 5(1), 69-80.
- Erensayın, E., & Güler, Ç. (2017). EBA platformundaki ders materyallerinin eğitsel yazılım değerlendirme ölçütlerine göre değerlendirilmesi. *Ahi Evran Üniversitesi Kırşehir Eğitim Fakültesi Dergisi*, 18(1), 657-678.
- Erişen, Y., & Çeliköz, N. (2010). *Eğitimde Bilgisayar Kullanımı*. Ö. Demirel & E. Altun (Ed.), *Öğretim Teknolojileri ve Materyal Tasarımı* (s. 113-146) içinde. Ankara: Pegem A Yayıncılık.
- Güven, S. (2006). Öğretim teknolojileri ve materyal geliştirme dersinin kazandırdığı yeterlikler yönünden değerlendirilmesi: İnönü Üniversitesi eğitim fakültesi örneği. *Türk Eğitim Bilimleri Dergisi*, 4(2), 165-179.
- Hannafin, M.S., & Peck, K.L. (1988). *The Design, Development and Evaluation of Instructional Software*. London: Collier Macmillan .
- Hinostroza, J. E., & Mellar, H. (2001). Pedagogy embedded in educational software design: report of a case study. *Computers & Education*, 37(1), 27-40.
- Hinostroza, J. E., Rehbein, L. E., Mellar, H., & Preston, C. (2000). Developing educational software: a professional tool perspective. *Education and Information Technologies*, 5(2), 103-117.
- Hwang, G. J., Tsai, C. C., & Yang, S. J. H. (2008). Criteria, strategies and research issues of context aware ubiquitous learning. *Educational Technology & Society*, 11(2), 81-91.
- Johnson, L. A., & Schleyer, T. K. (2003). Developing high-quality educational software. *Journal of Dental Education*, 67(11), 1209-1220.
- Kalaycı, S., & Bakır, E. (2018). 6. sınıf öğrencilerinin hücre konusundaki kavramları anlama düzeylerine alıştırmaya ve tekrar yazılımında hazırlanan materyalin etkisi *İnönü Üniversitesi Eğitim Fakültesi Dergisi*, 19(2), 67-81.
- Kaya, Z. (2005). *Öğretim Teknolojileri ve Materyal Geliştirme*. Ankara: Pegem A Yayıncılık.
- Kaya, Z., & Önder, H. H. (2002). İnternet yoluyla öğretimde ergonomi. *The Turkish Online Journal of Educational Technology*, 1(1), 48-54.
- Kazu, G. Y., & Yavuzalp, N. (2008). Öğretim yazılımlarının kullanımına ilişkin öğretmen görüşleri. *Eğitim ve Bilim Dergisi*, 33(150), 110-126.

- Keser, H. (1991). Eğitimde nitelik geliştirmede bilgisayar destekli eğitim ve ders yazılımlarının rolü. *Eğitimde Arayışlar*, 1, 13-14.
- Keskin, N. Ö. (2010, Şubat). *Mobil Öğrenme Teknolojileri ve Araçları*. Akademik Bilişim Konferansında sunulan bildiri, Muğla Üniversitesi, Muğla. Erişim adresi: https://ab.org.tr/ab10/kitap/keskin_AB10.pdf
- Külahçı, G., & Gürol, M. (1991). Eğitim aracı olarak bilgisayar ilişkin öğretmen görüş ve tutumları. *Eğitim ve Bilim*, 60, 10-11.
- Mutiara, D., Zuhairi, A., & Kurniati, S. (2007). Designing, developing, producing and assuring the quality of multi-media learning materials for distance learners: lessons learnt from Indonesia's Universitas Terbuka. *Turkish Online Journal of Distance Education*, 8(2), 95 - 112.
- Öner, G., & Öztürk, M. (2019). Okul dışı öğrenme ve öğretim mekânları olarak bilim merkezleri: Sosyal bilgiler öğretmen adaylarının deneyimi. *Eskişehir Osmangazi Üniversitesi Sosyal Bilimler Dergisi*, 20, 1109-1135.
- Özkılıç, R., & İmer, G. (2009). Self-efficacy of teacher trainees toward educational software development. *Education Sciences*, 4(2), 422-434.
- Pala, F. K., & Doğan, N. (2009). Nette öğretmen: Eğitim yönetim sistemi. *International Journal of Informatics Technologies*, 2(3), 9-17.
- Prensky, M. (2001). Digital natives, digital immigrants. *On the Horizon, MCB University Pres*, 9(5), 1-6.
- Schwarzer R., & Fuchs R. (1996). *Self-efficacy and Health Behaviors*. In Conner M. & Norman P. (Ed.), Predicting Health Behavior. Research and Practice with Social Cognition Models (p. 163-193). Open University Press.
- Seferoğlu, S. S. (2006). *Öğretim Teknolojileri ve Materyal Tasarımı*. Ankara: Pegem A Yayıncılık.
- Subaşı, Ş., & Demir, B. (2009). Fakültelerin işletme bölümlerinde verilen muhasebe derslerinin durum analizi. *Muhasebe ve Finansman Dergisi*, 44, 127-137.
- Tat, M. M., & Şahinkaya, Y. (2019). Öğretim tasarımı dersinde proje tabanlı öğrenmenin bazı değişkenlere etkisi. *Mustafa Kemal Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 16(43), 60-82.
- Uşun, S. (2000). *Dünya'da ve Türkiye'de Bilgisayar Destekli Öğretim*. Ankara: Pegem A Yayıncılık.
- Uzun, A., Özkılıç, R., & Şentürk, A. (2013). The perception of educational software development self efficacy among undergraduate ceit teacher candidates. *Current Issues in Education*, 16(2), 1-12.
- Veletsianos, G., & Russell, G. S. (2014). *Pedagogical Agents*. In M. Spector, M. D. Merrill, J. Elen & M. J. Bishop (Ed.), Handbook of Research on Educational Communications and Technology (p. 759-769). New York: Springer.
- Wu, P. H., Hwang, G. J., & Tsai, W. H. (2013). An expert system-based context-aware ubiquitous learning approach for conducting science learning activities. *Journal of Educational Technology & Society*, 16(4), 217-230.
- Yenice, N., Sümer, Ş., Oktaylar, H. C., & Erbil, E. (2003). Fen bilgisi derslerinde bilgisayar destekli öğretimin dersin hedeflerine ulaşma düzeyine etkisi. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 24, 152-158.
- Yıldız, D., & Uzunsakal, E. (2018). Alan araştırmalarında güvenilirlik testlerinin karşılaştırılması ve tarımsal veriler üzerine bir uygulama. *Uygulamalı Sosyal Bilimler Dergisi*, 1, 14-28.