

**T.C.**  
**BALIKESİR ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**  
**BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ**  
**ANABİLİM DALI**



**PROGRAMLAMA DİLLERİ DERSİ İÇİN BİR ÖĞRETİM**  
**PLATFORMU GELİŞTİRİLMESİ VE PLATFORMUN ÖZ**  
**YETERLİK ALGISI İLE BAŞARIYA ETKİSİNİN**  
**İNCELENMESİ**

**YÜKSEK LİSANS TEZİ**

**ÖZLEM TÜLEK**

**BALIKESİR, HAZİRAN - 2017**

**T.C.**  
**BALIKESİR ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**  
**BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ**  
**ANABİLİM DALI**



**PROGRAMLAMA DİLLERİ DERSİ İÇİN BİR ÖĞRETİM**  
**PLATFORMU GELİŞTİRİLMESİ VE PLATFORMUN ÖZ**  
**YETERLİK ALGISI İLE BAŞARIYA ETKİSİNİN**  
**İNCELENMESİ**

**YÜKSEK LİSANS TEZİ**

**ÖZLEM TÜLEK**

**Jüri Üyeleri : Yrd. Doç. Dr. M. Emin KORKUSUZ (Tez Danışmanı)**

**Yrd. Doç. Dr. Zerrin Ayvaz REİS**

**Yrd. Doç. Dr. Ayşen KARAMETE**

**BALIKESİR, HAZİRAN - 2017**

## KABUL VE ONAY SAYFASI

**Özlem TÜLEK** tarafından hazırlanan “**PROGRAMLAMA DİLLERİ DERSİ İÇİN BİR ÖĞRETİM PLATFORMU GELİŞTİRİLMESİ VE PLATFORMUN ÖZ YETERLİK ALGISI İLE BAŞARIYA ETKİSİNİN İNCELENMESİ**” adlı tez çalışmasının savunma sınavı 14.06.2017 tarihinde yapılmış olup aşağıda verilen jüri tarafından oy birliği / oy çokluğu ile Balıkesir Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı Yüksek Lisans Tezi olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Danışman  
Yrd. Doç. Dr. M. Emin KORKUSUZ

Üye  
Yrd. Doç. Dr. Zerrin Ayvaz REİS

Üye  
Yrd. Doç. Dr. Ayşen KARAMETE

  
.....  
  
.....  
  
.....

Jüri üyeleri tarafından kabul edilmiş olan bu tez Balıkesir Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulunca onanmıştır.

Fen Bilimleri Enstitüsü Müdürü

Doç. Dr. Necati ÖZDEMİR

.....

## ÖZET

**PROGRAMLAMA DİLLERİ DERSİ İÇİN BİR ÖĞRETİM PLATFORMU  
GELİŞTİRİLMESİ VE PLATFORMUN ÖZ YETERLİK ALGISI İLE  
BAŞARIYA ETKİSİNİN İNCELENMESİ  
YÜKSEK LİSANS TEZİ  
ÖZLEM TÜLEK  
BALIKESİR ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ANABİLİM  
DALI  
(TEZ DANIŞMANI:YRD. DOÇ. DR. M. EMİN KORKUSUZ)**

**BALIKESİR, HAZİRAN - 2017**

Bu çalışmada, lisans öğrencilerinin PHP programlama dili öğrenmedeki başarılarını artırmak için web tabanlı bir platform geliştirilmesi ve geliştirilen platformun etkisinin incelenmesi amaçlanmıştır. Geliştirilen platform wtPHP olarak isimlendirilmiştir. Platform ile öğrencilerin herhangi bir zamanda, mekâna bağlı kalmadan, internete bağlı herhangi bir bilgisayardan giriş yaparak alıştırma yapmalarını sağlamak hedeflenmiştir. Platform web tabanlı olarak tasarlanmıştır. Platformun, öğrencilerin akademik başarıları ve programlama öz yeterliklerine olan etkisini incelenmiştir.

Araştırmada hem nicel hem nitel veriler toplanmıştır. Araştırma yöntemi olarak deneysel desen türlerinden deneme öncesi modellerden olan tek grup ön-test ve son-test modeli kullanılmıştır. Nitel veriler yarı yapılandırılmış görüşme tekniği ile elde edilmiştir.

Araştırmanın çalışma grubunu 2016-2017 güz yarıyılında Balıkesir Üniversitesi, Necatibey Eğitim Fakültesi, Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü'nde İnternet Tabanlı Programlama dersini alan 38 üçüncü sınıf öğrencisi oluşturmaktadır.

Araştırma sonucunda öğrencilerin başarı puanlarında ve öz yeterlik algılarında istatistiksel olarak anlamlı bir artış olduğu tespit edilmiştir. Öğrencilerin başarı puanlarındaki artış ile öz yeterlik algılarındaki artış arasında zayıf bir korelasyon gözlenmiştir. Öğrenciler uygulamayı kullanım özellikleri açısından sade, kolay anlaşılır ve pratik bulmuşlardır. Bunun yanında öğrencilerin çoğu, uygulamanın çalışmalarına katkısı olduğunu ve içerik özellikleri açısından yararlı bulduklarını belirtmişlerdir.

**ANAHTAR KELİMELER:**Programlama öğretimi, PHP, materyal geliştirme.

## **ABSTRACT**

### **DEVELOPING A TEACHING PLATFORM FOR THE PROGRAMMING LANGUAGE COURSE AND INVESTIGATION OF THE EFFECT OF THE PLATFORM ON SUCCESS AND SELF-EFFICACY**

**MSC THESIS**

**ÖZLEM TÜLEK**

**BALIKESİR UNIVERSITY INSTITUTE OF SCIENCE**

**COMPUTER EDUCATION AND INSTRUCTIONAL TECHNOLOGY**

**(SUPERVISOR:ASSIST. PROF. DR. M. EMİN KORKUSUZ )**

**BALIKESİR, JUNE 2017**

In this study, it is aimed to develop a web based platform and to investigate the effect of the developed platform in order to increase the success of undergraduate students learning PHP programming language. The developed platform was named wtPHP. With this application, it is aimed to let the students to practice by logging in from any computer connected to the internet, without being connected to the place at any time. The platform is web based. The impact of the platform on students' academic success and programming self-efficacy has been examined.

Both quantitative and qualitative data were collected in the study. As a research method, a single group pre-test and post-test model from pre-experimental models were used from experimental design types. Qualitative data were obtained by semi-structured interview technique.

The study group is Balıkesir University, Necatibey Education Faculty, Department of Computer and Instructional Technology Education in the fall semester of 2016-2017, 38 undergraduate students who take Internet Based Programming Course.

As a result of the research, it was determined that there was a statistically significant increase in students' achievement scores and self-efficacy perceptions. A weak correlation was observed between the increase in students' achievement scores and the increase in self-efficacy perceptions. Students have found practice simple, easy to understand and practical in terms of their use characteristics. Most of the students, on the other hand, stated that the practitioner contributed to their work and found them useful in terms of content features.

**KEYWORDS:** Programming teaching, PHP, material development.

# İÇİNDEKİLER

## Sayfa

<b>ÖZET</b> .....	<b>i</b>
<b>ABSTRACT</b> .....	<b>ii</b>
<b>İÇİNDEKİLER</b> .....	<b>iii</b>
<b>ŞEKİL LİSTESİ</b> .....	<b>v</b>
<b>TABLO LİSTESİ</b> .....	<b>vi</b>
<b>ÖNSÖZ</b> .....	<b>vii</b>
<b>1. GİRİŞ</b> .....	<b>1</b>
1.1 Araştırmanın Amacı .....	4
1.2 Araştırmanın Önemi .....	4
1.3 Araştırma Soruları .....	5
1.4 Varsayımlar .....	6
1.5 Sınırlılıklar .....	6
<b>2. LİTERATÜR</b> .....	<b>7</b>
2.1 Programlama Eğitimi .....	7
2.1.1 Programlama Eğitiminin Önemi.....	7
2.1.2 Programlama Eğitimi ile İlgili Yapılmış Araştırmalar .....	8
2.1.3 Programlama Dilleri Eğitiminde Eğitimi Destekleyici Materyallerin Kullanımı .....	10
2.2 Öğretim Tasarımı .....	12
<b>3. YÖNTEM</b> .....	<b>16</b>
3.1 Araştırma Modeli .....	16
3.2 Katılımcılar.....	16
3.3 Verilerin Toplanması.....	16
3.3.1 Nicel Verilerin Toplanması .....	17
3.3.2 Nitel Verilerin Toplanması .....	18
3.4 wtPHP'nin Özellikleri.....	20
3.4.1 Analiz Aşaması.....	20
3.4.2 Tasarım Aşaması .....	22
3.4.3 Geliştirme Basamağı .....	24
3.4.4 Uygulama ve Değerlendirme Basamağı .....	38
<b>4. BULGULAR</b> .....	<b>41</b>
4.1 Geliştirilen Platformun Öğrencilerin Programlama Başarısına Etkisi .....	41
4.2 Geliştirilen Platformun Öğrencilerin Programlamaya Yönelik Öz Yeterlik Algılarına Etkisi .....	42
4.3 Öğrencilerin Öz Yeterlik Algılarındaki Değişim İle Ders Başarısındaki Değişim Arasındaki İlişki.....	43
4.4 Geliştirilen Platformun Programlama Eğitiminde Kullanımına İlişkin Öğrenci Görüşleri .....	44
<b>5. SONUÇ, TARTIŞMA VE ÖNERİLER</b> .....	<b>50</b>
5.1 Sonuçlar.....	50
5.2 Tartışma.....	52
5.3 Öneriler.....	54
5.4 Platformun Geliştirilmesine İlişkin Öneriler .....	54
5.5 Uygulamaya İlişkin Öneriler .....	54
<b>6. KAYNAKLAR</b> .....	<b>56</b>

<b>7. EKLER</b> .....	<b>61</b>
EK A- Programlamaya İlişkin Öz Yeterlik Algısı Ölçeği.....	61
EK B - Başarı Testi .....	62
EK C- Yarı Yapılandırılmış Görüşme Formu (Öğrenciler ile) .....	67
EK D- Yarı Yapılandırılmış Görüşme Formu (Öğretim Elemanları ile) .....	68

## ŞEKİL LİSTESİ

### Sayfa

Şekil 1.1: Genişbant internet abone sayısı.....	1
Şekil 2.1: Bilişim tek. ve yazılım dersi 4. basamak öğrenme alanları.....	8
Şekil 2.2: ADDIE basamakları.....	13
Şekil 2.3: ADDIE öğretim tasarım modeli.....	14
Şekil 3.1: Nitel veriler toplanırken izlenen yol.....	19
Şekil 3.2: Sunucuların çalışması.....	24
Şekil 3.3: Duyurular ekranı.....	25
Şekil 3.4: Konular ekranı.....	26
Şekil 3.5: Alıştırmalar ekranı.....	27
Şekil 3.6: Örnek çıktı ekranı.....	29
Şekil 3.7: Hata mesajları örnek ekranı.....	31
Şekil 3.8: Hata ayıklama için örnek.....	32
Şekil 3.9: Öğretim elemanı için yönetim menüsü.....	33
Şekil 3.10: Duyurular ekranı.....	33
Şekil 3.11: Yeni içerik ekleme ekranı.....	34
Şekil 3.12: Tüm içeriklerin listelendiği ekran.....	35
Şekil 3.13: İçerik düzenleme ekranı.....	36
Şekil 3.14: Sisteme yeni öğrenci ekleme ekranı.....	37
Şekil 3.15: Sistemde kayıtlı kullanıcılar listesi.....	37
Şekil 3.16: Öğrencinin sayfalarda harcadığı süre.....	38
Şekil 3.17: Oturum açma-kapatma zamanı ve oturum süresi.....	39
Şekil 3.18: Araştırma süreci.....	40
Şekil 4.1: Kelime bulutu.....	44

## TABLO LİSTESİ

### Sayfa

<b>Tablo 4.1:</b> Öğrencilerin programlama başarılarındaki değişim. ....	41
<b>Tablo 4.2:</b> Öğrencilerin öz yeterlik algılarındaki değişim. ....	42
<b>Tablo 4.3:</b> Öğrencilerin öz yeterlik algılarındaki değişim (basit programlama görevlerini gerçekleştirme). ....	42
<b>Tablo 4.4:</b> Öğrencilerin öz yeterlik algılarındaki değişim (karmaşık programlama görevlerini gerçekleştirme). ....	43
<b>Tablo 4.5:</b> Öğrencilerin ders başarıları ile öz yeterlik algılarındaki değişim arasındaki ilişki.....	43
<b>Tablo 4.6:</b> İnternet tabanlı programla dersine ait öğrenci görüşlerine ilişkin temalar. ....	45
<b>Tablo 4.7:</b> İnternet tabanlı programla ile önceki programlama derslerinin karşılaştırılmasına ait öğrenci görüşlerine ilişkin temalar.....	45
<b>Tablo 4.8:</b> wtPHP'nin içerik özellikleri açısından değerlendirilmesine yönelik öğrenci görüşlerine ilişkin temalar. ....	46
<b>Tablo 4.9:</b> wtPHP'nin kullanım özellikleri açısından değerlendirilmesine yönelik öğrenci görüşlerine ilişkin temalar. ....	47

## ÖNSÖZ

Bu çalışmada bana rehberlik ederek yol gösteren değerli danışmanım ve hocam Yrd. Doç. Dr. M. Emin KORKUSUZ'a en içten duygularıyla teşekkürlerimi sunuyorum.

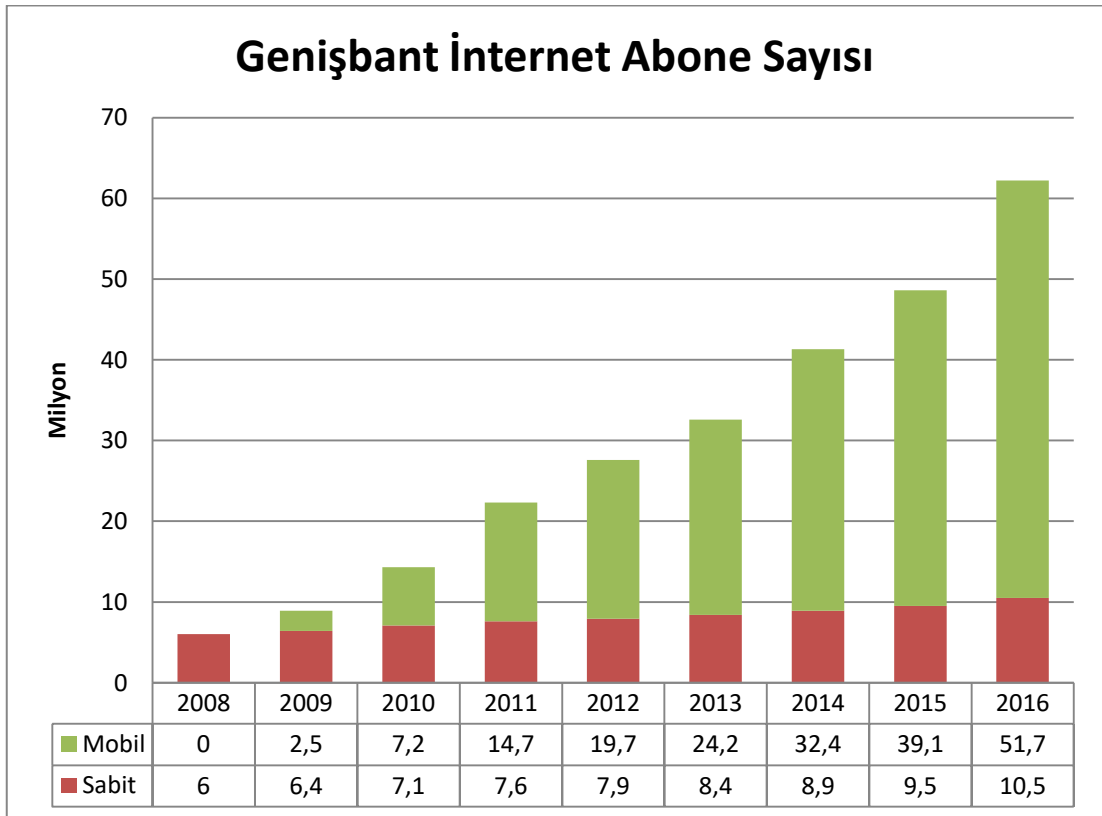
Çalışmam sırasında büyük bir anlayışla beni destekleyen başta daire başkanımız Ünal SIVACIOĞLU olmak üzere tüm mesai arkadaşlarıma, özellikle manevi desteklerinden dolayı Uzm. Dr. Öznur ÖZTUNÇ KAYMAK ve Uzm. Emine BAYATA'ya teşekkürü bir borç bilirim.

Çalışmam sırasında değerli görüş ve katkılarını esirgemeyen çok değerli arkadaşlarım Yrd. Doç. Dr. Hüseyin GÜNEŞ ve Uzm. Okan DURUSOY'a teşekkür ederim.

Beni büyütüp bugünlere getiren sevgili anne ve babama, ikiz çocuklarımla bakımı konusunda desteklerini hiçbir zaman esirgemeyen çok değerli kayınvalidem ve kayınpederime, her zaman yanımda olduğunu ve beni desteklediğini bildiğim kıymetli eşim Abdullah'a minnet borçluyum. İyi ki varsınız.

## 1. GİRİŞ

Ülkemizde bilgisayar ve internet kullanımını hızla artmaktadır. Bilgi Teknolojileri ve İletişim Kurumu (BTK) tarafından, telekomünikasyon sektöründe faaliyet gösteren firmalarca gönderilen veriler esas alınarak hazırlanan raporda 2008 yılında 6 milyon olan geniş bant internet abone sayısı, 2016 yılı aralık ayı itibariyle 62,2 milyonu aşmıştır (BTK, 2017). Türkiye İstatistik Kurumu (TÜİK) tarafından, 2016 yılı için yapılan Hanehalkı Bilişim Teknolojileri Kullanım Araştırması'na göre ülkemizde genişbant internet bağlantısına sahip hanelerin oranı %73,1 olmuştur. Ayrıca 16-74 yaş grubundaki bireylerde internet kullanım oranı %61,2 olarak tespit edilmiştir (TÜİK, 2016). İnternet ve iletişim teknolojilerinin kullanımının bu kadar çok yaygınlaşması, eğitimde de bu yenilikleri kullanma fırsatını doğurmuştur.



**Şekil 1.1:** Genişbant internet abone sayısı.

Artan genişbant bağlantı sayısı, bilgiye ulaşmayı kolaylaştırmakla birlikte özellikle gençlerin sadece üretilmiş içeriğin tüketicisi olması riskini doğurmaktadır.

Bu riski en aza indirmek ve aynı zamanda üretici konumuna gelebilmek için yeni çalışmaların yapılması gereklilik halini almıştır.

Eğitimde teknolojiden faydalanma birçok kolaylığı da beraberinde getirmiştir. Belli bir zaman diliminde ve sınıflarda yapılan yüz yüze eğitimde ses ve görüntüyü sınıflara taşıyan çoklu ortam araçları çeşitli imkanlar sunarken; web teknolojilerindeki gelişmeler, e-posta, internet vb. teknolojiler eğitimin zaman ve mekan sınırlarının dışına çıkmasını sağlamıştır.

Yüz yüze eğitimdeki sınırlı zaman aralığı ve sınıfların kalabalık olması, öğrencilerin öğretim elemanı ile yeterli düzeyde iletişim kurmasını zorlaştırabilmektedir. İnternet ve web tabanlı öğrenme ortamlarının sağladığı iletişim ve etkileşim ortamları sayesinde öğrenci-öğretmen iletişiminin sınıf dışında da mümkün olabileceği düşünülmektedir. Ayrıca öğrenciler, ders materyallerine ve ilave kaynaklara diledikleri zaman ulaşabilmekte ve konuları istedikleri kadar tekrar etme imkânını bulabilmektedir. Öğrencilerin interneti kullanma yoğunluğu dikkate alındığında, yüz yüze eğitimi destekleyen web tabanlı öğrenme ortamlarının, eğitimde akademik başarıyı ve öğrencilerin derse olan ilgi ve motivasyonu arttırmasında bir fırsat olarak görmek mümkündür.

Üniversitelerin farklı program ve bölümlerinde okutulan programlama dilleri dersinde öğrencilerin, konuları tekrar edip, bolca örnek uygulama yapması akademik başarıyı arttıran faktörlerden biridir. Yapılan araştırmalar bu ders için hazırlanmış çeşitli araçların ve web tabanlı öğrenme ortamlarının öğrencilerin derse olan ilgi ve isteğini arttırdığı ve programcılık performansını arttırmada destek olduğunu ifade etmektedir (Ersoy, 2003; Hadjerrouit, 2008; Yurdağül ve Gültekin, 2009; Dehmenoğlu, 2015).

Programlama dilleri dersinde, öğrencilerden hem öğretilen programlama diline has özellikleri öğrenmeleri hem de problemi analiz etme, çözüm için algoritmalar geliştirme ve bu çözüm yollarından hareketle programlama dilinin kurallarına uygun olarak programı yazma gibi üst düzey bilişsel becerileri kazanmaları beklenmektedir.

Öğrencilerin öğrenilen programlama dilinin söz dizimini ve kurallarını bilmesi, o derste başarılı olması için yeterli gelmemektedir. Öğrendikleri konuları

içeren örnek uygulamalar ile kazandığı problem çözme becerisi ve edindiği deneyim başarısını arttırmada önemli bir faktördür. Bilgisayar programlamada deneyim kazanmak için sürekli uygulama yapmak gerekir. Yapılan araştırmalar programlama dilleri derslerinin, programlamaya yeni başlayan öğrenciler tarafından zor olarak kabul edildiğini göstermektedir (Aşkar ve Davenport, 2009). Programlama deneyimi az olan öğrencilerin programlamaya yönelik öz yeterlik algıları da düşüktür ve bu durum onların programcılık performansını olumsuz yönde etkilemektedir. Öz yeterlik algısı bir kişinin, kendisine verilen görevleri yerine getirirken kendi başarısına ilişkin yargılarıdır ve kişinin performansı ile doğru orantılıdır (Mazman ve Altun, 2013).

Öğrencilerin bilgisayar programlamadaki başarısızlığının başlıca sebebi olarak çalışma ve uygulama yapma eksikliği olduğu belirtilmektedir (NG, Choy, Kwan ve Chan, 2005; Hawi, 2010; Uzun ve Özkılıç, 2012). Programlama dersleri için ayrılan sürede yapılan örnek uygulamalar bir programlama dilinin öğrenilmesinde yetersiz kalmaktadır. Geleneksel yöntemlerle sınıf ortamında yapılan programlama derslerinde sınıfın kalabalık olması ve öğretim elemanının öğrenciler için yeterli zaman ayıramaması önemli bir sorundur. Kalabalık sınıflarda öğrencilerin istenen örnek uygulamaları yapmasını sağlamanın, derse olan ilgi ve motivasyonlarını yüksek tutmanın oldukça zor olduğu görülmektedir.

Programlama dilleri eğitimi öğrenciler tarafından zor bir süreç olarak kabul edilmektedir. Bu durum derse olan ilgi ve motivasyonu da azaltmaktadır. Öğrencilerin oldukça fazla zaman geçirdikleri internet ortamında onların ilgisini çekebilecek bir öğrenme platformunun yüz yüze eğitimi destekleyeceği ve akademik başarıyı arttıracığı düşünülmektedir.

## 1.1 Araştırmanın Amacı

Bu çalışmanın amacı; lisans düzeyinde okutulan internet tabanlı programlama dersi için web tabanlı, öğrencilerin konularla ilgili çeşitli uygulama örnekleri bulabilecekleri, kendi kendilerine uygulama yapabilecekleri ve yaptıkları uygulamalarla ilgili aldıkları hata mesajlarının olası sebepleri konusunda kendi dillerinde ipuçları alabilecekleri bir platform oluşturmaktır. Oluşturulan bu platform ile öğrencilerin ders saatleri dışında, istedikleri mekânda programlama çalışmalarını sağlamak amaçlanmaktadır. Ayrıca, öğrencilerin programlama dilleri dersindeki akademik başarısında ve programlamaya yönelik öz yeterlik algısında herhangi bir değişiklik olup olmadığı ve geliştirilen platform hakkında görüşlerinin belirlenmesi amaçlanmıştır.

## 1.2 Araştırmanın Önemi

Ülkemizde programlama dilleri dersleri farklı programlarda, farklı sınıflarda ve farklı düzeylerde verilse de bu dersle ilgili problemler hemen hemen aynıdır. Eğitim fakültelerinde yer alan Bilgisayar ve Öğretim Teknolojileri Eğitimi (BÖTE) bölümü de bu bakımdan benzer problemleri yaşamaktadır.

Programlama dilleri eğitimiyle ilgili başlıca problem her yarıyılıda farklı bir programlama dilinin seçilmesidir. Bir yarıyıl yaklaşık 3 ay sürmektedir ve bu yarıyıl sona erdiğinde programlama dilinin öğretimi de sona ermektedir. Yeni yarıyılıda farklı bir programlama dili öğretime başlanmaktadır. Öğrenciler eğitimleri süresince birçok programlama dili dersi olsa da bu dilleri çalışma hayatlarında kullanabilecek düzeyde öğrenememektedir. Bu tür platformlar, öğrencilerin programlama dili öğrenme çalışmalarına ders dışında devam etme becerisi kazandırılarak, eğitimini aldığı programlama diline daha uzun süreler çalışıp gerekli bilgiyi edinmesi için bir fırsat oluşturma potansiyeli taşımaktadır.

Bir diğer problem ise öğrencilerin çalışma alışkanlıkları ile ilgilidir. Bir programlama dilini iyi bir düzeyde öğrenebilmek için haftada birkaç saat yapılan bir ders ve bir dönemde yapılan 2-3 sınav için yapılan hazırlık yeterli değildir. Bu çalışmanın en önemli hedeflerinden birisi öğrencilerin ders içi etkinlikleri ile

yetinmeyip programlama çalışmak için zaman ayırmalarını sağlamaktır. Çalışma öğrenciye hem okulda hem de okul dışında çalışması için düzenlenmiş bir ortam sunmaktadır. Öğrenci yapacağı çalışmalarla programlama dilini daha iyi öğrenebilme imkânı bulacaktır. Bu aynı zamanda öğrenciye, kendi eğitimi hakkında sorumluluk alma davranışı kazandırması bakımından da çok önemlidir. Çünkü öğrencinin okul dışı etkinliklerde çalışması için bir sorumluluk duygusuna sahip olması gereklidir.

Bu çalışma ile öğrencilerin, öğrendikleri programlama dili için daha fazla çalışma yapmaları amaçlanmaktadır. Öğrencilerin bir yarıyıldaki yapılan derslerle sınırlı olan çalışmalarını, ders saatleri dışında da yapmaları sağlanarak çalışma süresini arttırmak, daha çok uygulama yapmalarını sağlamak ve sonuçta akademik başarıyı arttırmak hedeflenmektedir. Öğrenciden, platformda çalışırken kendi başına karar vermesi ve öğrenme sorumluluğunu üstlenmesi beklenmektedir. Bu durumun öğrencideki sorumluluk duygusunu geliştireceği düşünülmektedir.

Bu çalışmada kullanılacak olan platform (wtPHP), PHP programlama dili için hazırlanmıştır. Yapılan literatür taramasında PHP programlama dili için hazırlanmış eğitim amaçlı kullanılabilir bir uygulamaya rastlanılmamıştır. Alanda görülen bu boşluğa katkıda bulunması bakımından yapılan çalışmanın önemli olduğu düşünülmektedir.

### **1.3 Araştırma Soruları**

1. Geliştirilen platformun çalışmanın katılımcıları olan öğrencilerin programlama başarısına etkisi nedir?
2. Geliştirilen platformun çalışmanın katılımcıları olan öğrencilerin programlamaya yönelik öz yeterlik algılarına etkisi nedir?
3. Öğrencilerin öz yeterlik algılarındaki değişim ile ders başarısındaki değişim arasında anlamlı bir ilişki var mıdır?
4. Geliştirilen platform hakkındaki katılımcı görüşleri nelerdir?

## 1.4 Varsayımlar

Bu arařtırmada, kullanılan ölçme araçlarının arařtırmanın amacına uygun olduđu ve çalışmanın katılımcıları olan öğrencilerin veri toplama araçlarını içtenlikle ve doğru bir şekilde yanıtladıkları kabul edilmiştir.

## 1.5 Sınırlılıklar

- Arařtırma, Balıkesir Üniversitesi, Necatibey Eğitim Fakültesi, Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü'nde öğrenim gören toplam 38 öğrenci ile gerçekleştirilmiştir.
- Arařtırmada, PHP programlama diline ait *Şartlı İfadeler* ve *Döngüler* konuları ele alınmıştır.
- Arařtırma 4 (dört) haftalık bir zaman diliminde gerçekleştirilmiştir.

## 2. LİTERATÜR

### 2.1 Programlama Eğitimi

Elektronik cihazlarda bulunan işlemcilerin farklı amaçlara hizmet edebilmesi için çeşitli komutlarla kontrol edilmeleri gerekmektedir. Özel kelime ve sembollerden oluşan bu komutlar bütününe programlama dili denmektedir. Bilgisayarlar için farklı işlev ve görevlere göre dizayn edilmiş çok sayıda programlama dili bulunmaktadır. ADA, C, PHP, ASP gibi diller bunlardan sadece bazılarıdır (Güngören, 2001). Bu kadar çok dilin olması ve bilgisayar programlarının artan önemi ile programlama dillerinin öğrenilmesi ve öğretilmesi hem bireylerin hem de eğitim sistemlerinin konusu haline gelmiştir.

#### 2.1.1 Programlama Eğitiminin Önemi

Bir program ya da daha genel bir ifadeyle yazılım, elektronik araçların belirli bir işi yapabilmesini sağlayan, özel dillerde yazılmış komutlardır. Mutfaktaki buzdolabından, hastanedeki röntgen cihazına, cep telefonundan ulaşım araçlarına kadar sadece günlük yaşamda kullandığımız bilgisayarlarda değil her alanda (ulaşım, haberleşme, sağlık, savunma, imalat vb.) elektronik parçaları içeren tüm araç ve makinelerde yazılımlar bulunmaktadır. Teknolojik ilerlemeyi hedefleyen ve sanayi üretimine önem veren ülkelerin programlama eğitimine önem verdiği ve öğrencilere sadece yüksek öğrenimde değil, ilköğretim ve lise düzeyinde programlama eğitimi vermeye başladıkları görülmektedir (Demirer ve Sak, 2016).

Ülkemizde de önemi anlaşılan programlama eğitiminin ilköğretim çağından itibaren verilmesi amacıyla Millî Eğitim Bakanlığı (MEB) birtakım düzenlemeler yapmıştır. Buna göre ortaokul seviyesinde 5, 6, 7 ve 8. sınıflarda okutulan “Bilişim Teknolojileri ve Yazılım” dersi için belirlenen öğrenme alanlarının 4. basamağında programlama eğitimine temel oluşturabilecek konular belirlenmiştir (TTKB, 2012).

- 4. Problem Çözme, Programlama ve Özgün Ürün Geliştirme**
- 4.1. Problem Analiz ve Çözme Yaklaşımları
  - 4.2. Algoritma ve Strateji Geliştirme (algoritma oluşturma mantığı, sözde kod, akış şemaları vb.)
  - 4.3. Programlama
    - 4.3.1. Program ve Programlama Dilleri
    - 4.3.2. Programlama Araçları
    - 4.3.3. Animasyon ve Makrolar
    - 4.3.4. Sosyal Kodlama Ortamları
    - 4.3.5. Kullanıcı Etkileşimli Program Hazırlama
  - 4.4. Yazılım Projesi Geliştirme, Uygulama ve Yayınlaştırma

**Şekil 2.1:** Bilişim tek. ve yazılım dersi 4. basamak öğrenme alanları.

Bunun yanında mesleki ve teknik liselerde bulunan Bilişim Teknolojileri Alanı altında Web Programcılığı ve Veritabanı Programcılığı dallarında programlama eğitimi verilmektedir.

Programlama eğitiminin farklı bir boyutu daha bulunmaktadır. Programlama öğrenmenin, öğrencilerin bakış açılarını değiştirdiği, yaratıcı düşünme, olaylar arasındaki ilişkileri görebilme, problemlerin çözüm yolları konusunda çözüm üretme ve sistematik düşünme becerilerini geliştirme gibi olumlu bir etkisinin olduğu bilinmektedir (Akpınar ve Altun, 2014). Bilgisayarın günümüze değin gelişiminde çok önemli bir yere sahip olan ve Apple'ın kurucularından Steven Paul Jobs "*Bu ülkedeki herkes bilgisayar programlamayı öğrenmeli; çünkü bu size düşünmeyi öğretir.*" diyerek programlamanın öğrencilerin bilişsel gelişimine yaptığı önemli etkiye vurgu yapmıştır (Code.org, 2013).

### **2.1.2 Programlama Eğitimi ile İlgili Yapılmış Araştırmalar**

Bu bölümde programlama eğitimi hakkında yapılmış çeşitli araştırmalar konularına göre incelenmiştir.

Programlama eğitiminin öğrencilere kazandırdığı problem çözme ve analitik düşünme becerisi, bu derslerin küçük yaştaki öğrencilere de verilmeye başlamasına sebep olmuştur; ancak programlama öğrenme sürecinin karmaşık ve zor bulunması, eğitimcileri bu dersleri küçük sınıflardaki öğrencilerin daha kolay anlayacağı ve daha eğlenceli bulacağı şekilde verilmesi için arayışlara yöneltmiştir.

Literatüre bakıldığında özellikle küçük yaştaki öğrencilere programlamayı sevdirmeyi, herhangi bir dili öğretmekten ziyade temel programlama mantığını kavratmayı hedefleyen çeşitli yazılımlar, web siteleri, organizasyonları inceleyen araştırmalar bulunmaktadır (Lewis, 2010; Maloney, Resnick, Rusk, Silverman ve Eastmond, 2010; Kukul ve Gökçearslan, 2014; Berksoy, Sözcü, Armağan ve Arslan, 2016). Bu çalışmalarda programlama eğitiminin öğrencilerin bilişsel gelişimine yaptığı katkıya vurgu yapıp, erken yaşlarda bu eğitimin verilmeye başlamasının gereklilikleri ortaya konulmaktadır. Bu kapsamda erken yaştaki öğrencilere programlama eğitimi verilirken sade, karmaşık kod yapıları yerine görsel öğeleri içeren, öğrencilerin yazım hataları ile uğraşmak yerine problemi analiz etme, çözüm yolları konusunda fikirler üretme ve yaratıcılık yönlerinin gelişmesi konusunda destekleyici yazılımlar kullanılmasının yararlı olacağı belirtilmektedir.

Massachusetts Teknoloji Enstitüsü (MIT) tarafından geliştirilen, programlamaya yeni başlayan kullanıcıların kod ifadeleri yerine, görsel kod blokları ile program geliştirmesini hedefleyen Scratch yazılımının programlama eğitiminde kullanımını konu alan 32 makaleyi inceleyen Çatlak, Tekdal ve Baz (2015) çalışmalarda Scratch yazılımının kod yazmaya gerek olmadan kolay bir şekilde proje üretmeyi mümkün kılması, görsel öğeler ile desteklenmesi, zor kavramları daha somut hale getirerek anlaşılmasını kolaylaştırdığından dolayı öğrenci başarısını arttırdığı yönünde bulguların yer aldığını ifade etmişlerdir.

Program yazmak karmaşık bir süreçtir. Verilen bir problemi bilgisayar programı haline getirirken önce problem analiz edilir, çözüm yolları için algoritmalar oluşturulur ve sonra program kodlanır. Yani hem problem analiz etme ve çözme gibi düşünsel becerilere sahip olmayı hem de programlama diline has kavramlara ve yapılara hâkim olmayı gerektirir. Program yazabilmek deneyim gerektirmektedir. Bu sebeple bir programlama dilini öğrenmek özellikle yeni başlayan, acemi öğrencilere çok zor gelmektedir. Literatürde programlamaya yeni başlayan öğrencilerin yaşadıkları zorluklarla ilgili yapılmış birçok araştırmaya rastlanmaktadır (Lahtinen, Ala-Mutka ve Järvinen, 2005; Butler ve Morgan, 2007; Mhashi ve Alakeel, 2013).

Lahtinen, Ala-Mutka ve Järvinen (2005) yaptıkları çalışmada programlama ile ilgili kavramların öğrenilmesinin zor olduğunu belirtip, örnek uygulamalar yapmanın kavramları somutlaştırdığı ve öğrenmeyi sağladığını belirtmektedir.

Çalışmada, acemi programcılarının programlama diline ait kavram ve yapıların mantığını ilk başta tamamen anlamasının zor olduğunu, yapılan örnek uygulamalar sayesinde bunların anlaşılmasının kolaylaşacağı ifade edilmektedir.

Mhashi ve Alakeel (2013), Tabuk Üniversitesi'nde 165 öğrencinin katılımı ile yaptıkları çalışmada öğrencilerin bilgisayar programlamayı öğrenirken karşılaştıkları problemleri şöyle ifade etmektedir:

- Öğrenciler yeterli miktarda örnek uygulama yapmamaktadırlar. Ancak programlama diline ait kavramların yeterince öğrenilebilmesi için örnek uygulamalar yapmak gerekmektedir.
- Uygulama ve pratik yapmak için bilgisayar laboratuvarlarını kullanmak her zaman mümkün olmamaktadır.
- Programlamaya yeni başlayan öğrenciler kendi başına çalışırken birtakım sorunlarla karşılaşmaktadır. Bu onların motivasyonunu ve özgüvenlerini azaltmaktadır. Ancak öğrenciler çalışırken ihtiyaç duydukları rehberlik ve yardımı sağlayacak öğretim elemanları ya da yardımcılara ulaşamamaktadırlar.

### **2.1.3 Programlama Dilleri Eğitiminde Eğitimi Destekleyici Materyallerin Kullanımı**

Programlama dilleri eğitiminde eğitimi destekleyici materyaller (mobil öğrenme aracı, çoklu ortama dayalı eğitim yazılımı, web tabanlı öğrenme ortamları, bilgisayar destekli öğrenme ortamı vb.) kullanarak, materyallerin etkinliğini tespit eden çeşitli araştırmalar bulunmaktadır. Bu çalışmalar incelendiğinde materyallerin, programlama eğitimine katkıda bulunduğu, öğrencilerin derse olan ilgisi ve ders başarılarına olumlu yönde etki ettiği görülmektedir.

Dehmenoğlu (2015) yüksek lisans tezinde mesleki ve teknik Anadolu lisesi bilişim teknolojileri alanında okutulan programlama temelleri dersi için mobil öğrenme aracını geliştirmiştir. Uygulama sonucu elde edilen verilere göre mobil öğrenme aracının öğrencilerin akademik başarıları üzerinde istatistiki olarak anlamlı bir etkisinin olduğu, mobil öğrenme aracı ile desteklenen gruptaki öğrencilerin,

sadece yüz yüze eğitim alan öğrencilerden daha yüksek başarı sağladığı sonucuna varılmıştır.

Vural ve Taşdelen (2014), web tasarımı ve programlama dersi için tasarladıkları web tabanlı öğrenme modülünün, öğrencilerin istedikleri yer ve zamanda ders çalışıp uygulama yapabilmelerini ve uygulamalarının çıktılarını kolay bir şekilde görebilmelerini sağladığını ifade edip, çalışma sonucu elde ettikleri verilere göre materyali uyguladıkları deney grubu öğrencilerinin ders başarılarının, kontrol grubu öğrencilerinden daha iyi olduğunu belirtmişlerdir.

Uzun (2013), lisans düzeyinde okutulan programlama dilleri dersi için hazırlanan bilgisayar destekli öğrenme ortamına ilişkin öğrenci görüşlerini araştırdığı çalışmada öğrencilerden büyük çoğunluğunun dersin bilgisayar destekli olarak yürütülmesinden memnun kaldıkları sonucuna ulaşmıştır. Bu çalışmada yüz yüze yapılan dersi desteklemek maksadıyla oluşturulan öğrenme ortamının şu özellikleri taşıması gerektiği vurgulanmıştır:

- Öğrenme ortamı, öğrencilerin istedikleri zaman, istedikleri yerden çalışabilmelerine imkân sağlamalıdır.
- Öğrenme ortamı, zengin çoklu ortam materyalleri barındırmalıdır. İçeriğin anlatılması derinlemesine ve iyi detaylandırılmış olmalıdır. İçerik çok sayıda örnek uygulama içermelidir.
- Öğrenme ortamı, öğrencilerin kendi aralarında ve öğretmenleriyle iletişim kurmasına ve bireysel ya da arkadaşlarıyla çalışma yapabilmelerine imkân sağlamalıdır.

Yurdugül ve Gültekin (2009) çoklu ortama dayalı öğretim yazılımının programlama öğreniminde öğrenci başarıları üzerine etkisini araştırmak amacıyla yaptıkları çalışmada; programlama öğreniminde çoklu ortama dayalı öğretim yazılımının, alışlagelmiş öğretim yöntemine kıyasla öğrenci başarısı üzerinde istatistiksel olarak anlamlı bir etkisi bulunduğu saptamıştır.

Hadjerrouit (2008) lisans düzeyinde okutulan programlama dersi (Java) için yaptığı çalışmada yüz yüze yapılan normal öğretimin yanında içerik yönetim sistemi (LMS) ile çevrimiçi kaynakların bulunduğu bir öğrenme ortamı oluşturmuştur. Öğrenciler, bu çevrimiçi kaynakların Java ile ilgili kavramları

anlamada ve programcılık performanslarını arttırmada destek olduğunu belirtmişler; ayrıca motivasyon sağladığını ve programlama bilgilerini inşa etmede yardım ettiğini belirtmişlerdir.

Al-Imamy, Alizadeh ve A. Nour (2006) C programlama dilinin öğretiminde kullanılmak üzere web tabanlı bir materyal geliştirmişler ve materyalin, programlama bilgisi farklı seviyelerde olan öğrencilerden oluşan bir sınıfta, öğrenciler arasındaki farklılıkları kapatarak, hepsinin aynı seviyeye ulaşmasında yararlı olduğunu belirtmişlerdir.

Ersoy (2003) yüksek lisans tezi için yaptığı çalışmada, lisans düzeyinde okutulan programlama dersi için geleneksel yüz yüze öğretimi, web tabanlı öğretim materyali (web sitesi) ile destekleyerek öğrencilerden anket yoluyla aldığı verilere göre materyalin öğrencilerin derse olan ilgisini ve isteğini arttırdığını belirtmiştir.

## 2.2 Öğretim Tasarımı

Eğitim kurumlarında yapılan eğitimin, planlı ve programlı yapılması esastır. Bunu sağlamak için eğitim programları hazırlanmaktadır. Eğitim programları, öğrenene kazandırılması hedeflenen davranışların nasıl kazandırılacağını planlamaktadır. Eğitim programları, eğitimin hedeflerini ortaya koyan, bu hedeflere ulaşmak için gerekli eğitim durumlarını belirten ve eğitimin sonunda hedeflere ne derecede ulaşıldığını anlamak için değerlendirme süreçlerini içeren bir plandır (Seferoğlu, 2016). Bu plan, değerlendirme sonuçları ve oluşan yeni ihtiyaçlara göre sürekli olarak güncellenmektedir. Eğitim programları genel olarak ülkedeki tüm eğitim kurumları için hazırlanır. Belirli bir öğrenci grubu için, belirli bir ders ya da konunun öğretilmesine yönelik olarak, öğretimin planlanmasında öğretim tasarımı rehber rolündedir.

Öğretim tasarımı ile öğretimin kalitesini ve etkisini arttırmak için en uygun yöntemler belirlenmektedir. Öğretim tasarımı, öğrenmeyi destekleyen öğrenme ortamı ve aktivitelerin planlanmasıdır (İşman, Çağlar, Dabaj ve Ersözlü, 2005). Bu planlama, birtakım ilkelere göre duruma en uygun olan öğretim yöntemlerinin seçilmesini içermektedir (Çakır ve Karataş, 2012). Öğretim tasarımı ile kazandırılmak istenen davranışlar ve hedefler ortaya koyulur, öğrenen özellikleri

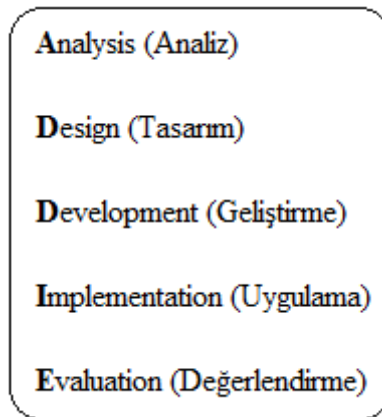
belirlenir, öğrenmeyi etkileyen faktörler ele alınır, uygun yöntem ve stratejiler belirlenir ve öğrenmenin ne düzeyde gerçekleştiğini tespit etmek için değerlendirme araçları belirlenir.

Pek çok öğretim tasarım modeli bulunmaktadır. Bunlardan bazıları şunlardır:

- ADDIE Modeli
- ASSURE Modeli
- ARCS Modeli
- Dick ve Carrey Modeli
- Seels-Glasgow Modeli
- Kemp, Morrison ve Ross Modeli
- Gerlach ve Ely Modeli
- Gagné, Briggs ve Wagner Modeli
- Smith ve Ragan Modeli

Bu çalışmada derslerin planlanması ve öğretim platformunun geliştirilmesi sürecinde ADDIE öğretim tasarım modeli tercih edilmiştir.

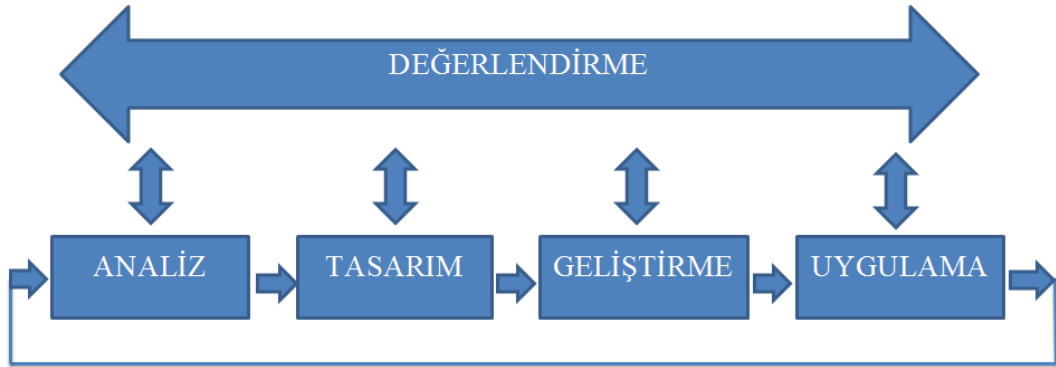
ADDIE öğretim tasarım modeli ilk olarak Florida Eyalet Üniversitesi tarafından askeri hizmet içi eğitimleri planlamak amacıyla geliştirilmiştir (Muruganatham, 2015). ADDIE öğretim tasarım modeli beş basamaktan meydana gelmektedir ve basamak isimlerinin baş harfleri ADDIE kelimesini oluşturmaktadır (McGriff, 2000):



**Şekil 2.2:** ADDIE basamakları.

ADDIE öğretim tasarım modelinde bir basamağın sonuçları, bir sonraki basamağın girdileri olmaktadır. Her basamağın sonunda değerlendirme

yapılmaktadır ve eğer eksiklikler ya da problemler varsa ilgili basamağa geri dönlmektedir.



Şekil 2.3: ADDIE öğretim tasarım modeli.

### **ADDIE Öğretim Tasarım Modelinin Basamakları:**

**Analiz (Analysis) Basamağı:** Bu basamakta ihtiyaç ve sınırlılıklar belirlenir. Hedefler ve amaçlar ortaya koyulur. Öğrencilerin özellikleri saptanıp, mevcut ön bilgi ve becerileri tespit edilir (Aydın, 2013). Analiz basamağında şu sorulara cevap aranır (Çakır ve Karataş, 2012; Forest, 2014):

1. Hedef kitle/öğrenciler kimdir ve bunların özellikleri nelerdir? Bu basamakta öğrenciler ile ilgili edinilmesi gereken bilgilerden bazıları şunlardır:

- Öğrencilerin hedefleri
- Öğrencilerin ihtiyaçları
- İlgi ve yetenekleri
- Öğrenilecek konu hakkındaki ön bilgi ve becerileri
- Çalışma alışkanlıkları
- İçinde buldukları sosyal ve kültürel ortam

2. Öğretilecek olan içerik nedir? İçeriğe ilişkin hedefler nelerdir? Hedefler kazandırılırken kullanılacak yöntem ve teknikler nelerdir? Hedeflere ulaşılıp ulaşılmadığını anlamak için hangi ölçme ve değerlendirme araçları kullanılmalıdır?

3. Öğretim nerede ve ne zaman gerçekleşecek? Öğretim eylemleri için ne kadarlık bir zaman dilimi gerekli? Yapılacak etkinlikler için nasıl bir ortam gerekli?

**Tasarım (Design) Basamağı:** Analiz basamağında elde edilen verilerden yararlanarak içeriğin öğretilmesi ile ilgili süreçler bu basamakta planlanmaktadır. Bu basamakta şu sorulara cevap aranır:

1. Öğretilmesi planlan içerik hangi sırada ve hangi öğretim stratejileri kullanılarak öğretilicektir?
2. Hangi etkinlikler yapılabilir? Hangi sunum yöntemleri kullanılabilir?
3. Yapılacak her bir etkinlik için ne kadar zaman ayrılmalı?
4. Mevcut araçlar ve kaynaklar nelerdir? Bunlar nasıl kullanılabilir?
5. Öğretimin hedeflerine ulaşıp ulaşılamadığını kontrol etmek için kullanılacak ölçme ve değerlendirme araçları nelerdir?

**Geliştirme (Development) Basamağı:** Geliştirme basamağı, üretim basamağıdır. Bu basamakta yapılan öğretimin hedeflerine ulaşmak için gerekli olan etkinlikler ve süreçler oluşturulur (Çakır ve Karataş, 2012). Ayrıca öğretim esnasında kullanılacak materyaller ve araçlar edinilir ya da üretilir.

**Uygulama (Implementation) Basamağı:** Uygulama basamağında yapılan tasarım, planlanan öğretim süreçleri ve geliştirilen ürünler öğrencilerle birlikte test edilir. Tasarımın uygulanması esnasında karşılaşılan durumlar yeniden gözden geçirilmektedir.

**Değerlendirme (Evaluation) Basamağı:** Değerlendirme basamağında yapılan tasarımın belirlenen hedefleri ve öğrencilerin ihtiyaçlarını ne ölçüde karşıladığı araştırılmaktadır. Tespit edilen eksiklik veya sorunlara göre ilgili basamaklara geri dönüş yapıp, düzeltmeler yapılmaktadır (Arkün, Baş, Avcı, Çevik, ve Gürcan, 2009).

### **3. YÖNTEM**

Bu bölümde araştırma modeli, katılımcılar, verilerin toplanması ile araştırma için geliştirilen uygulama hakkında bilgi verilmiştir.

#### **3.1 Araştırma Modeli**

Araştırmada deneysel desen türlerinden deneme öncesi (pre-experimental) modellerden olan “Tek Grup Ön Test ve Son Test Modeli” kullanılmıştır. Bu model ile yapılan çalışmanın etkisi tüm sınıf (tek bir grup) üzerinde yapılan çalışmayla tespit edilmektedir. Gruba bağımsız değişken uygulanıp, uygulamanın hem başında hem de sonunda ölçme yapılmaktadır. Son test puanları, ön test puanlarından daha büyük ise bağımsız değişkenin etkisinden söz edilebilmektedir (Büyüköztürk, Çakmak, Akgün, Karadeniz ve Demirel, 2009).

#### **3.2 Katılımcılar**

Araştırmaya Balıkesir Üniversitesi, Necatibey Eğitim Fakültesi, Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü 3. Sınıf öğrencileri (N=38) katılmıştır. Katılımcıların 21’i erkek (%55) ve 17’si kızdır (%45).

Çalışma grubu belirlenirken seçkisiz olmayan örnekleme yöntemlerinden uygun (kazara) örnekleme yöntemi kullanılmıştır. Uygun (kazara) örnekleme; zaman, maddi kaynaklar, işgücü vb. sınırlılıklar sebebiyle örneklemin kolay ulaşılabilir bir gruptan seçilmesidir (Büyüköztürk, Örnekleme Yöntemleri, 2012).

#### **3.3 Verilerin Toplanması**

Araştırmada nicel ve nitel araştırma yöntemleri ile veri toplanmıştır. Bu bölümde verilerin toplanmasına dair ayrıntılı bilgiler yer almaktadır.

### 3.3.1 Nicel Verilerin Toplanması

Araştırmada nicel veriler araştırmacı tarafından hazırlanan başarı testi ve Altun ve Mazman (2012) tarafından Türkçe'ye uyarlanmış olan Programlamaya İlişkin Öz Yeterlik Algısı Ölçeği ile elde edilmiştir. Elde edilen veriler istatistiksel veri analizi programında (SPSS) analiz edilmiştir.

#### 3.3.1.1 Öz Yeterlik Algısı ve Programlamaya İlişkin Öz Yeterlik Algısı Ölçeği

Araştırmada Altun ve Mazman (2012) tarafından Türkçe'ye uyarlanmış olan Programlamaya İlişkin Öz Yeterlik Algısı Ölçeği kullanılmıştır.

Ölçeğin orijinali Ramalingam ve Wiedenbeck (1998) tarafından C++ programlama dili için geliştirilmiş olup, 7'li likert tipi 32 madde ve 4 başlıktan oluşmaktadır. Bu başlıklar şunlardır:

- Bağımsızlık ve kararlılık
- Basit programlama görevlerini gerçekleştirme
- Karmaşık programlama görevlerini gerçekleştirme
- Öz düzenleme

Ölçeğin Türkçe uyarlaması ise 7'li Likert tipi 9 madde ve 2 başlıktan (basit programlama görevlerini gerçekleştirme, karmaşık programlama görevlerini gerçekleştirme) oluşmaktadır. Ölçeğin orijinali C++ programlama dili için maddeler içermekle birlikte, Türkçe uyarlamasında genel programlama becerilerini ifade eden maddeler bulunmaktadır [EK-A].

Altun ve Mazman (2012) tarafından yapılan çalışmada geçerlik ve güvenilirlik çalışması için 152 üniversite öğrencisine uygulanan ölçeğin tümü için Cronbach Alpha iç tutarlılık katsayısı 0.928 olarak bulunmuştur. Geçerlik ve güvenilirlik çalışmaları kapsamında yapılan açımlayıcı faktör analizine göre ölçeği oluşturan 9 madde toplam varyansın %80,814'ünü açıklamıştır. Doğrulayıcı faktör analizi ile de ölçeğin geçerli ve güvenilir olduğu doğrulanmıştır.

### 3.3.1.2 Başarı Testi

Başarı testi, yapılan uygulamanın öğrencilerin başarı düzeyine etkisini görmek amacıyla ön-test ve son-test olarak uygulanmıştır.

Araştırma için hazırlanan başarı testinde PHP diline özgü şartlı ifadeler ve döngüler konularını içeren 20 adet test sorusu bulunmaktadır. Bunlardan 10 tanesi şartlı ifadeler, 10 tanesi de döngüler konusu hakkındadır [EK-B]. Sorular if yapısı, if else yapısı, if elseif yapısı, switch case yapısı, for döngüsü, while döngüsü, do while döngüsü ve foreach döngüsü konuları ile ilgili bilgileri ölçmeye yönelik olarak hazırlanmıştır.

Başarı testindeki sorular araştırmacı tarafından oluşturulmuştur. Soruların geçerli ve güvenilir olmasını sağlamak için alan uzmanlarına danışılmış ve bu kişilerin görüşleri doğrultusunda düzeltmeler yapılmıştır. Alan uzmanları soruları incelerken, soruların hem PHP dili açısından doğruluğunu hem de dersin amaçlarına uygunluğunu dikkate almışlardır. Ayrıca soruların şartlı ifadeler ve döngüler ünitelerindeki tüm kavramları kapsayıp kapsamadığı da araştırılmıştır. Alan uzmanlarının görüşleri doğrultusunda sorular düzeltildikten sonra test, bilgisayar ve öğretim teknolojileri eğitimi alanında lisansüstü eğitim alan 6 tane öğrenciye uygulanmıştır. Öğrenciler iki tane sorunun net olarak anlayamadığını belirtmişlerdir. Bu görüşler doğrultusunda problemlen sorularda değişikliğe gidilmiş ve test son halini almıştır.

### 3.3.2 Nitel Verilerin Toplanması

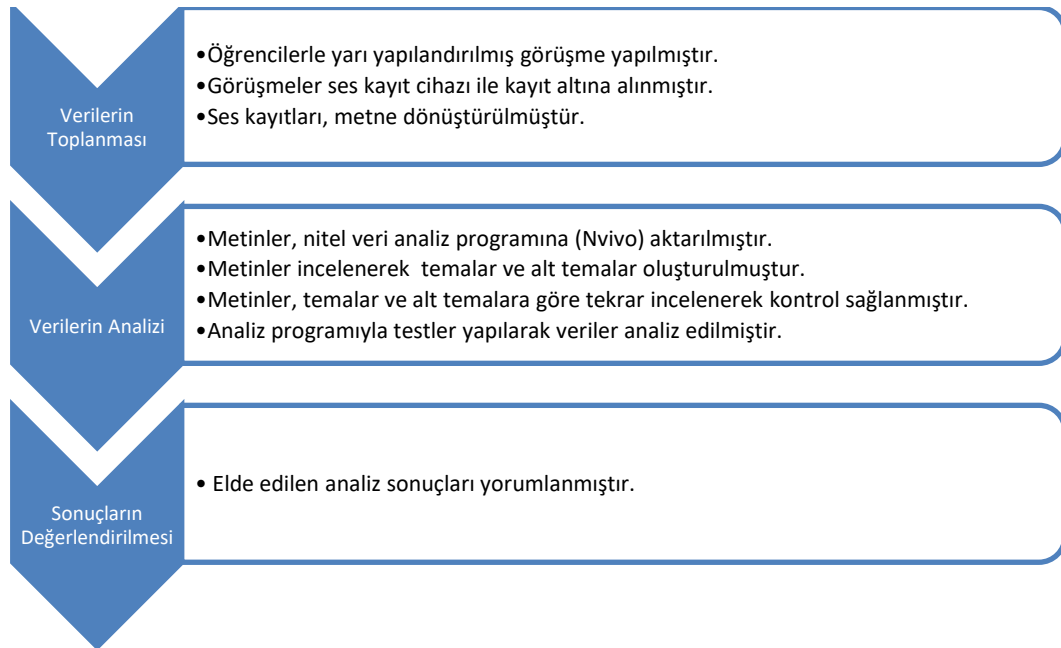
Araştırmada nitel veriler, araştırmaya katılan öğrencilerden gönüllü olan 12 öğrenci ile görüşme yapılarak elde edilmiştir.

Görüşme, herhangi bir konu hakkında kişilerin duygu, düşünce, algı ya da düşünme tarzlarını öğrenmek ya da birtakım davranışları niçin yaptıklarını anlayabilmek için kişilere sorular sorularak bilgi alma tekniğidir (Türnüklü, 2000). Üç tür görüşme yapılmaktadır:

- **Yapılandırılmamış Görüşme:** Bu görüşme türünde araştırmacı, herhangi bir görüşme protokolü hazırlanmadan, kişinin yanıtlarına göre sorularını o anda hazırlamakta ve kişiye yönelmektedir.
- **Yapılandırılmış Görüşme:** Bu görüşme türünde araştırmacı, edinmek istediği bilgiler ışında görüşme öncesinde sorularını hazırlamakta ve tüm görüşme yapacağı kişilere aynı soruları sormaktadır.
- **Yarı Yapılandırılmış Görüşme:** Bu görüşme türünde araştırmacı, yapılandırılmış görüşme türünde olduğu gibi görüşme öncesinde sorularını hazırlamaktadır. Ancak görüşme esnasında görüşmenin akışına bağlı olarak kişinin konuyu detaylandırmasını sağlayabilmek için alt sorular sorabilmektedir (Çetin, 2012).

Öğrencilere wtPHP hakkında, aracın kullanım özelliklerini, içerik özelliklerini, eksik ya da yetersiz kalan yönlerini vb. tespit edebilmek için sorular yöneltilmiştir [EK-C]. Öğrencilerle yapılan görüşmeler kayıt altına alınmıştır.

Nitel veriler toplanırken izlenen yol aşağıdaki şemada gösterilmektedir:



**Şekil 3.1:** Nitel veriler toplanırken izlenen yol.

### 3.4 wtPHP'nin Özellikleri

Bu araştırma için geliştirilen platform, ADDIE öğretim tasarım modeli temel alınarak geliştirilmiştir. Bu bölümde öğretim tasarımı modelinin aşamalarına göre platformun tasarlanması ve geliştirilmesi sürecinden bahsedilmiştir.

#### 3.4.1 Analiz Aşaması

Analiz aşamasında ders saati, derslik ve laboratuvarların yeterliliği, konu seçimi, hedef kitle, hedef kitlenin temel özellikleri, problemler ve sınırlılıkları tespit edebilmek için Balıkesir Üniversitesi'nde farklı bölümlerde (bilgisayar mühendisliği, bilgisayar ve öğretim teknolojileri eğitimi, bilgisayar programcılığı) derse giren 6 öğretim elemanı ile yarı yapılandırılmış görüşme yapılmıştır [EK-D].

Yapılan görüşmelerde öğretim elemanları, programlama dersleri için 4 ders saatinin çok uygun olduğunu, bu derslerin en az 3 ders saati olması gerektiğini belirtmişlerdir. Derslik ve laboratuvarların bu dersi işlemek için uygun olduğunu, bu konuda herhangi bir problem yaşamadıklarını belirtmişlerdir.

Yazılım için öncelikli olarak programlama dillerine ait iki temel konuda (şartlı ifadeler ve döngüler) karar kılınmıştır. Konu seçimi yapılırken öğretim elemanlarına yöneltilen *“Bu derste öğrenciler en çok hangi konu/konuları öğrenmede zorluk çekiyorlar?”* sorusuna verilen cevaplar dikkate alınmıştır. Öğretim elemanların çoğu, döngüler konusunun öğrencilerin anlamakta en çok zorlandıkları konulardan biri olduğunu belirtmişlerdir. Üç tane öğretim elemanı ise öğrencilerin şartlı ifadeler ile döngülerin bir arada kullanılmasını gerektiren örneklerde zorlandıklarını ifade etmişlerdir.

Görüşme yapılan öğretim elemanlarının hedef kitle (öğrenciler) hakkındaki en belirgin tespitinin, sınıfların homojen olmaması yönünde olduğu görülmüştür. Sınıflarda hem ortaöğretim kurumlarından (düz lise olarak anılan) gelen öğrencilerin hem de mesleki ve teknik ortaöğretim kurumlarının bilişim teknolojileri alanlarından gelen öğrencilerin bulunması dersin hangi seviyeden başlanması, hangi hızda ilerlenmesi gerektiği vb. problemleri beraberinde getirdiği dile getirilmiştir.

Öğretim elemanlarının “*Dersinizi işlerken karşılaştığınız problemler var mı?*” sorusuna verdikleri cevaplar iki boyutta ele alınabilir: Öğrenci kaynaklı problemler ve dersin yapısı ile ilgili problemler.

Öğretim elemanlarının gözlemlerine göre öğrenci kaynaklı problemler çoğunlukla öğrencilerin derse olan ilgi ve motivasyonu hakkındadır. Yapılan değerlendirmelere göre öğrenciler, özellikle soyut kavramlar (nesnelere, işaretçiler vb.) anlatılırken, ilgi ve motivasyonlarını çok çabuk kaybetmekte ve dersin/konunun gerekliliği konusunda sorgulamaya başlamaktadır.

Dersin yapısı ile ilgili başlıca problem ise, program yazabilmek için sadece programlama diline has kural ve yapıları bilmenin yetersiz olmasıdır. Öğrenci programlama dilinin kural ve yapılarını bilmenin yanında problem çözme vb. becerilere de sahip olmalıdır. Öğretim elemanları öğrencilerin derste yapılan örnekleri kendi başlarına uygulayabildiklerini; ancak örnek bir yönüyle değiştirilerek tekrar sorulduğunda pek çok öğrencinin bu örneği yapamadığını belirtmişlerdir. Bir öğretim elemanı bu durumun, öğrencilerin çoğunun Bloom’un taksonomisine göre yüksek seviyeli zihinsel düşünme becerilerine (analiz, sentez ve değerlendirme) yeterli düzeyde sahip olmadıkları için meydana geldiği yönünde değerlendirmiştir.

Öğretim elemanları, öğrencilerin pek çoğunun derste yapılan örneklerle yetindiklerini, kendi başlarına ders saatleri dışında pratik yapmaya uğraşmadıklarını ifade etmişler ve öğrencilerin kendi kendilerine program yazmak için çaba sarf etmedikçe bu derste başarılı olmanın mümkün olmadığını vurgulamışlardır. Lahtinen, Ala-Mutka ve Järvinen (2005) de yaptıkları çalışmada öğretim elemanlarının görüşleriyle paralel olarak, teorik bilginin programlama öğrenmede çok önemli olduğunu, ancak öğrencilerin konuları öğrenebilmek için pratik tecrübeler ihtiyacı duyduğunu belirtmişlerdir.

wtPHP’de, *öğrencilerin programlamada başarılı olabilmeleri için çok sayıda örnek uygulama yapmaları gerekmektedir* fikrinden yola çıkılarak öğretilmesi amaçlanan konularla ilgili çok sayıda alıştırmalar bulunması planlanmıştır. Öğrencilerin çok sayıda alıştırmaların, hem onların PHP dilinin kural ve yapılarını öğrenmelerini sağlayacağı hem de problem çözme becerilerini arttıracığı düşünülmektedir.

wtPHP ile öğrencileri çalışmaktan vazgeçirebilecek zorluklar ile ilgili tedbirler almak hedeflenmektedir. Bu tedbirler;

- Öğrencilerin herhangi bir program kurmadan istedikleri bilgisayarda çalışabilmeleri sağlanmalıdır.
- Öğrencilerin çalışma dosyalarını okuldan eve, evden okula taşımak için uğraşmamaları sağlanmalıdır.
- Yazdıkları kodları çalıştırdıklarında, yaptıkları hatanın ne olduğunu daha iyi anlamaları sağlanmalıdır.

Analiz aşaması ile üretilecek olan wtPHP'nin taşınması gereken özellikler belirlenmiştir. Buna göre platformun; internet olan her ortamda ulaşılabilir olması için web tabanlı olması, içeriğin bu dersi ilk defa alan öğrencilere de daha önceden lisede almış olan öğrencilere de hitap edecek şekilde düzenlenmesi, ilgili konularla ilgili çok sayıda örnek ve alıştıırma içermesi gerektiği tespit edilmiştir.

### **3.4.2 Tasarım Aşaması**

Tasarım aşamasında, üretilecek olan platform hakkında, öğretim elemanlarının da görüşleri dikkate alınarak aşağıda belirtilen konularda planlamalar yapılmıştır:

- İçerik ve konu analizi, içeriklerin sunulması, alıştıırma ve örneklerin belirlenmesi
- Ekran tasarımları, öğrenci ve öğretim elemanı için menülerin belirlenmesi
- İletişim öğelerinin belirlenmesi
- Değerlendirme öğelerinin belirlenmesi

*İçerik ve konu analizi, içeriklerin sunulması, alıştıırma ve örneklerin belirlenmesi*

İnternet Tabanlı Programlama dersinde ilk olarak şartlı ifadeler ve o üniteden sonra da döngüler ünitesi işlenmektedir. Şartlı ifadelerde konular if yapısı, if else yapısı, if elseif yapısı ve son olarak da switch case yapısı olarak sıralanmıştır. Döngüler ünitesinde ise for döngüsü, while döngüsü, do while döngüsü ve foreach

döngüsü konuları sırasıyla işlenmektedir. wtPHP'de de konu anlatımlarının bu sırada olması planlanmıştır.

Yapılacak platformda uzun uzun konuyu anlatmak yerine kısa, öz bilgiler vermek, ilgili yapı ile ilgili olarak çeşitli kullanım örnekleri sunmak hedeflenmiştir. Örneklerde kod satırlarının yanına, açıklama satırı şeklinde notlar düşülmesi planlanmıştır. Bu sayede örneğin daha iyi anlaşılacağı düşünülmüştür.

wtPHP'de hem konular kısmında kullanılan örnekler hem de araştırmalar kısmında sorulan sorular için çeşitli kaynaklardan araştırma yapılmalıdır. Bunun için önceki yıllarda bu dersi (başka programlama dilleri dersleri de olabilir) veren öğretim elemanlarından derslerde kullandıkları örnekler ile sınavlarda sordukları soruların edinilmesi ve basılı kaynaklar ile internet kaynaklarından yararlı olabilecek, farklı örneklerin araştırılması planlanmıştır.

*Ekran tasarımları, öğrenci ve öğretim elemanı için menülerin belirlenmesi:*

wtPHP'nin çıkış noktası öğrencilerin ders saatleri dışında da programlama çalışmasını teşvik etmek, onların çalışması için uygun ortamı sağlamaktır. Bunun için yazılımda, öğrencilerin kolayca örnek ve alıştırmalara ulaşım çalışabilecekleri, öğretim elemanının da çok pratik bir şekilde yeni içerikleri ekleyebileceği sade, kolay anlaşılır arayüzler düşünülmüştür.

*İletişim öğelerinin belirlenmesi*

wtPHP'de öğrencinin, diğer öğrencilerle ya da öğretim elemanı ile iletişime geçebilmesi için bir mesaj gönderme arayüzü planlanmaktadır. Burada öğrenci mesajı göndereceği kişiyi seçip, mesajın konusunu ve içeriğini ekleyerek mesajını iletebilmelidir.

Öğretim elemanının tüm sınıfa duyuru yapabilmesi için ise sadece öğretim elemanın menüsünde bulunan duyuru gönderme arayüzü düşünülmüştür. Bu özellik öğrenci menüsünde bulunmayacaktır.

### *Değerlendirme öğelerinin belirlenmesi*

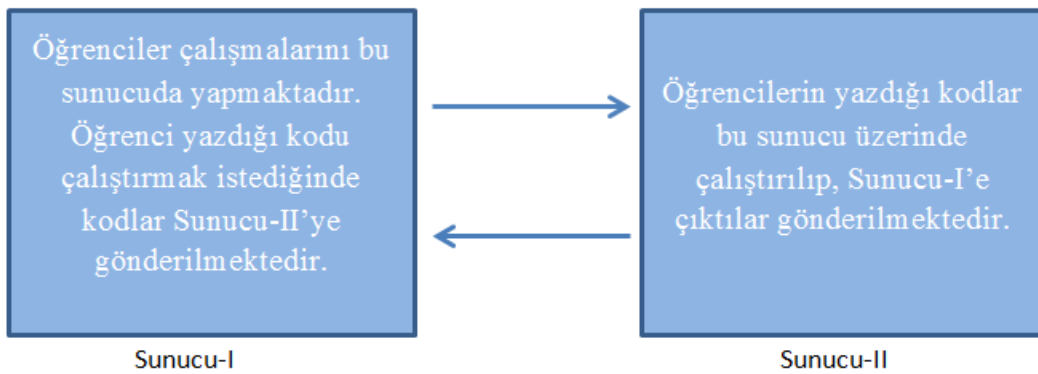
wtPHP için başlangıçta herhangi bir değerlendirme aracı düşünülmemektedir. Ancak bu platformun pratikte kullanılmaya başlanmasıyla, gelen olumlu-olumsuz değerlendirmeler ile birlikte çeşitli değerlendirme araçları platforma eklenebilir.

### **3.4.3 Geliştirme Basamağı**

Tasarım aşamasında yapılan planlamalar bu bölümde gerçekleştirilmeye çalışılmıştır.

#### *Sistemin Kurulması:*

wtPHP, PHP web programlama dilinde geliştirilmiş, verilerin kaydedilmesi için MYSQL veritabanı yönetim sistemi kullanılmıştır. Yazılımın çalıştırılması için Balıkesir Üniversitesi Bilgi İşlem Daire Başkanlığı'nda devamlı olarak çalışmak üzere bir bilgisayar hizmete alınmış ve bu bilgisayar üzerinde 2 adet sanal sunucu kurulmuştur. Sunucularda Ubuntu Server 14.04.03 sürümü kullanılmaktadır. Sunuculardan birinde istemcilerin login olarak çalışmalarını yaptıkları program (web yazılımı) çalışmaktadır. Diğer sunucuda ise öğrencilerin yazdıkları kodların işlenip sonucunun gönderildiği program çalışmaktadır.



**Şekil 3.2:** Sunucuların çalışması.

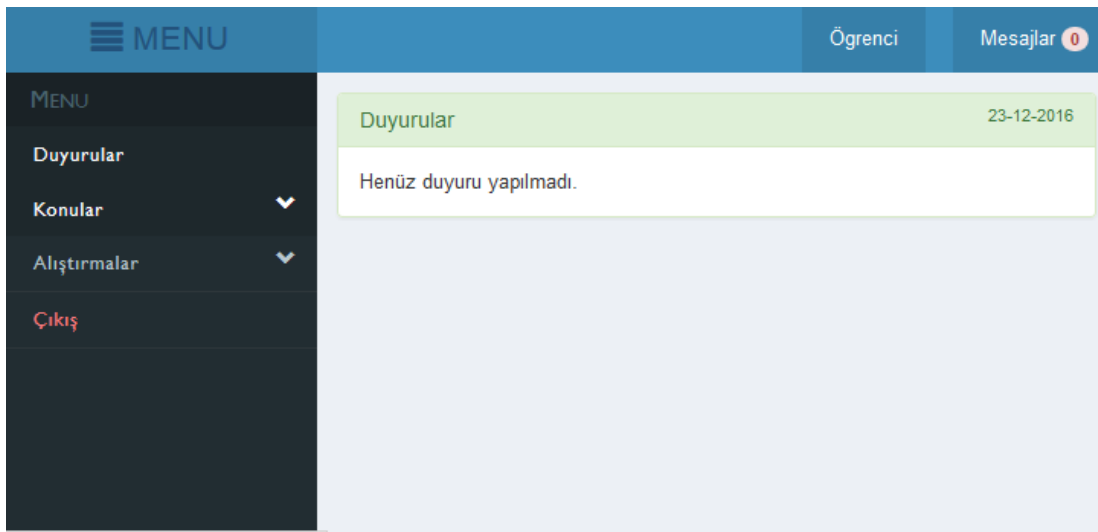
Sunucu üzerinde kullanıcının yazdığı PHP kodunu direk olarak çalıştırmak birtakım tehditleri beraberinde getirmektedir. Bunların başında öğrencilerin istemeden, yani test etmeye çalıştıkları komutların nasıl bir işlem yapacağını tam kestirememeleri ya da kişilerin hacking konusundaki bilgilerini test etmeye

çalışmaları sonucunda sunucuları çökertmeleri gelmektedir. Bu sebeple bazı tedbirler alarak bu tehditleri en aza indirmek amaçlanmıştır:

- wtPHP, sadece sorumlu öğretim elemanın sisteme kayıt ettiği kullanıcıların kullanımına açıktır. Bunun dışında başka herhangi bir kişi sisteme giriş yapamamaktadır. Bu sayede kötü niyetli kişilerin, sistemde açıklar arayıp, sistemi çökertmeye çalışması engellenmeye çalışılmıştır.
- PHP programının konfigürasyon dosyası olan php.ini dosyasında yer alan disable\_functions özelliği ile PHP öğretimini engellemeyecek, ancak izin verilmesi durumunda sistem için açıklar oluşturacak komutlar engellenmiştir.
- Tüm sistemin yedeği alınmış, ayrıca her 10 dakikada bir veritabanının yedeğinin alınması için sunucuya zamanlanmış görev eklenmiştir. Yedeklerden 5 günden daha eski olanların silinmesi için de gerekli komutlar eklenmiştir.

#### ***wtPHP'de Öğrenciler için Kullanıcı Arayüzleri:***

wtPHP için sade, basit bir arayüz oluşturulmuştur. wtPHP'ye kullanıcı adı ve şifreyle giriş yapılabilir. Öğretim elemanı öğrencileri sisteme kaydedip, kullanıcı adı ve şifrelerini öğrencilere vermektedir. Sisteme giriş yapan öğrenci, ilk olarak duyuru ekranıyla karşılaşmaktadır. Öğretim elemanının tüm sınıfa yapmak istediği duyurular bu ekranda listelenmektedir.

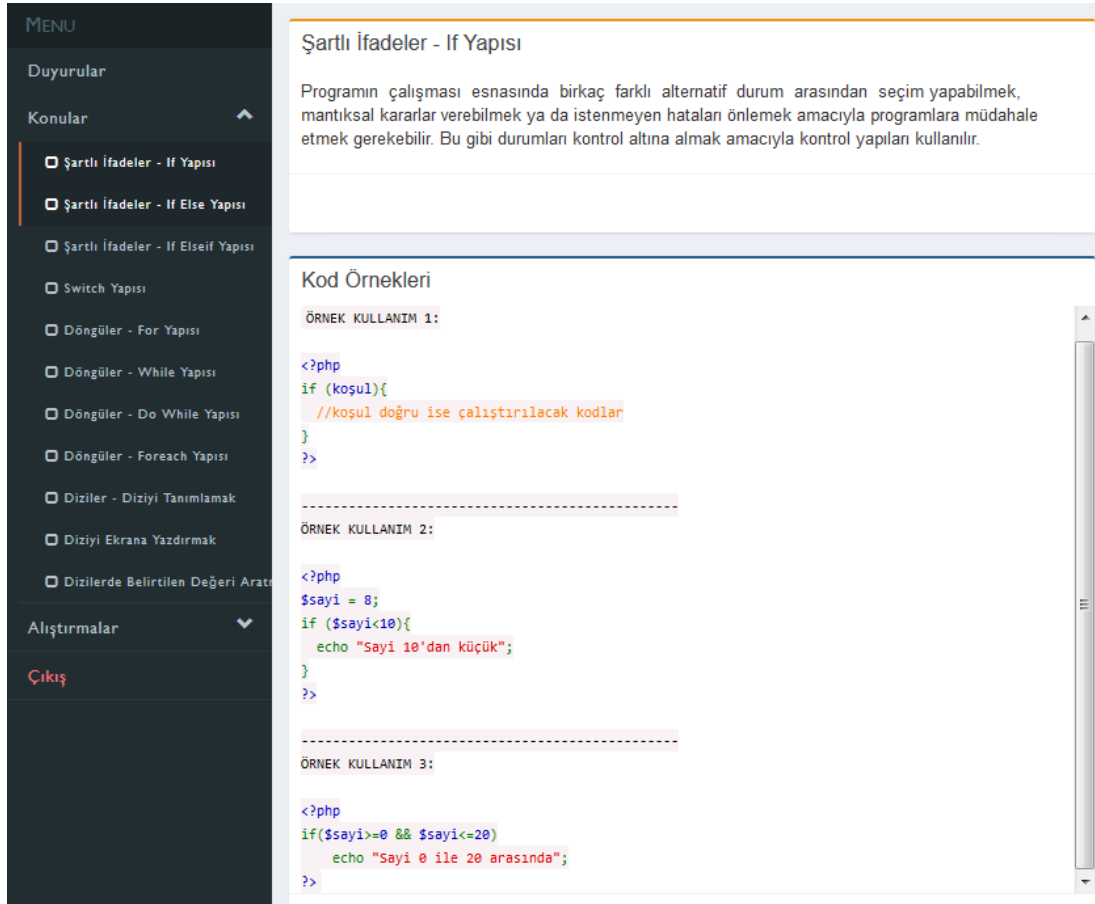


**Şekil 3.3:** Duyurular ekranı.

Duyuru başlığının yanında, duyurunun girildiği tarih ve saat yer almaktadır. *Duyurular* en son yapılan duyurudan daha eskiye doğru sıralanarak verilmiştir.

Menüde *Konular* başlığı altında anlatılmak istenen kavram ile ilgili çok kısa, özet bilgi verilmekte ve ilgili kavramın nasıl kullanıldığına dair örnekler sunulmaktadır. Kullanılan örneklerin, kavram ile ilgili farklı kullanımları içeriyor olmasına özen gösterilmiştir. Bu bölümde tüm konuyu uzun uzun anlatmak yerine, öğrencinin alıştırmaya yaparken takıldığı noktada bu bölüme bakarak yardım alması hedeflenmektedir.

Konular, uygulamaya eklendiği sırada listelenmektedir.



**MENU**

Duyurular

**Konular**

- Şartlı İfadeler - If Yapısı
- Şartlı İfadeler - If Else Yapısı
- Şartlı İfadeler - If Elseif Yapısı
- Switch Yapısı
- Döngüler - For Yapısı
- Döngüler - While Yapısı
- Döngüler - Do While Yapısı
- Döngüler - Foreach Yapısı
- Diziler - Diziyi Tanımlamak
- Diziyi Ekran Yazdırmak
- Dizilerde Belirtilen Değeri Aratmak

Alıştırmalar

Çıkış

### Şartlı İfadeler - If Yapısı

Programın çalışması esnasında birkaç farklı alternatif durum arasından seçim yapabilmek, mantıksal kararlar verebilmek ya da istenmeyen hataları önlemek amacıyla programlara müdahale etmek gerekebilir. Bu gibi durumları kontrol altına almak amacıyla kontrol yapıları kullanılır.

#### Kod Örnekleri

ÖRNEK KULLANIM 1:

```
<?php
if (koşul){
    //koşul doğru ise çalıştırılacak kodlar
}
?>
```

ÖRNEK KULLANIM 2:

```
<?php
$sayi = 8;
if ($sayi<10){
    echo "Sayı 10'dan küçük";
}
?>
```

ÖRNEK KULLANIM 3:

```
<?php
if($sayi>=0 && $sayi<=20)
    echo "Sayı 0 ile 20 arasında";
?>
```

Şekil 3.4: Konular ekranı.

Öğrenci, herhangi bir konu başlığını tıkladığında açılan ekranın üst kısmında konu ile ilgili bilgi alanı, bu alanın altında da konu ile ilgili kod örnekleri alanı yer almaktadır. Kod örnekleri alanında yer alan örnek uygulamalar çalıştırılmamaktadır. Burada yer alan örnekler, ilgili kavram ya da yapının nasıl kullanıldığını göstermektedir.

Menüde *Alıştırmalar* başlığı altında ise öğrencinin kendi kendine çalışabileceği alıştırmalar sunulmaktadır. Öğrenci, bu başlık altından istediği alıştırmayı seçebilmektedir. Alıştırmalar, uygulamaya eklendiği sırada listelenmektedir.

Öğrenci istediği alıştırmayı tıkladığında aşağıdaki ekran karşısına çıkmaktadır:

MENU

Duyurular

Konular

Alıştırmalar

Kullanıcı Adı, Şifre Kontrolü

T.C Kontrolü

Doğum Günü

Üçgenin Çeşidi

Suyun Katı-Sıvı-Gaz Hali

Ehliyet Alabilir mi?

İki Sayı Arasında mı?

Sınava Girebilir mi?

Otopark Ücreti

IP Sınıfı

Harf Notu

Koddaki hatayı düzeltme-1

Şifre - Limit Koyma

Veri Girişi - Harf ve Rakamlar

Geçerli Eposta Deseni Kontrolü

Koddaki hatayı düzeltme-2

Dizi İçinde Var mı?

Adını Yazdırma

Geçerli Eposta Deseni Kontrolü

Bir kayıt formunda **e-posta** alanına girilen verinin e-posta desenine uygun olup olmadığını kontrol edip, uygun olmaması durumunda "**Geçersiz e-posta**" uyarısını ekrana yazdıran kodu yazınız.

**NOT:** PHP 5.2 sürümüyle gelen `filter_var()` fonksiyonu bu kontrolü yapmanın kolay bir yoludur.

KONSOL

```
1 <?php
2 $eposta = "deneme@deneme.com";
3
4 ?>
```

ÇIKTI

Sıfırla Kaydet Çalıştır

Şekil 3.5: Alıştırmalar ekranı.

Bu ekranda üstte sorunun başlığı bulunmaktadır. Başlığın altında ise sorunun içeriği verilmektedir. Bazı alıştırmalarda soru ile birlikte soru hakkında ipuçları da verilmiştir. Bu ipuçları ile öğrencilerin, karşılaştığı problemi çözmek için araştırma yapmayı öğrenmesi istenilmektedir. Aynı zamanda problemin çözümünün tek olamayacağı, aynı işi yapmak için farklı kod ve fonksiyonların olabileceği fikrinin de benimsetilmesi amaçlanmaktadır.

Sorunun sorulduğu bölümün altında Konsol ve Çıktı başlıklarını taşıyan iki alan bulunmaktadır. Konsol alanında öğrenci, sorunun çözümü için kodlarını

yazmaktadır. Bu alanın altında yer alan Çalıştır butonu tıkladığında yazılan kodların sonuçları yandaki Çıktı alanında görülmektedir. Kaydet butonu tıkladığında yazılan kodlar kaydedilmektedir. Sıfırla butonu ise Konsol alanını temizlemektedir.

### *Yazılan Kodların Çalıştırması*

PHP, yapısı itibarıyla sunucu taraflı (server side) bir dildir. PHP kodları web sunucusunda kurulu olan PHP programı tarafından yorumlanır. PHP programı, PHP kodlarını html biçimine çevirmektedir. Bu sayede web tarayıcıları ile ekran çıktıları görüntülenebilmektedir.

Öğrencilerin, PHP dilinde yazdığı bir kodu çalıştırabilmesi için, kodlarını web sunucusuna atıp, web tarayıcısından ilgili dosya adresini çağırması gerekmektedir. Ancak öğrencilerin bir web sunucusuna erişimi mümkün değilse, kendi bilgisayarlarını bir web sunucusu gibi kullanmalarını sağlayan, Apache web sunucusu, Mysql veritabanı sunucusu ve PHP programını içeren programlar (örneğin Windows için Wamp Server, EasyPHP gibi) kullanmak zorunda kalmaktadırlar.

Öğrenciler, yazdığı kodda her yaptığı değişiklik için dosyayı kaydedip, tekrar web sayfasını yenileyerek yaptığı değişikliğin sonucunu görebilmektedir. wtPHP'de, öğrencinin yazdığı kodda echo, print gibi komutlarla ekrana yazdırdığı ifadeler, Çalıştır butonuna basılmasıyla Çıktı alanında gösterilmektedir. Her defasında PHP dosyasını web sunucusuna atıp, web sayfasını yenilemek gerekmediğinden bu gösterim pratik bir kullanım sağlamaktadır.

### 10X10 Tablo - Arkaplan Rengi

10X10 tabloyu, çift numaralı satırlarda satır arkaplan rengini sarı olacak şekilde yapınız.

#### KONSOL

```
1 <?php
2 echo "<table border='1'>";
3 for($i=1; $i<=10; $i++){
4     if($i%2==0){
5         echo "<tr bgcolor='yellow'>";
6     }else
7         echo "<tr>";
8     for($j=1; $j<=10; $j++){
9         echo "<td> Sutun";
10        echo "</td>";
11    }
12    echo "</tr>";
13 }
14 echo "</table>";
15 ?>
```

#### ÇIKTI

Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun
Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun
Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun
Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun
Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun
Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun
Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun
Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun
Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun
Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun	Sutun

SıfırlaKaydetÇalıştır

Şekil 3.6: Örnek çıktı ekranı.

#### *Hata Mesajları:*

Hata mesajları çoğunlukla yazılan koddaki problemin ne olduğunu ve nerede olduğunu programcıya ifade etmektedir, ancak öğrenciler bu mesajları biraz daha zor anlamaktadır. Bu durumun sebeplerinden birisi hata mesajlarının kendi dilinde olmamasıdır. Örneğin aşağıdaki kod bloğu çalıştırıldığında alt kısımda yer alan hata mesajı üretilmektedir:

```
<?PHP
$user = "yonetici";
$password = "12345678";
if ($user == "Admin" && $password == "741852963"){
    echo "Giriş başarılı";
}else
    echo "Kullanıcı adı ya da şifreniz hatalı";
?>
```

**Parse error: syntax error, unexpected '\$user' (T\_VARIABLE), expecting '(' in /eva/akademik/ozlemtulek/public\_html/dene.PHP on line 10**

Bu hata mesajı, tecrübeli bir programcı için hatanın türü, nasıl çözülebileceği ve hatayı hangi satırlarda araması gerektiği konularında ipuçları vermektedir. Ancak tecrübesiz programcılar, hata mesajları okuma ve anlama konusunda daha çekingendir.

wtPHP'de, öğrencinin yazdığı kodda herhangi bir yazım hatası yapması durumunda hata mesajları Çıktı alanında gösterilmektedir. Hata mesajları mükemmel bir Türkçe'yle ifade edilemese de öğrencilerin anlayabileceği bir şekilde Türkçe'ye çevrilmiştir.

Yukarıdaki kod, wtPHP'de çalıştırıldığında, öğrenci aşağıdaki hata mesajını görmektedir:

**AYRIŞTIRMA HATASI:***Sözdizimi hatalı, '\$user' beklenmiyor (T\_VARIABLE), beklenen '(', hatalı satır 4.*

Mesajın ilk kısmı (*Sözdizimi hatalı*) yazılan kodda yazım yanlışı yapıldığını ifade etmektedir.

İkinci kısım (*'\$user' beklenmiyor (T\_VARIABLE), beklenen '('*) ise mesajın en sonunda belirtilen hatalı satırda "\$user" (değişken ismi) yerine "(" işaretinin kullanılması gerektiğini ifade etmektedir. Türkçe hata mesajlarında, yanlış görünen ifade ve yanlış ifade yerine kullanılması gereken ifade kırmızı renkle vurgulanarak öğrencilerin dikkatini bu noktaya vermelerini sağlamak amaçlanmıştır.

PHP hata mesajlarında hata yapılan satır numarası söylenmektedir. Bununla birlikte yapılan hata her zaman belirtilen satır numarasında olmayabilir. Örneğin; bir üst satırın sonuna konulması gereken noktalı virgül (;) unutulduğunda, hatalı satır numarası bir sonraki satır numarası olarak gösterilmektedir. Hatanın nerede olduğunu tespit edebilme becerisi, zaman içinde edinilen tecrübeler sonucu kazanılmaktadır.

Programlama öğrenmeye yeni başlayan öğrencilerin, hata ayıklama (debugging) becerisini kazanmak için hata mesajlarını okuyabilmeyi ve anlayabilmeyi öğrenmeleri gerekmektedir. Bunun için hata mesajlarının öğrencilerin çok daha iyi anlayabilecekleri kendi adillerinde olmasının, onların gözünü korkutmaması açısından faydalı olabileceği düşünülmektedir.

### Kullanıcı Adı, Şifre Kontrolü

Bir formdan gelen kullanıcı adı ve şifre bilgisine göre kullanıcı adı **Admin** ve şifresi **741852963** ise **"Giriş başarılı"** mesajını, değilse **"Kullanıcı adı ya da şifreniz hatalı"** mesajını ekrana yazdırınız.

#### KONSOL

```
1 <?php
2 $form_kullanici = "yonetici";
3 $form_sifre = "12345678";
4 if($form_kullanici == "Admin" && $form_
5   echo "Giriş başarılı"
6 }else
7   echo "Kullanıcı adı ya da şifreniz ha
8 ?>
```

#### ÇIKTI

**AYRIŞTIRMA HATASI:** Sözdizimi hatası, '?' beklenmiyor, beklenen ',' veya ';', hatalı satır 6.

Sıfırla Kaydet Çalıştır

Şekil 3.7: Hata mesajları örnek ekranı.

Öğrencilerin hata ayıklama konusunda deneyim edinmesi için bazı alıştırmalar koddaki hatayı düzeltme şeklinde planlanmıştır. Kodda yapılan hatayı bulup düzeltebilen öğrencilerin zaman içinde kendi yazdıkları kodlarda da bu tür hatalar yapmama konusunda daha dikkatli olacakları düşünülmektedir.

### Koddaki hatayı düzeltme-1

Aşağıdaki koda yazım yanlışı (syntax hatası) bulunmaktadır. Bunu düzeltiniz.

#### KONSOL

```
1 <?php
2 $message="This is a message";
3 if (strlen($message<=7){
4     echo "asd";
5 }
6 else{
7     echo "qwe";
8 }
9 ?>
```

#### ÇIKTI

**AYRIŞTIRMA HATASI:** Sözdizimi hatalı, '{' beklenmiyor, hatalı satır 3.

Sıfırla Kaydet Çalıştır

**Şekil 3.8:** Hata ayıklama için örnek.

*wtPHP'de Öğretim Elemanı için Kullanıcı Arayüzleri:*

wtPHP'de, öğretim elemanı aşağıdaki işlemleri gerçekleştirebilmektedir:

- Duyuru ekleme, eklenmiş duyuruları güncelleme, duyuru silme
- Yeni içerik ekleme, eklenmiş içerikleri güncelleme, içerik silme
- Yeni öğrenci ekleme, öğrenci bilgilerini güncelleme, öğrenci silme



Şekil 3.9: Öğretim elemanı için yönetim menüsü.

Öğretim elemanı kullanıcı adı ve şifreyle giriş yaptığında *Duyurular* ekranıyla karşılaşmaktadır. Bu ekranda yeni duyuru girişi yapabilmekte, daha önceden eklemiş olduğu duyuruları güncelleyebilmekte ya da istediği duyuruları silebilmektedir. Eklenen duyurular en yeni duyurudan, en eski duyuruya doğru sıralanmaktadır.

### Yeni duyuru ekle

**Başlık**

**Duyuru**

Dosya ▾ Düzenle ▾ Görüntüle ▾

Biçim ▾

Biçimler ▾ **B** *I*



≡ ≡ ≡ ≡

≡ ≡ ≡ ≡

p

**Kaydet**

### Kullanıcı Hesapları

#ID	Başlık	İşlem
#2	Yeni Alıştırmalar	 

İlk (1) « 1 » Son (1)

Şekil 3.10: Duyurular ekranı.

Öğretim elemanı iki tür içerik ekleyebilmektedir: konu ve alıştırma. Admin Menü'den *Yeni İçerik Ekle* seçeneğini tıkladığında gelen pencerede öncelikle içeriğin türü seçilmektedir. Sonra içeriğin başlığı ve içerik metni girilmektedir. Ayrıca bu pencerede kod yazma alanı bulunmaktadır. İçerikte herhangi bir PHP kodu verilmek istendiğinde bu alana girilmektedir.

Yeni içerik ekle

İçerik Türü

Konu

Başlık

İçerik

Dosya ▾ Düzenle ▾ Görüntüle ▾ Biçim ▾

← → Biçimler ▾ **B** *I* [List Icons]

p

Kod

1

Kaydet

**Şekil 3.11:** Yeni içerik ekleme ekranı.

Öğretim elemanı daha önceden eklemiş olduğu bir içeriği silmek ya da içerik üzerinde değişiklik yapmak istediğinde Admin Menü'den *Tüm İçerikler* seçeneğini seçmektedir. Bu ekranda girilmiş olan tüm konu ve alıştırma içerikleri, ilk eklenenden en son eklenen içeriğe doğru sıralanmaktadır.

Tüm İçerikler			
#ID	TÜR	Başlık	İşlem
#65	konu	Dizilerde Belirtilen Değeri Aratma	 
#66	alistirma	Matris Örneği	 
#67	alistirma	Matrisin Satırları Toplamı	 
#68	alistirma	Matrisin Sütunlarını Toplatma	 

İlk (1) « 1 2 3 4 » Son (4)

**Şekil 3.12:** Tüm içeriklerin listelendiği ekran.

Öğretim elemanı değiştirmek ya da güncelleştirmek istediği içeriğin, türünü (konu ya da alıştırma), başlığını, içerik metnini ya da varsa kodlarında değişiklik yapabilmektedir.

"**Matris Örneđi**" başlıklı içeriđi düzenle.

**İçerik Türü**

Alıştırma

**Başlık**

Matris Örneđi

**İçerik**

Dosya ▾ Düzenle ▾ Görüntüle ▾ Biçim ▾

↶ ↷ Biçimler ▾ **B** *I* ≡ ≡ ≡ ≡ ≡ ≡ ≡ ≡ ≡ ≡

Bir matrisin her bir elemanına 5 ekleyerek yeni bir matris oluşturan php kodlarını yazınız.

p

**Kod**

```
1 <?php
2 $matris1 = array(
3   array(0=>5,1=>4,2=>0),
4   array(0=>2,1=>7,2=>3)
5 );
6
7 ?>
```

Kaydet

**Şekil 3.13:** İçerik düzenleme ekranı.

Öğretim elemanı sisteme yeni öğrenci eklemek istediğinde Admin Menü'den *Yeni Kullanıcı Ekle* seçeneğini seçmektedir. Her bir öğrenci tek tek eklenebileceği gibi, bir dosyadan (.csv olmalıdır) içe aktarmak da mümkündür.

### Yeni Kullanıcı Ekle

**Kullanıcı adı**

**Şifre**

**Ad Soyad**

**Telefon**

**E-mail**

### Excel İçe Aktarma

.csv dosyanızı yükleyin

### Bilgilendirme

\* Aynı kullanıcı adı veya email ile kayıt yapamazsınız.

**Şekil 3.14:** Sisteme yeni öğrenci ekleme ekranı.

Öğretim elemanı, sisteme eklediği kullanıcıları Admin Menü'deki *Kullanıcılar* seçeneği ile listeleyebilmektedir. Bu ekranda seçilen kullanıcı hesabı ile ilgili hesap bilgilerini güncelleme/değiştirme, hesabı silme ya da öğrencinin çalışma istatistiklerini görme işlemleri yapılabilmektedir.

Kullanıcı Hesapları			
Kullanıcı Adı	Ad Soyad	Telefon	İşlem
Ogrenci	Ogrenci	02666121400	<input type="button" value="İstatistik"/> <input type="button" value="Güncelle"/> <input type="button" value="Sil"/>
ogrenci1	Ogrenci1	05551112233	<input type="button" value="İstatistik"/> <input type="button" value="Güncelle"/> <input type="button" value="Sil"/>
ogrenci2	Ogrenci2	05551112244	<input type="button" value="İstatistik"/> <input type="button" value="Güncelle"/> <input type="button" value="Sil"/>
ogrenci3	Ogrenci3	05551112255	<input type="button" value="İstatistik"/> <input type="button" value="Güncelle"/> <input type="button" value="Sil"/>

**Şekil 3.15:** Sistemde kayıtlı kullanıcılar listesi.

İstatistik sayfasında öğrencilerin hangi sayfada ne kadar vakit harcadığı bilgisi bulunmaktadır.



**Şekil 3.16:** Öğrencinin sayfalarda harcadığı süre.

#### 3.4.4 Uygulama ve Değerlendirme Basamağı

Bu başlık altında wtPHP'nin üretilmesi sürecinde yapılan test uygulamalar ve değerlendirme süreçlerinden bahsedilmektedir.

Uygulama basamağı, üretilen yazılımın test edildiği aşamadır. Burada üretilen yazılımın kullanılabilirliği test edilmekte, eksikleri ve hataları tespit edilmekte, tekrar düzeltme işlemleri yapılmaktadır.

ADDIE öğretim tasarım modeli temel alınarak üretilen eğitim yazılımları için değerlendirme iki şekilde yapılmaktadır. Birincisi tüm üretim süreci boyunca, basamakların ve basamaklar arası geçişin değerlendirildiği süreç değerlendirme, ikincisi ise üretilen yazılımın değerlendirildiği ürün değerlendirmedir.

Süreç değerlendirme, yazılım üretilirken bitirilen her bir basamağın, ulaşılmak istenen hedeflere uygunluğunu tespit etmek için yapılmaktadır. wtPHP üretilirken yapılan süreç değerlendirmelerinde şu eksiklik tespit edilmiştir:

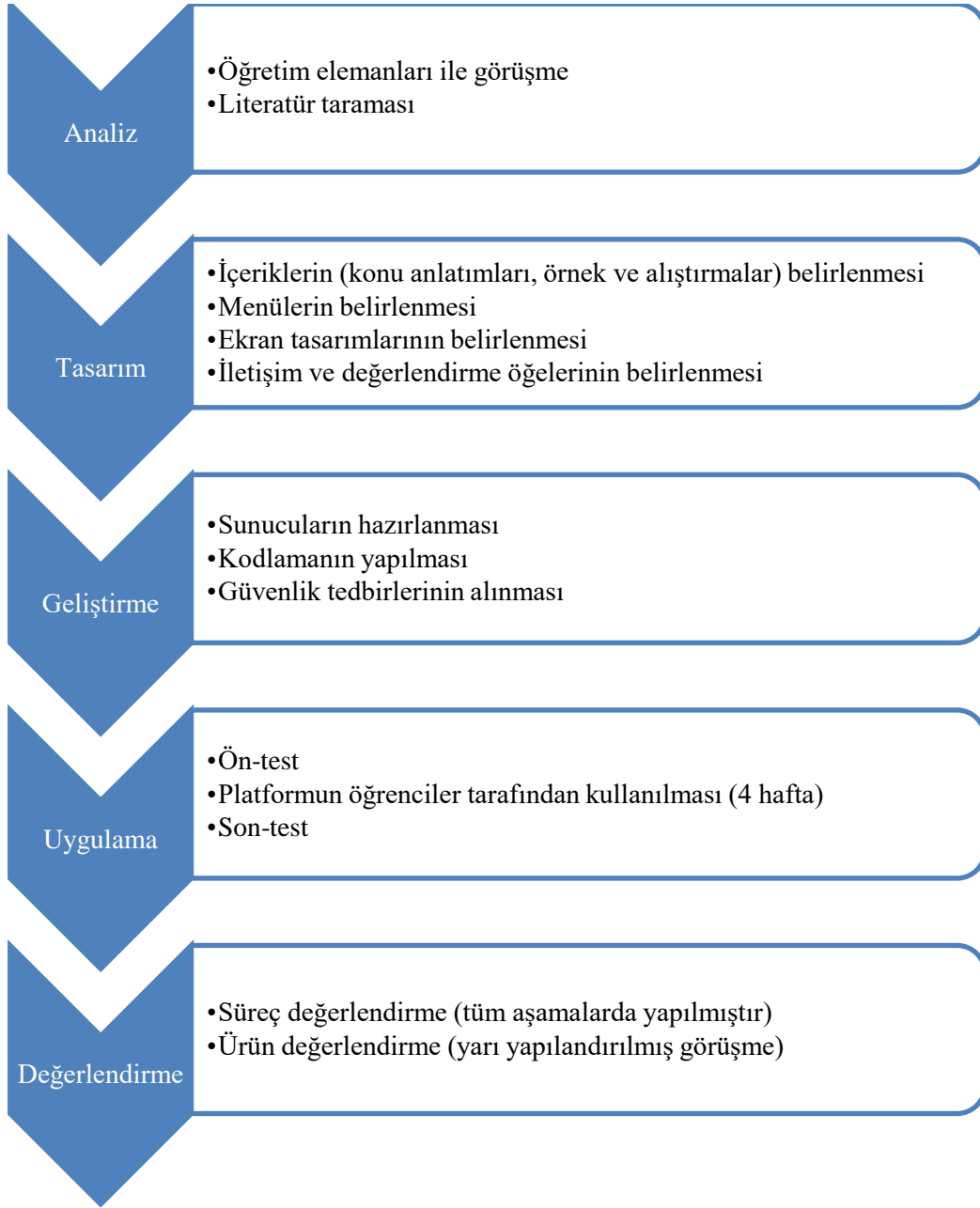
Öğrencilerin programlama derslerinde başarısız olma nedenlerinden biri olan ders etkinlikleri dışında programlama çalışmaları durumuyla ilgili olarak, üretilen yazılımın öğrencileri ders saatleri dışında da programlamaya teşvik edip etmediğine dair herhangi bir kaydın tutulmadığı belirlenmiştir. Bu sebeple yazılıma, öğretim

elemanına ait Admin Menü altında yer alan Kullanıcılar seçeneğindeki istatistik bilgileri ekranına oturum açma-kapatma bilgileri ile o oturumun toplam süresi hakkında bilgiler listelenmiştir.

Oturum Bilgileri (Son 10 oturum bilgisi)		
Oturum Başlangıç	Son Aktivite	Toplam Süre
09-11-2016 18:42:36	09-11-2016 19:39:09	57 dakika
09-11-2016 16:32:29	09-11-2016 16:32:40	0 dakika
09-11-2016 15:54:26	09-11-2016 16:19:37	25 dakika
09-11-2016 14:12:15	09-11-2016 15:47:29	95 dakika
09-11-2016 13:58:44	09-11-2016 13:58:54	0 dakika
09-11-2016 13:10:11	09-11-2016 13:55:41	46 dakika
09-11-2016 12:06:48	09-11-2016 12:07:29	1 dakika
08-11-2016 20:00:19	08-11-2016 20:01:21	1 dakika
28-10-2016 16:02:33	28-10-2016 16:07:04	5 dakika
28-10-2016 14:01:43	28-10-2016 14:15:55	14 dakika

**Şekil 3.17:** Oturum açma-kapatma zamanı ve oturum süresi.

Yapılan iyileştirmelerden sonra wtPHP'yi öğrencilerin kullanması sağlanmıştır. wtPHP 4 hafta boyunca internet tabanlı programlama dersinde kullanılmıştır. Ders süresi haftalık 5 saattir. Bilgisayar laboratuvarında yapılan yüz yüze derslerde, araştırmacı tarafından şartlı ifadeler ve döngüler konuları işlenirken platformda yer alan alıştırmalar ve örnekler kullanılmıştır. Bunun yanında öğrencilerin ders dışında da platformda yer alan alıştırmaları yapmaları beklenmiştir. Uygulama sonunda öğrencilerle yapılan yarı yapılandırılmış görüşmelerle platformu değerlendirmeleri sağlanmıştır. Bulgular bölümünde değerlendirme sonuçları ayrıntılı olarak verilmiştir.



**Şekil 3.18:** Araştırma süreci.

## 4. BULGULAR

Bu bölümde araştırma sonucu elde edilen bulgulara yer verilmiştir.

### 4.1 Geliştirilen Platformun Öğrencilerin Programlama Başarısına Etkisi

Araştırmada, yapılan uygulamanın öğrenci başarılarına herhangi bir etkisi olup olmadığını tespit edebilmek amacıyla ilişkili örneklem için t-testi kullanılmıştır.

İlişkili örneklem için t-testi, aynı gruba farklı zamanlarda yapılan tekrarlı ölçümler arasında anlamlı bir değişim olup olmadığı tespit edebilmek amacıyla kullanılmaktadır (Büyüköztürk, 2003).

Araştırmanın başında ve sonunda yapılan başarı testinden elde edilen verilerin t-testi ile analizinin yapılabilmesi için iki testten elde edilen puanlar arasındaki farkın normal bir dağılım gösterip göstermediği tespit edilmiştir. Ön-test ve son-test başarı puanları ile testlerden elde edilen puanların farkının Shapiro-Wilk testinin sonuçlarına göre normal dağılım gösterdiği görülmüştür (Sig. değeri 0.05'ten büyüktür).

T-testi yapılarak elde edilen, ön-test ve son-teste dair veriler, aşağıdaki tabloda gösterilmektedir:

**Tablo 4.1:** Öğrencilerin programlama başarılarındaki değişim.

Ölçüm (Başarı Testi)	N	$\bar{X}$	S	sd	t	p
Ön-test	38	68.82	14.632	37	-2.351	.000
Son-test	38	74.61	12.807			

Tablo 4.1'de öğrencilerin son-test başarı puanlarının ( $\bar{X}=74.61$ ,  $S=12.807$ ), ön-test başarı puanlarından ( $\bar{X}=68.82$ ,  $S=14.632$ ) daha yüksek olduğu görülmektedir. Uygulama sonunda öğrencilerin programlama başarı puanlarında istatistiksel olarak anlamlı bir artış olduğu tespit edilmiştir ( $t_{(37)}=2.351$ ,  $p<0.05$ ).

## 4.2 Geliştirilen Platformun Öğrencilerin Programlamaya Yönelik Öz Yeterlik Algılarına Etkisi

Araştırmanın başında ve sonunda uygulanan programlamaya ilişkin öz yeterlik algısı ölçeğinden elde edilen verilerde anlamlı bir artış olup olmadığını tespit etmek üzere t-testi yapabilmek için öncelikle verilerin normal dağılım gösterip göstermediği incelenmiştir. Yapılan testlere göre elde edilen sonuçların normal dağılım gösterdiği görülmüştür (shapiro-wilk testinin sonuçlarına göre sig. değeri 0.05'ten büyüktür).

İki başlıktan oluşan programlamaya ilişkin öz yeterlik algısı ölçeğinden elde edilen verilerin tamamına bakıldığında anlamlı bir artış olduğu görülmektedir.

**Tablo 4.2:** Öğrencilerin öz yeterlik algılarındaki değişim.

Ölçüm (Öz Yeterlik)	N	$\bar{X}$	S	sd	t	p
Ön Test	38	39.03	10.239	37	-3.866	.000
Son Test	38	44.47	7.521			

Tablo 4.2’de öğrencilerin öz yeterlik algısı ön-test puanlarının ( $\bar{X}$ =39.03, S=10.239), son-test puanlarından ( $\bar{X}$  =44.47, S=7.521) daha düşük olduğu görülmektedir. Programlamaya ilişkin öz yeterlik algısı ölçeğinden elde edilen verilere bakıldığında anlamlı bir artış olduğu görülmektedir ( $t_{(37)}=3.866$ ,  $p<0.05$ ).

**Tablo 4.3:** Öğrencilerin öz yeterlik algılarındaki değişim (basit programlama görevlerini gerçekleştirme).

Ölçüm (Öz Yeterlik)	N	$\bar{X}$	S	sd	t	p
I. Kısım						
Ön Test	38	18.79	4.021	37	-2.092	.043
Son Test	38	20.10	1.331			

Tablo 4.3’de öğrencilerin öz yeterlik algısı I. kısım ön-test puanlarının ( $\bar{X}$ =18.79, S=4.021), son-test puanlarından ( $\bar{X}$ =20.10, S=1.331) daha düşük olduğu görülmektedir. Programlamaya ilişkin öz yeterlik algısı ölçeğinden elde edilen verilere bakıldığında anlamlı bir artış olduğu görülmektedir ( $t_{(37)}=2.092$ ,  $p<0.05$ ).

Programlamaya ilişkin öz yeterlik algısı ölçeğinin ikinci kısmından (karmaşık programlama görevlerini gerçekleştirme) elde edilen verilere bakıldığında anlamlı bir artış olduğu görülmektedir.

**Tablo 4.4:** Öğrencilerin öz yeterlik algılarındaki değişim (karmaşık programlama görevlerini gerçekleştirme).

Ölçüm (Öz Yeterlik)	N	$\bar{X}$	S	sd	t	p
II. Kısım						
Ön Test	38	20.24	7.754	37	-4.191	.000
Son Test	38	24.368	7.193			

Tablo 4.4'te öğrencilerin öz yeterlik algısı II. kısım ön-test puanlarının ( $\bar{X}=20.24$ ,  $S=7.754$ ), son-test puanlarından ( $\bar{X}=24.368$ ,  $S=7.193$ ) daha düşük olduğu görülmektedir. Programlamaya ilişkin öz yeterlik algısı ölçeğinden elde edilen verilere bakıldığında anlamlı bir artış olduğu görülmektedir ( $t_{(37)}=4.191$ ,  $p<0.05$ ).

### 4.3 Öğrencilerin Öz Yeterlik Algılarındaki Değişim İle Ders Başarısındaki Değişim Arasındaki İlişki

Öğrencilerin ders başarılarındaki değişim ile öz yeterlik algılarındaki değişim arasında anlamlı bir ilişki olup olmadığını tespit edebilmek için Pearson korelasyon analizi yapılmıştır.

**Tablo 4.5:** Öğrencilerin ders başarıları ile öz yeterlik algılarındaki değişim arasındaki ilişki.

		Öz Yeterlik	Başarı Testi
Başarı Testi	Pearson K.	.224	1
	Anlamlılık	.177	
	N	38	38
Öz Yeterlik	Pearson K.	1	.224
	Anlamlılık		.177
	N	38	38

Tablo 4.5'ten elde edilen sonuçlara göre öğrencilerin ders başarılarındaki artış ile öz yeterlik algılarındaki artış arasında zayıf bir korelasyon gözlenmiştir ( $r=.224$ ,  $p>0.05$ ,  $r^2=0.05$ )



**Tablo 4.6:** İnternet tabanlı programla dersine ait öğrenci görüşlerine ilişkin temalar.

	<b>Temalar</b>	<b>f</b>
Olumlu	Alanımızda kendimizi geliştirmek için gerekli bir ders	4
	İş hayatımız için faydalı olacağını düşünüyorum	4
	PHP'nin mantığını iyi kavrayabildim	1
	Web tasarımı ve PHP ile işe yarar bir ürün üretebiliyoruz	2
Olumsuz	Yapısı karışık	1
	Öğrenmesi zor	3

Öğrencilerin bu ders hakkında çoğunlukla olumlu düşündükleri ve gelecekte bu derste öğrendikleri konuların kendilerine faydalı olacağına inandıkları görülmektedir. Verilen olumsuz yanıtlarda ise dersin karışık ve zor olduğu öğrencilerce ifade edilmiştir.

2. İnternet Tabanlı Programlama dersi ile daha önceden almış olduğunuz diğer programlama dersleri arasında bir fark görüyor musunuz?

Öğrenciler internet tabanlı programlama dersinde öğrendikleri PHP programlama dilinin yapısını daha önce öğrendikleri diğer dillere benzetmekle birlikte bir kısmı (f=3) daha kolay olduğunu ifade etmişlerdir.

**Tablo 4.7:** İnternet tabanlı programla ile önceki programlama derslerinin karşılaştırılmasına ait öğrenci görüşlerine ilişkin temalar.

<b>Temalar</b>	<b>f</b>
PHP ile uygulama yapmak için önceki bilgilerimizi de kullanıyoruz	1
PHP ile kullanışlı programlar üretebiliyoruz	1
Diğer öğrendiğimiz dillerden daha kolay	3
Kodları farklı ama mantık olarak diğer dillerle aynı	7

3. wtPHP'yi ne kadar kullandınız?

Öğrencilerin tamamına yakını platformu internet tabanlı programlama dersini aldıkları süre boyunca kullandıklarını söylemiştir. Öğrencilerin bazıları platformdaki

alıştırmaları yapmanın yanında, platformu kod editörü gibi de kullandıklarını belirtmişlerdir.

4. wtPHP'yi içerik özellikleri açısından değerlendirir misiniz?

**Tablo 4.8:** wtPHP'nin içerik özellikleri açısından değerlendirilmesine yönelik öğrenci görüşlerine ilişkin temalar.

	<b>Temalar</b>	<b>f</b>
Olumlu	Ders başarıma katkısı oldu	2
	Örnek ve alıştırmalar sayı olarak yeterli	3
	Örnek ve alıştırmalar ders ile örtüşüyor	3
	İçerikler PHP'yi öğrenmek için faydalı	5
Olumsuz	Yapılması beklenen alıştırmalar, örneklerden daha zor	1
	Derste anlatılan tüm konuları içermiyor	3

Görüşme yapılan öğrenciler, platformda yeterli sayıda örnek ve alıştırma olduğunu ve içeriklerin ders içeriğiyle örtüştüğünü belirtip, içeriklerin ders başarısına ve PHP'yi öğrenmeye katkısı olduğunu ifade etmişlerdir. wtPHP'de uygulamanın yapıldığı süre zarfında anlatılan konular yer almaktadır. Bununla birlikte öğrencilerin bir kısmı ders kapsamında anlatılan diğer konuların da wtPHP'ye eklenmesi gerektiğini belirtmişlerdir.

- ogr2: *Alıştırma yaparken, mesela bir döngünün parantezlerini nereye koyacağımı unuttuğum zaman hemen konu anlatımı kısmına bakıp, rahatlıkla nasıl yapacağımı bulabiliyorum.*
- ogr6: *İçerik açısından güzel. Çok alıştırma var. Sınavlarıma katkısı oldu.*
- ogr8: *İçerik olarak konu anlatımı kısmı iyi olmuş. Alıştırma sorularını yapmaya çalışırken aynı pencerede sonuçları görmek kolaylık sağlıyor. Her bir konu başlığında bir sürü alıştırma sorusu var, o açıdan güzel.*
- ogr9: *Örnekler ve alıştırmalar güzel, oradan çalıştım zaten çoğu zaman, başka bir yere bakmadım.*

- ogr11: *Konu anlatımı ve alıştırmaları yapmak benim için yararlı oldu. Konuların arkasından alıştırmaları yapmak faydalı oldu tekrar etmek açısından.*

Öğrencilerden biri, konu anlatımında kolay örnekler olduğunu, ancak alıştırmalar bölümünde daha zor soruların sorulduğunu iletmiştir.

- ogr7: *Örnekler basit geldi, ancak sorduğunuz alıştırmalar daha zordu. Yani alıştırmalar kısmı zor ama kendimiz örneklerle açıklıyoruz ya oralar daha kolaydı.*

5. wtPHP'yi kullanım özellikleri açısından değerlendirir misiniz?

**Tablo 4.9:** wtPHP'nin kullanım özellikleri açısından değerlendirilmesine yönelik öğrenci görüşlerine ilişkin temalar.

Temalar	f
Çalışmayı kaydetme konusunda uyarılar eksik	1
Kullanımı kolay ve pratik	5
Menü başlıkları ve pencereler net ve anlaşılır	4
Editör gibi kullanmak mümkün	3
Kullanışlı, işe yarar bir program	5

Tablo 4.9'da öğrencilerin platformun kullanım özelliklerini değerlendirdikleri soruda platformun sade bir arayüzü olduğunu, menülerin anlaşılır ve kolay erişilebilir olduğunu ifade etmişlerdir.

- ogr3: *Kullanımı çok kolay, çok rahat. Her şey menü menü belirtilmiş, çok rahat kullanılabilir.*
- ogr8: *Bir öğrenci için ders çalışırken nereden başlayayım, nasıl gideyim, nasıl devam edeyim sorularına cevap veren bir uygulama olmuş.*
- ogr9: *Kullanışlı bir site olmuş. Her şey açık ve net. Soldaki menülerde gerekli maddeler ve örnekler bulunduğu için problem yaşamıyorum.*
- ogr12: *Gayet iyiydi, kesinlikle kolaydı.*

Bazı öğrenciler PHP kodunu yazıp, aynı pencerede ekran çıktılarını görebilmenin pratik bir kullanım sağladığını belirtip, bu özellikten memnun kaldıklarını dile getirmişlerdir.

- ogr10: *Kullanım basitti, hoştu. PHP kodunu yazdığımızda orada çalıştırabiliyorduk. Bence bu güzel bir şeydi.*

Görüşmelerden öğrencilerin platformu, kod editörü gibi de kullandıkları anlaşılmaktadır. Öğrenciler, platformda yer alan alıştırma ile ilgili kodları silip, kendi yaptıkları programa dair kodları orada denediklerini söylemişlerdir.

- ogr5: *Bence kesinlikle iyiydi. Oradaki alıştırmalar dışında kendi kendimize alıştırma yapabiliyorduk.*
- ogr7: *Çok işime yaradı. Çünkü bilgisayarına Wamp Server programını kurarken bir problem oluştu ve bilgisayarına bu programı kuramadım. Ben de uygulamada (wtPHP) alıştırmaya dair kodları silip, kendi yazdığım örnek kodlarımı denedim.*

#### 6. wtPHP'de gördüğünüz eksiklikler nelerdir? Bu eksiklikler nasıl giderilebilir?

Görüşmelerde öğrencilere platform ile ilgili herhangi bir eksik yön olup olmadığı, eğer varsa bu eksikliğin nasıl giderilebileceği sorulmuştur.

Öğrencilerden bazıları platformun mevcut haliyle iyi olduğunu, herhangi bir düzeltme ya da eklemeye ihtiyaç duymadığını belirtmişlerdir.

- ogr1: *Bu haliyle gayet kullanışlı.*
- ogr3: *Bence program zaten tamamen uygun.*
- ogr12: *Herhangi bir eksiklik görmüyorum.*

Bir öğrenci, platformda alıştırma yaparken başka bir menüye tıkladığında, örneğin konu anlatımı kısmına bakmak istediğinde yaptığı alıştırmayı kaydetmek isteyip istemediğinin sorulmadığını, bazen dalgınlıkla kaydetmeyi unuttuğunda yaptıklarının boşa gittiğini söylemiştir. Alıştırma yaparken başka bir menüye tıklanması halinde yazdığı kodları kaydetmesi için bir uyarı konulmasının ya da yazdığı kodları çalıştırdığı zaman otomatik olarak kaydedilmesinin uygulama için iyi bir özellik olabileceğini belirtmiştir.

- ogr7: *Bazen ben konuya bakmak istiyorum ama tıkladığımda yaptıklarım gidiyor. Çalıştır'a tıkladığı zaman kaydetme yada başka menüye geçerken kaydetmek istiyor musun şeklinde bir uyarı iyi olur.*

Bir öğrenci, alıştırma yaparken öğretim elemanına, o alıştırma ile ilgili soru sorabilmek için alıştırmanın hemen altına bir soru sorma alanı oluşturulabileceğini belirtmiştir. Öğretim elemanı da soruya aynı yerden cevap verebilmelidir.

- ogr6: *Daha etkili olması için alıştırmanın altına mesaj gönderme alanı eklenebilir. Böylece hocamıza alıştırma ile ilgili sorularımızı sorabiliriz. Hocamızda bize oradan cevap verebilir. Onun dışında gayet kullanışlı.*

Öğrencilerden bir kısmı konu anlatımlarında yer alan örneklerle ilgili olarak örneklerin sayı olarak arttırılmasının, kimi örneklerin biraz da karmaşık ve ayrıntılı olmasının ve bu örneklerin de videolu anlatım şeklinde olmasının konu anlatımını zenginleştirebileceğini ifade etmişlerdir.

- ogr2: *Konu anlatımında yer alan örnekler arttırılabilir.*
- ogr5: *Biraz daha kompleks örnekler olabilir. Bu örnekler video şeklinde gösterilebilir. Kısa videolar şeklinde programın nasıl yapıldığıyla ilgili anlatımlar içerir bu videolarda.*

## 5. SONUÇ, TARTIŞMA VE ÖNERİLER

Bu çalışmanın amacı, PHP programlama dilinin eğitiminde kullanılacak web tabanlı, öğrencilerin konularla ilgili çeşitli uygulama örnekleri bulabilecekleri, kendi kendilerine uygulama yapabilecekleri ve yaptıkları uygulamalarla ilgili aldıkları hata mesajlarının olası sebepleri konusunda kendi dillerinde ipuçları alabilecekleri bir platform oluşturmaktır. Oluşturulan platform ile öğrencilerin ders saatleri dışında, istedikleri mekânda PHP programlama dilini çalışmalarını sağlamak amaçlanmaktadır.

Bu bölümde, araştırma sonucunda elde edilen bulguların sonuçlarına ve bu sonuçlar doğrultusunda ileride yapılacak araştırmalara ilişkin önerilere yer verilmiştir.

### 5.1 Sonuçlar

PHP programlama dili için eğitimi destekleyici materyal olarak üretilen wtPHP 4 hafta boyunca, şartlı ifadeler ve döngüler konuları işlenirken sınıf ortamında kullanılmıştır. Bunun yanında öğrencilerin ders dışında da platformda yer alan alıştırmaları yapmaları beklenmiştir. Bu çalışmadan elde edilen veriler ışığında ulaşılan sonuçlar şöyledir:

- Öğrencilerin programlama dilleri dersinde başarılı olabilmesi için derste yaptıkları çalışmalarla yetinmeyip, ders dışında da bolca örnek uygulama yapmaları gerekmektedir. Buna göre; öğrencilerin platform üzerindeki hareketlerini gösteren veriler incelendiğinde, çalışmanın amacına uygun olarak öğrencilerin ders saati dışında platforma giriş yapıp, alıştırmaları yaptıkları görülmüştür.
- wtPHP'ye öğrencilerin problem çözme becerilerini geliştirebilmesi ve araştırma yaparak çözüm yolları araması için derste henüz görmedikleri konularla ilgili fonksiyonları da içeren alıştırmalar eklenmiştir. Öğrencilerin çalışma verileri incelendiğinde görülmüştür ki derste başarılı öğrenciler bu tür

alıştırmaları yapmak için epey uğraşmışken, ders başarısı daha düşük öğrenciler bu alıştırmaları yapmak için ya hiç vakit ayırmamışlar ya da çok az uğraşmışlardır.

- Öğrencilerin çalışma verileri incelendiğinde fibonacci serisini ekrana yazdırma ya da ikinci dereceden bir bilinmeyenli denklemlerin çözümü gibi matematik konularını içeren alıştırmaların daha az yapıldığı görülmüştür.
- Uygulama sonunda, öğrencilerin programlama başarı puanlarında istatistiksel olarak anlamlı bir artış olduğu tespit edilmiştir.
- Uygulama sonunda, öğrencilerin programlamaya ilişkin öz yeterlik algılarında anlamlı bir artış olduğu görülmüştür. Ayrıca programlamaya ilişkin öz yeterlik algısı ölçeğinin hem birinci kısmından (basit programlama görevlerini gerçekleştirme) hem de ikinci kısmından (karmaşık programlama görevlerini gerçekleştirme) elde edilen veriler ayrı ayrı incelendiğinde öğrencilerin öz yeterlik algılarında anlamlı bir artış olduğu görülmektedir.
- Öğrenciler, wtPHP'nin programlama çalışmalarına katkısı olduğunu ve içerik özellikleri açısından yararlı bulduklarını belirtmişlerdir. Öğrencilerin çoğu, ders kapsamında anlatılan tüm konuların uygulamaya eklenmesini istemişlerdir.
- Öğrenciler wtPHP'yi kullanım özellikleri açısından sade, kolay anlaşılır ve pratik bulmuşlardır. Öğrenci görüşlerinden ve öğretim elemanının gözlemlerinden anlaşıldığı kadarıyla öğrenciler, wtPHP'de yer alan alıştırmaları yapmanın yanında başka programlama çalışmaları için wtPHP'yi editör gibi kullandıkları ortaya çıkmıştır.
- Bu çalışmada öğrencilerin wtPHP'de yer alan alıştırmaları çözmek için ne kadar süre harcadığına dair bilgilerde kullanılmak istenmiş; ancak öğrencilerin wtPHP'yi editör gibi kullanmaları sebebiyle kullanılamayacak kadar çok veri birikmiştir.
- Öğrencilerin bir kısmı videolu örneklerle ya da diğer görsel öğelerle konu anlatımının zenginleştirilebileceğini belirtmişlerdir.
- Bazı öğrenciler alıştırma yapılan ekranda öğretim elemanına soru sormak için alanların eklenmesinin uygulamayı daha faydalı yapabileceğini söylemişlerdir.

## 5.2 Tartışma

Bu bölümde, araştırmadan elde edilen sonuçlar, benzer araştırmalardan elde edilen sonuçlar ile karşılaştırılmıştır.

Dehmenoğlu (2015) mesleki ve teknik Anadolu lisesi bilişim teknolojileri alanında eğitim görüp, programlama temelleri dersini alan 62 tane 10. sınıf öğrencisi ile yaptığı çalışmada geliştirdiği mobil öğrenme aracının, öğrencilerin akademik başarıları üzerinde istatistiki olarak anlamlı bir etkisinin olduğu, mobil öğrenme aracı ile desteklenen gruptaki öğrencilerin, sadece yüz yüze eğitim alan öğrencilerden daha yüksek başarı sağladığını belirtmiştir.

Yurdugül ve Gültekin (2009) çoklu ortama dayalı öğretim yazılımının programlama öğreniminde öğrenci başarıları üzerine etkisini araştırmak amacıyla, bilgisayar ve öğretim teknolojileri eğitimi bölümünde eğitim alan 34 tane birinci sınıf öğrencisinin katılımıyla yaptıkları çalışmada; programlama öğreniminde çoklu ortama dayalı öğretim yazılımının, alışlagelmiş öğretim yöntemine kıyasla öğrenci başarıları üzerinde istatistiksel olarak anlamlı bir etkisi bulunduğu saptamıştır.

Bu araştırmada da yapılan testler ile öğrencilerin ders başarılarında istatistiksel olarak anlamlı bir artış olduğu görülmüştür. Ancak bu araştırmada kontrol grubu olmadığından wtPHP'nin ders başarıları üzerindeki etkisi konusunda kesin bir şey söylemek mümkün değildir. Bununla birlikte dersi veren öğretim elemanının öznel yorumuna göre, önceki yıllara kıyasla bu dönem yapılan çalışmanın, öğrencilerin ders başarıları üzerinde olumlu bir etkisi olduğu belirtilebilir.

Vural ve Taşdelen (2014), web tasarımı ve programlama dersi için tasarladıkları web tabanlı öğrenme modülünün öğrencilerin ders başarıları üzerindeki etkisini tespit etmek için yaptıkları çalışmada, modülün uygulandığı gruptaki öğrencilerin ders başarılarının, modülün uygulanmadığı diğer gruptaki öğrencilerin ders başarılarına göre daha iyi olduğunu belirtmiştir. Hem web tabanlı öğrenme modülünde hem de bu araştırma için geliştirilen wtPHP'de, öğrencilerin istedikleri yer ve zamanda ders çalışıp uygulama yapabilecekleri, uygulamalarının çıktılarını kolay bir şekilde görebilecekleri bir ortam tasarlanmaya çalışılmıştır. Bununla birlikte web tabanlı öğrenme modülünde değerlendirme araçları da bulunmaktadır. wtPHP'de herhangi bir değerlendirme aracı mevcut değildir.

Uzun (2013), bilgisayar ve öğretim teknolojileri eğitimi bölümde eğitim alan ve programlama dilleri dersini alan 60 öğrencinin katılımıyla yaptığı çalışmada geleneksel yüz yüze eğitime destek olması için tasarlanan internet destekli öğrenme ortamına ilişkin katılımcıların görüşlerini almıştır. Hadjerrouit (2008) ise 11 lisans öğrencisinin katılımı ile yaptığı çalışmasında Java programlama dili için tasarlanmış öğrenme ortamının etkisini ortaya koymaya çalışmıştır. Her iki çalışmada programlama öğretimi için tasarlanan materyallerin (bilgisayar destekli öğrenme ortamı, öğrenme ortamı) programlama diline ait kavramları anlamada ve programcılık performanslarını arttırmada öğrencilere destek olduğunu; ayrıca motivasyon sağladığını ve programlama bilgilerini inşa etmede yardım ettiği belirtilmiştir. Bu araştırmada da öğrenci görüşlerinden elde edilen sonuçlara göre wtPHP'nin, öğrencilerin programlama çalışmalarına destek olduğu ve yararlı bir araç olduğu söylenilebilir.

Al-Imamy, Alizadeh ve A. Nour (2006) yaptıkları çalışmada web tabanlı bir materyal geliştirmişlerdir. Araştırmacılar uygulamayı ikişer haftalık 3 periyot halinde C dersi alan 20 öğrenci ile uygulamışlardır ve elde ettikleri veriler sonucunda geliştirdikleri materyalin, programlama öğretiminde etkili bir araç olduğunu ifade etmişlerdir. Araştırmacılar materyalin, programlama bilgisi farklı seviyelerde olan öğrencilerden oluşan bir sınıfta, öğrenciler arasındaki farklılıkları kapatarak, hepsinin aynı seviyeye ulaşmasında yararlı olduğunu belirtmişlerdir. wtPHP'de da en temelden başlayarak konu ve alıştırmalar düzenlenmeye çalışılmıştır.

### 5.3 Öneriler

Bu bölümde ileride yapılacak arařtırmalar için öneriler sunulmuřtur.

### 5.4 Platformun Geliřtirilmesine İliřkin Öneriler

Öğrenci görüşleri ve arařtırmacının deneyimlerinden elde sonuçlar neticesinde platformun geliřtirilmesine iliřkin ařađıda yer alan öneriler sunulmuřtur:

- Her konu için, öğrencinin konuyu ne kadar öğrendiđini ölçmeye yarayacak konu deđerlendirme araçları eklenebilir.
- Ders kapsamında öğretilecek olan tüm konuları içeren genel deđerlendirme araçları eklenebilir.
- Programlama dili gibi zor olduđu düşünölen bir derste öğrencilerin ilgisini çekip ve motivasyonu sađlamak zordur. Bunun için materyale çeřitli eđence unsurları eklemek faydalı olabilir.
- Platformda konu anlatım kısmında görsel unsurlar ve çoklu ortam nesnelere mümkün olduđunca yer vermek faydalı olabilir.
- Öğrencilerin, öğretim elemanı ya da arkadaşlarıyla görüşebilecekleri anlık mesajlařma özelliđi eklenebilir.
- Her alıřtırmanın altına öğrencilerin soruları sorup görüşlerini iletebileceđi bir yazıřma alanı (yorum alanı gibi) yapılabilir.
- Öğrencilerin uygulamada yer alan alıřtırmalar dıřında da programlama çalışabilmeleri için serbest çalışma ekranı yapılabilir.

### 5.5 Uygulamaya İliřkin Öneriler

- Bu arařtırmada platformun etkinliđi tek bir grup üzerinde ve 4 hafta süresince denenmiřtir. Gelecek arařtırmalarda, farklı bölgelerde yer alan daha fazla grup üzerinde ve daha uzun sürede deneme yapılması daha sađlıklı sonuçlar verebilir.

- Öğrencilerin platform üzerindeki çalışmaları sık sık kontrol edilip, sorun yaşayan öğrencilerin problemlerine hemen çözüm bulmak gerekmektedir.
- Platformu az kullanan ya da hiç kullanmayan öğrenciler tespit edilip, çalışmaları konusunda teşvik edilmelidir.
- Bilgisayarı ya da internet erişimi olmayan ya da bu konuda sorunlar yaşayan öğrencilerin çalışabilmeleri için uygun ortamlar sağlanmalıdır.
- Uygulamayla başlarken öğrencilerin programlama, yapılan uygulama ya da konular ile ilgili her türlü fikrini tartışabileceği bir ortam (forum) da kullanıma sunulabilir. Bu ortamdan veriler sağlanabilir.
- Öğrencilerin istedikleri zaman çalışabilmeleri için sunucuların her zaman erişilebilir olması gerekmektedir.

## 6. KAYNAKLAR

Akpınar, Y. ve Altun, A. (2014). Bilgi Toplumu Okullarında Programlama Eğitimi Gereksinimi. *İlköğretim Online*, 13(1), 1-4.

Al-Imamy, S., Alizadeh, J. ve A. Nour, M. (2006). On the Development of a Programming Teaching Tool: The Effect of Teaching by Templates on the Learning Process. *Journal of Information Technology Education*, 5, 271-286.

Altun, A. ve Mazman, S. G. (2012). Programlamaya İlişkin Öz Yeterlilik Algısı Ölçeğinin Türkçe Formunun Geçerlilik ve Güvenirlilik Çalışması. *Eğitimde ve Psikolojide Ölçme ve Değerlendirme Dergisi*, 3(2), 297-308.

Arkün, S., Baş, T., Avcı, Ü., Çevik, V. ve Gürcan, T. (2009). Addie Tasarım Modeline Göre Web Tabanlı Bir Öğrenme Ortamı Geliştirilmesi. *Eğitimin Değişen Yüzü: Yeni Paradigmalar 25.Yıl Konferansı*. Ankara.

Aşkar, P. ve Davenport, D. (2009). An Investigation of Factors Related to Self-Efficacy for Java Programming Among Engineering Students. *The Turkish Online Journal of Educational Technology, TOJET*, 8(1), 26-32.

Aydın, E. (2013). Öğretim Tasarım Modelleri. [https://www.academia.edu/2454177/%C3%96%C4%9Fretim\\_Tasar%C4%B1m\\_Modeleri](https://www.academia.edu/2454177/%C3%96%C4%9Fretim_Tasar%C4%B1m_Modeleri) (Erişim Tarihi: 08.02.2017)

Berksoy, İ., Sözcü, Ö. F., Armağan, E. ve Arslan, A. (2016). Algoritma ve Programlama Eğitiminde Scratch Programı Üzerine Yapılan Çalışmaların Analizi. *XVIII. Akademik Bilişim Konferansı*. Aydın.

BTK. (2017). Türkiye Elektronik Haberleşme Sektörü Üç Aylık Pazar Verileri Raporu. Bilgi Teknolojileri ve İletişim Kurumu: [https://www.btk.gov.tr/File/?path=ROOT%2F1%2FDocuments%2FSayfalar%2FPazar\\_Verileri%2F2016-Q4.pdf](https://www.btk.gov.tr/File/?path=ROOT%2F1%2FDocuments%2FSayfalar%2FPazar_Verileri%2F2016-Q4.pdf) (Erişim Tarihi: 02.04.2017)

Butler, M. ve Morgan, M. (2007). Learning Challenges Faced by Novice Programming Students Studying High Level and Low Feedback Concepts. *Providing Choices for Learners and Learning*, 99-107. Singapur: ICT.

Büyüköztürk, Ş. (2012). Örneklem Yöntemleri. <http://cv.ankara.edu.tr/duzenleme/kisisel/dosyalar/21082015162828.pdf> (Erişim Tarihi: 20.01.2017)

Büyüköztürk, Ş., Çakmak, E. K., Akgün, Ö. E., Karadeniz, Ş. ve Demirel, F. (2009). *Bilimsel Araştırma Yöntemleri*. Ankara: Pegem Akademi, 187-196.

Büyüköztürk, Ş. (2003). *Sosyal Bilimler için Veri Analizi El Kitabı*. Ankara: Pegem A Yayıncılık.

Code.org (2013). What Most Schools Don't Teach. <https://www.youtube.com/watch?v=nKIu9yen5nc> (Erişim Tarihi: 17.02.2017)

Çakır, H. ve Karataş, S. (2012). Öğretim Sistemleri Geliştirilmesi Sürecine Bir Bakış. *Eğitim Teknolojisi Kuram ve Uygulama*, 2(1), 19-35.

Çatlak, Ş., Tekdal, M. ve Baz, F. Ç. (2015). Scratch Yazılımı ile Programlama Öğretiminin Durumu: Bir Doküman İnceleme Çalışması. *Journal of Instructional Technologies & Teacher Education*, 4(3), 13-25.

Çetin, O. (2012). Nitel Bir Araştırma Tekniği: Görüşme. <http://oguzcetin.gen.tr/nitel-bir-arastirma-teknigi-gorusme.html> (Erişim Tarihi: 02.04.2017)

Dehmenoğlu, C. (2015). Programlama temelleri dersine yönelik mobil öğrenme aracının geliştirmesi, Yüksek Lisans Tezi. Bahçeşehir Üniversitesi, İstanbul.

Demirer, V. ve Sak, N. (2016). Programming Education and New Approaches Around The World and In Turkey. *Journal of Theory and Practise in Education*, 12(3), 521-546.

Ersoy, H. (2003). Blending online instruction with traditional instruction in the programming language course: A case study, Yüksek Lisans Tezi. Orta Doğu Teknik Üniversitesi, Ankara.

Forest, E. (2014). The ADDIE Model: Instructional Design. <http://educationaltechnology.net/the-addie-model-instructional-design/> (Erişim Tarihi: 09.02.2017)

Güngören, B. (2001). *C++ ile Nesne Tabanlı Programlama*. Ankara: Seçkin Yayıncılık.

Hadjerrouit, S. (2008). Towards a Blended Learning Model for Teaching and Learning Computer Programming: A Case Study. *Informatics in Education*, 7(2), 181–210.

Hawi, N. (2010). Causal Attributions of Success and Failure Made by Undergraduate Students in an Introductory-Level Computer Programming Course. *Computers & Education*, 54, 1127–1136.

İşman, A., Çağlar, M., Dabaj, F. ve Ersözlü, H. (2005). A New Model for the World of Instructional Design: A New Model. *The Turkish Online Journal of Educational Technology*, 4(3), 33-39.

Kukul, V. ve Gökçearsan, Ş. (2014). Scratch ile Programlama Eğitimi Alan Öğrencilerin Problem Çözme Becerilerinin İncelenmesi. *8th International Computer & Instructional Technologies Symposium*, 58-63. Edirne.

Lahtinen, E., Ala-Mutka, K. ve Järvinen, H. (2005). A Study of the Difficulties of Novice Programmers. *ITiCSE '05 Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, 14-18. Monte de Caparica, Portugal: ACM.

Lewis, C. M. (2010). How Programming Environment Shapes Perception, Learning and Goals: Logo vs. Scratch. *SIGCSE'10 Proceedings of the 41st ACM Technical Symposium on Computer Science Education*, 346-350. Wisconsin, USA: ACM.

Maloney, J., Resnick, M., Rusk, N., Silverman, B. ve Eastmond, E. (2010). The Scratch Programming Language and Environment. *ACM Transactions on Computing Education*, 10(4).

Mazman, S. G. ve Altun, A. (2013). Programlama–I Dersinin BÖTE Bölümü Öğrencilerinin Programlamaya İlişkin Öz Yeterlilik Algıları Üzerine Etkisi. *Journal of Instructional Technologies & Teacher Education*, 2(3), 24-29.

McGriff, S. J. (2000). Instructional System Design (ISD): Using the ADDIE Model. <https://www.lib.purdue.edu/sites/default/files/directory/butler38/ADDIE.pdf> (Erişim Tarihi: 15.01.2017)

Mhashi, M. M. ve Alakeel, A. M. (2013). Difficulties Facing Students in Learning Computer Programming Skills at Tabuk University. *Recent Advances in Modern Educational Technologies*, 15-24. Morioka City, Iwate, Japan: WSEAS Press.

Muruganatham, G. (2015). Developing of E-content Package by using ADDIE model. *International Journal of Applied Research*, 1(3), 52-54.

NG, S. C., Choy, S. O., Kwan, R. ve Chan, S. F. (2005). A Web-Based Environment to Improve Teaching and Learning of Computer Programming in Distance Education. *International Conference on Web-based Learning*, 279-290.

Ramalingam, V. ve 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.

Seferoğlu, S. S. (2016). Eğitimde Program Geliştirme Süreci ve Eğitim Durumlarının Düzenlenmesi. [http://members.comu.edu.tr/msalahli/EgitimTeknolojileri/2.Seferoglu\\_Egitimde-Program\\_Durumlar.pps](http://members.comu.edu.tr/msalahli/EgitimTeknolojileri/2.Seferoglu_Egitimde-Program_Durumlar.pps) (Erişim Tarihi: 28.12.2016)

TTKB. (2012). Bilişim Teknolojileri ve Yazılım Dersi ( 5, 6, 7 ve 8. Sınıflar) Öğretim Programı. <http://ttkb.meb.gov.tr/program2.aspx?islem=1&kno=196> (Erişim Tarihi: 26.12.2016)

TÜİK. (2016). Hanehalkı Bilişim Teknolojileri Kullanım Araştırması, 2016. Türkiye İstatistik Kurumu (TÜİK) Haber Bülteni: <http://www.tuik.gov.tr/PreHaberBultenleri.do?id=21779> (Erişim Tarihi: 09.09.2016)

Türnüklü, A. (2000). Eğitimbilim Araştırmalarında Etkin Olarak Kullanılabilecek Nitel Bir Araştırma Tekniği: Görüşme. *Kuram ve Uygulamada Eğitim Yönetimi*, 24, 543-559.

Uzun, A. (2013). BÖTE Öğretmen Adaylarının İnternet Destekli Öğrenme Ortamına İlişkin Görüşleri. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 28(3), 400-416.

Uzun, A. ve Özkılıç, R. (2012). Programlama Dilleri Dersi için Hazırlanan Karma Öğrenme Ortamına İlişkin Öğrenci Görüşleri. *e-Journal of New World Sciences Academy*, 7(2), 638-646.

Vural, H. G. ve Taşdelen, K. (2014). Web Tasarımı ve Programlama Dersi İçin Web Tabanlı Eğitimin Öğrenme Üzerine Etkisi. *International Conference on Education in Mathematics, Science & Technology*, 248-252. Konya.

Yurdugül, H. ve Gültekin, K. (2009). Çokluortamın Bilgisayar Programlama Başarısı Üzerine Etkisi. *9th International Educational Technology Conference*, 449-457. Ankara.

# **EKLER**

## 7. EKLER

### EK A- Programlamaya İlişkin Öz Yeterlik Algısı Ölçeği

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

## EK B - Başarı Testi

### BAŞARI TESTİ

AD-SOYAD:

NO:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	

1. `if ($yas>18 && $cinsiyet=="erkek")` ifadesinin anlamı nedir?

- A) Yaşı 18'den büyük ve cinsiyeti erkek ise
- B) Yaşı 18'den büyük veya cinsiyeti erkek ise
- C) Yaşı 18'den küçük ve cinsiyeti erkek ise
- D) Yaşı 18'den küçük veya cinsiyeti erkek ise

2. `$x=7;`

Yukarıdaki `x` değişkeni tanımlamasına göre aşağıdaki kod bloklarından hangisi ekrana çıktı vermez?

A) <code>if(\$x&gt;=6 &amp;&amp; \$x&lt;=8)</code> <code>echo "Merhaba";</code>	B) <code>if(\$x==7    \$x&gt;=8)</code> <code>echo "Merhaba";</code>
C) <code>if(!\$x==6)</code> <code>echo "Merhaba";</code>	D) <code>if(\$x&gt;=6 &amp;&amp; \$x&lt;=0)</code> <code>echo "Merhaba";</code>

3. `if (($a+$b)<0 && ($a*$b)>0)` ifadesinin anlamı nedir?

- A) `$a` ve `$b` değişkenlerinin toplamı 0'dan küçük veya çarpımı 0'dan büyük ise
- B) `$a` ve `$b` değişkenlerinin toplamı 0'dan küçük ve çarpımı 0'dan büyük ise
- C) `$a` ve `$b` değişkenlerinin toplamı 0'dan büyük ve çarpımı 0'dan küçük ise
- D) `$a` ve `$b` değişkenlerinin toplamı 0'dan büyük veya çarpımı 0'dan küçük ise

4. Bir sayının pozitif, negatif ya da 0 (sıfır) olduğunu ekrana yazdıran kod bloğu aşağıdakilerden hangisidir?

A) <code>if(\$sayi&gt; 0)</code> <code>echo "Sayı pozitif";</code> <code>elseif(\$sayi&lt; 0)</code> <code>echo "Sayı negatif";</code> <code>else</code> <code>echo "Sayı 0'dır";</code>	B) <code>if(\$sayi&gt; 0)</code> <code>echo "Sayı pozitif";</code> <code>elseif(\$sayi&lt; 0)</code> <code>echo "Sayı negatif";</code> <code>elseif</code> <code>echo "Sayı 0'dır";</code>
---	---

<pre>C) if(\$sayi &gt; 0){     echo "Sayı pozitif"; } else{     echo "Sayı negatif"; } echo "Sayı 0'dır";</pre>	<pre>D) if(\$sayi &gt;= 0)     echo "Sayı pozitif"; elseif(\$sayi &lt;= 0)     echo "Sayı negatif"; else     echo "Sayı 0'dır";</pre>
---	---

5. Kullanıcı tarafından girilen bir sayı 50 - 100 aralığında ise karesini, değilse karekökünü ekrana yazdıran kod bloğu aşağıdakilerden hangisidir?

<pre>A) if(\$sayi &gt;= 50    \$sayi &lt;= 100)     \$sayi=\$sayi*\$sayi; else     \$sayi=sqrt(\$sayi); echo \$sayi;</pre>	<pre>B) if(\$sayi &lt;= 50 &amp;&amp; \$sayi &lt;= 100)     \$sayi=\$sayi*\$sayi;     echo \$sayi; else     \$sayi=sqrt(\$sayi);     echo \$sayi;</pre>
<pre>C) if(\$sayi &gt;= 50 &amp;&amp; \$sayi &lt;= 100)     \$sayi=\$sayi*\$sayi; else     \$sayi=sqrt(\$sayi); echo \$sayi;</pre>	<pre>D) if(\$sayi &gt;= 50 &amp;&amp; \$sayi &lt;= 100)     \$sayi=sqrt(\$sayi); else     \$sayi=\$sayi*\$sayi; echo \$sayi;</pre>

6. Aşağıdaki kod bloğunun ekran çıktısı nedir?

<pre>\$not = 45; if(\$not &lt;= 0    \$not &gt;= 100)     echo "Yanlış not girişi yapılmıştır."; elseif(\$not &gt;= 50)     echo "Başarılı"; else     echo "Başarısız";</pre>	<p>A) Ekran çıktısı yoktur.</p> <p>B) Yanlış not girişi yapılmıştır.</p> <p>C) Başarılı</p> <p>D) Başarısız</p>
---	---

7. Aşağıdaki kod bloğu çalıştırıldığında kullanıcının girdiği kullanıcı adı ve şifre doğru ise "Giriş başarılı", değilse "Kullanıcı adı ya da şifre hatalı" şeklinde çıktı vermesi beklenmektedir. Fakat istenen sonucu vermemektedir. Aşağıdaki şıklardan hangisinde hata doğru bir şekilde ifade edilmiştir?

```
if($sifre == $girilen_sifre || $kullanici_adi == $girilen_kullanici_adi)
    echo "Giriş başarılı";
else
    echo "Yanlış kullanıcı adı ya da şifre";
```

A) Kodda parantezler kullanılmalı, kod aşağıdaki gibi olmalıdır.

```
if($sifre == $girilen_sifre || $kullanici_adi == $girilen_kullanici_adi){
    echo "Giriş başarılı";
}
else{
    echo "Yanlış kullanıcı adı ya da şifre";
}
```

B) Kodda eşitlikler parantez içine alınmalıdır.

```
if(($sifre == $girilen_sifre) || ($kullanici_adi == $girilen_kullanici_adi))
```

C) Kodda || yerine && kullanılmalıdır. || kullanıldığında şartlardan biri dahi doğru olsa şart sağlanmış olur. Hâlbuki kullanıcı adı ve şifrenin ikisinin de doğru olması gerekir.

```
if($sifre == $girilen_sifre&& $kullanici_adi == $girilen_kullanici_adi)
```

D) Kodda else kısmı kullanılmamalıdır, kod aşağıdaki gibi olmalıdır.

```
if($sifre == $girilen_sifre || $kullanici_adi == $girilen_kullanici_adi)
    echo "Giriş başarılı";
else
    echo "Yanlış kullanıcı adı ya da şifre";
```

8. Aşağıdaki kod bloğunun gerçekleştirdiği işlem nedir?

<pre>switch (\$var) { case "abc": case "def": case "hij":     echo "evet";     break; default:     echo "hayır"; }</pre>	<p>A) \$var değişkeninin değeri hij ise ekrana evet yazar, diğer durumlarda hayır yazar.</p> <p>B) \$var değişkeninin değeri abc, def ve hij ise ekrana evet yazar, diğer durumlarda hayır yazar.</p> <p>C) Kodlarda hata olduğu için hata mesajı döndürür.</p> <p>D) default değer verildiğinden her durumda ekrana hayır yazar.</p>
--	---

9. Aşağıdaki kod bloğunun ekran çıktısı nedir?

<pre>\$x = 7; \$y = 2; if(\$x&gt;\$y &amp;&amp; \$x%2==0)     \$y--; echo "\$x \$y";</pre>	<p>A) 7 2</p> <p>B) 7 1</p> <p>C) 72</p> <p>D) \$x \$y</p>
--	--

10. Aşağıdaki kod bloğunun ekran çıktısı nedir?

<pre>\$x = 1; if(!empty(\$x))     \$x++; else     \$x--; echo \$x;</pre>	<p>A) 0</p> <p>B) 1</p> <p>C) 2</p> <p>D) null</p>
--	--

11. Aşağıdaki kod bloklarından hangisi çıktı olarak ekrana 100 kez "Bu bir döngü" yazısını yazdırır?

<pre>A) \$x = 1; while(\$x&lt;=100){     echo " Bu bir döngü "; }</pre>	<pre>B) \$x = 100; while(\$x&lt;=1){     echo " Bu bir döngü "; }</pre>
<pre>C) \$x = 100; while(\$x&lt;=1){     echo " Bu bir döngü ";     \$x--; }</pre>	<pre>D) \$x = 1; while(\$x&lt;=100){     echo " Bu bir döngü ";     \$x++; }</pre>

12. 1'den 100'e kadar olan doğal sayıların toplamını (1+2+3+4+5+6+...) bulmak için yazılan aşağıdaki kod bloğunda bir hata bulunmaktadır. Bu hata hangisidir?

<pre>\$x = 1; \$toplam = 0; while(\$x&lt;=100){     \$toplam+=\$x; } \$x++; echo "Sayıların toplamı=".\$toplam;</pre>	<p>A) <math>x \leq 100</math> değil <math>x \geq 100</math> olmalıdır.</p> <p>B) <math>\\$toplam += \\$x</math>; değil <math>\\$toplam = \\$toplam + \\$x</math>; olmalıdır.</p> <p>C) Sayaç, döngünün içinde artırılmalıdır (<math>\\$x++</math>;) )</p> <p>D) <math>\\$toplam</math>'ın ilk değeri 0 değil 1 olmalıdır (<math>\\$toplam=1</math>;) )</p>
---	--

**13. Aşağıdaki kod bloğunun ekran çıktısı ne olur?**

<pre>for(\$i=0; \$i&lt;4; \$i++){     echo "X"; }</pre>	<p>A) X</p> <p>B) XX</p> <p>C) XXX</p> <p>D) XXXX</p>
---	---

**14. Aşağıdaki kod bloğunun ekran çıktısı ne olur?**

<pre>\$i = 10; do {     echo "merhaba ";     \$i++; } while(\$i &lt; 5)</pre>	<p>A) merhaba</p> <p>B) merhaba merhaba</p> <p>C) merhaba merhabamerhaba</p> <p>D) merhaba merhabamerhabamerhaba</p>
---	--

**15. Aşağıdaki kod bloğunun ekran çıktısı ne olur?**

**NOT:** explode() fonksiyonu string ifadeyi, belirtilen ayırıcı karaktere göre parçalayıp, bir dizi olarak döndürür. Bu kod için string ifade (\$metin) boşluklardan bölünmüştür.

<pre>\$metin="Bu bir cümledir."; \$kelimeler=explode(" ", \$metin); foreach(\$kelimeler as \$kelime) echo \$kelime."&lt;br /&gt;";</pre>	<p>A) Ekran çıktısı yoktur.</p> <p>B) Bu</p> <p>C) Bubircümledir</p> <p>D) Bu bir cümledir.</p>
--	---

**16. Aşağıdaki döngü çalıştırıldığında ekrana 5 kez *Merhaba* yazması beklenmesine rağmen istenilen sonucu vermemektedir. Buradaki hata nedir?**

<pre>for(\$i=1; \$i=5; \$i++){     echo "Merhaba"; }</pre>	<p>A) Sayaç (\$i), döngüde kullanılmadan önce tanımlanmalıdır.</p> <p>B) Sayaç artırılmamalı, azaltılmalıdır (\$i++ değil, \$i-- olmalı)</p> <p>C) Dönme şartı yanlıştır. <math>i \leq 5</math>; olmalıdır.</p> <p>D) Ekrana yazdırmak için echo değil, print kullanılmalıdır.</p>
--	--

**17. Aşağıdaki kod bloğunun ekran çıktısı ne olur?**

<pre>for (\$i=15; \$i!=10; \$i--){     echo "\$i "; }</pre>	<p>A) 15 14 13 12 11</p> <p>B) 11 12 13 14 15</p> <p>C) 15</p> <p>D) 10</p>
---	---

**18. Aşağıdaki döngülerden hangisi 1 ile 50 arasındaki tek sayıları ekrana yazdırır?**

<pre>A) \$i=1; while(\$i&lt;=50){     echo "\$i ";     \$i+=2; }</pre>	<pre>B) \$i=0; while(\$i&gt;=50){     echo "\$i ";     \$i+=2; }</pre>
<pre>C) \$i=0; while(\$i&gt;=50){     echo "\$i ";     \$i-=2; }</pre>	<pre>D) \$i=1; while(\$i&lt;=50){     echo "\$i ";     \$i-=2; }</pre>

19. Aşağıdaki kod bloğu çalıştırıldığında ekran çıktısı ne olur?

<pre>\$x = 0; while(\$x&lt;5){     \$x++;     if(\$x==3){         break;     }     echo "\$x "; }</pre>	<p>A) 0</p> <p>B) 1</p> <p>C) 1 2</p> <p>D) 1 2 3</p>
---	---

20. Aşağıdaki kod bloğu çalıştırıldığında ekran çıktısı ne olur?

<pre>for (\$i=3; \$i&gt;=1; \$i--) echo "&lt;h\$i&gt;Merhaba&lt;/h\$i&gt;";</pre>	<p>A) Merhaba</p> <p>B) 1Merhaba12Merhaba23Merhaba3</p> <p>C) MerhabaMerhabaMerhaba</p> <p>D) Merhaba Merhaba Merhaba</p> <p><b>Merhaba</b></p>
---	---

## **EK C- Yarı Yapılandırılmış Görüşme Formu (Öğrenciler ile)**

### **GÖRÜŞME FORMU**

- İnternet Tabanlı Programlama dersi hakkındaki düşünceleriniz nelerdir?
- İnternet Tabanlı Programlama dersi ile daha önceden almış olduğunuz diğer programlama dersleri arasında bir fark görüyor musunuz?
- wtPHP'yi ne kadar kullandınız?
- wtPHP'yi kullanım özellikleri açısından değerlendirir misiniz?
- wtPHP'yi içerik özellikleri açısından değerlendirir misiniz?
- wtPHP'de gördüğünüz eksiklikler nelerdir?
  - Bu eksiklikler nasıl giderilebilir?
- Ekleme istediğiniz bir şey var mı?

## **EK D- Yarı Yapılandırılmış Görüşme Formu (Öğretim Elemanları ile)**

### **GÖRÜŞME FORMU**

- Programlama dilleri dersini nasıl işliyorsunuz?
- Bu dersin işlenmesi için ayrılan süre sizce yeterli midir?
- Sizce okulunuzdaki laboratuvar ya da derslikler bu dersin işlenmesi için uygun mu?  
Bu konuda herhangi bir problem yaşıyor musunuz?
- Dersinizi işlerken karşılaştığınız problemler var mı?
  - Bu problem neden kaynaklanıyor olabilir?
  - Bu problemi çözmek için neler yapılabilir?
- Öğrenci profiliniz genel olarak nasıl?
- Bu derste öğrenciler en çok hangi konu/konuları öğrenmede zorluk çekiyorlar?
  - Elinizde nasıl bir materyal olsaydı bu konu/konular daha kolay anlatılabilirdi/anlaşılabilirdi?
- PHP'nin temel konularını içeren bir öğretim materyali sizce nasıl olmalıdır?
- Böyle bir materyalde öğrencilerin güdülenmesini ve motivasyonunu arttırabilecek ne tür öğeler, araçlar kullanılabilir?
- PHP'nin temel konuları ile ilgili (önceki sorularda bahsedilen özelliklerde) etkinlikler içeren bir öğretim materyalini derslerinizde kullanır mısınız?
  - Böyle bir materyalin öğrenci başarılarını arttırmada faydalı olabileceğini düşünüyor musunuz?