

# Türkiye’de Çevik Yazılım Geliştirme Üzerine Bir İnceleme

Esra Çetin<sup>1</sup>, Pınar Onay Durdu<sup>2</sup>

<sup>1</sup> Bilgisayar Mühendisliği Bölümü, Fen Bilimleri Enstitüsü, Kocaeli Üniversitesi, Kocaeli, Türkiye

<sup>2</sup> Bilgisayar Mühendisliği Bölümü, Kocaeli Üniversitesi, Kocaeli, Türkiye

<sup>1</sup>esrasolguncetin@gmail.com, <sup>2</sup>pinar.onaydurdu@kocaeli.edu.tr

**Özet.** Günümüz iş dünyasının rekabetçi koşulları özellikle yazılım endüstrisindeki organizasyonların geliştirdikleri ürünleri hızla pazara sunmalarını, aynı zamanda değişen gereksinimlere karşılık vermelerini gerektirmektedir. Çevik yaklaşımlar düşük maliyet, yüksek üretkenlik ve kalite vaatleri göz önünde bulundurularak pek çok yazılım organizasyonu tarafından son yıllarda uygulanmaya başlanmıştır. Çevik yazılım geliştirme yöntemlerinin kullanımını ve vaat ettiği avantajları ne kadar sağladığını içeren güncel durumunu belirlemeye yönelik uluslararası düzeyde çeşitli çalışmalar mevcuttur. Bu çalışma kapsamında, Türkiye yazılım endüstrisindeki organizasyonlarda çevik ve geleneksel yaklaşımların ne kadar kullanıldığı ve yazılım geliştiricilerin bu yaklaşımlarla ilgili algılarının nasıl olduğunu belirlemeye yönelik bir anket hazırlanarak çeşitli sektörlerden 74 katılımcıdan veri toplanmıştır. Katılımcılardan alınan anket sonuçlarına göre Türkiye’de çevik yöntemlerin uygulanma oranı geleneksel yöntemlerle birbirine yakındır, ancak giderek çevik yaklaşımlara yönelim gözlemlenmektedir. Yazılım geliştiriciler çevik yaklaşımların verimliliği, kaliteyi ve müşteri memnuniyetini arttırdığını, ancak daha çok küçük takımlar için uygun olduğunu ve proje yönetiminin sağlanmasının zor olduğunu düşünmektedirler.

**Anahtar Kelimeler:** Yazılım geliştirme yöntemleri, Çevik yazılım geliştirme, Geleneksel yazılım geliştirme, Türkiye’deki yazılım endüstrisi uygulamaları

## 1 Giriş

Günümüzde yazılım projelerinin birçoğu istenen başarıyla tamamlanamamaktadır [1,2]. Standish Group Bilgi Teknolojileri (BT) alanında yaşanan proje başarısızlıkları ile ilgili düzenli çalışmalar [3] yürütmektedir. Yayınladığı “Chaos” raporlarında projeler başarılı, zorlanmış (ek maliyetli) (zaman, bütçe ya da proje gereksinimlerinin tümünü karşılayamamış) ve başarısız olarak sınıflandırılmaktadır. Tablo 1’de görüldüğü üzere, 1994’ten günümüze dek olan raporların sonuçlarına göre, başarılı proje oranında artış yaşanırken başarısız proje oranlarında düşüş yaşanmaktadır [3]. Ancak

Tablo 1 dikkatlice incelendiğinde 2002'ye kadar başarısız proje oranında düşüş yaşandığı, sonrasında ise yeniden artış olduğu görülmektedir. Başarılı projelerde gözlemlenen artış ise gelişen teknoloji, sahip olunan daha fazla bilgi birikimi, daha iyi araç ve teknikler göz önünde bulundurulduğunda oldukça düşük kalmaktadır. Bunda proje karmaşıklıklarının artması ve teslimat sürelerinin kısalması gibi faktörlerin etkisi de oldukça fazladır [3]. Türkiye'de de buna benzer bir durum izlenmekte ve yazılım projelerinde başarı oranı % 50'nin altında olduğu düşünülmektedir [4].

**Tablo 1.** Chaos raporu verileri [3]

	1994	1996	1998	2000	2002	2004	2006	2009	2011
Başarılı	16%	27%	26%	28%	34%	29%	35%	32%	37%
Ek maliyet	53%	33%	46%	49%	51%	53%	46%	44%	42%
Başarısız	31%	40%	28%	23%	15%	18%	19%	24%	21%

Yazılım projelerinin hedeflenen kapsamda, zamanda ve bütçede tamamlanması organizasyonlar için çok önemlidir. Projelerin başarılı olması için en iyi uygulamaların veya sistem geliştirme modellerinin organizasyon içerisinde uygulanıyor olması gerekmektedir. Yazılım projelerinde olumsuzlukların yaşanmaması için yazılım geliştirme süreci günümüz rekabet koşullarına uygun şekilde yürütülmelidir. Bu amaçla geleneksel yazılım geliştirme yaşam döngüsü modellerine alternatif olarak çevik yöntemler [5] geliştirilmiştir. Çevik yazılım geliştirme yöntemleri hantal ve bürokratik olarak eleştirilen geleneksel, plana dayalı yazılım geliştirmeye karşılık 2001 yılında bir grup araştırmacı tarafından yazılım geliştirme için daha iyi bir yol olarak önerilmiştir. Çevik yöntemlerin yazılım kalitesini arttırdığı [6], paydaşlar arası iletişim [7] ve koordinasyonu [8] iyileştirdiği, üretkenliği arttırdığı [9] ve günümüzde yazılım sistemlerini geliştirme, sürdürme ve destekleme için en iyi yol olduğu [10] iddia edilmektedir. İddia edilen bu özelliklerine rağmen çevik yazılım geliştirme yöntemleri tüm organizasyonlar tarafından henüz uygulanmamaktadır [11].

Çevik yöntemlerin organizasyonlar tarafından ne kadar uygulandığına yönelik farklı ülkelerde gerçekleştirilmiş çalışmalar vardır. Bu çalışma kapsamında Türkiye'deki yazılım organizasyonlarında çevik ve geleneksel yaklaşımların ne kadar kullanıldığı ve yazılım geliştiricilerin bu yaklaşımlarla ilgili algılarının nasıl olduğu, geliştirilen bir anket aracılığıyla belirlenmeye çalışılmıştır.

Çalışmanın ilerleyen bölümlerinde öncelikli olarak gerçekleştirilen benzer çalışmalara değinilmektedir. Çalışmada kullanılan anketin tasarımı ve uygulama süreci 3. Bölümde açıklanmaktadır. 4. Bölümde anket bulguları ve analizleri sunulmaktadır. Son bölümde ise bulgular özetlenmekte, çalışmanın kısıtları ve gelecek çalışma önerileri verilmektedir.

## 2 İlgili Çalışmalar

Çevik yazılım geliştirme yöntemlerine yönelik ülkemizde ve yabancı ülkelerde gerçekleştirilmiş çeşitli büyüklükte çalışmalar bulunmaktadır. Çevik yazılım geliştirme ile ilgili yapılan anketlerin özeti, Tablo 2’de belirtilmektedir. Tablo 2’de izlenen çalışmalar genel olarak 2006 ve 2014 yılları arasında gerçekleştirilmiştir.

**Tablo 2.** Çevik yazılım geliştirme ile ilgili çalışmaların özeti

Araştırma Referansı	Bölge / Anketi yapan	Yıl	Katılımcı sayısı	Hedef/Odak alanı
[12]	Çeşitli ülkeler	2006	128 organizasyon - 136 yönetici	Çevik yaklaşımlar ve kullanımları önündeki engeller
	Digital Focus Firması			
[13]	Çeşitli ülkeler	2008	98 katılımcı	Çevik yazılım geliştirme adaptasyonu ve kullanımı
	Vijayasarathy ve Turk			
[14]	Avrupa	2008	13 yazılım organizasyonu	Çevik yöntemlerin durumu - XP, SCRUM
	Salo ve Abrahamsson			
[15]	Avusturya	2008	Raporlanmadı	Çevik yazılım geliştirme yöntem ve pratiklerinin durumu
	Schindler			
[16]	35 farklı ülke	2011	120 yazılım organizasyonu- 121 katılımcı	Çevik yaklaşım yöntemleri ve birlikte kullanılan araçlar
	Azizyan, Magarian, Kajko-Mattson			
[17]	Raporlanmadı	2013	Raporlanmadı	Çevik yöntemler ve yazılım geliştirme durumu
	VersionOne			
[18]	Hollanda, Endonezya	2012	279 katılımcı	Çevik yazılım geliştirme başarısı
	Warma			
[19]	Kuzey İrlanda	2010-2012	24 katılımcı + 30 katılımcı	Çevik yazılım geliştirme prensip ve pratiklerinin uygulanma değişiklikleri
	Bustard, Wilkie, Greer			
[20]	Çeşitli ülkeler	2010	108 katılımcı	Çevik projelerin başarı durumu
	Ambler			
[21]	Çeşitli ülkeler	2014	114 katılımcı	Çevik yazılım geliştirme adaptasyonu
	Ambler			
[22]	Türkiye	2011	30 katılımcı	Çevik yöntemler- SCRUM
	Baytam, Kalıpsız			
[23]	Türkiye	2013	540 katılımcı	Yazılım üretkenlik raporu
	Agile Türkiye			

Digital Focus [12] firmasının 2006 yılında gerçekleştirdiği araştırmanın sonuçlarına göre çalışmaya katılan organizasyonların % 81’i çevik yaklaşımları ya kullanmakta ya da kullanmaya başlamayı düşünmektedirler. Katılımcılardan % 51’i ise çevik yaklaşımları uygulamanın önündeki engel olarak bilgi ve beceri eksikliğini göstermektedirler.

Vijayasathay ve Turk'un [13] çevik yazılım geliştirmenin adaptasyonu ve kullanımını erken benimseyenler açısından değerlendirdikleri çalışmalarında, XP ve test-first en yaygın yazılım geliştirme yaklaşımı olarak belirtilirken, SCRUM dördüncü sırada gelmektedir. Katılımcılarının büyük çoğunluğu (%75) projelerinde çevik yöntemleri kullandıklarını ifade etmişlerdir.

Salo ve Abrahamsson [14] gömülü yazılım geliştirme organizasyonlarındaki çevik yöntemlerden özellikle XP ve SCRUM'ın durumuna belirlemek için bir anket uygulamışlardır. Çalışmanın sonucunda gömülü yazılım endüstrisinde de çevik yöntemlerin kullanımının gerçekleştiği ve bu yöntemlere karşı olumlu algının yaygınlaştığı gözlemlenmiştir.

Schindler [15] bilişim endüstrisindeki organizasyonlar içerisinde çevik yöntemlere yönelik farkındalığın bulunduğu ve gelecek planları içinde çevik yöntemler ve özellikle eş programlamayı kullanmayı planladıkları sonucunu elde etmiştir. Çevik yaklaşımlardan en yaygın kullanılanlar, XP (%46) ve SCRUM (%32,8) olarak izlenmektedir.

Azizyan, Magarian ve Kajko-Mattson [16] daha çok çevik yaklaşımlar ile kullanılan araç ihtiyacı ve kullanımına yönelik olarak gerçekleştirdikleri çalışmalarında SCRUM (%54) ve XP'nin (%32) en yaygın kullanılan çevik yöntemler olduğu sonucunu elde etmişlerdir. En yaygın kullanılan araçlar ise klasik duvar-yapışkan kağıt (%26), standart ofis araçları (Excel, vb.) (%23) ve MS Project (%8) olarak izlenmektedir.

VersionOne [17] firması 2007 yılından bu yana çevik geliştirmenin durumunu tespit etmeye yönelik araştırma raporları yayınlamaktadır. En son 2013 yılında yayınlanan raporunda çevik yöntemleri gelecek projelerde adapte etme oranı % 69'dan % 83'e çıkmış, çevik yöntemler içerisinde en yaygın kullanılan ise % 72 ile SCRUM olarak seçilmiştir. Katılımcıların yaklaşık % 90'ı çevik yöntemleri kullanmanın değişen öncelikleri yönetme becerilerini artırdığını, % 84'ü ise çevik yöntemler ile sağlanan görünürlüğün faydalı olduğunu vurgulamışlardır. En yaygın olarak kullanılan standart ofis yazılımları, Microsoft Project ve VersionOne gibi yazılımlar olarak izlenirken, en çok tavsiye edilen VersionOne (%93) olarak ön plana çıkmaktadır.

Warma [18], çevik yöntemler konusundaki farkındalığın % 79 oranında olduğunu tespit etmiş ve katılımcıların yarı yarıya çevik yazılım geliştirme projelerinde yer aldıklarını belirlemiştir. En yaygın olarak kullanılan çevik yazılım geliştirme yöntemi olarak % 90 ile SCRUM birinci sırada gelmektedir. Anket katılımcıları tarafından çevik yazılım geliştirmenin yazılım üretkenliği ve kalitesini olumlu yönde etkilediği düşünülmektedir. % 60'dan fazla katılımcı çevik projelerin başarı oranlarının % 50'den fazla olduğunu belirtmektedir.

Çevik yazılım geliştirme prensip ve pratiklerinin 2010-2012 yılları arasında Kuzey İrlanda bölgesindeki yazılım organizasyonlarında uygulanmasının ne yönde değişiklik gerçekleştirdiğini inceleyen çalışma [19], çevik yöntemlerin geleneksel çağlayan paradigmasının yerini almaya başladığını ortaya koymaktadır. Çevik yaklaşımların özellikle geliştirilen yazılımların fonksiyonelliğini ve kullanılabilirliği artırma olumlu etkisinin olduğu yönünde artan bir algı gelişmektedir.

Ambler [20,21] 2007 yılından bu yana çevik yazılım geliştirme adaptasyonu ve çevik projelerin başarı oranları konuları ile ilgili anketler yapmaktadır. 2010 yılındaki anket sonuçlarına göre, katılımcıların % 62'si çevik projeleri planlanan zamana göre teslim etmenin, % 34'ü ise hazır olduğunda teslim etmenin önemli olduğunu

belirtmişlerdir. 2014 yılında gerçekleştirilen başka bir ankete göre ise, katılımcıların yaklaşık % 15'i çevik teknikleri benimsemeye başladıklarını, % 26'sı takımlarının yarısından azında çevik teknikleri uyguladıklarını, % 49'u ise takımlarının yarısından fazlasında çevik teknikleri uyguladıklarını belirtmişlerdir.

Türkiye'de çevik yöntemlerden SCRUM ile ilgili gerçekleştirilen bir anket çalışmasının sonuçlarına [22] göre organizasyonların bu yöntemi kullanmaya başlamaları henüz erken aşamalarda yer almakta ve projeler küçük takımlar halinde gerçekleştirilmektedir. Yaygın olarak kullanılan araçlar duvar-yapışkan kağıt ya da Excel gibi yazılımlar şeklinde listelenmektedir.

Güncel olarak Agile Türkiye [23] tarafından 2013 yılında gerçekleştirilen Yazılım Üretkenlik Raporu sonuçlarına göre de Türkiye'deki projelerin % 64'ünde çevik yöntemler uygulanmaktadır. En yaygın olarak kullanılan çevik yaklaşımın da SCRUM ve türevleri olduğu bilgisi paylaşılmaktadır.

### 3 Yöntem

Bu çalışmada, Türkiye'deki yazılım organizasyonlarında çevik ve geleneksel yazılım geliştirme yöntemlerinin ne kadar kullanıldığı ve yazılım geliştiricilerin çevik yaklaşımlarla ilgili algılarının nasıl olduğu belirlenmeye çalışılmıştır. Araştırma ankete dayalı tanımlayıcı bir inceleme çalışması şeklinde gerçekleştirilmiştir. Tanımlayıcı incelemede amaç, değişkenler arası ilişkilerin belirlenmesinden çok örneklemin ne kadarının belirli bir fikirde olduğu veya bazı olayların ne sıklıkla olduğunun belirlenmesidir [24].

Bu kapsamda, yazılım geliştirme yöntemleri ile ilgili 39 maddelik bir anket hazırlanmıştır. Anket üç temel bölümden oluşmaktadır. Birinci bölüm katılımcılarla ilgili demografik bilgileri almaya yönelik hazırlanmıştır. Bu bölüm sonunda katılımcılara organizasyonlarında hangi yazılım geliştirme yöntemini kullandıkları sorularak, çevik yöntemleri seçenlerin ikinci bölümde yer alan çevik yazılım geliştirme odaklı soruları yanıtlamaları sağlanırken, geleneksel yöntemi seçenlerin ise üçüncü bölümde yer alan geleneksel yöntemle yönelik soruları cevaplamaları sağlanmıştır. Bu sorular, geleneksel yazılım geliştirme yöntemlerini kullanan organizasyonlarda yer alan katılımcıların çevik yaklaşımlar konusundaki görüşlerini almaya yönelik olarak hazırlanmıştır. Çevik yöntemlere yönelik hazırlanan 23 maddeden 19 tanesi Ambler'in [25] çalışmasından Türkçeye çevrilerek adapte edilmiştir. Geleneksel yöntemlere yönelik hazırlanan sorulardan 6 maddeden üç tanesi Baytam ve Kalıpsız'ın [22] çalışmasından ve 1 tanesi yine Ambler'in [25] çalışmasından adapte edilmiştir. Hazırlanan sorular konu ile ilgili daha önce anket tasarlamış ve bu konu da eğitim veren kişiler tarafından gözden geçirmeleri için iletilmiş ve verdikleri geribildirimler doğrultusunda güncellemeler yapılmıştır. Böylece çalışmanın yüzeysel ve içerik geçerliliği sağlanmıştır [26]. Anket sorularının tamamına çevrim içi olarak ulaşılabilmektedir [27].

Hazırlanan anket, Surveey isimli bir çevrimiçi anket sağlama hizmeti kullanılarak 2014 Şubat, Mart ve Nisan ayları boyunca üç ay süreyle katılımcıların erişimine açılmıştır. Katılımcılar çeşitli sosyal ağlar, e-mail grupları kullanılarak davet edilmişlerdir. Bilişim sektöründe yer alan birçok farklı kuruma ulaşılarak, 74 yazılım geliştiricinin görüşleri alınmıştır.

## 4 Anket Sonuçları ve Bulguları

### 4.1 Demografik özellikler

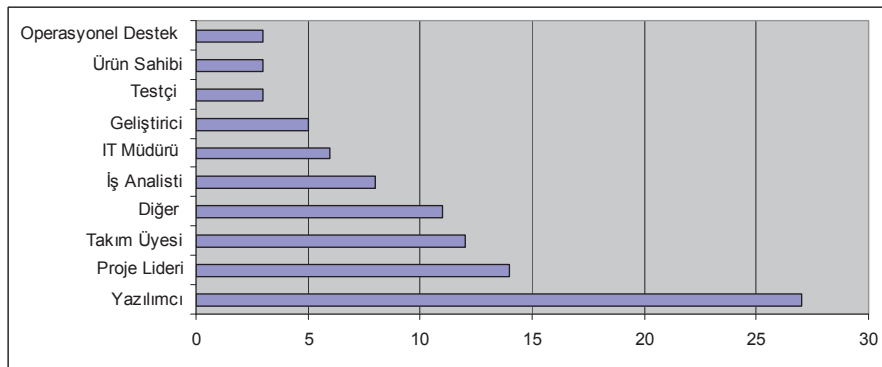
Ankete katılan 74 katılımcının % 31'i kadınlardan, % 69'u ise erkeklerden oluşmaktadır. Anket katılımcılarının % 42'si üniversite % 33'ü ise yüksek lisans ve % 17'si doktora derecesine sahipken, %8'i ön lisans mezunudur.

Katılımcıların yazılım alanındaki iş tecrübesinin dağılımı Tablo 3'te yıl bazında gösterilmektedir. Ortalama değer yaklaşık 8 yıldır. Katılımcıların % 34 ve % 40'ının, iş deneyimi 5 ila 10 yıl ve altındadır. Bu da katılımcıların çoğunluğunu genç bir kitlenin oluşturduğunu bize göstermektedir. Tablo 3'te ayrıca katılımcıların BT sektöründeki toplam iş tecrübeleri de gösterilmektedir. Katılımcıların bu sektördeki iş tecrübesi genel iş tecrübeleri ile büyük oranda paralellik göstermektedir.

**Tablo 3.** Katılımcıların yıl bazındaki tecrübeleri

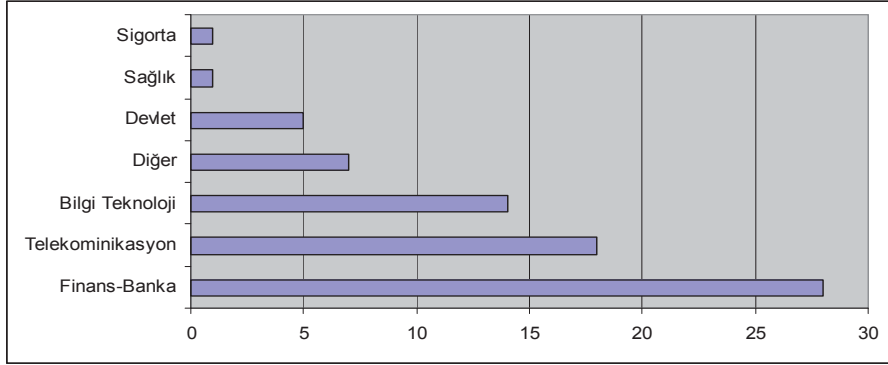
	Ort.	<=5	6-10	11-15	16-24	>25
		%	%	%	%	%
Katılımcıların iş deneyimleri	8,2	34	40	20	5	1
Katılımcıların BT sektöründeki toplam iş tecrübeleri	8,2	37	35	22	5	1
Katılımcıların buldukları kurumdaki iş tecrübesi	4,7	71	19	9	0	1

Katılımcıların çalıştıkları firmalarda görev aldıkları pozisyonlar Şekil 1'de gösterilmektedir. Ankette yer alan ve 10 kişiden fazla katılımcının tercih ettiği "diğer" alanı ise; genel müdür yardımcısı, sistem mühendisi, sistem yöneticisi gibi pozisyonları içermektedir. Bu soru çoklu seçimli bir soru olduğu için, katılımcıların 4'ü 2 farklı pozisyonda, yine 4'ü 3 farklı pozisyonda görev aldığını belirtmiştir. 3 kişi ise 4 farklı pozisyonda görev almaktadır. 63 katılımcı ise sadece tek pozisyonda görev aldıklarını bildirmiştir.



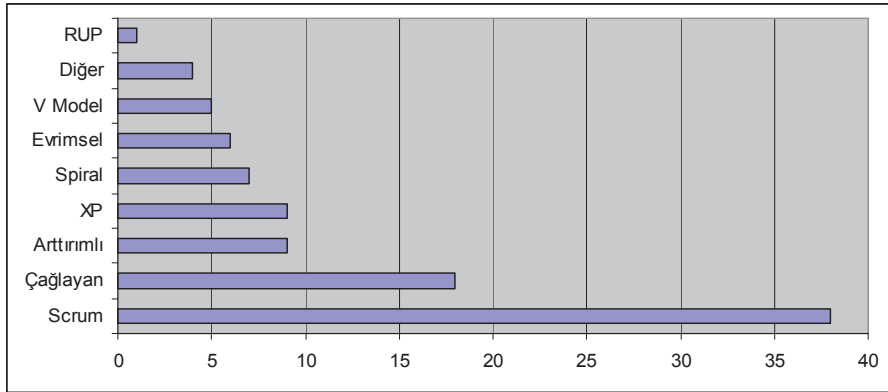
**Şekil 1.** Katılımcıların çalıştıkları firmalarda görev aldıkları pozisyonlar

Şekil 2’de görüldüğü üzere, katılımcıların görev aldıkları firmaların hizmet verdiği sektörleri sırasıyla Finans-Banka, Telekomünikasyon ve Bilgi Teknolojileri sektörleri oluşturmaktadır. 5 katılımcı Devlet sektöründe hizmet verirken, Sağlık ve Sigorta sektörleri ise ankette en az katılımcının yer aldığı sektörler olarak izlenmektedir. Ankette yer alan diğer alanı ise; robotik teknolojiler, hayvancılık, savunma sanayi, otomasyon sistemleri, üniversite ve sigorta sektörlerinin tümünü içermektedir.



Şekil. 2. Katılımcıların görev aldıkları organizasyonların hizmet verdiği sektörler

Katılımcıların organizasyonlarında hangi yazılım geliştirme yöntemini kullandıkları sorusuna alınan yanıtlara göre, % 54 çevik yazılım geliştirme, % 46 ise geleneksel yazılım geliştirme yöntemlerini kullanmaktadır. Şekil 3’te görüldüğü üzere, en yüksek tercih çevik yazılım geliştirme yöntemleri içerisinde Scrum üzerine yoğunlaşmaktadır. Toplam 38 katılımcı Scrum yazılım geliştirme modelini kullanırken, 18 katılımcı geleneksel yazılım geliştirme yöntemlerinden Çağlayan modelini kullandıklarını belirtmişlerdir.

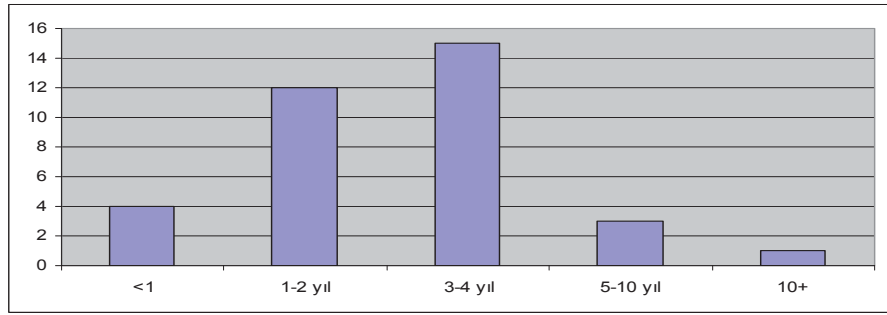


Şekil. 3. Organizasyonlarda kullanılan yazılım süreç modelleri

## 4.2 Çevik Yazılım Geliştirme Yöntemleri

Katılımcılara organizasyonlarında hangi yazılım geliştirme yöntemlerini kullandıkları sorulmuştur. Çevik yöntemleri kullandıklarını belirten katılımcılara anketin ikinci bölümünde yer alan sorular yöneltilmiştir.

Organizasyonlarda çevik yaklaşımlar ile proje yürütülme süresine yönelik bulgular Şekil 4'te gösterilmektedir. Anket katılımcılarının % 43'ü 3-4 yıldır, % 34'ü ise 1-2 yıldır çevik yaklaşım projelerinde görev aldıklarını belirtmektedir. Grafikte izlenebildiği üzere çevik yaklaşımların geçmişi birçok şirket için 3-4 yılı geçmemektedir.



Şekil. 4. Organizasyonlarda çevik yöntemler ile proje yürütülme süresi (yıl bazında)

Organizasyonlarda çevik yaklaşımların kullanıldığı proje sayısı için en yüksek değer Tablo 4'te görüldüğü üzere % 56 ile 1 ile 5 projededir. % 23'lük kısım ise şimdiye kadar en çok 6 ile 10 projede çevik yaklaşımı kullandıklarını belirtmektedir. Bu verilere göre çevik yaklaşımın birçok şirket için yeni uyum sağlamaya çalıştıkları bir model olduğu söylenebilir.

Çevik yaklaşım projelerinde çoğunlukla SCRUM metodu ile çalışan katılımcıların görev aldıkları en büyük takım %23 ile 6 ile 10 kişilik takımlar ve yine % 23 ile 51 kişi üzerinde olan takımlar olarak izlenmektedir. Anket katılımcılarının çevik yaklaşımlarda 51 kişi ve üzeri takımlar ile çalışmasının sebebi olarak, birçoğunun 1000 ve üzeri personel çalıştıran firmalarda görev aldıkları gösterilebilmektedir.

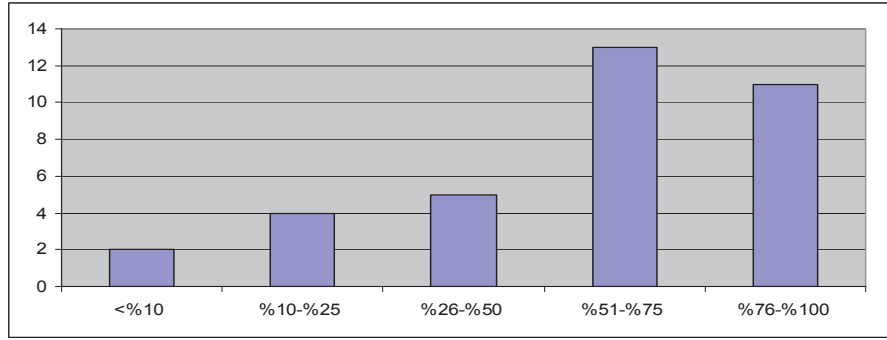
Çevik projelerde genellikle tercih edilen takım büyüklüğü, % 51 ile 6 - 10 kişi arasındadır. Bu verilere göre, çevik yaklaşımların genellikle en fazla 10 kişilik takımlar için tercih edildiğini söyleyebiliriz.

Tablo 4. Çevik yöntem kullanılan proje bilgileri

	1-5	6-10	11-20	21-50	51+
	%	%	%	%	%
Kurumunuzda çevik yaklaşımları kullandığımız proje sayısı	56	23	6	9	6
Kurumunuzda çevik yaklaşımlar ile çalışan en büyük projede çalışan sayısı	14	23	20	20	23
Kurumunuzda çevik projelerde ortalama proje takımlarındaki kişi sayısı	34	51	9	3	3



Organizasyonlar içerisinde yazılım geliştirme sürecini destekleme için kullanılan en yaygın araçlar olarak VersionOne ve TFS (Team Foundation Server - Takım ortak bilgi yönetim sunucusu) öne çıkmıştır. Katılımcılara çevik yazılım geliştirme yöntemleri ve başarısına yönelik sorular yöneltilmiştir. Şekil 5'te izlendiği üzere % 69 katılımcı çevik yaklaşımlar için proje başarı oranlarının % 50'nin üzerinde olduğunu düşünmektedir.



Şekil 5. Kurumunuzda gerçekleştirilen çevik projelerin başarı oranı

Tablo 5'te yazılım geliştiricilerin çevik yaklaşımlarla ilgili görüşleri çeşitli ifadeler ile belirlenmeye çalışılmıştır. Katılımcıların çoğu, çevik yaklaşımın verimliliği, kaliteyi ve müşteri memnuniyetini artırdığını; ancak sadece bir arada çalışan küçük takımlar için uygun olduğunu ve proje yönetiminin zor olduğunu belirtmektedir.

Tablo 5. Katılımcıların çevik yazılım geliştirme yöntemlerine yönelik düşünceleri

	Katılıyor	Kararsız	Katılmıyor
	%	%	%
Çevik yaklaşımlar verimliliğimizi arttırdı.	43	23	34
Çevik yaklaşımlar üretilen sistemlerinin kalitesini arttırdı.	34	37	29
Çevik yaklaşımlar geliştirme maliyetini azalttı.	40	26	34
Çevik yaklaşımlar müşteri memnuniyetini arttırdı.	34	40	26
Çevik takımlar yeterli dokümantasyon hazırlamaz.	31	38	31
Çevik takımlar yeterli mimari tasarım yapmaz (address architecture) yoktur.	20	26	54
Çevik takımlar yeterli analiz yapmaz.	17	29	54
Çevik takımlar yeterli planlama yapmaz.	23	23	54
Çevik yazılım geliştirme disiplinsizdir.	14	20	66
Çevik yazılım geliştirme sadece geçici bir hevestir.	20	11	69
Çevik yazılım geliştirme sadece bir arada çalışan takımlara uygundur.	43	26	31
Çevik yazılım geliştirme sadece küçük takımlar içindir.	37	29	34
Çevik yazılım geliştirme sonucu düşük kalitedir.	11	29	60
Çevik yazılım geliştirme projelerini yönetmek zordur.	43	20	37

### 4.3 Geleneksel Yazılım Geliştirme Yöntemleri

Organizasyonlarında geleneksel yöntemleri kullandıklarını belirten katılımcılara geleneksel yöntemin avantaj ve dezavantajları ile çevik yöntemleri kullanım isteklerine yönelik sorular sorulmuştur. Tablo 6'da görüldüğü üzere geleneksel yöntemleri kullanan katılımcıların büyük çoğunluğu (%42) kullandıkları yöntemin takip edilebilirlik özelliğini avantajlı özellik olarak vurgulamışlardır. Diğer taraftan yöntemin esnekliği ise az tercih edilen (%16) özellikler arasında yer almıştır.

Katılımcılara takip ettikleri yöntem ne tür bir özellik eklemek istedikleri sorulmuş; alınan yanıtlar çoğunlukla esneklik (%42) ve kullanılabilirlik (%25) özellikleri olmuştur. Bir yazılım geliştirme destek aracında olması gereken özellikler sorulduğunda ise esneklik katılımcıların büyük çoğunluğu (%40) tarafından en çok tercih edilen özellik olarak tanımlanmıştır.

Takip ettikleri yazılım sürecinin istemedikleri özellikleri olarak ise en fazla sürecin kolay kullanılabilir olmaması, süreç tekrarlarının desteklenmemesi, gereksiz dokümantasyon yapılması gibi konuları listelemişlerdir. Geleneksel süreci takip eden katılımcılara organizasyonlarının çevik yöntemleri ne kadar sürede adapte edebileceklerini düşündükleri sorulduğunda ise, adaptasyon süreci için en fazla ortalama 1 ila 2 yıl arasında bir süre tahmin etmişlerdir.

**Tablo 6.** Geleneksel yöntem kullanılan proje bilgileri

	Esneklik	Kullanılabilirlik	Takip edilebilirlik	Öğretici bilgiler	Diğer
	%	%	%	%	%
Geleneksel yöntemin avantajlı özelliği	16	29	42	3	10
Geleneksel yöntem eklenmesi gereken özellik	42	25	8	8	17
Bir yazılım geliştirme araçındaki olması gereken özellik	40	24	28	4	4

## 5 Sonuç, Kısıtlar ve Gelecek Çalışmalar

Bu çalışma kapsamında Türkiye'deki organizasyonların çevik yöntemleri ne oranda kullandıkları ve yazılım geliştiricilerin çevik yöntemlere yönelik düşünceleri belirlenmeye çalışılmıştır. Anket sonuçlarına göre, Türkiye'de çevik yöntemleri kullanan organizasyonların bu yöntemleri kullanma süreleri ortalama 2 yıldan fazladır; bu da Türkiye'de çevik yöntemlerin kullanımının çok yaygın olmadığını, ancak bu yöntemlerin kullanımına yönelik bir eğilim olduğunu göstermektedir. En yaygın olarak kullanılan yaklaşım olarak SCRUM ön plana çıkmaktadır. Sonuçlar Agile Türkiye'nin [23] 2013 yılında gerçekleştirmiş olduğu çalışmanın sonuçları ile önemli oranda örtüşmektedir.

Çalışmaya katılan yazılım geliştiricilerin çevik yaklaşımlar konusundaki tutumları oldukça olumludur. Çevik yaklaşımların verimliliği, kaliteyi ve müşteri

memnuniyetini arttırdığını düşünmektedirler. Ancak daha çok küçük takımlar için uygun olduklarını ve proje yönetiminin sağlanmasının zor olduğunu düşünenler de çoğunluktadır.

Anket sonuç değerlendirmeleri, tarafsızlığın korunabilmesi adına, tespit edilen istatistiki bulgulara göre yapılmıştır; ancak pek çok deneysel çalışmada olduğu gibi bu çalışmada da önemli kısıtlar yer almaktadır. Öncelikli olarak ankete cevap veren katılımcılar Türkiye çapında rastgele örneklem ile seçilememiştir. Böylece sonuçlar tüm Türkiye'deki duruma genellenemez, ancak Türkiye'deki genel eğilimlere yönelik yazılım geliştirici ve araştırmacılara bir bakış açısı sağlayabilir. Gelecek çalışmalar kapsamında araştırmanın daha geniş bir örnekleme kapsayacak şekilde daha uzun zaman aralığında yaygınlaştırılması ya da anketin periyodik aralıklar ile tekrar uygulanması bulguların genişletilmesi açısından faydalı olacaktır.

## 6 Kaynaklar

1. Marnewick, C., Labuschagne, L.: Factors that influence the outcome of information technology projects in South Africa: an empirical investigation. In: Acta Commercii, vol. 9, No 1, 78-89. (2009)
2. Galorath, D.: Software Project Failure Costs Billions. Better Estimation & Planning Can Help. <http://www.galorath.com/wp/software-project-failure-costs-billions-better-estimation-planning-can-help.php> (2012)
3. Dominguez, J.: The Curious Case of the CHAOS Report 2009. <http://www.projectsmart.co.uk/the-curious-case-of-the-chaos-report-2009.html> (2009)
4. NTVMSNBC News Site, Yazılım projeleri çöpe gidiyor. <http://www.ntvmsnbc.com/id/25327974/> (2012)
5. Agile Alliance, The Agile Manifesto, <http://www.agilealliance.org/the-alliance/the-agile-manifesto/> (2001)
6. Livermore, J. A.: Factors that impact implementing an agile software development methodology. In: SoutheastCon Proceedings, IEEE, pp. 82- 86 (2007)
7. Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., Still, J.: The impact of agile practices on communication in software development. Empirical Software Engineering, 13 (3), pp. 303-337. (2008)
8. Strode, D. E., Huff, S. L., Hope, B., Link, S.: Coordination in collocated agile software development projects. The Journal of Systems and Software, 85 (6), pp. 1222-1238. (2012)
9. Sutherland, J., Jakobsen, C.R., Johnson, K.: Scrum and CMMI Level 5: The Magic Potion for Code Warriors. In: Proceedings of the 41st Annual Hawaii International Conference on System Sciences (2008)
10. Suganya, G., Mary, S.A.: Progression towards agility: A comprehensive survey. In: 2010 International Conference on Computing Communication and Networking Technologies, pp. 1-5. (2010)
11. Srinivasan, J., Lundqvist, K.: Using Agile Methods in Software Product Development: A Case Study. In: Sixth International Conference on Information Technology: New Generations, pp. 1415 – 1420. (2009)
12. Preuss, D.H.: Digital Focus Unveils Market Survey Results at Agile 2006. <http://www.infoq.com/news/Digital-Focus-Unveil-Survey-2006> (2006)

13. Vijayasaraty, L. R., Turk, D.: Agile software development: A survey of early adopters. *Journal of Information management*, 19 (2), pp. 1-8. (2008)
14. Salo, O., Abrahamsson, P.: Agile methods in European embedded software development organizations: a survey on the actual use and usefulness of Extreme Programming and Scrum. *Software, IET*, 2 (1), pp. 58-64. (2008)
15. Schindler, C.: Agile software development methods and practices in Australian IT-industry: Results of an empirical study. In: *International Conference on Computational Intelligence for Modelling Control & Automation*, pp. 321-326. (2008)
16. Azizyan, G., Magarian, M.K., Kajko-Mattson, M.: Survey of agile tool usage and needs. In: *Agile Conference 2011*, pp. 29-38. (2011)
17. VersionOne, 7<sup>th</sup> annual state of agile development survey. <http://www.versionone.com/pdf/7th-Annual-State-of-Agile-Development-Survey.pdf>
18. Warma, R.: *The Success of Agile Software Development*, Yayınlanmamış Mezuniyet Tezi, Fontys University of Applied Sciences, Eindhoven (2012)
19. Bustard, D., Wilkie, G., Greer, D.: The maturation of agile software development principles and practice: Observations on successive industrial studies in 2010 and 2012. In: *20th IEEE International Conference and Workshops on the Engineering of Computer Based Systems (ECBS)*, pp.139-146. (2013)
20. Ambler, S.: 2010 Agile Project Success Rates Survey Results. <http://www.ambysoft.com/downloads/surveys/AgileSuccess2010.pdf> (2010)
21. Ambler, S.: 2014 Agile Adoption Mini-Survey. <http://www.ambysoft.com/downloads/surveys/AgileAdoption2014.pdf> (2014)
22. Baytam, V., Kalıpsız O.: *Scrum Yazılım Geliştirme Metodolojisi için Yönetim Sistemi Tasarımı ve Gerçeklenmesi*. Yayınlanmamış Master Tezi, Yıldız Teknik Üniversitesi, İstanbul (2011)
23. Agile Türkiye, 2013 Software Productivity Report. <http://www.agileturkey.org/> (2013)
24. Oppenheim, A.N.: *Questionnaire design, interviewing and attitude measurement*, Pinter Publishers (1996)
25. Ambler, S.: Agile Adoption Rate Survey Results. <http://www.ambysoft.com/downloads/surveys/AgileAdoption2008.pdf> (2008)
26. Black, J. A., Champion, D. J.: *Methods and issues in social research*. John Wiley & Sons, Inc., New York (1976)
27. Çevik Yaklaşımlar Anket Çalışması, <https://docs.google.com/file/d/0B6Uc3xMvXFRkTlJWb2lQM2VZeDQ/edit> (2014)