

Mikroservisler için İşgücü Kestirimi: Bir Durum Çalışması

Effort Prediction for Microservices: A Case Study

Hüseyin Ünlü
Bilgisayar Mühendisliği Bölümü
İzmir Yüksek Teknoloji Enstitüsü
İzmir, Türkiye
huseyinunlu@iyte.edu.tr

Tuna Hacaloğlu
Bilişim Sistemleri Mühendisliği
Atılım Üniversitesi
Ankara, Türkiye
Enformatik Enstitüsü
Orta Doğu Teknik Üniversitesi
Ankara, Türkiye
tuna.hacaloglu@atilim.edu.tr

Onur Leblebici
Univera Bilgisayar Sistemleri
İzmir, Türkiye
onur.leblebici@univera.com.tr

Onur Demirörs
Bilgisayar Mühendisliği Bölümü
İzmir Yüksek Teknoloji Enstitüsü
İzmir, Türkiye
onurdemirors@iyte.edu.tr

Özetçe—Yazılım büyüklük ölçümü efor, maliyet ve takvim kestirimi gibi önemli proje yönetimi adımlarını gerçekleştirmek için bir girdi olması açısından oldukça kritiktir. İşlevsel büyüklük ölçüm (İBÖ) yöntemleri işlevsel gereksinimler üzerinden yazılım yaşam döngüsünün erken fazlarında uygulanabilir olmaları, sistematik ve tekrar edilebilir bir yöntem sunmaları açısından faydalıdır. Ancak çevik yöntemlerle geliştirilen projelerde, özellikle gereksinimlerin dokümantasyonu Şelale Modeli gibi geleneksel yöntemlere nazaran daha minimum seviyede tutulduğundan ve proje ilerledikçe detaylandırıldığından, erken fazlarda gereksinimlerden İBÖ yöntemlerinin ölçüm bileşenlerini ayırtmak zorlayıcı olabilmektedir. Bunun yanında, var olan İBÖ yöntemleri veri-tabanı ve işlem odaklı olmaları dolayısıyla veriden daha çok davranışa doğru evrilmekte olan günümüz mimari yapıları ile tam olarak uyumlu olmamaktadırlar. Bu çalışmada, hem çevik yöntemlerle geliştirilen hem de mikroservis mimarisinin kullanıldığı bir örnek durum çalışmasında COSMIC İBÖ yöntemi ile olay-odaklı büyüklük ölçümü yöntemi kullanılarak proje büyüklüğü ölçülmüş ve efor kestirim modelleri oluşturulmuştur. Projede gereksinimlerin kısıtlı detay seviyesinde yazılmasından ötürü ölçerler her iki yöntemi uygulamakta da zorlanmıştır. Ancak, olay-odaklı yöntemin, COSMIC İBÖ yöntemine göre daha az hata ile eforu tahminlediği görülmüştür.

Anahtar Kelimeler—COSMIC, efor, büyüklük ölçümü, çevik yazılım geliştirme, agile, olay odaklı ölçüm

Abstract—Software size measurement is critical as an input to perform important project management processes such as effort, cost and schedule estimation. Functional size measurement (FSM) methods are beneficial in terms of being applicable in the early phases of the software life cycle over functional requirements and providing a systematic and repeatable method. However, in agile organizations, it can be challenging to separate measurement components of FSM methods from requirements in the early phases as the documentation is kept to a minimum compared to traditional methods such as the Waterfall Model and is detailed as the project steps. In addition, the existing FSM methods are not fully compatible with today's architectural structures, which are from being data-driven and to evolve into a behaviour-oriented structure. In this study, we performed a case study which includes a project developed with agile methods and using

microservice-based architecture to compare the effectiveness of COSMIC FSM and event-based software size measurement. For this purpose, we measured the size of the project and created effort estimation models based on two methods. The measurers had difficulty in applying both methods due to the limited detail level of the requirements in the project. However, the event-based method was found to estimate effort with less error than the COSMIC FSM method.

Keywords—COSMIC, effort, size measurement, agile software development, event-based measurement

I. GİRİŞ

2001 yılında geleneksel yazılım geliştirme modellerine karşılık bir manifesto olarak ortaya çıkan Çevik Yazılım geliştirme yöntemi, günümüzde bir çok yazılım şirketi ve takımları tarafından sıklıkla kullanılmaktadır [1]. Çevik Yazılım geliştirme, bireyler ve onların iletişimlerini, süreçler ve araçlardan; çalışan yazılım ürünü ortaya çıkarmayı, detaylı dokümantasyondan; müşteri ile iletişim halinde olmayı, sözleşmelerden; ve değişikliklere cevap verebilmeyi, bir planı takip etmekten daha değerli görmektedir [2]. Bununla birlikte, bir yazılım projesinin başarılı olarak değerlendirilmesi için müşterilerin gereksinimlerini karşılamanın yanında, zamanında ve öngörülen bütçe dahilinde ortaya çıkmış olması beklenmektedir [3]. Bu bağlamda, yazılım proje yönetimi açısından hem takvim hem de maliyet kestirimi için tutarlı efor kestiriminde bulunmak oldukça kritik bir işidir. Efor kestirimi formel yöntemler ve uzman yargıları olmak üzere iki ana grupta değerlendirilebilir [4]. Çevik yazılım geliştirmede işlevsel büyüklük ölçümü (İBÖ) gibi formel yöntemleri kullanarak bir kestirim yapmak yerine daha öznel yapıda olan Hikaye Puanı (HP) ve Kullanım Durumu Puanı (KDP) kestirimleri kullanılmaktadır [5], [6]. Ancak çevik yazılım geliştirme, yazılım projeleri için önemli bir adım olan efor kestirimi açısından bir takım sorunlarla karşılaşmaktadır.

Özellikle Hikaye Puanının (HP) çevik projelerde kestirim yapan açısından efor olarak tanımlanması hedeflenmiş olmasına ve yoğunlukla kullanılmasına rağmen, büyüklük [7], karmaşıklık [8] gibi farklı algılar yaratması, öznel [9] ve kullanıldığı takımın dinamiklerine özgü olup takımlar arası

kullanımının zor olması [10] yüzünden eleştirildiği gözlemlenmektedir [11].

Buna karşın işlevsel büyüklük ölçümü kullanımı çevik projelerde çeşitli çalışmalarda denenmiş ve kimi zaman başarılı sonuçlara [12]–[14] ulaşılmış olsa da zorlukları çeşitli çalışmalarda [7],[13] belirtilmiştir. Hacaloglu ve Demiors [15] tarafından 2019 yılında bir çoklu durum çalışması kapsamında ikinci nesil bir İBÖ yöntemi olan COSMIC İBÖ yönteminin uygulanması sırasında karşılaşılan zorluklar sunulmuştur. Yazarlar çalışmayı çevik yazılım geliştirilen üç ayrı kurumun projelerinde gerçekleştirmişlerdir. Bu zorluklar şu şekilde özetlenebilir: COSMIC İBÖ yöntemi veri hareketi sayma üzerine kuruludur [16] ve veri sayma hareketi veri yapısı hakkında bir miktar bilgi sahibi olmayı gerektirir. Bu bağlamda COSMIC İBÖ yöntemi, ilişkisel veri analizi gibi geleneksel veri analizi yöntemlerinin kullanımını ölçüm açısından önermektedir [17]. Ancak günümüzde, veri akışının yerine istek ve cevapların çoğunlukta olduğu projelerde artık geleneksel veri analizi yöntemleri kullanımı git gide azalmıştır [15].

Ayrıca, çevik yöntemlerle geliştirilen yazılım projelerinde gereksinimler kullanıcı hikayeleri gibi daha kısa formatlarda yazılmakta ve gereksinim dokümantasyonu oldukça asgari düzeyde tutulmaktadır. Bu durumun, COSMIC İBÖ ölçümü yöntemi kullanımını, ölçüm bileşenlerini ayırt etmekte zorluklar yaratması dolayısıyla zorlaştırdığı saptanmıştır [15].

Özellikle yeni nesil mimarilerin “monolitler”de olduğu gibi veri tabanı ve işlem odaklı yapıdan mikroservis ve benzeri mimarilerde kullanılan daha davranışsal ve olay odaklı yapılar evrilmeleri, özellikle çevik projelerde zaten zorlukları bulunan işlevsel büyüklük ölçümü uygulamayı daha da zorlaştırmıştır [18]–[20].

Yazılım büyüklüğü ölçümü efor kestiriminin yanında yöneticiler için projeyi yönetmek, iyileştirmek, projeler arası kıyaslamalar yapabilmek açılarından da önemlidir [21].

Bu çalışmada, hem yeni nesil mimarilerin hem de çevik yazılım geliştirme yönteminin uygulandığı bir yazılım firmasında hem işlevsel büyüklük ölçümünü uygulamayı hem de yeni nesil mimariler için yenilikçi bir yaklaşım olarak önerilen olay odaklı ölçüm yönteminin (Olay Puanı) başarısını efor kestirim modelleri oluşturarak gözlemlemek hedeflenmiştir.

Bu bildirinin devamında, 2. Bölümde Araştırma Yöntemi, 3. Bölümde durum çalışmasının uygulanış detayları, 4. Bölümde sonuçlar, 5. Bölümde çıktılarının tartışılması ve 6. Bölümde sonuçlar ve ileriki çalışmalar için öneriler sunulmuştur.

II. ARAŞTIRMA YÖNTEMİ

Bu çalışmada, keşfedici durum çalışması araştırma yöntemine başvurulmuştur [22]. Durum çalışması tasarımı ve durum tanıtımı içeren durum çalışması detayları aşağıdaki bölümlerde açıklanmıştır.

A. Durum Çalışması Tasarımı

Bu çalışmada, çevik yazılım geliştirme yöntemini uygulayan organizasyonlarda yazılım büyüklük ölçümü ve efor kestirimi süreçlerinde karşılaşılan zorlukları belirlenmesi ve bu belirlenen zorlukları aşabilmek ve süreci iyileştirmek için önerilerde bulunulması amacı ile çevik yazılım geliştiren bir organizasyonda durum çalışması yapılmış ve sonuçları

sunulmuştur. Bu amaç doğrultusunda araştırma soruları aşağıdaki şekilde belirlenmiştir:

- Çevik yazılım geliştiren organizasyonlarda yeni nesil mimarilerin yazılım büyüklük ölçümü ve efor kestirimi süreçlerinde karşılaşılan zorluklar nelerdir?
- Olay Puanı ve COSMIC yöntemleri çevik yazılım geliştiren organizasyonlarda yenilikçi mimariler ile geliştirilen projelerin büyüklük ölçümü ve efor kestiriminde ne derece başarılıdır?

Durum Seçim Kriteri: Bu çalışma için seçilecek projenin en önemli kriteri çevik yazılım geliştirme yöntemi ile geliştirilmiş olmasıdır. Ayrıca, projenin ölçülebilir olması için belirli düzeylerde gereksinimlerinin belirli olması gerekmektedir. Efor kestiriminin yapılabilmesi için ise projenin parçalarına ait efor verisinin kaydedilmiş olması gerekmektedir.

Veri Toplama Yöntemi: Ölçerler projenin ölçülebilir parçalarına ait COSMIC işlevsel büyüklük, Olay Puanı büyüklük, gerçek efor ve ölçüm sürecinde karşılaşılan zorlukları toplayıp kaydetmeyi hedeflemiştir.

Ölçüm Planı: Projenin COSMIC İşlevsel Büyüklük Ölçümü ve Olay Puanı Büyüklük ölçümlerinin bu projenin herhangi bir sürecinde görev almamış bağımsız iki ayrı ölçer tarafından ölçerlerin birinin COSMIC diğerinin ise Olay Puanı büyüklük ölçüm yöntemi ile yapacağı şekilde planlanmıştır. Bu sayede, her ölçer gerçekleştirdiği ölçümlere eşit uzaklıkta olma şansına sahip olacak ve yaptığı varsayımlarda tutarlı olabilecektir. Projenin ölçümlerinin her iki yöntemde de tutarlı olabilmesi için ayrıca ölçüm sonrasında ölçerler tarafından toplantı yapılması planlanmıştır. Ölçümler yapıldıktan sonra ise efor kestirim modellerinin iki ölçerin de katılacağı toplantıda oluşturulması planlanmıştır. COSMIC yöntemini kullanan ölçerin 3 sene COSMIC ölçüm tecrübesi bulunmaktadır. Olay Puanı yöntemini kullanan ölçer ise bu yöntemi geliştiren bir araştırmacı olmakla beraber bu yöntemin geçerliliğini çeşitli projelerde değerlendirmiştir.

COSMIC İşlevsel Büyüklük Ölçüm yöntemi veri hareketini baz alır. Bu amaç doğrultusunda yöntem, fonksiyonel kullanıcı gereksinimlerinin (functional user requirements – FUR) ve bu gereksinimlere ait fonksiyonel süreçler (functional process – FP) ile bu süreçlerin içinde yer alan veri hareketlerinin (data movement – DM) belirlenmesini önerir [16]. Bu veri hareketleri tek bir İlgi Nesnesi (Object of Interest - OOI) tanımlayan veri gruplarının akışı olarak belirlenir [17]. Bir başka deyişle, bir hareketin veri hareketi (DM) olarak belirlenebilmesi için tek bir OOI ile ilişkili olması gerekmektedir. DM 4 ayrı formda görülebilir: Girdi (Entry – E), Çıktı (Exit – X), Okuma (Read – R) ve Yazma (Write – W) [16]. COSMIC İBÖ yöntemini detaylı bir şekilde açıklayan manuel [16] bulunmaktadır. Ayrıca, Abran tarafından özet bir şekilde geliştirilen bir manuel de bulunmaktadır [23]. Ancak, özet manuele dayalı yapılan ölçüm sonuçlarının aynı projenin yazılım büyüklüğünü ölçerken farklılık yaratabileceği görülmüştür [24].

Olay odaklı ölçüm yöntemi [25], gereksinimlerden büyüklük ölçüm yapmayı hedeflemektedir. Yazılım geliştirme yaşam döngüsünün erken fazlarında ölçüm yapabilmeyi hedeflemesi dolayısıyla, yöntemi kullanarak erken dönemde efor, maliyet ve takvim kestirimi gerçekleştirilebilmesi beklenmektedir.

Yöntem temelinde proje gereksinimlerini olay odaklı süreç modelleme yöntemini kullanarak modellemeyi ve oluşan olay sayılarını saymayı hedefler. Ölçüm birimi olarak olay puanı kavramını ortaya atar. Ölçüm sonucunda bir gereksinimin büyüklüğü içerdiği olayların toplam sayısı olarak bulunur. Dolayısıyla bir yazılım projesinin büyüklüğü gereksinimlerinin toplam sayısı kadardır [25].

B. Durum Tanıtımı

Durum çalışması için seçilen proje, işletmelerin, sağladıkları saha servis operasyonlarını kontrol etmelerine ve daha verimli şekilde yönetmelerine olanak sağlama amacı ile geliştirilmektedir. İlk olarak iklimlendirme sektöründe faaliyet gösterecek olup, ilerleyen zamanlarda farklı iş alanlarında da hizmet vermesi hedeflenmiştir. Farklı paket seçenekleri ile çeşitli müşteri kitlesine hitap etmekte olup mobil ve web versiyonları ile çeşitli fonksiyonlar sağlamaktadır. Proje, olay odaklı mimaride geliştirilmektedir.

III. DURUM ÇALIŞMASININ UYGULANMASI

Sürecin ilk aşaması olarak organizasyon ile bir toplantı düzenlenmiş ve bu toplantıda organizasyonda uygulanan süreç hakkında bilgi edinilmiştir. Organizasyonda Scrum metodolojisi uygulanmakta ve Azure DevOps kullanılmaktadır. Sahadan gelen talepler bir taslak ile alınıp planlama sonrasında İş Analistleri bu maddeler ile ilgili çalışmaya başlamaktadır. İş Analisti tarafından çıkarılan gereksinimler açılan Epik maddelerine bağlanmaktadır. Çevik yazılım geliştirmede epik, müşterilerin veya son kullanıcıların ihtiyaçlarına/taleplerine göre belirli görevlere bölünebilen bir çalışma bütünüdür. Proje yöneticisi ve yazılım geliştiriciler ile birlikte yapılan bir toplantıda Epik maddeleri Ürün İş Listesi Öğeleri'ne (Product Backlog Item – PBI) parçalanmakta ve PBI'lar projede yer alan yazılım geliştiricilerine atanmaktadır. Her bir PBI için organizasyon tarafından geliştirilen bir yöntem ile efor hesaplaması yapılmaktadır. Ayrıca, her bir yazılım geliştiricisinin geçmiş performansı göz önünde bulundurularak bir iş gününde eritebileceği efor o kişinin performans puanı olarak hesaplanmakta ve bu performans puanına göre ekip üyelerine Sprint içinde PBI ataması yapılmaktadır.

Toplantıda edilen bilgiler doğrultusunda ilk olarak PBI'ların COSMIC ve Olay Puanı yöntemleri ile ölçülebilir olup olmadığı incelenmiştir. Bunun için, her PBI iki ölçer tarafından açıklama ve ekler başta olmak üzere incelenmiş ve ölçüm için yeterli açıklama (gereksinim dökümanı, kullanım senaryosu, vb.) içerip içermediği not edilmiş ve bir tablo oluşturulmuştur. Ancak, bu aşamada çevik yazılım geliştiricinin bir sonucu olarak birçok PBI için açıklama bulunamamıştır. Bu inceleme sürecinde ürün iş listesi öğelerinin Epikler (Epic) altında gruplandırıldığı görülmüştür. Ürün iş listesi öğelerinin fonksiyonel büyüklük ölçümü için fazla teknik detayda parçalanmış olması göz önünde bulundurularak epikler üzerinden bir inceleme yapılması planlanmıştır.

Epik incelemesi öncesi organizasyon ile bir toplantı yapılmış ve bu toplantıda ölçüm için incelenecek epikler belirlenmiştir. İlk olarak, seçilen epiklere ait ölçüm için yeterli açıklama olup olmadığı incelenmiş ve epiklere ait açıklama bulunamamıştır. Epik incelemesinde ikinci aşama olarak her bir epik altındaki ürün iş listesi öğeleri incelenmiş ve her iki yöntem ile de ölçülebilir olup olmadığı not edilmiştir. Bu incelemenin sonucunda ölçülecek ürün iş listesi öğeleri epikler altında belirlenmiştir.

Sonraki aşamada, birinci ölçer COSMIC ile ikinci ölçer ise Olay Puanı yöntemi ile belirlenen öğeleri ölçüm değerleri tablo üzerinde kaydetmiştir. Ölçüm sonrasında ölçerler tarafından düzenlenen toplantıda her bir öğe üzerinden geçilerek yapılan varsayımlar tartışılmış ve sonuca bağlanarak ölçümler güncellenmiştir. Ayrıca, değişiklik olarak belirlenen öğeler ölçüm listesinden çıkarılmıştır.

Efor kestirim modeli oluşturabilmek için öğelerin gerçek efor verisine ihtiyaç duyulmaktadır. Ancak, organizasyon ile yapılan toplantıda gerçek efor bilgisinin Sprint ve kişi bazlı tutulduğu öğrenilmiştir. Epikler ile Sprints karşılaştırıldığında ise bir epikin birçok Sprint'e ait öğe bulundurduğu gözlenmiştir. Bu nedenle, her bir ürün iş listesi öğesine ait efor verisine ihtiyaç duyulmuştur. Öğe bazlı gerçek efor bulunmadığından Sprint bazlı gerçek efor ve organizasyon tarafından gerçekleştirilen öğe bazlı tahmini efor verisi kullanılarak her bir PBI için efor verisi hesaplanmıştır.

Bu veriler kullanılarak epiklere dayalı efor kestirim modeli COSMIC ve Olay Puanı yöntemi için iki ölçerin de katıldığı bir toplantıda oluşturulmuştur. Model üzerinde 2 epikin oluşturulan modele göre uç değer olduğu görülmüş (modele uyum sağlamayan değerler) ve bu epiklere ait öğeler organizasyon ile düzenlenen bir toplantıda tartışılmıştır. Bu toplantı sonucunda bir epik altındaki öğelerin değişiklik olduğu ve ekibin bu işi sıklıkla yaptığı; bu nedenle çok hızlı gerçekleştiği, diğer epik altındaki öğelerin ise ekip için yeni bir iş olduğunu ve daha önce bu işi hiç yapmadıkları belirtilmiştir. Sonuç olarak, model üzerinde bu iki epik uç değer olarak görülmüş ve çıkarılarak modeller yeniden oluşturulmuştur. Oluşturulan modeller ve bu modeller ile yapılan tahminler aşağıdaki bölümde verilmiştir.

IV. SONUÇLAR

Ölçüm 16 Epik üzerinden yapılmıştır. Seçilen Epikler minimum 1, maksimum 6 ve ortalama 2.56 PBI içermektedir. Çalışmanın sonuçları 2 aşamada sunulmuştur. Öncelikle bütün ölçülen veri seti üzerinden korelasyon ve regresyon analizi yapılmış modelin hassasiyeti değerlendirilmiştir. Daha sonra proje yöneticisi ile bu sonuçlar tartışılmış ve 2 tane epik'in uç değer oldukları öğrenilmiştir. Ardından uç değerler elenmiş ve korelasyon ve regresyon analizi tekrar çalıştırılmıştır. Bu sonuçlar A ve B bölümlerinde sunulmuştur.

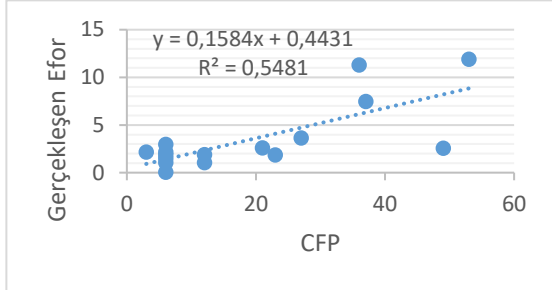
A. Uç değerler elenmeden önce yapılan analiz sonuçları

Büyük ölçümü ve efor arasındaki ilişkiyi gözlemlemek için korelasyon analizi yapılmıştır. Hem COSMIC Fonksiyon Nokta (COSMIC Function Point – CFP) ile efor arasında ($R=0,74$) hem de Olay puanı ile efor arasında ($R=0,76$) güçlü korelasyon olduğu saptanmıştır [26].

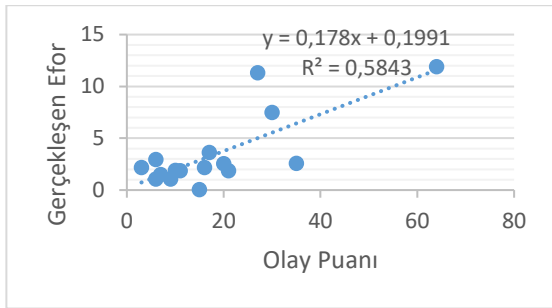
Korelasyon analizinin ardından, büyüklük değerleri ile kurulan efor kestirim modellerinin başarısını değerlendirmek üzere regresyon analizi yapılmıştır. Şekil 1'de CFP ve efor dağılımı; Şekil 2 de ise olay puanı ve efor dağılımı sunulmuştur.

Kurulan efor kestirim modellerine göre olay puanı ile kurulan efor kestirim modelinin R^2 değerinin ($R^2=0,58$), CFP ile kurulan efor kestirim modelindeki R^2 değerinden ($R^2=0,55$) az farkla daha fazla olduğu gözlenmiştir. Yani, olay puanının efordaki %58 oranındaki değişikliği, CFP'nin ise %55 oranındaki değişikliği açıkladığı söylenebilir.

Efor kestirimi modellerinin başarısını değerlendirmek üzere Bağıl Hata Büyüklüğü (MRE), Ortalama Bağıl Hata Büyüklüğü (MMRE) ve PRED(30) metrikleri kullanılmıştır. Bu sonuçlar CFP ile yapılan efor kestirim modeli için Tablo II'de, Olay Puanı ile yapılan efor kestirim modeli için de Tablo III'de sunulmuştur. Buna göre, MMRE değeri her iki büyüklük ölçümü ile kurulan efor modeli için oldukça yüksek oldukları, modellerin ikisinin de tahminleme başarılarının çok düşük oldukları saptanmıştır.



Şekil 1. Efor kestirim modeli (CFP vs Gerçekleşen Efor – Uç değerler elenmeden)



Şekil 2. Efor kestirim modeli (Olay Puanı vs Gerçekleşen Efor – Uç değerler elenmeden)

TABLO II. ANALİZ SONUÇLARI (COSMIC – UÇ DEĞERLER ELENMEDEN)

Epik No	CFP	Gerçekleşen Efor (Kişi-Gün)	Tahminlenen Efor (Kişi Gün)	MRE
1	53	11.9	8.84	0.26
2	6	2.96	1.39	0.53
3	49	2.56	8.20	2.20
4	37	7.48	6.30	0.16
5	6	1.05	1.39	0.33
6	6	0.05	1.39	26.87
7	3	2.16	0.92	0.57
8	21	2.59	3.77	0.46
9	6	1.49	1.39	0.06
10	23	1.87	4.09	1.19
11	6	1.86	1.39	0.25
12	27	3.64	4.72	0.30
13	36	11.3	6.15	0.46
14	12	1.89	2.34	0.24
15	6	2.17	1.39	0.36
16	12	1.07	2.34	1.19
MMRE				2.21
PRED (30)				0.38

Elde edilen sonuçlar, proje yöneticisi ile tartışılmıştır. Proje yöneticisi ile yapılan görüşmede 6 ve 13 no'lu epiklerin diğerlerinden farklı talepleri içerdikleri anlaşılmıştır. 6 nolu epikin sıklıkla yapılan bir değişiklik olduğundan boyutuna göre gerçekleşen eforun az olduğu, 13 nolu epikte yer alan gereksinimin ise yazılım takımı tarafından daha önce yapılmamış bir iş olduğundan dolayı boyutuna göre gerçekleşen eforun fazla olduğu öğrenilmiştir. Bu iki epik uç değer olarak kabul edilmiş ve analizler bu uç değerler çıkarıldıktan sonra tekrarlanmıştır.

TABLO III. ANALİZ SONUÇLARI (OLAY PUANI – UÇ DEĞERLER ELENMEDEN)

Epik No	Olay Puanı	Gerçekleşen Efor (Kişi-Gün)	Tahminlenen Efor (Kişi Gün)	MRE
1	64	11.9	11.59	0.03
2	6	2.96	1.27	0.57
3	20	2.56	3.76	0.47
4	30	7.48	5.54	0.26
5	6	1.05	1.27	0.21
6	15	0.05	2.87	56.38
7	3	2.16	0.73	0.66
8	35	2.59	6.43	1.48
9	7	1.49	1.45	0.03
10	21	1.87	3.94	1.11
11	11	1.86	2.16	0.16
12	17	3.64	3.23	0.11
13	27	11.3	5.01	0.56
14	10	1.89	1.98	0.05
15	16	2.17	3.05	0.40
16	9	1.07	1.80	0.68
MMRE				3.95
PRED (30)				0.44

B. Uç değerler elendikten sonra yapılan analiz sonuçları

Yeni durumda, CFP ile efor arasında korelasyon katsayısı $R=0,73$ olarak bulunurken Olay Puanı ile efor arasında bu değer $R=0,87$ olarak bulunmuştur. Bu korelasyon değerleri yüksek korelasyon olarak kabul edilmektedir [26]. Buna göre Olay puanının, CFP'ye göre efor ile daha yüksek korelasyona sahip olduğu gözlemlenmiştir. CFP ile efor dağılımı ve Olay Puanı ile efor dağılımı sırasıyla Şekil 3 ve Şekil 4'te sunulmuştur.

Doğrusal regresyon analizi kullanılarak efor kestirim modelleri oluşturulmuştur. Bu durumda olay puanını $R^2=0,75$ değeri ile efordaki %75 oranındaki değişikliği açıklayabildiği görülmüştür. Bu değer CFP için %54 olarak bulunmuştur.

Efor kestirim modellerinin hassasiyetine bakıldığında olay puanı ile oluşturulan modelin ortalama bağıl hata değerinin $MMRE=0,41$ olduğu ve CFP ile oluşturulan efor kestirim modelinin ortalama bağıl hata değerinin ise $MMRE=0,51$ olduğu görülmüştür.

Hastings ve Sajeev [27] 0,20'den az olan MMRE değerlerini tahminleyici; 0,20 ile 0,50 arasında bulunan MMRE değerlerini kabul edilebilir bulurken, 0,50'nin üzerindeki MMRE değerleri ile oluşturulan modelleri kabul edilemez olarak değerlendirmektedir. Bu durumda, olay puanı

MMRE=0,41 değeri ile kabul edilebilir bir hata oranı sunmaktadır.

Olay puanı ile oluşturulan modelin PRED(30) değeri CFP ile oluşturulan değerden PRED(30) değerinden yüksek olsa da, bu değerler başarılı kestirim için yeterli olmamaktadır. MMRE ve PRED(30) değerleri sırasıyla Tablo IV ve Tablo V'te CFP ve Olay Puanı için görülebilir.

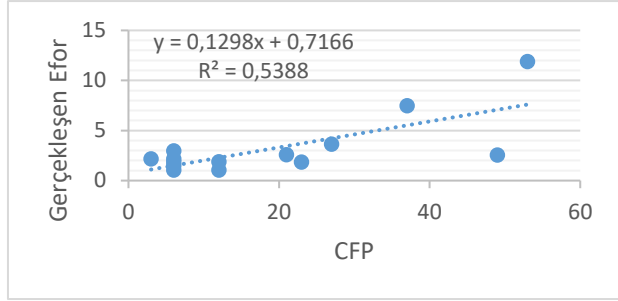


Fig. 1. Efor kestirim modeli (CFP vs Gerçekleşen Efor – Uç değerler elendikten sonra)

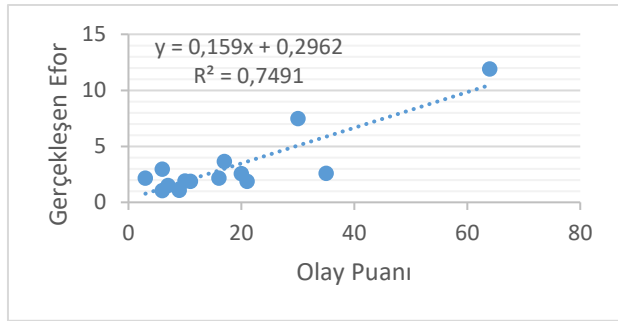


Fig. 2. Efor kestirim modeli (Olay Puanı vs Gerçekleşen Efor – Uç değerler elendikten sonra)

TABLO IV. ANALİZ SONUÇLARI
(COSMIC– UÇ DEĞERLER ELENDİKTEN SONRA)

Epik No	CFP	Gerçekleşen Efor (Kişi-Gün)	Tahminlenen Efor (Kişi Gün)	MRE
1	53	11.9	7.60	0.36
2	6	2.96	1.50	0.49
3	49	2.56	7.08	1.76
4	37	7.48	5.52	0.26
5	6	1.05	1.50	0.42
7	3	2.16	1.11	0.49
8	21	2.59	3.44	0.33
9	6	1.49	1.50	0.00
10	23	1.87	3.70	0.98
11	6	1.86	1.50	0.20
12	27	3.64	4.22	0.16
14	12	1.89	2.27	0.20
15	6	2.17	1.50	0.31
16	12	1.07	2.27	1.13
MMRE				0.51
PRED (30)				0.31

TABLO V. ANALİZ SONUÇLARI
(OLAY PUANI – UÇ DEĞERLER ELENDİKTEN SONRA)

Epik No	Olay Puanı	Gerçekleşen Efor (Kişi-Gün)	Tahminlenen Efor (Kişi Gün)	MRE
1	64	11.9	10.47	0.12
2	6	2.96	1.25	0.58
3	20	2.56	3.48	0.36
4	30	7.48	5.07	0.32
5	6	1.05	1.25	0.19
7	3	2.16	0.77	0.64
8	35	2.59	5.86	1.26
9	7	1.49	1.41	0.05
10	21	1.87	3.64	0.94
11	11	1.86	2.05	0.10
12	17	3.64	3.00	0.18
14	10	1.89	1.89	0.00
15	16	2.17	2.84	0.31
16	9	1.07	1.73	0.61
MMRE				0.41
PRED (30)				0.43

V. TARTIŞMA

Bu çalışmada, yeni nesil mimariler için yenilikçi bir yaklaşım olarak önerilen olay odaklı ölçüm yönteminin (Olay Puanı) başarısı ile COSMIC İBÖ başarısı karşılaştırılmıştır. Olay Puanı yöntemi ile daha hassas bir efor kestirimi yapılmış olmasına rağmen her iki yöntem için de yeteri kadar hassas bir efor kestirim modeli oluşturulamadığı gözlemlenmiştir. Bunun nedeni ise çevik yöntemlerle geliştirilen projelerin özelliklerine bağlı olarak aşağıdaki şekilde açıklanmıştır.

Organizasyonda çevik yazılım geliştirme ilkelerine bağlı olarak iş analistleri tarafından belirlenen gereksinimler yazılım ekibinin de katıldığı toplantılarda ürün iş listesi öğelerine (PBI) parçalamakta ve her bir PBI bir yazılım geliştiriciye atanmaktadır. Ancak, çevik yazılım geliştiricinin doğası gereği bu öğelerin bir çoğunda gerekli açıklama bulunmamaktadır. Ayrıca, her yazılım geliştirici bir Sprint içinde birden çok PBI üstlenebileceği için fonksiyonel büyüklük ölçümü için fazla teknik düzeyde olabilmektedir. İşlevsel büyüklük ölçümü yöntemleri için gereksinimlerin fonksiyonel seviyede yazılması, kullanıcıların işlevsel adımlarının belirtilmesi gerekmektedir. Organizasyonda, epikler fonksiyonel düzeyde olup ürün iş listesi öğelerine bölünmüş olmasına rağmen gereksinimler epik düzeyinde açıklayıcı olmadığı için ölçüm her iki yöntemle de daha teknik seviyede yapılmıştır. Bu durum, ölçümde olası sapmaları mümkün kılmıştır. Örneğin, ön yüz ile ilgili bir iş ile arka yüz ile ilgili bir iş aynı gereksinime bağlı olmasına ve boyutu aynı olmasına rağmen efor olarak farklılık gösterebilir. Bu konuda, organizasyona epiklerin fonksiyonel düzeyde açıklama içermesi veya gereksinim dokümanı ile ilişkilendirilmesi önerisinde bulunulmuştur.

Efor kestiriminin istenen düzeyde hassas sonuçlanmamasının bir diğer nedeni ise organizasyonda efor

verisinin Sprint bazlı tutulması olarak görülmektedir. Ölçüm, PBI'lar üzerinden yapıldığı için bu PBI'ların gerçek efor verisine gereksinim duyulmuş; ancak, PBI bazlı efor verisi tüm öğelerde bulunmadığı ve epikler farkı Sprintlere ait öğeler içerdiği için Sprint bazlı gerçek efor verisi ve organizasyonun PBI bazlı tahmini efor verisi kullanılarak her bir PBI için efor hesaplaması yapılmış ve bu hesaplanan efor, model girdisi olarak kullanılmıştır. Bu nedenle, COSMIC ve Olay Puanı yöntemi ile gerçekleştirilen efor kestiriminin organizasyonun efor kestirimi sonuçlarına göre daha düşük hassasiyette olması beklenen bir sonuçtur. Efor kestirim modellerinin doğru bir şekilde oluşturulabilmesi için çevik yazılım geliştiren organizasyonlarda gerçek efor verisinin PBI gibi en küçük öğeler için de kaydedilmesinin önemi büyüktür.

Çevik yazılım geliştirme ilkelerini benimsemiş ve organizasyonlarda yenilikçi mimarilerde yazılım büyüklük ölçümü ve efor kestirimi süreçlerinde karşılaşılan zorlukların yanında Olay Puanı ve COSMIC yöntemlerinin bu organizasyonlarda ölçüm için uygunluğunun tartışılması da bu çalışmada amaçlanmıştır. COSMIC İBÖ yöntemi ile ölçüm yapan ölçer teknik detaydaki gereksinimlerin ölçümü esnasında ilgi nesnesi ve veri gruplarını belirlemede zorlandığını belirtmiştir. Olay Puanı yöntemini kullanan ölçer ise bu yöntemin getirdiği kolaylık sayesinde veri akışlarını düşünmeden, ilgi nesnesi ve veri gruplarını belirlemeden olaylara odaklanarak sayım yaptığı için bir zorluk yaşamadığını belirtmiştir.

VI. SONUÇ VE GELECEK ÇALIŞMALAR

Nesnel bir işlevsel büyüklük ölçüm yöntemi olan ve veri hareketi sayma üzerine kurulu olan COSMIC günümüz projelerinde kullanımı bazı zorluklar içermektedir. Günümüz projelerinde geleneksel veri analizi yöntemleri kullanımının git gide azalması ve yazılım tasarımın nesnelere yerine olaylar bazında oluşturulması bu zorluğun temel nedenleridir. Olay Puanı büyüklük ölçüm yöntemi ise veri hareketini saymadan oluşan olay sayılarını saymayı hedefler. Bu çalışmada, COSMIC ve Olay Puanı yöntemlerinin başarısı karşılaştırılmıştır. Ayrıca, çevik yazılım geliştiren ve yenilikçi mimariler kullanan organizasyonlarda tutarlı bir efor kestirim modeli oluşturabilmek için iki öneride bulunulmuştur: (1) gereksinimlerin fonksiyonel düzeyde açıklama içermesi ve (2) gerçekleşen eforun Sprint bazlı kaydedilmesine ek olarak ürün iş listesi öğeleri seviyesinde de kaydedilmesi. Önerilerin yanı sıra, iki yöntemin kullanım kolaylığı da incelenmiş ve benzer projelerde Olay Puanı yönteminin daha kolay ve hızlı bir ölçüm sağladığı görülmüştür.

Bu çalışmanın ışığında daha tutarlı bir efor kestirim modeli oluşturabilmek için yapılan önerilerle iyileştirilen süreç sonucunda ölçümlerin tekrarlanması planlanmıştır. Ayrıca, günümüz çevik yazılım geliştiren organizasyonlarında sıklıkla kullanılan ürün iş listesi öğeleri (PBI) bazında ölçüm yapılabilmesi için bir yöntem ihtiyacı duyulduğu da saptanmıştır. Çalışmanın gelecek hedeflerinin arasında aşamalı bir şekilde detaylandırılan çevik yazılım gereksinimleri için erken büyüklük ölçümü ve efor kestirimi için yöntem araştırmaları da bulunmaktadır.

REFERENCES

- [1] L. Barroca, H. Sharp, D. Salah, K. Taylor, and P. Gregory, "Bridging the gap between research and agile practice: an evolutionary model," *Int. J. Syst. Assur. Eng. Manag.*, vol. 9, no. 2, pp. 323–334, 2018.
- [2] M. Fowler and J. Highsmith, "The agile manifesto," *Softw. Dev.*, vol. 9, no. 8, pp. 28–35, 2001.
- [3] T. S. G. International, "Standish Grup Chaos Report," 1995.
- [4] M. Jørgensen, B. Boehm, and S. Rifkin, "Software development effort estimation: Formal models or expert judgment?," *IEEE Softw.*, vol. 26, no. 2, pp. 14–19, 2009.
- [5] M. Usman, E. Mendes, F. Weidt, and R. Britto, "Effort estimation in agile software development: a systematic literature review," in *Proceedings of the 10th International Conference on Predictive Models in Software Engineering*, 2014, pp. 82–91.
- [6] C. Ertaban, S. Gezgin, S. Bağrıyanık, E. Albey, and A. Karahoca, "Cevik Yöntemlerde Cosmic İsv Puanı ve Hikaye Puanının Birlikte Kullanımı (Using Cosmic Function Points and Story Points Together in Agile Frameworks)," in *UYMS*, 2017, pp. 378–390.
- [7] S. Kang, O. Choi, and J. Baik, "Model-based dynamic cost estimation and tracking method for agile software development," in *2010 IEEE/ACIS 9th International Conference on Computer and Information Science*, 2010, pp. 743–748.
- [8] M. M. Hohman, "Estimating in actual time [extreme programming]," in *Agile Development Conference (ADC'05)*, 2005, pp. 132–138.
- [9] H. Huijgens and R. van Solingen, "A replicated study on correlating agile team velocity measured in function and story points," in *Proceedings of the 5th International Workshop on Emerging Trends in Software Metrics*, 2014, pp. 30–36.
- [10] L. Buglione and S. Trudel, "Guideline for sizing agile projects with COSMIC," *Proc. IWSM/MetriKon/Mensura, Stuttgart, Ger.*, 2010.
- [11] T. Hacaloğlu and O. Demirörs, "Challenges of using software size in agile software development: A systematic literature review," 2018.
- [12] E. Ungan, N. Cizmeli, and O. Demirörs, "Comparison of functional size based estimation and story points, based on effort estimation effectiveness in SCRUM projects," in *Proceedings - 40th Euromicro Conference Series on Software Engineering and Advanced Applications, SEAA 2014*, 2014.
- [13] C. Commeyne, A. Abran, and R. Djouab, "Effort Estimation with Story Points and COSMIC Function Points-An Industry Case Study," 2016.
- [14] M. Salmanoğlu, T. Hacaloğlu, and O. Demirörs, "Effort Estimation for Agile Software Development: Comparative Case Studies Using COSMIC Functional Size Measurement and Story Points," *ACM Int. Conf. Proceeding Ser.*, 2017.
- [15] T. Hacaloğlu and O. Demirörs, "Measurability of Functional Size in Agile Software Projects: Multiple Case Studies with COSMIC FSM," in *2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 2019, pp. 204–211.
- [16] COSMIC, "The COSMIC Functional Size Measurement Method Measurement Manual (The COSMIC Implementation Guide for ISO/IEC 19761:2017) Version 4.0.2," 2017.
- [17] COSMIC, "Guideline for Sizing Business Application Software v1.3," 2017.
- [18] B. Bilgin, H. Unlu, and O. Demirörs, "Analysis and Design of Microservices: Results from Turkey," in *2020 Turkish National Software Engineering Symposium (UYMS)*, 2020, pp. 1–6.
- [19] H. Unlu, B. Bilgin, and O. Demirörs, "Türkiye'deki Yazılım Organizasyonlarının Mikroservis Tabanlı Mimaride Uyguladığı Analiz ve Tasarım Yöntemleri Üzerine Bir Araştırma," *EMO Bilim. Dergi*.
- [20] H. Unlu, S. Tenekeci, A. Yıldız, and O. Demirörs, "Event Oriented vs Object Oriented Analysis for Microservice Architecture: An Exploratory Case Study," in *47th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 2021.
- [21] A. Abran, C. Symons, C. Ebert, F. Vogelesang, and H. Soubra, "Measurement of software size: Contributions of cosmic to estimation improvements," in *The International Training Symposium, At Marriott Bristol, United Kingdom*, 2016, pp. 259–267.
- [22] R. K. Yin, *Case study research and applications: Design and methods*. Sage publications, 2017.
- [23] A. Abran, "Software Development Velocity with COSMIC Function Points," *Common Softw. Meas. Int. Consort.*, 2019.
- [24] T. Hacaloğlu, H. Unlu, O. Demirörs, and A. Abran, "COSMIC Light vs COSMIC Classic Manual: Case Studies in Functional Size Measurement," in *IWSM-Mensura*, 2020.
- [25] T. Hacaloğlu, "Event Points: A Software Size Measurement Model, Phd. Thesis," Middle East Technical University, 2021.
- [26] P. Schober, C. Boer, and L. A. Schwarte, "Correlation coefficients: appropriate use and interpretation," *Anesth. Analg.*, vol. 126, no. 5, pp. 1763–1768, 2018.
- [27] T. E. Hastings and A. S. M. Sajeev, "A vector-based approach to software size measurement and effort estimation," *IEEE Trans. Softw. Eng.*, vol. 27, no. 4, pp. 337–350, 2001.