

Yazılım Projelerinde Bütünleşik Efor Tahmini için Farklı Veri Tabanlarının Senkronizasyonu

İlknur Gür Nalçacı¹ ve Kazım Kıvanç Eren² ve Burak Bilge³

¹ Ar-Ge Merkezi Yöneticisi, İdea Teknoloji Çözümleri, İstanbul, Türkiye

² Ar-Ge Mühendisi, İdea Teknoloji Çözümleri, İstanbul, Türkiye

³ Proje Uygulama Uzmanı, İdea Teknoloji Çözümleri, İstanbul, Türkiye

ilknur.gur@ideateknoloji.com.tr

kivanc.eren@ideateknoloji.com.tr

burak.bilge@ideateknoloji.com.tr

Özet. Yazılım hizmeti veren firmalar müşterileri ile olan ilişkilerini yönetmek ve içerde yazılım süreçlerini izleyebilmek adına birbirinden bağımsız proje yönetim araçlarından faydalanmaktadır. Farklı uygulamaların kullanılması, platformlar arası senkronizasyon problemlerine sebep olmaktadır. Uçtan uca süreç bazlı efor tahminlerinin yapılabilmesi için projenin farklı aşamalarının yönetildiği bağımsız uygulamaların senkronize edilmesi deneyimi bu yazının konusunu oluşturmaktadır. Bu çalışma İdea Teknoloji Çözümleri TEYDEB 1501 destekli "NOTICE: Katmanlı Mimari Yazılım Projelerinde Efor Tahmin Modeli" projesi kapsamında gerçekleştirilmiştir

Anahtar Kelimeler: Müşteri Talep Yönetimi, Yazılım Proje Yönetimi, Proje Yönetim Araçları, Efor Tahmini

Synchronization of Different Databases for Integrated Effort Estimation in Software Projects

Abstract. Software services provider companies benefit from independent project management tools in order to manage their relationships with customers and to monitor software processes inside. Usage of different applications causes synchronization problems. Synchronizing the independent applications by which the different stages of the project are managed, is necessary for making end-to-end process-based effort estimates. Experience of the synchronizing process is the subject of this article. This study was carried out within the scope of the project named "NOTICE: Effort Estimation Model for Layered Architectural Software Projects" which is supported by TUBITAK TEYDEB 1501, developed in Idea Technology Solutions.

Keywords: Customer Demand Management, Software Project Management, Project Management Tools, Effort Estimation

1 Giriş

Yazılım projelerinin efor tahmininin tutarlılığı müşteri ve firmanın ilişkilerinin gidişatı açısından önemli bir yere sahiptir. Yazılım geliştirme teknolojisinin sürekli değişen senaryolar içinde olması, firmaların kullandığı yazılım dilleri, altyapısı personel kadrosu gibi birçok etkene göre efor konusundaki tahminlerde kayda değer farklılıklara sebep olmaktadır.

Yazılım maliyeti hesaplanmasında birçok metot geliştirilmesine rağmen firma için en uygun olanın seçilmesi ise yeni bir problemi beraberinde getirmektedir. Mevcutta efor hesaplanmasında birçok model/metot geliştirilmiş ve modeller sınırlı bir veri seti üzerinden oluşturulmuştur. Her metodun zayıf ve güçlü yönleri mevcuttur. Model başarımları firmalara göre farklılık gösterebilmektedir. Bu sebeple projeye ve firmaya özgü parametrelerle modeller geliştirilmesinin daha tutarlı sonuçlar vereceği öngörülmektedir.

Proje yazılım yaşam döngüsü incelendiğinde talep sürecinin tetikleyici olduğu; analiz ve yazılım süreçlerinden oluştuğu gözlenmiştir. Bir projeye ait bir efor tahmini yapılabilmesi için bu süreçlerin bütünlük şeklinde ele alınması gerektiği gereksinimi belirlenmiştir.

Firmamız bünyesinde müşteri taleplerinin alındığı, analizinin gerçekleştiği CA [18] ve yazılım süreçlerinin yönetildiği JIRA [14] iki ayrı sistem mevcuttur. Projeye ait farklı süreçlerin yönetildiği ve farklı birimlerin kullandığı bu iki sistemin arasında herhangi bir entegrasyon bulunmamaktadır.

Geliştirilecek olan efor tahmin modeli; süreci bütünlük ele alabilmek adına her iki sistemden gelecek olan verilerle eğitilecektir. Yazılım yaşam döngüsüne ait uçtan uca bir efor tahmini yapabilmek için birbirinden bağımsız olan bu iki aracın senkronizasyonunun yapılması bu yazının konusu olacaktır.

Bu çalışma yazılım yaşam döngüsünde müşteri talepleri kaynaklı değişiklikler veya geliştirmelerin hem müşteri tarafından daha şeffaf bir şekilde takip edilmesi, hem de yazılım geliştirici tarafından daha hızlı sürece alınmasını hedefleyen ve firmaya özgü efor tahmini yapılmasını amaçlayan "NOTICE: Katmanlı Mimari Yazılım Projelerinde Efor Tahmin Modeli" projesi kapsamında gerçekleştirilmiştir.

2 Literatür Taraması

2.1 Yazılım Efor Tahmini

Literatürde yazılım efor tahmini ile ilgili çok çeşitli çalışmalar bulunmaktadır. En temel iki yöntem Yapıcı Maliyet Modeli (Constructive Cost Model – COCOMO) ve İşlev Noktası (Function Point – FP) yöntemleridir [1]. COCOMO yöntemi, Boehm tarafından geliştirilmiştir [2]. Model yazılan kod miktarının belirli parametreler kullanan matematiksel bir fonksiyondur. İşlev Noktası yöntemi ise yazılım probleminin işlevselliğine göre önsel olarak problemin zorluğu ve karmaşıklığını ölçmektedir [1]. Bu yöntemler algoritmik yöntemler olarak adlandırılmaktadır [3]. Algoritmik

yöntemler belirli parametreleri kullanarak efor değerinin çıktı olarak elde edildiği matematiksel bağıntılar olarak tanımlanabilir. Algoritmik Olmayan Yöntemler olarak adlandırılan diğer efor tahmin yöntemleri ise geçmiş projelerden faydalanılarak güncel bir yazılım efor tahmin problemini çözmeye çalışmaktadır. Uzman kararı yöntemi [4], analogi tabanlı yöntemler, price-to-win [2] gibi daha eski yöntemler ile günümüzde gittikçe kullanımı yaygınlaşan makine öğrenmesine dayalı [5] ve bulanık tabanlı yöntemler [6] bu grup altında gösterilebilir.

Literatürdeki çalışmalarda belli başlı veri setleri sıklıkla kullanılmakta ve farkı çalışmalar için karşılaştırma altyapısı oluşturmaktadır. Cocomo81 [2], Albrecht [7], Kemerer [8], Maxwell [9], Desharnais [10], Kitchenham [11], Tukutuku [12] ve ISBSG [13] veri setleri sıklıkla karşımıza çıkmaktadır. Kullanılan veri setleri farklı yazılım projelerine ait programlama dili, donanım bilgileri, yazılan kaynak kodu değeri, uygulama tipi gibi farklı özelliklerden oluşmaktadır.

2.2 Proje Yönetim Araçları

Günümüzde yazılım projelerinin büyüklüğü göz alınırsa projelerin yönetimi esnasında karşılaşılan zorluklar gün geçtikçe artmaktadır. Projelerin yönetiminin düzenlenmesi ve kolaylaştırılması amacıyla çok farklı proje yönetim araçları bulunmaktadır. Atlassian tarafından geliştirilen JIRA, bir hata, sorun izleme ve proje yönetim anlamında kullanılan en popüler araçtır [14]. 2002 yılında ilk sürümü çıkarılmış ve Java programlama dili ile yazılmıştır. JIRA ile proje geliştirme esnasındaki karşılaşılan birbirinden farklı durumların yönetimini (farklı iş paketlerinin organizasyonu, hata veya sorun izleme, çalışanların çalışma durumlarını gözetleme gibi) kolaylıkla yapılabilmektedir. Primevera, sorun yönetim araçları arasında JIRA'dan sonra en çok kullanılan yönetim aracı konumundadır [15]. Projelerin planlanması, kaynakların ayrılması ve yönetilmesi, proje performansları hakkında sunduğu görselleştirme avantajları sayesinde oldukça yaygın olarak kullanılmaktadır [16]. 2008 yılında Oracle tarafından satın alınmıştır. MS-Project ise Microsoft tarafından geliştirilen bir proje ve kaynakların planlanması, izlenmesi amacıyla kullanılan bir yönetim aracıdır. Sunduğu hazır şablonlar ile projeye dair kolayca grafikler oluşturarak görselleştirme imkânları sunmaktadır [17]. Projelerin takvim programlarının gerçekleştirilmesi, sunduğu gerçek zamanlı raporlama sistemi gibi özellikleri ile günümüzdeki en popüler proje planlama aracıdır [15]. CA Servis Masası Yöneticisi (CA_Service Desk Manager) ise firmaların müşterileri ile olan ilişkilerinin yönetildiği bir talep yönetim platformudur[18]. Müşterilerden gelen talep veya olayların tutulduğu ve yönetildiği bir sistemdir. Müşterilerin istekleri doğrultusunda gerçekleştirilecek olan geliştirmelerin kayıt altına alınması ve yönetilmesi CA platformu üzerinden gerçekleştirilmektedir.

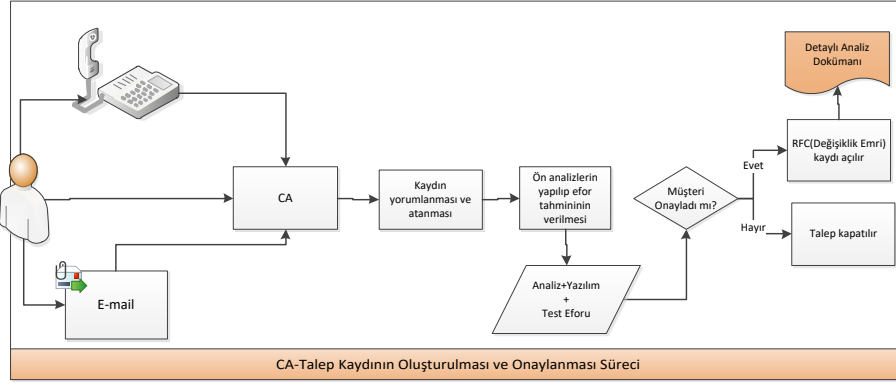
3 Yazılım Talep ve Geliştirme Süreci Yönetimi

Yazılım hizmeti veren firmalar müşterileri ile olan ilişkilerini ve yazılım süreçlerini yönettikleri platformlar mevcuttur. Farklı amaçlar için kullanılan platformlar proje

süreçlerinin farklı aşamaları için kullanılmaktadır. İlgili platformlar üzerinden yapılan işlemler aşağıda açıklanmış; müşteri talep yönetimi Şekil 1 ve yazılım süreçleri yönetimi Şekil 2 üzerinde modellenmiştir. Her iki proje yönetim aracı farklı kullanıcı grupları tarafından kullanılmakta ve aralarındaki bilgi akışı herhangi bir senkronizasyon olmadan kişi bağımlı olarak ilerlemektedir.

3.1 Müşteri Talep ve Onay Süreci

Projenin ortaya çıkmasının en büyük tetikleyicisi müşteri ihtiyacıdır. Müşterilerle iletişimin sağlıklı şekilde yürütülmesi, taleplerin düzgün şekilde alınıp değerlendirilerek projelendirilmesi için firmaların faydalandıkları talep yönetim araçları mevcuttur. Çalışmanın gerçekleştirildiği firmada müşteri talep yönetimi için CA [18] aracı kullanılmaktadır. Müşteri talep ve onay süreci Şekil 1 üzerinde modellenmiştir.



Şekil 1. Müşteri Talep ve Kayıt Süreci

Bu platform üzerinden;

- Müşteri talep kayıtları açabilir
- Açılan kayıtların firma tarafından değerlendirilmesi yapılır
- Analiz, yazılım ve test tarafındaki toplam eforlar hesaplanır
- Efor tahmini yapıp efor değeri müşteri onayına sunulur
- Müşteri efor değerini onaylamazsa talep kapatılır
- Müşteri efor değerini onayladığında ise değişiklik emri (RFC: Request for change) açılarak yazılım sürecinin başlatılabilmesi için detaylı bir analiz dokümanı hazırlanır.

İlgili doküman müşteri isteklerinin detaylı şekilde analiz edilerek yazılım tarafındaki gereksinimleri ortaya koyabilecek nitelikte hazırlanmaktadır. Bu doküman aynı zamanda yazılım ve test süreçlerine kaynaklık edecek dokümandır.

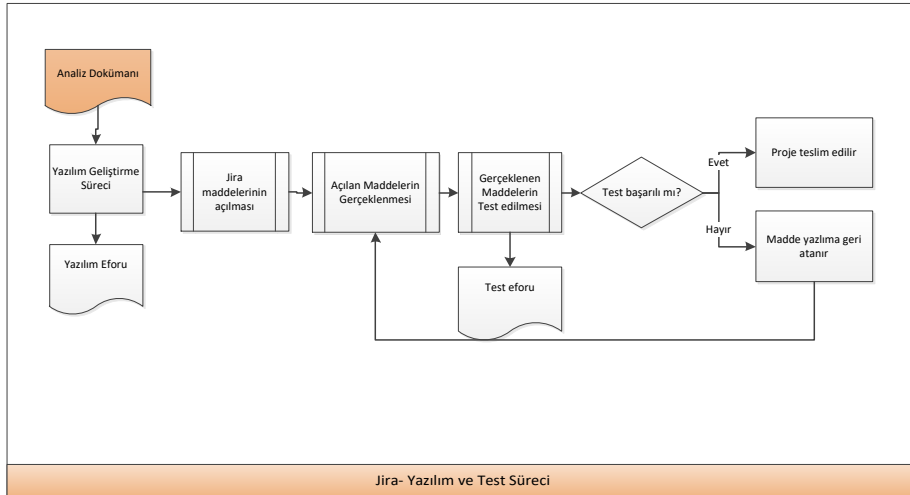
3.2 Yazılım Geliştirme ve Test Süreci

Gereksinim ve kayıt analizi süreci tamamlandıktan sonra yazılım ve test süreçleri ayrı bir aşama olarak değerlendirilip farklı metrikler üzerinden sürecin takip edildiği yazılım proje yönetimi için farklı araçlar kullanılmaktadır.

Çalışmanın yapıldığı firmada yazılım ve test süreçleri takibi için JIRA [14] platformu kullanılmaktadır. Bu platform üzerinden takip edilen adımlar aşağıdadır;

- Müşteri gereksinim analizinden sonra hazırlanan detaylı analiz dokümanına göre yazılım süreçleri başlatılır. Yazılım sürecini gereksinim süreci ile ilişkisi hazırlanan analiz dokümanı üzerinden yani kişi bağımlı olarak iletilemektedir.
- Gereksinimlere maddeler halinde JIRA üzerinde açılır
- Planlanan tarih ve gerçekleştirmelere göre eforlar bu platform üzerinden takip edilir
- Yazılım süreci tamamlanan JIRA maddesi yazılım test süreçlerine tabi tutulur.

JIRA platformu üzerinde takip edilen yazılım ve test süreci Şekil 2 'de modellenmiştir.



Şekil 2. Yazılım ve Test Süreci

4 Veri Tabanlarının Senkronizasyonu Çalışması

İlgili firmada müşteri talep yönetimi ile yazılım ve test süreçleri bağımsız iki platform üzerinden yönetilmektedir. Her iki platform da birbirinden bağımsız işletiliyor olsa da bir projenin farklı aşamalarına ait metrikleri içermektedir. Bu sebeple bir projeye ait genel bir çıkarım ve bütünlük bir efor hesabı yapılabilmesi için iki platformun senkronize edilmesi gerekliliği doğmuştur.

CA ve JIRA uygulamaları birbirinden bağımsız veri tabanları kullanmaktadır. Bu veri tabanlarındaki bilgiler dinamik olarak güncellenmektedir. Bu iki veri tabanını tek

bir veri tabanı üzerinde birleştirmek ve periyodik olarak güncellemek amacıyla bir Windows servis uygulaması geliştirilmiştir. Bu servis CA ve JIRA veri tabanlarındaki ilgili tablolardaki yeni verileri çekip NOTICE veri tabanındaki kopyası tabloları güncellemektedir. Bu servis 2 saat periyodunda çalışmaktadır. Servisin senkronize ettiği tablolar ve içerik bilgileri Tablo 1’de gösterilmektedir.

Tablo 1. Senkronize Edilen Tablolar

Tablo Adı	Veri Tabanı	İçerik
BI_D_REQS	CA	Talep, olay ve problem kayıtları
BI_D_CHGS	CA	Değişiklik emri kayıtları
JIRA_F_REQUEST	JIRA	JIRA madde kayıtları

Senkronize edilmiş tablolardaki ilgili verileri ID bilgisini kullanarak birleştiren ikinci bir Windows servis çalışmaktadır. Bu servis, üç tablodan elde edilen verileri JIRA_CA isimli bir tablo üzerinde birleştirmektedir. Servis, 5 saat periyodunda çalışmaktadır.

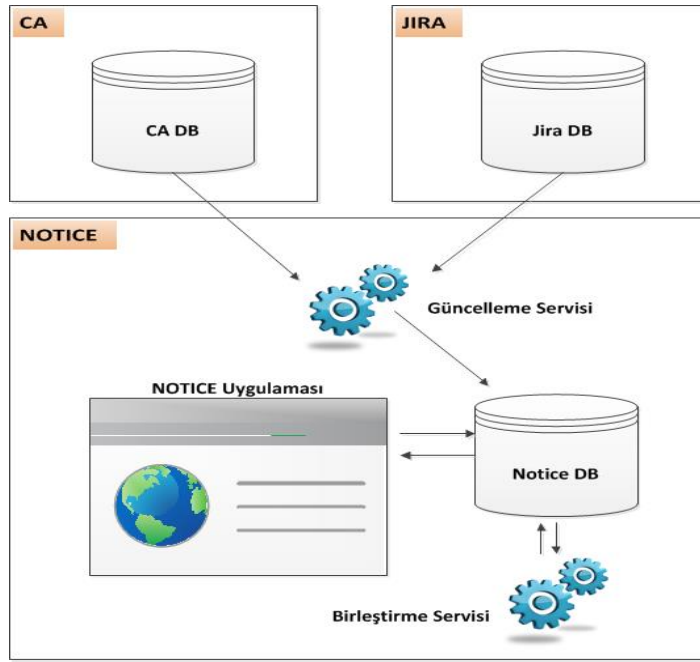
Platformlar birleştirilirken efor tahmininde kullanılacak olan özellik seçimi gerekmektedir. Belirlenen özellikler seçilirken 2.1. literatür taramasında bahsi geçen veri setlerinden faydalanılmıştır. Bu veri setleri üzerinde efor tahmininde işe yarayabilecek ve aynı zamanda firma bünyesindeki JIRA ve CA kayıtlarından doğrudan veya dolaylı olarak elde edilebilecek özelliklerin seçimine özen gösterilmiştir. Tablo 2’de farklı veri setlerinde efor tahmini için kullanılan, firma bünyesindeki JIRA ve CA platformlarından elde edilebilecek özellikler listelenmiştir.

Tablo 2 Efor Tahmininde Kullanılacak Öz nitelikler ile Veri Tabanındaki Karşılıkları

Yazılım Tahmin Modeli	Öznitelik Açıklaması	Referans Veri Setleri	Karşılık Gelen Veri Tabanı Alanı	Veri Tabanı ile İlgili Açıklama
Müşteri	Hangi müşteri şirket	ISBSG	CHG_ORGANIZATIONNAME	İsteğin ilgili olduğu konfigürasyonun organizasyon bilgisini göstermektedir.
Uygulama Türü	Uygulama çeşidi	Tukutuku, Maxwell, ISBSG		
İstek Tipi	Müşteriden gelen istek tipi	Kitchenham, ISBSG	CHG_TYPE	Her İstek için otomatik olarak 'Değişiklik Emri' stringi atılmaktadır.
Talebi Açan Kişinin Niteliği	Talebi açan kişi şirket bünyesinde bir çalışan mı değil mi?	-	CHG_REPORTER	İsteği raporlayan kişiyi göstermektedir.
Proje Yılı	Proje yılı	ISBSG	CHG_OLUSTURULMA_AY	İstek kaydının açıldığı tarihin ayını göstermektedir.
Proje Ayı	Proje ayı	-	CHG_OLUSTURULMA_YILI	İstek kaydının açıldığı tarihin yılını göstermektedir.
Description	Müşteriden gelen mesaj	-	CHG_DESCRIPTION	CA'deki talep kaydının açıklaması
İstek Grubu	Efor tahmini yapılacak isteği gerçekleştiren takımın bulunduğu grup	-	CHG_GRUP	İsteğin atandığı grubu raporlama gibi göstermektedir.
Departman	İsteği gerçekleştirecek grubun bağlı olduğu departman	-	CHG_GRUP_JOBTITLE	İsteğin atandığı analiz grubunun bağlı olduğu departmanı (Proje Uyarılama vs. gibi) göstermektedir.
Talebin Açılması için Geçen Süre	Destek ekiplerinin isteği alması ve kaydını oluşturma için geçen ilk süre	-	CHG_TIMESPENT_DK	İsteğin harcama sürenin dakika cinsinden değerini göstermektedir. CA'de loglarda bulunan time spent değerlerinin toplamını almaktadır.

Analiz Eforu	Analiz sürecinde harcanan gerçek efor	Kitchenham, ISBSG	TICKET_TIMESPENT_DK	İsteğin analizi için geçen süre
Yazılımcı Eforu	Yazılım sürecinde harcanan gerçek efor		JIRA_TIMESPENT_DK	Jira kayıtdaki Time Spent dakika cinsinden gösterir.
Toplam Efor	İsteğin tamamlanması için geçen süre	Tüm veri setleri	TOPLAM_SURE_SAAAT	$CHG_TIMESPENT_DK + JIRA_TIMESPENT_DK + TICKET_TIMESPENT_DK / 60$ formülü ile hesaplanan değeri göstermektedir.

Senkronize edilen sisteme ait mimari Şekil 3'te gösterilmektedir.



Şekil 3. Veri Senkronizasyonu Sistem Mimarisi

NOTICE veri tabanı bir projenin baştan sona tüm süreçlerinin takip edilebileceği şekilde tasarlanmıştır. CA ve JIRA kayıtları projeye ait referans değeri üzerinden birbiri ile ilişkilendirilmiştir. Böylelikle bir projeye tüm bileşenler ilgili platformlar üzerinden merkezi bir yapıda elde edilebilmektedir. Oluşturulan veri tabanı ve içerdiği veriler proje yönetim süreçleri için karar destek sistemine katkı sağlayacak araçların geliştirilmesine olanak sağlayacaktır.

5 Sonuç

Yazılım projelerindeki en büyük sorun efor tahminindeki sapmalar ve süreçlerin sağlıklı şekilde izlenememesi yönündedir. Proje müşteri ihtiyacı ile başlatılır, firma içinde değerlendirilir, müşteriden teyitler alındıktan sonra geliştirme süreci başlatılır. Firmamızda müşteri ilişkileri ve yazılım süreçleri farklı araçlar üzerinden

yürütülmektedir. Bu araçların bütünleşik değerlendirilmemesi ve aralarında bir senkronizasyon olmamasından dolayı bir projeye ait uçtan uca sağlıklı veri elde edilememektedir. Aradaki aktarım kişiler aracılığıyla yapılması verinin kişi bağımlı ve hataya açık olması anlamına gelmektedir.

Veritabanı senkronizasyonu yönünde çalışmanın yapılması ile müşteri talebi, analizi ve yazılım geliştirme süreçleri referans numarası ile ilişkilendirilmiş, tek platform üzerinden bir proje ait bütünleşik verinin temin edilmesi sağlanmıştır. Belli periyotlarla ilgili uygulamalardan, platforma veri aktarımı sağlanmakta böylelikle süreç dinamik bir şekilde işletilerek güncel verilerle beslenmektedir. Firmaya özgü tahmin modellerinin oluşturulabilmesi için firmanın gerçekleşen süreçlerinden verilerin toplanmış olması büyük önem taşımaktadır. Efor tahmini talebin niteliğine, firma iç dinamiklerine, talebin açılma zamanı, personel niteliği gibi birçok niteliğe bağlı olarak değişkenlik göstermektedir. Mevcut efor tahmin modelleri ise sınırlı veri setleri ile eğitilmiş ve sabit modeller olarak oluşturulmuştur. Ancak sağlıklı bir efor tahminin yapılabilmesi firma iç dinamikleri ve gerçekleşen süreçler üzerinden toplanan veriler üzerinden oluşturulan modellerle mümkün olacaktır. Firmaya özgü efor tahmini yapılması amacı ile başlatılan TEYDEB 1501 destekli NOTICE projesi kapsamında bütünleşik efor tahmini yapabilmek adına bir projenin farklı aşamalarının yönetildiği platformların entegrasyonu çalışması yapılmıştır.

Gelecekte yapılması planlanan çalışmalar ise senkronize edilmiş sistem üzerinden toplanan veriler üzerinden makine öğrenmesi tabanlı firmaya özgü efor tahmin modellerinin geliştirilmesi yönünde olacaktır.

Referanslar

1. Kumari, K., Comparison and Analysis of Different Software Cost Estimation Methods. *International Journal of Advanced Computer Science and Applications (IJACSA)* 4(1), 153-157 (2013).
2. B.W.Boehm, *Software Engineering Economics*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1981.
3. Shekhar, S., Review of Various Software Cost Estimation Techniques. *International Journal of Computer Applications* 141(11), 31-34 (2016).
4. Jørgensen, M., A review of studies on expert estimation of software development effort. *The Journal of Systems and Software* 70(1-2), 37-60 (2004).
5. Attarzadeh, I., Ow, S.H.: Proposing a New Software Cost Estimation Model Based on Artificial Neural Networks. *International Conference on Computer Engineering and Technology 2010, ICCEMS*, vol. 3, pp. 487–491, Chengdu, China (2010).
6. Attarzadeh, I., Ow, S.H.: Improving the Accuracy of Software Cost Estimation Model Based on a New Fuzzy Logic Model. *World Applied Sciences Journal* 8(2), 117-184, 2010.
7. Albrecht A., *Software Function, Source Lines of Code, and Development Effort Prediction: A Software Science Validation*. *IEEE Transactions on Software Engineering* 9(6), 639-648, 1985.

8. Kemerer, F.C., An empirical validation of software cost estimation models. *Communications of the ACM*, 30 (5), 416–429, 1987.
9. Sentas, P., Software productivity and effort prediction with ordinal regression. *Information and Software Technology*, 47(1), 17-29, 2005.
10. Desharnais, J.M., Analyse statistique de la productivité des projets informatiques à partir de la technique des points de fonction, University of Montreal, Master's Thesis, 1989.
11. Kitchenham, B., An empirical study of maintenance and development estimation accuracy. *Journal of Systems and Software* 64(1), 57-77, 2005.
12. Ferrucci, F., Web effort estimation: The value of cross-company data set compared to single-company data set. *Proceedings of the 8th International Conference on Predictive Models in Software Engineering*, pp. 29-38, Lund, Sweden, 2012.
13. Guevara F., The usage of ISBSG data fields in software effort estimation: A systematic mapping study. *Journal of Systems and Software*, Vol. 113, 188-215, 2016.
14. Atlassian Homepage, <https://www.atlassian.com/software/jira>, Erişim tarihi 2018/06/12.
15. Project Management Zone Homepage, <https://project-management.zone/>, Erişim tarihi 2018/06/12.
16. Primavera Homepage, <https://www.oracle.com/tr/applications/primavera/products/project-management.html>, Erişim tarihi 2018/06/15.
17. MS Project Homepage, <https://products.office.com>, Erişim tarihi 2018/06/12.
18. CA Technologies Homepage, <https://www.ca.com>, Erişim tarihi 2018/06/12