

TADES CAR: Yazılım Kalite Maliyetlerini Düşürmeye Yönelik Kök-Neden Analizi Durum Çalışması

Sezen Erdem¹

¹ASELSAN A.Ş. SST-KKYTM P.K.1 06172, Yenimahalle/Ankara, Türkiye

¹erdem@aselsan.com.tr

Özetçe. Üretim alanında yaygın kullanımı bulunan Kalite Maliyetleri yazılım geliştirme alanında da projelerde yazılım kalitesine yönelik yapılan faaliyetler hakkında yol gösterici bir ölçüm olarak değerlendirilebilir. Yazılım kalite maliyetlerinin toplam proje maliyetine oranı ve yazılım kalite maliyetlerinin kendi içerisindeki dağılımı yazılım geliştirme ekiplerine izledikleri süreçler ile ilgili analizler yapabilmek adına değerli geri beslemeler sağlamaktadır. Bu geri beslemeler sonucunda izlenen yazılım geliştirme süreçleri ile ilgili iyileştirme faaliyetleri başlatılarak kaynakların daha verimli kullanılması ve daha kaliteli yazılımlar geliştirilmesi yönünde adımlar atılmaktadır. Yazılım kalite maliyetleri ölçümleri hata yoğunluğu ölçümleri ile birlikte değerlendirilerek kalite maliyetleri en uygun seviyelere çekilirken hata yoğunluğunun da en az seviyeye indirilmesi hedeflenmelidir. Bu çalışmada ASELSAN Savunma Sistem Teknolojileri Sektör Başkanlığı bünyesindeki Komuta Kontrol Yazılım Tasarım Bölümü'nde yürütülen TADES Yazılım Ürün Hattı geliştirme projesinde kalite maliyetlerinin düşürülmesine yönelik başlatılan kök-neden analizi çalışması (TADES CAR (Causal Analysis and Resolution)), izlenen yöntem, alınan kararlar, uygulama ve sonuç değerlendirmeleri anlatılmaktadır.

Anahtar Kelimeler: Yazılım Kalite Maliyeti, Hata Yoğunluğu, Yazılım Ürün Hattı, Kök-Neden Analizi

1 Giriş

Yazılım Kalite Maliyeti ve Hata Yoğunluğu yazılım geliştirme projelerinde etkinlik ve verimlilik ölçümleri için sıklıkla kullanılan yöntemlerdir. Kalite maliyetleri en uygun seviyeye çekilirken hata yoğunluğunun da en az seviyeye indirilmesi yazılım projelerinin başarısı için kritik olan zaman, maliyet, kalite dengesinin kurulması açısından önem taşımaktadır. Yazılım kalite maliyetlerinin analizi izlenen yazılım geliştirme sürecinde iyileştirmeler yapmak için çok faydalı geri bildirimler sağlamaktadır. Hem yazılım kalite maliyetinin toplam geliştirme maliyeti içerisindeki oranı hem de kendi içerisindeki alt başlıklara dağılımının analiz edilmesi ile yazılım kalitesinin artırılması, kaynakların daha verimli ve etkin kullanılması sağlanabilir. Bu çalışmada ASELSAN Savunma Sistem Teknolojileri (SST) Sektör Başkanlığı bünyesindeki Komuta Kontrol Yazılım Tasarım Müdürlüğü (KKYTM) Bölümü'nde

yürütülen TADES Yazılım Ürün Hattı geliştirme projesinde kalite maliyetlerinin düşürülmesine yönelik başlatılan kök-neden analizi çalışması, izlenen yöntem, alınan kararlar, uygulama ve sonuç değerlendirmeleri anlatılmaktadır.

Makalenin 2. bölümünde Yazılım Kalite Maliyeti ve Hata Yoğunluğu ölçüm yöntemlerinden, 3. bölümünde TADES Yazılım Ürün Hattı yaklaşımında Yazılım Kalite Maliyeti ölçümleri ve sonuçlarının değerlendirilmesinden ve 4. bölümde de uygulanan kök-neden analizi çalışması ve bu çalışmanın sonuçlarından bahsedilecektir.

2 Yazılım Kalite Maliyeti ve Hata Yoğunluğu

Üretim alanında yaygın olarak kullanımı olan Kalite Maliyeti kavramı 1950’li yıllarda ortaya atılmıştır. Kalite maliyeti Juran tarafından “kaçınılabilecek maliyetler ya da madendeki altın” olarak tanımlanmaktadır [1]. Başka bir deyişle kalite maliyeti bir işi ilk seferde tam ve hatasız yapamamanın maliyeti olarak tanımlanabilir. Crosby, kalite maliyetlerini Uyumluluk Maliyetleri ve Uyumsuzluk Maliyetleri olmak üzere iki kategoriye ayırmıştır [2]. Uyumluluk Maliyetleri, gereksinimlere uygun ürün oluşturabilmek için yapılan önleyici ve değerlendirme maliyetlerini, Uyumsuzluk Maliyetleri ise oluşan iç ve dış hataların giderilmesine ait maliyetleri içermektedir. Toplam kalite maliyeti ise uyumluluk ve uyumsuzluk maliyetlerinin toplanması ile elde edilmektedir. Kalite maliyetleri süreç izleme, yönetim ve iyileştirme faaliyetlerinde sıklıkla kullanılan bir metriktir [1][2].

Üretim alanında yaygın ve etkin kullanımı olan kalite maliyetlerinin yazılım geliştirme alanında da kullanımına yönelik çalışmalar yapılmıştır [3][4]. Üretim ile kıyaslandığında daha dinamik bir yapısı olan yazılım geliştirme alanında kalite maliyeti alt başlıkları yazılım geliştirme sürecinin doğasına uygun olarak tekrar tanımlanmıştır. Yazılım kalite maliyetlerini oluşturan temel kalemler kısaca şu şekilde tanımlanabilir [5].

- Önleyici Faaliyet Maliyetleri: Yazılım geliştirme sürecinin tüm aşamalarında hataların oluşmasını önlemeye yönelik olarak geliştirme aşamasında yapılan faaliyetleri içerir. Kalite planlarının hazırlanması, yazılım test planlarının hazırlanması, eğitimlerin alınması, prototip hazırlama faaliyetleri önleyici faaliyetler kapsamında değerlendirilebilir.
- Değerlendirme Maliyetleri: Yazılım kalitesinin değerlendirilmesine yönelik faaliyetleri içerir. Gözden geçirme, birim test, doğrulama ve geçerli kılma faaliyetleri değerlendirme maliyeti olarak sayılmaktadır.
- İç Hata Maliyetleri: Yazılım geliştirme ve doğrulama aşamalarında tespit edilen hataların giderilmesine yönelik harcanan efor maliyetidir.
- Dış Hata Maliyetleri: Kullanıcı tarafından tespit edilen hataların giderilmesine yönelik olarak harcanan efor maliyetidir.

Yazılım kalite maliyetleri yazılım projelerinin performans ölçümlerinde ve süreç iyileştirme çalışmalarında yaygın olarak kullanılmaktadır [6][7]. Yazılım kalite maliyetlerinin yazılım kalitesinin değerlendirilmesi için tek başına kullanımı çok

anlamalı bir değerlendirme olmayacaktır. Yazılım kalitesine yönelik olarak kalite maliyetleri ölçümlerini tamamlayıcı bir başka ölçüm de hata yoğunluğudur. Hata yoğunluğu yazılımda bulunan hata sayılarının yazılım büyüklüğüne oranı olarak tanımlanmaktadır. Hata yoğunluğu tespit edilen hataların önem derecelerine, gerçekleşme ve tespit dönemlerine göre sınıflandırılabilir. Yazılım kalitesinin artırılması ve müşteri memnuniyetinin yükseltilmesi için hata yoğunluğu ölçümü önemli bir metriktir.

Organizasyonların Yazılım Kalite Maliyeti ve Hata Yoğunluğu ölçümlerini birlikte almaları ve analiz etmeleri yazılım kalitesinden taviz vermeden kalite maliyetlerini düşürmelerini sağlamaktadır. Bu açıdan bu iki ölçümün birlikte yapılması organizasyonların kaynaklarını hem daha verimli kullanmaları hem de daha kaliteli ürünler çıkarmaları konusunda yol gösterici olacaktır.

3 TADES Yazılım Ürün Hattı Yazılım Kalite Maliyeti ve Hata Yoğunluğu Ölçümleri

Bir önceki bölümde anlatıldığı gibi yazılım kalite maliyetleri ölçümlerinin analizi izlenen yazılım geliştirme süreci ile ilgili olarak geliştirme yapılabilecek alanlar için ipuçları vermektedir. TADES, ASELSAN SST Sektör Başkanlığı bünyesindeki KKYTM Bölümü'nde yürütülen Teknik Ateş Destek Sistemleri yazılımlarının yeniden kullanım temelli üretilmesi çalışmasıdır [8]. TADES kapsamında 30 adet ortak kullanım varlığı geliştirilmiş ve geliştirilen ortak varlıklar kullanılarak TADES Yazılım Ürün Hattı'ndan bugüne kadar 9 farklı proje kapsamında 19 yazılım ürünü geliştirilmiştir. TADES Yazılım Ürün Hattı'nın performansının ölçülmesine yönelik bir takım metrikler belirlenmiş olup, düzenli ölçümlerle performans analizleri yapılmaktadır. Yazılım Kalite Maliyeti ve Hata Yoğunluğu ölçümleri de düzenli olarak takip edilen ve istatistiksel kontrol altına alınmaya çalışılan ölçümler içerisindedir.

ASELSAN SST Yazılım Tasarım Bölümleri, yazılım projeleri işçilik verilerini toplamak için Microsoft Team Foundation Server (TFS) aracını kullanmaktadır. Projeler TFS aracında tanımlanmakta ve yazılım geliştirme ekipleri iş dağılım ağaçlarını TFS üzerinde tanımlayarak projelerinin takibini yapmaktadır. TFS'ten toplanan işçilik verileri kullanılarak Yazılım Kalite Maliyeti hesaplamaları ve analizleri yapılmaktadır. Hata yoğunluğu ölçümleri için IBM Rational ClearQuest aracı ve kaynak kod satır sayıları kullanılmaktadır.

TADES geliştirme projesi için alınan ölçümlerde yazılım kalite maliyetlerinin çok yüksek değerlere ulaştığı görülmüştür. Bu maliyetlerinin detaylarının incelenmesi ve hata yoğunluğu ile ilgili hedeflerden sapmadan maliyetlerin düşürülmesine yönelik olarak bir kök neden analizi çalışması başlatılmıştır. ASELSAN SST Yazılım Tasarım Bölümlerinde yazılım geliştirme alanında CMMI yüksek olgunluk seviyelerine yönelik olarak çalışmalar başlatılmıştır. TADES Yazılım Ürün Hattı projesi yazılım kalite maliyetlerinin düşürmeye yönelik olarak başlatılan kök-neden analizi çalışması CMMI Seviye 5 süreç alanlarından birisi olan "Casual Analysis Resolution (CAR)"

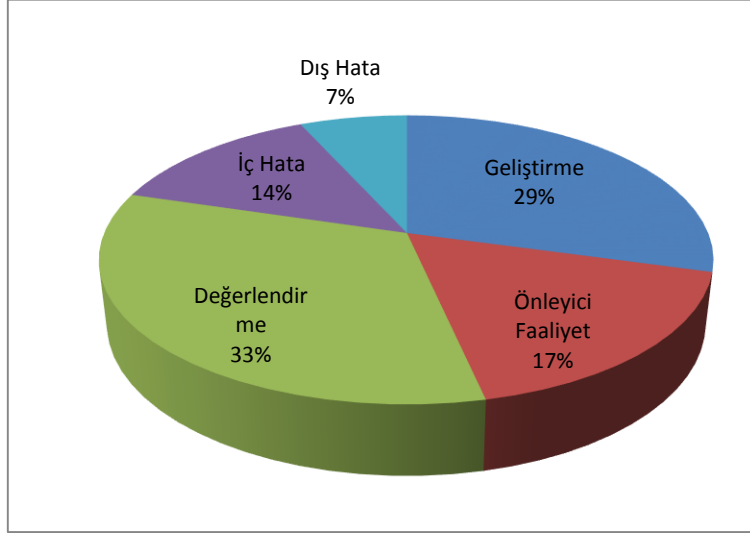
süreç alanı çalışmaları kapsamında yürütülmüştür. CMMI CAR süreç alanı kaliteyi ve üretkenliği artırmak amacı ile süreç kaynaklı problemlerin çözümüne yönelik sistematik bir yaklaşım tanımlamaktadır [9]. Makalenin 4. bölümünde yapılan çalışmanın detayları ve sonuçları anlatılmıştır.

4 TADES CAR Kök-Neden Analizi Durum Çalışması

ASELSAN SST Yazılım Tasarım Müdürlükleri'nde yazılım kalite maliyetlerinin ölçülmeye başlanması ile birlikte projelerdeki yazılım kalitesine yönelik harcanan efor bilgileri sayısal olarak değerlendirmeye alınmaya başlanmıştır. Alınan ölçümler sonucunda TADES Yazılım Ürün Hattı geliştirme projesine ait yazılım kalite maliyet değerleri 2014 yılı sonu itibari ile hedef değerlerin üzerinde ölçülmüştür. Elde edilen ölçüm bilgileri, geliştirme ve yazılım kalite maliyeti dağılım oranları Tablo-1 ve Şekil-1'de verilmiştir.

Faaliyet	Değer
Geliştirme	% 29
Değerlendirme	% 33
Önleyici Faaliyet	% 17
İç Hata	% 14
Dış Hata	% 7

Tablo-1. 2014 Yıl Sonu TADES Geliştirme ve Kalite Maliyeti Dağılımı

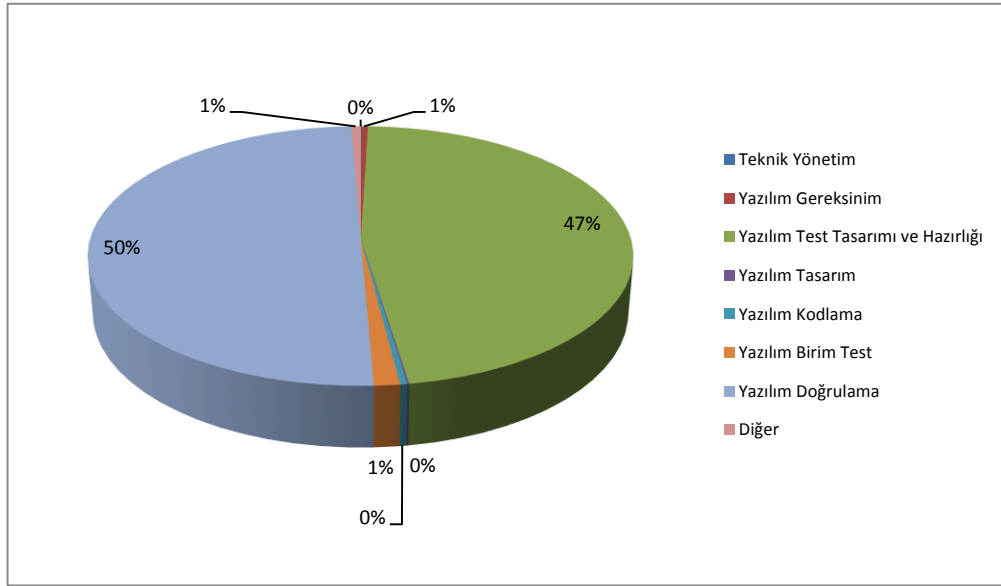


Şekil-1. 2014 Yıl Sonu TADES Geliştirme ve Kalite Maliyeti Dağılımı

Ölçüm sonuçlarından da görülebileceği üzere Değerlendirme Maliyetleri yazılım kalite maliyetleri içerisinde en büyük kalemi oluşturmaktadır. Değerlendirme Maliyetleri daha detaylı analiz edildiğinde Değerlendirme Maliyetlerinin kendi içerisindeki dağılımı Tablo-2 ve Şekil-2’de verilmiştir.

Faaliyet	Değer
Teknik Yönetim	% 0
Yazılım Gereksinim	% 1
Yazılım Tasarım	% 0
Yazılım Test Tasarımı ve Hazırlığı	% 47
Yazılım Kodlama	% 0
Yazılım Birim Test	% 1
Yazılım Doğrulama	% 50
Diğer	% 1

Tablo-2. 2014 Yılı Sonu TADES Kalite Maliyet Dağılımı



Şekil-2. 2014 Yılı Sonu TADES Kalite Maliyet Dağılımı

Bu sonuçlar analiz edildiğinde “Yazılım Test Tasarımı ve Hazırlığı” ve “Yazılım Doğrulama” adımlarının Değerlendirme Maliyetlerini artıran esas adımlar olduğu görülmektedir. Bu adımlara yönelik yapılacak iyileştirme çalışmalarının yazılım kalite maliyetlerini azaltmada etkili olacağı değerlendirilerek kök neden analizi çalışması başlatılmıştır. Başlatılan kök neden analizi çalışması adımları Şekil-3’de verilmiştir.



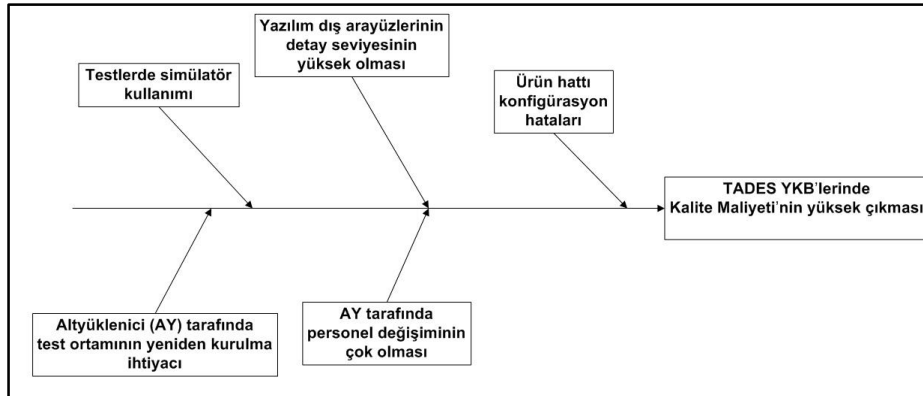
Şekil-3. Kök Neden Analizi Çalışması Adımları

İzlenecek adımlar Balık Kılıçığı yöntemi kullanılarak değerlendirme maliyetlerinin yüksek olmasına sebep olan kök nedenlerin araştırılması ile başlamaktadır. Belirlenen kök nedenlerin değerlendirilmesinin ardından çözüm önerileri ortaya atılmış ve çözüm önerilerinin uygulanması ile elde edilecek kazanımlar için maliyet-kazanım analizi yapılarak uygulanacak çözümlere karar verilmiştir. Alınan kararlara uygun olarak çözümler uygulamaya alınmış ve sonuçların değerlendirilmesi ile çalışma tamamlanmıştır.

4.1 Balık Kılıçığı Yöntemi Uygulaması

Kök Neden Analiz çalışmalarına Değerlendirme Maliyetlerinin yüksek olmasına sebep olan kök nedenleri araştırmak için çalışmalara Balık Kılıçığı yöntemi ile başlanmıştır. Bu çalışma sonucunda elde edilen kök nedenler şu şekilde belirlenmiştir (Şekil-4).

- Kök Neden 1 (KN1): Alt Yüklenici (AY) tarafında test ortamlarının kurulma ihtiyacı
- Kök Neden 2 (KN2): Testlerde simülator (mock) kullanımı
- Kök Neden 3 (KN3): Yazılım dış arayüzlerinin detay seviyesinin yüksek olması
- Kök Neden 4 (KN4): AY tarafında personel değişiminin çok olması
- Kök Neden 5 (KN5): Ürün hattı konfigürasyon hataları



Şekil 4. Kök Neden Analizi Balık Kılıçığı Diyagramı

4.2 Çözüm Önerilerinin Üretilmesi

Belirlenen kök nedenler için çözüm önerileri oluşturulmuş. Ortaya atılan çözümler ve ilişkili oldukları kök neden Tablo-3’de verilmiştir.

Öneri	İlişkili Kök Neden
Detaylı yazılım dış arayüzlerine ait testlerin Beyaz Kutu olarak geliştirici tarafından yapılması	KN1,KN3, KN4
Test Mühendisleri tarafından sadece Kara Kutu-Gösterim gereksinimlerinin doğrulaması	KN1,KN3, KN4
Simülatör (Mock) yerine gerçek Yazılım Konfigürasyon Birimi (YKB) kullanımı	KN2
Dış bağımlılıkları yüksek YKB’lerin testlerinin Takım Yazılımlar içerisinde yapılması	KN2
Grafiksel Kullanıcı Arayüzü (GKA) testlerinin otomatik olarak yapılması için alt yapının kurulması	KN1, KN4
Takım yazılım yapılandırma dosyalarının araç yardımı ile hazırlanması	KN5

Tablo 3. Çözüm Önerileri

Ortaya atılan çözüm önerilerinden “Takım yazılım yapılandırma dosyalarının araç yardımı ile hazırlanması” önerisinin kısa vadede yapılamayacağı değerlendirilmiş ve bu çözüm önerisi Kök Neden Analizi çalışması kapsamında çıkarılmıştır.

4.3 Maliyet-Kazanım Analizi

Çözüm önerilerinin uygulamaya alınması ile elde edilecek kazanım oranları hesaplanmıştır. Planlanan kazanım değerleri kullanılarak Yazılım Kalite Maliyeti hesaplamaları yapıldığında ilk aşamada Yazılım Kalite Maliyetlerinin % 8 oranında düşürülebileceği öngörülmüştür. Yazılım Test Tasarımı ve Hazırlığı adımı ile ilgili bazı alt yapı oluşturma faaliyetlerinin (test yazılımların hazırlanması vb.) sadece ilk uygulama sırasında yapılacağı ve bir sonraki doğrulama faaliyetlerinde bu adımlar için bu seviyede işçilik harcanmayacağı değerlendirildiğinde Yazılım Kalite Maliyetlerinin ilerleyen zamanlarda daha düşük seviyelere indirilebileceği değerlendirilmiştir.

Elde edilmesi planlanan kazanımlar Tablo-4’de verilmiştir.

Hedeflenen Kazanım Oranı
Yazılım Test Tasarımı ve Hazırlığı + % 50 Yazılım Doğrulama

Tablo 4. Kazanım Analizi

4.4 Karar

Çözüm önerilerinin pilot uygulama yapılarak değerlendirilmesine ve yapılan değerlendirme sonunda yaygınlaştırma konusunda adım atılmasına karar verilmiştir. Bu kapsamda TADES Yazılım Ürün Hattı içerisinde bulunan 5 adet yazılım konfigürasyon biriminin geliştirme faaliyetlerinde belirlenen çözüm önerileri uygulanarak sonuçları değerlendirmeye alınmıştır.

4.5 Uygulama

Çözüm önerilerinin pilot YKB'lere uygulaması sonuca "Yazılım Test Tasarımı ve Hazırlığı" ve "Yazılım Doğrulama" faaliyetleri maliyetlerinde elde edilen kazanım oranları Tablo-5'de verilmiştir.

YKB	Kazanım Oranı
Pilot YKB 1	% 46
Pilot YKB 2	%57
Pilot YKB 3	%6
Pilot YKB 4	%29
Pilot YKB 5	%42

Tablo-5. YKB Değerlendirme Maliyeti Efor Kazanım Oranı

Bu çalışma sonucunda YKB testlerinde tespit edilen hatalar, YKB testleri sonrası gerçekleştirilen doğrulama faaliyetlerinde tespit edilen hatalar da değerlendirmeye alınmış ve YKB hata yoğunluğu hedefleri ile ilgili bir kayma olmadığı görülmüştür. Tespit edilen hataların tespit dönemi içerisindeki dağılımları da göz önüne alınarak analizler yapıldığında uygulanan çözümün yazılım kalitesi ile ilgili olarak olumsuz bir durum oluşturmadığı gözlemlenmiştir.

5 Sonuç

Yazılım Kalite Maliyeti ölçümleri, yazılım kalitesi ile ilgili analizler yapmak ve ilerleyen dönemde süreçlerde iyileştirme yapılabilecek alanları tespit etmek için değerli bilgiler sağlamaktadır. Bu makalede ASELSAN SST bünyesindeki Komuta Kontrol Yazılım Tasarım Bölümü'nde yürütülen TADES Yazılım Ürün Hattı geliştirme projesinde Yazılım Kalite Maliyetlerinin düşürülmesine yönelik olarak uygulanan kök neden analizi çalışması anlatılmıştır. Bu çalışma ASELSAN STT Yazılım Tasarım Müdürlükleri tarafından konulan CMMI Yüksek Olgunluk seviyeleri hedefi doğrultusunda yürütülmüş bir iyileştirme faaliyetidir. Faaliyet, CMMI Seviye 5 süreç alanlarından Causal Analysis Resolution (CAR) süreç alanı kapsamında, kalite maliyetlerinin yazılım kalitesinden ödün vermeden düşürülmesi için gerçekleştirilmiştir. Pilot YKB'ler üzerinde yapılan çalışma sonucunda uygulanan

özüm önerilerinin genel olarak başarılı olduđu gözlemlenmiř ve önümüzdeki dönemde tüm YKB’lerde çözümün uygulanmasına karar verilmiřtir.

Kaynaka

1. Juran, J.M., *Quality Control Handbook*. 1st ed. 1951, New York, NY: McGraw-Hill
2. Crosby, P.B., *Quality is Free*. 1979, New York, NY: McGraw-Hill
3. Houston D. "Cost of Software Quality: Selling Software Process Improvement to Managers". *Software Quality Journal*,
4. Mandewille W.A., "Software Cost of Quality",*IEEE Journal on Selected Areas in Communications*, Vol.8, No:2, 1990, February.
5. Demirors, O.; Yildiz, O.; Selcuk Guceglioglu, A., "Using cost of software quality for a process improvement initiative," *Euromicro Conference 2000*. Proceedings of the 26th , vol.2, no., pp.286,291 vol.2, 2000
6. Dion, Raymond. "Process Improvement and the Corporate Balance Sheet", *IEEE Software* (July 1993) 28-35.
7. Price Waterhouse.. "Software Quality Standards: The Costs and Benefits. A review for the Department of Trade and Industry" London: Price Waterhouse Management Consultancy, 1988
8. Barak E., Erdem S., Yılmaz H., "TADES: Komuta Kontrol Alanında bir Yazılım Ürün Hattı alışması", *Ulusal Yazılım Mühendisliđi Konferansı (UYMK)*, 2010
9. Chrissis M.B., Konrad M., Shrum S., *CMMI for Development*, Addison Wesley, 2011 ISBN:0321711505