

Yazılım Eş Düzey Gözden Geçirmelerde Araç Kullanımı

Tool Deployment in Software Peer Reviews

Celal Ademoğlu
Yazılım Test ve Yönetim Birimi
ASELSAN MGEO, Ankara
ademoglu@mgeo.asehsan.com.tr

Ömer Karaduman
Yazılım Test ve Yönetim Birimi
ASELSAN MGEO, Ankara
okduman@mgeo.asehsan.com.tr

Özet

Bu bildirinin amacı yazılım eş düzey gözden geçirmelerinin yazılım kalitesi ve yazılım geliştirme süreçlerindeki önemini vurgulamak, eş düzey gözden geçirmelerin geliştirilen bir araç ile sistematik ve etkin bir şekilde nasıl gerçekleştirildiğini paylaşmaktır.

ASELSAN MGEO grubu bünyesinde yazılım süreçleri kapsamında ve ilgili proje gereksinimleri uyarınca bir takım gözden geçirmeler hali hazırda yürütülmektedir. Fakat pratikte bu gözden geçirmelerin yürütülmesi ve kayıtlarının tutulması ile ilgili bir takım güçlükler yaşanmakta olduğu gözlemlenmiştir. Gözden Geçirme Yönetim aracı, pratikte karşılaşılan bu sorunları çözmek, kaliteli ve güvenilir yazılım geliştirme yolunda gözden geçirmeleri etkin bir şekilde gerçekleştirmek için geliştirilmiştir.

Abstract

The aim of this paper is to emphasize the importance of software peer reviews in the software quality and the development processes and to share how the peer reviews are performed systematically and effectively through a dedicated tool.

Software peer reviews are performed in conjunction with software processes and the requirements of the related projects in ASELSAN MGEO division. But it has been observed that some difficulties had been experienced in the execution of peer reviews and maintaining the records of them. Review Manager tool has been developed to resolve these issues and perform the software peer reviews effectively in order to improve the software quality and reliability.

1. Giriş

Önceden elle yapılan birçok görevin hızı, doğruluğu, güvenilirliği, kararlılığı ve üretkenliği arttırabileceği düşüncesiyle yazılımla otomatize edildiği günümüzde kaliteli yazılım, şirketlerin ayakta kalmasında ve başarısında önemli rol oynamaktadır [1]. Her ne kadar bilgi teknolojilerinin oturmuş bir disiplin olduğu

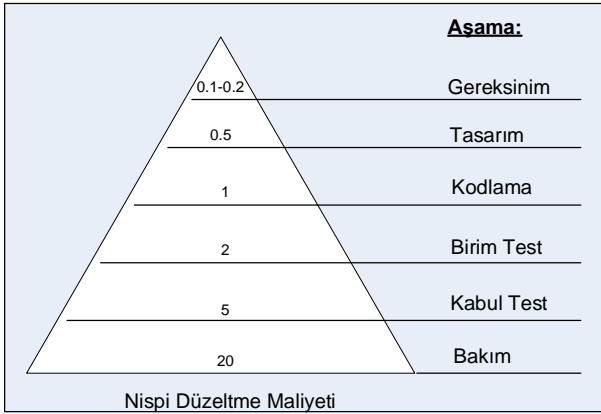
kanaati oluşmuş olsa da yazılım projeleri hala başarısızlığa açıktır [2]. Hatta bir yazılım projesi başarısız olarak sınıflandırılmamış olsa bile ortaya çıkan genel yazılım kalitesi birçok iyileştirmeyi gerekli kılabilir [3]. Yazılım gözden geçirme veya inceleme yazılım kalitesini iyileştirmek için kullanılan önemli tekniklerden birisidir [4].

Yazılım gözden geçirmenin amacı tasarım ve geliştirme aşamasında oluşturulan ara çıktıların incelenerek ürünün kalitesinin iyileştirilmesidir [5]. Gözden geçirme sayesinde hatalar erken tespit edilmiş ve ürün çıkmadan önce giderilmiş olacaktır [6]. Birçok araştırmacı gözden geçirmenin maliyet azaltma, üretkenlik ve kalite iyileştirme yönleriyle en etkin yazılım mühendisliği tekniklerinden biri olduğu konusunda hemfikirdir Gözden geçirme sayesinde:

- Yazılım geliştirme yaşam döngüsü içerisinde kavramsal tasarım, gerçekleştirim ve test evreleri boyunca hatalar tespit edilebilir. Hatalar geliştirme sürecinde ne kadar erken tespit edilirse giderilmeleri ve düzeltilmeleri o kadar kolay ve az maliyetli olacaktır
- Yazılım geliştirme yaşam döngüsünün daha sonraki aşamalarında bulunması güç veya imkansız olan hatalar erken safhalarda belirlenebilir.
- Yazılım geliştirme esasında bir insan aktivitesi olduğundan, yazılım ekibi içerisinde öğrenmeyi ve iletişim geliştirir [7]

Yazılım gözden geçirmenin diğer bir getirisi de geliştirme maliyetlerini azaltması ve üretkenliği arttırmasıdır. Çünkü yazılım projelerinde en yaygın yapılan ve maliyeti arttıran hatalardan biri de yazılım problemlerini tespit etme ve düzeltme aktivitelerinin projenin sonuna ertelenmesidir [3]. Yazılım geliştirme yaşam döngüsünün değişik aşamalarından kaynaklanan hataların düzeltme maliyetleri ile ilgili yapılan araştırmalar göstermiştir ki hataların daha erken safhalarda bulunması maliyetleri büyük oranda azaltmaktadır. Aşağıda Şekil 1'de de görülebileceği gibi gereksinim aşamasında hataları bulup düzeltmekle

bakım aşamasında bulup düzeltmek arasında maliyet açısından 200:1 gibi bir oran söz konusudur [8].



Şekil 1: Yazılım Geliştirme Sürecinde Aşamalara Göre Hata Düzeltme Maliyetleri [9]

Eş düzey gözden geçirme, gözden geçirilecek iş ürünün yazarı dışında bir veya daha fazla kişi tarafından muhtemel hataları ve iyileştirme noktalarını bulmak amacıyla incelenmesi aktivitesidir [10]. Yazılım projesinde gözden geçirme, yazılım yaşam döngüsünün ilgili safhalarında, belirli bir ürünün, tanımlı bir gözden geçirme ekibi tarafından incelenerek ürünün olgunluğunun ve bir sonraki faza geçiş için yeterliliğinin tanımlanmış ölçütlere göre değerlendirilmesi, varsa hatalarının tespit edilmesi ve gereken önlemlerin alınması aktivitelerini içerir [11]. Gözden geçirme, yazılım iş ürününün hem teknik yeterliliğinin hem de ilgili gereksinim ve standartlara uyumluluğunun değerlendirilmesine olanak sağlar. Gözden geçirme kayıtları, ilgili gözden geçirme aktivitesi için doğrulama sonuçlarını oluşturur.

ASELSAN MGEO grubunda uygulanan yazılım geliştirme süreçlerinde aşağıdaki gözden geçirmelerin uygulanması kararlaştırılmıştır:

- Yazılım Gereksinimleri Gözden Geçirme
- Yazılım Tasarım Gözden Geçirme
- Test Durumları Gözden Geçirme
- Test Hazırlığı Gözden Geçirme
- Test Sonuçları Gözden Geçirme

Proje gereksinimleri doğrultusunda aşağıdaki gözden geçirmeler de uygulanmaktadır:

- Kod Gözden Geçirme

- Yazılım Entegrasyonu Gözden Geçirme
- Yapısal Kapsama Analizi Gözden Geçirme

Bu gözden geçirmelerin her biri, ilgili gözden geçirmeye özgü önceden tanımlanmış kontrol listesi adı verilen bir soru listesi vasıtasıyla yerine getirilmektedir. Gözden geçirme toplantısından önce katılımcılar tarafından incelenmiş olan yazılım iş ürünü kontrol listesindeki sorular kapsamında değerlendirilmekte, açık kalan, hatalı bulunan noktalar ve iyileştirme önerileri soru listesinin altına not edilen işlem maddeleri ile kayıt altına alınarak düzeltilmesi amaçlanmaktadır.

Yazılım gözden geçirmesi insan odaklı bir doğrulama aktivitesi olduğundan uygulamada yazılım gözden geçirme aktivitelerinin etkinliğini azaltacak bir takım güçlüklerle karşılaşmaktadır. Aynı zamanda gözden geçirmelerin etkinliğini etkileyen diğer bir faktör ise gözden geçirme süreci organizasyonu ve gözden geçirmelerde kullanılan altyapıdır [12].

Gözden geçirmelerin yürütülmesi, kontrol listeleri vasıtasıyla kayıtlarının tutulması, yazılım iş ürünü ile ilgili eksiklikleri, hataları, uyumsuzlukları ve iyileştirme noktalarını belirleyen işlem maddelerinin tanımlanması ve takibinde pratikte bir takım güçlükler ile karşılaşıldığı yaşanan tecrübeler sonucu gözlemlenmiştir. Gözden geçirmeler ile ilgili uygulamada karşılaşılan bu güçlükler şu şekilde özetlenebilir:

- Gözden geçirme kayıtlarının düzenli olarak, bütün proje çalışanlarının erişebileceği ortak bir alanda saklanamaması (Gözden geçirme kayıtlarının kişisel bilgisayarlarda tutulması ve gerektiğinde kayıtların bulunmasında karşılaşılan güçlükler)
- Aynı tip gözden geçirme için standart dışı farklı kontrol listeleri ile karşılaşıyor olması
- Gözden geçirmelerde kayıt altına alınan işlem maddelerinin kapatılıp kapatılmadığını kontrolünde karşılaşılan güçlükler
- Gözden geçirmelerde kayıt altına alınan işlem maddelerinin sorumlular bazında takip edilmesinde karşılaşılan güçlükler
- Gözden geçirmelerde kayıt altına alınan işlem maddelerine miat verilememesi sonucunda işlem maddesinin gerçekleşme zamanının kontrol edilememesi

- Gözden geçirmelerde kayıt altına alınan işlem maddelerinin kapatılması sırasında nasıl bir işlem yapıldığına dair izlenebilirliğin bulunmaması

2. Gözden Geçirme Yönetim Aracı

Karşılaşılan bu problemlere çözüm getirmek ve gözden geçirmelerin etkin bir şekilde yürütülebilmesini sağlamak amacıyla Görüntü İşleme Müdürlüğü bünyesinde yapılan çalışmalar sonucunda tüm yazılım geliştirme projelerinde kullanılacak bir gözden geçirme yönetim aracı geliştirilmiştir. Bu araç temel olarak aşağıdaki gereksinimler doğrultusunda geliştirilmiştir:

- Gözden geçirme kayıtlarının merkezi bir veri tabanında tutulabilmesi,
- Kayıtların istendiğinde rahatça bulunabilmesi
- Standart kontrol listelerinin kullanılabilmesi
- İşlem maddeleriyle ilgili olarak
 - Miatlarının ve açık/kapalı durumlarının takip edilebilmesi
 - Kişi bazında işlem maddelerinin takip edilebilmesi
 - İşlem maddelerinin kontrollü olarak kapatılabilmesi

Bu bildiri ile bu aracın özellikleri, kullanımı ve getirdiği faydalar konusunda bilgi verilmesi amaçlanmıştır.

Gözden geçirme yönetim aracı yukarıda da belirtildiği gibi temel olarak yazılım süreçlerinde tanımlanmış eş düzey gözden geçirmelerin bilgisayar ortamında kayıtlarının tutulabilmesi, yukarıda anlatılan pratikte karşılaşılan sorunların giderilmesi amaçlanarak geliştirilmiştir.

Gözden geçirme yönetim aracı geliştirilirken mevcut süreçler ile tanımlanmış ve form olarak dokümantasyon sistemine girmiş gözden geçirme kontrol listeleri esas alınmıştır. Araç geliştirilirken gereksiz detaylardan kaçınılarak kullanım kolaylığı ön planda tutulmuştur. Aşağıdaki bölümlerde sırasıyla araç ile ilgili genel bilgiler, kullanıcı ismi tanımlama, kontrol listesinin doldurulması, işlem maddelerinin doldurulması ve takibi konusunda bilgiler örnekleriyle birlikte verilmiştir.

3. Gözden Geçirme Yönetim Aracının Özellikleri

3.1 Genel

Gözden geçirme yönetim aracı (GGYA), PHP ve MySQL kullanılarak yazılmış, yerel ağdaki bir web sunucusunda çalışan bir arayüzdür. Kullanıcı kayıt ve giriş işlemleri için Joomla alt yapısı kullanılmış, görsel düzenlemeler ve sayfa zaman aşımaları ile ilgili olarak Joomla içerisinde değişiklikler yapılmıştır.

GGYA, merkezi bir sunucuda bulunmakta ve kullanılmak isteyen bütün çalışanlar <http://gimserver2/rev> gibi basit bir adres vasıtasıyla uygulamaya erişebilmektedir.

Gözden geçirme kayıtları araç vasıtasıyla istenildiğinde yazıcıdan bastırılabilir veya istenen kullanıcıya e-posta yoluyla ulaştırılabilir.

3.2 Kullanıcı Oluşturma

Arayüzün amacına uygun bir şekilde kullanılabilmesi, özellikle işlem maddesi takibinin yapılabilmesi için tüm kullanıcıların birer kullanıcı adı oluşturması gerekmektedir. Bunun için ana sayfada “Kayıt Olma” bağlantısı kullanılmaktadır. Bu bağlantı ile kullanıcı adı yerel ağ e-posta adresi ve şifre girerek kayıtlı kullanıcı olunabilmektedir. Kullanıcı şifresi unutulduğu hallerde kullanıcı adı veya e-posta adresi kullanılarak aracın bulunduğu sunucu üzerinden e-posta gönderimi yoluyla kullanıcı bir doğrulama göstergesi vasıtasıyla yeni bir şifre oluşturup uygulamayı kullanmaya devam edebilmektedir.

3.3 Ana Sayfa

Action Items:

[Manage your action items](#)

You have 4 open action items and none of them are overdue.

Show action items on user:

Filter Checklists:

Manager: Project: Checklist Type:

Create new checklist:

Checklist Type: Project:

#	Checklist Identifier	Manager	Project	Modification Date	Type	Actions
1	dfsdf	İzzet Keviravus	ASELPOD	08:24 13.04.09	tr	read print delete
2		Ömer Karaduman	ATAK	16:11 10.04.09	sr	read print modify delete
3		Ömer Karaduman	ATAK	14:19 10.04.09	sr	read print modify delete
4		Ömer Karaduman	ATAK	15:58 08.04.09	sr	read print modify delete
5		Celal Ademözü	ATAK	11:44 07.04.09	sr	read print delete
6	ASELPOD PYY	Deriis Aköz	ASELPOD	12:22 03.04.09	sr	read print delete

Şekil 2: Açılış Sayfası (Ana Sayfa)

Açılış sayfası Şekil 2’de de görüldüğü gibi dört bölümden oluşmaktadır, bunlar:

1. *Action Items*
2. *Filter Checklists*
3. *Create Checklist*
4. *Checklist List*

3.4 İşlem Maddeleri (Action Items)

Bu bölümde, giriş yapmış olan kullanıcı kendi üzerindeki işlem maddeleri ile ilgili özet bilgiyi görebilmektedir.

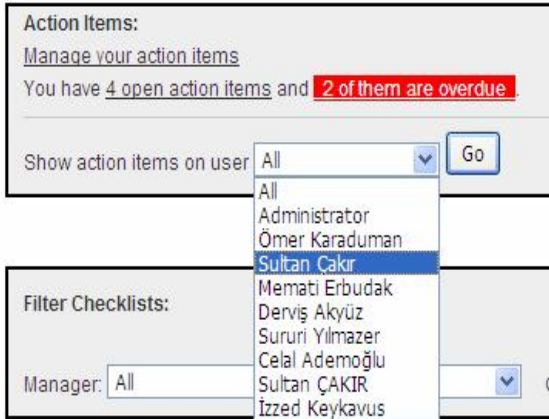
Giriş yapmış olan kullanıcı üzerinde, süresi dolduğu halde halen kapatılmamış işlem maddeleri varsa kullanıcı, Şekil 3’deki gibi bilgilendirilecektir.



Şekil 3: İşlem Maddeleri Özeti

Altı çizili metinler ilgili bağlantılar olup “Manage your action items” ile aktif kullanıcı, üzerindeki açık/kapalı/onaylı bütün işlem maddelerini görebilecek, diğer bağlantılarla da açık bütün işlem maddelerini veya açık ve teslim tarihi geçmiş işlem maddelerini listeleyebilecektir.

Yine bu bölümde, kullanıcılar üzerindeki işlem maddelerinin listelenebilmesi sağlanmıştır. Seçim kutusundan kullanıcı seçilerek (bknz:Şekil 4) o kullanıcıya ait bütün işlem maddeleri görülebileceği gibi “All” seçilerek, tanımlı bütün işlem maddeleri de listelenebilir.



Şekil 4: Kullanıcıya Göre İşlem Maddesi Listeleme

3.5 İşlem Maddesi Listesi

Herhangi bir şekilde işlem maddeleri listelendiğinde Şekil 5’deki gibi bir sayfa açılacaktır.



ID	Action Item	Assignee	Status	Due Date	Comment	Project	Actions
953	ee	Ömer Karaduman	Open	31/12/2009		ATAK	Close view checklist
957		Ömer Karaduman	Open	31/12/2009		ATAK	Close view checklist
954		Ömer Karaduman	Open	16/12/2009		ATAK	Close view checklist
952	ee	Ömer Karaduman	Open	10/12/2009		ATAK	Close view checklist
951	ee	Ömer Karaduman	Closed	16/04/2009	das dasdqe21 13123	ATAK	view checklist
956	a sdas	Ömer Karaduman	Closed	15/04/2009	d as131231231232	ATAK	view checklist
958		Ömer Karaduman	Closed	15/04/2009	asd asdasd	ATAK	view checklist
949		Ömer Karaduman	Closed	11/04/2009	sdasdasd	ATAK	view checklist

Şekil 5: İşlem Maddeleri Listesi

Siteye giriş yapmış olan kullanıcıya ait işlem maddelerinin “Comment” sütununda, işlem maddesi ile ilgili yorum girilebilmesini sağlayan bir kutu görüntülenecek, diğer kullanıcılara ait veya kapalı işlem maddelerinde ise bu sütunun değeri değiştirilememektedir. Kullanıcı, kapatmak istediği işlem maddesine bir yorum yazıp “Close” tuşuna basarak bu işlem maddesini kapatabilmektedir.

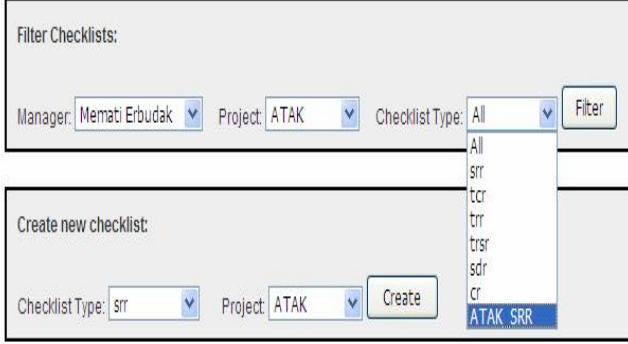
Kapatılan bir işlem maddesinin kullanıcı tarafından yeniden açılması mümkün olmamakta, sistemde bu hak yalnızca ilgili kontrol listesini oluşturan kişiye ve proje kalite temsilcisine verilmiş durumdadır. Sorumlusu tarafından kapatılan işlem maddelerini onaylama yetkisi sadece ilgili projenin kalite temsilcisinde bulunmaktadır. Böylelikle işlem maddelerinin düzgün, zamanında ve tanımlı süreçlere uygun bir şekilde kapanıp kapanmadı kontrolünün bağımsız bir otorite tarafından yapılması sağlanmıştır.

Bu sayfadaki tabloda, “Due Date” sütunun başlığına tıklanarak, işlem maddeleri teslim tarihine göre artan veya azalan şekilde sıralanabilmektedir.

“Assignee” kolonundaki bir isme tıklandığında, o kişi üzerindeki bütün işlem maddeleri gösterilmektedir. “Status” kolonundaki “Open”, “Closed” veya “Approved” bağlantılarından birisine tıklanarak ise seçili kullanıcı üzerindeki, tıklanan “Status”teki bütün işlem maddelerinin listelenmesi sağlanmaktadır.

3.6 Kontrol Listesi Filtreleme (Filter Checklists)

Gözden geçirme kontrol listeleri, liste yöneticisi (listeyi oluşturan, gözden geçirilmiş olan yazılım ürününün sorumlusu), proje ve liste tipi parametreleri ile filtrelenebilir. Seçimleri yapıp Şekil 6'daki gibi "Filter" tuşuna basıldığında yapılmış olan seçime uygun listeler gösterilecektir.



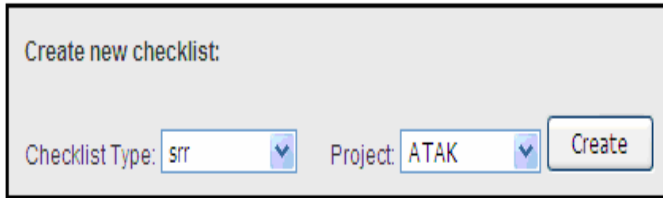
Şekil 6: Kontrol Listesi Filtreleme

Kontrol listeleri tablosunun en sağ ("Actions") sütununda, o satırdaki liste ile ilgili yapılabilecek işlemler gösterilmektedir. Giriş yapmış kullanıcı tarafından oluşturulmuş listeler için değiştirme ("modify") seçeneği bulunurken diğer listelerde sadece okuma ve yazdırma hakkı gösterilmektedir.

Bu listede de, "Modification Date" e başlığına tıklayarak listelerin değiştirilme tarihine göre artan veya azalan şekilde sıralanabilir.

Ayrıca, "Manager", "Project" ve "Type" sütunlarındaki verilere tıklayarak, tıklanılan veriye göre filtreleme yapılabilir. (örneğin "ATAK" bağlantısına tıklayınca ATAK projesine ait bütün doğrulama listeleri listelenmektedir)

3.7 Kontrol Listesi Oluşturma (Create New Checklist)



Şekil 7: Kontrol Listesi Oluşturma

Şekil 7'de gösterilen kutulardan, oluşturulacak Kontrol Listesinin tipi ve hangi projeye ait olduğu seçilerek Şekil 8'da gösterildiği gibi yeni bir gözden geçirme kontrol listesi oluşturma ekranı açılabilir.

aselsan Microelectronics, Guidance and Electro- optics Division	ATAK Software Requirements Review Checklist ATAK	Date: 13.04.2009 Page:
---	---	---------------------------

Checklist Identifier	
Version/Release Date	
Standards Used	
Reviewer(s)	

Checklist Item	Yes/No	Comments
1. Are all requirements allocated to software from the System Specification covered and defined?		
2. Are all high-level software requirements traceable? (i.e. all system requirements allocated to software are addressed by one or more high-level software requirements and all non-derived		

Şekil 8: Yeni Kontrol Listesi

Bilgi girişi yapılan yerler genelde metin kutusu şeklindedir. "Checklist Identifier", "Version/Release Date" ve "Standards Used" bölümleri düz metin olarak girilebilir. "Reviewers" kısmı içinse farklı bir yöntem izlenmektedir. "Reviewers" kısmına girilen metin, kullanıcı adları içinde aranacak ve bu metni içeren kullanıcı isimleri Şekil 9'daki gibi listelenecektir.

Reviewer(s)	ad Administrator-admin Ömer Karaduman-gomercin Celal Ademoğlu-cademoğlu
Checklist Item	Yes/No

Şekil 9: Kullanıcı İsmi Listeleme

Eklenmek istenen kullanıcı bu listeden seçilip tıkladığında, kullanıcı katılımcı olarak eklenmektedir. Kullanıcıları bu şekilde eklenmesi ile gözden geçirme katılımcı listesi oluşturulmaktadır. Eğer, eklenmek istenen kullanıcı veritabanında bulunamazsa (kayıtlı değilse), "ismi ekle" diye bir seçenek gösterilmekte ve isim, "ismi ekle" bağlantısına tıklanması suretiyle listeye eklenebilmektedir.

Eklenmiş bir ismi listeden silmek için ismin üzerine tıklamak yeterli olacaktır. Üzerine geline isim kırmızı arka plan rengi ile işaretlenecektir (bknz: Şekil 10) ve tıkladığında listeden atılacaktır.

Reviewer(s)	Ömer Karaduman, Memati Erbudak, Sururi Yilmazer,
-------------	--

Şekil 10: Katılımcı Listesinden İsim Silme

Kontrol listesindeki sorulara cevaplar normal metin olarak verilebilir. (“YES/NO” başlığındaki kısım da giriş biçiminde esneklik sağlamak için düz metin olarak bırakılmıştır). Metin kutuları, girilen içeriğe göre otomatik olarak büyümektedir.

3.8 İşlem Maddesi Oluşturma Listesi İşlemleri

İşlem maddeleri listesi tablosunda solda görünen “Add” bağlantısına tıkladığında Şekil 11’deki gibi yeni, boş bir işlem maddesi satırı eklenecektir.

Action items						
Item ID	Review Comments	Assignee Comment	Assignee	Due Date (gg/aa/yyyy)	Action(s) Taken	Remove Item
				13/4/2009	Açık	Remove

Şekil 11: Yeni İşlem Maddesi Ekleme

Item ID: Kullanıcı tarafından değiştirilemeyen, işlem maddesine ait eşsiz numaradır. Liste kaydedildiğinde doldurulacaktır

Review Comments: İşlem maddesi ile ilgili yorumlardır, düz metin olarak girilir.

Assignee Comment: İşlem maddesinin atandığı kişinin yorumudur. Toplantıda boş bırakılabilir, işlem maddesinin atandığı kişi tarafından, işlem maddesi kapatılırken doldurulacaktır

Assignee: İşlem listesinin atandığı kişidir. Her bir işlem maddesi için yalnızca tek bir kişi atanabilir. Aynı işlem maddesi için birden fazla kişi atanması gerekirse her bir işlem maddesi sahibi için yeni bir satır açılması gerekecektir. Burada da, kutuya girilen metin kullanıcılar içerisinde aranacak ve uyan kullanıcı isimleri Şekil 12’de gösterildiği gibi listelenecektir.

Action items	
Assignee	Due Date (gg/aa/yyyy)
Sul	13/4/2009

Sultan Çakır -cakir
SubnSururi Yilmazer -chrome
Sultan ÇAKIR -sucakir

Şekil 12: İşlem Maddesi Sorumlusu Atama

Yine “Reviewers” kısmındaki gibi istenen kullanıcıya tıklanarak bu kişi işlem maddesine atanabilir. Kullanıcı

atandıktan sonra metin kutusu kaybolacak ve yalnızca kullanıcı ismi kalacaktır. Farklı bir isim atamak istenirse mevcut kullanıcı, isminin üstüne tıklanarak silinebilir. Silme işleminden sonra metin kutusu geri gelecektir.

Due Date: İşlem listesinin bitirilmesi istenen tarih girilmelidir. Başlangıç değeri o günün tarihidir, Kutuya tıkladığında “pop-up” bir pencere açılır ve buradan istenilen tarih seçilebilir. Bir tarih seçildiğinde pencere kapanır ve seçilen tarih kutuya girilmiş olur.

Actions Taken: İşlem maddesi ile ilgili yapılan işlemi gösterecektir, başlangıçta “Açık” olacaktır. Kullanıcı tarafından kapatılmışsa “kapalı” olarak gösterilecektir. “Kapalı” durumuna getirilmiş işlem maddeleri yine buradan “onaylı” hale getirilebilecektir.

Remove Item: İşlem maddesi listesi oluşturulurken, eklenen işlem listesi satırının silinmesi için kullanılabilir. “Submit” tuşuna basılana kadar silme işlemi sadece görünürde gerçekleşecek, eğer “Submit” tuşuna basılmadan sayfadan çıkılırsa, sayfa yeniden açıldığında herhangi bir silme işlemi yapılmamış olacaktır (Bu durum sayfadaki bütün veriler için geçerlidir)

En alttaki “Submit” tuşuna basılarak sayfa kaydedilebilir. “Submit” tuşuna basıldığında aynı sayfaya geri dönülecek ve liste oluşturulmaya devam edilebilecektir.

4. Sonuç

GGYA, kullanılmaya başlandığı yaklaşık iki yıllık süre zarfında ASELSAN MGEO bünyesinde tüm yazılım projelerinde etkin olarak kullanılmaktadır. Bu bildirinin yazıldığı tarih itibarıyla araçta yüzün üzerinde eş düzey gözden geçirme kaydı bulunduğu gözlemlenmiştir. Uygulama ile ilgili kullanıcılar tarafından yapılan değerlendirmelerin olumlu olduğu da bilinmektedir. GGYA süreçler çerçevesinde tanımlı olan gözden geçirmelerin sistematik bir şekilde yapılmasını, kayıtlarının güvenilir bir şekilde tutulmasını, her bir gözde geçirme sonucunda kayda alınan işlem maddelerinin kolaylıkla takibini ve zamanında kapatılmasını sağlamıştır. Bu sayede hem gözden geçirilen yazılım gereksinimleri, yazılım tasarımı, yazılım test durumları gibi ara ürünlerin standartlara uyumlu çıkmasını hem de kalite açısından iyileşmelerini sağlamış, hem de nihai üründe ortaya çıkabilecek hataların erken aşamalarda bulunup düzeltilmesini sağlamıştır.

Netice itibarı ile GGYA, bildirinin giriş kısmında anlatıldığı gibi yazılım geliştirme sürecinin önemli bir parçasını oluşturan ve sistematik bir şekilde uygulandığında büyük faydalar getirdiği gözlemlenen gözden geçirmelerin insan odaklı bir aktivite olmasından kaynaklanan uygulama güçlüklerini minimuma indirmiştir. Halen devam etmekte olan

projelerde DO-178B gibi standartlara uyumlu geliştirme süreçleri takip edildiği düşünüldüğünde GGYA üzerindeki gözden geçirme kayıtları standartların uyum kayıtları olarak müşteri tarafından da incelenmekte ve kabul görmektedir.

5. Kaynaklar

- [1] Humphrey, W. S. *Winning with Software: An Executive Strategy, How to Transform Your Software Group Into A Competitive Asset*, Boston, Addison-Wesley, 2002
- [2] Sommerville, I., *Software Engineering* (6th ed.). Harlow, UK: Addison-Wesley, 2001
- [3] Boehm, B. W., & Basili, B. R.. Software Defect Reduction Top10 List. *IEEE Computer*, 135-137, January 2001
- [4] Fagan, M. E., *Advances in Software Inspections*. IEEE Transaction on Software Engineering, 12(7), 1986
- [5] Fagan, M. E., . Design and Code Inspections to Reduce Errors in Program Development. *IBM System Journal*, 15(3), 182-211, July 1976
- [6] Xu, J.,. On Inspection and Verification of Software with Timing Requirement. *IEEE Transaction on Software Engineering*, 29(8), 705-720, August 2003
- [7] O'Neill, D.,. Issues in Software Inspection. *IEEE Software*, 18-19, January, 1997
- [8] Davis, A. M. (1993). *Software Requirement: Objectives, Functions, and States*. Englewood Cliffs, NJ: Prentice-Hall
- [9] Leffingwell, D., & Widrig, D., *Managing Software Requirements: A Unified Approach*. NJ: Addison Wesley, 2000
- [10] Karl E. Wiegers, "Peer Reviews in Software A Practical Guide," Addison-Wesley Information Technology Series, 2002
- [11] T. Saraç, "Yazılım Ürün Gözden Geçirmelerinin Önemi, Hazırlık Süreci ve Bir Uygulama Örneği", YKGS-2008.
- [12] Briand L. C., Freimut B., Vollei F., Using Multiple Adaptive Regression Splines to Understand Inspection Data and Identify Optimal Inspection Rates, Fraunhofer Institut, Report No. 062.00/E, January 2001