

BASAMAKLI CMMI MODELİ ile EXTREME PROGRAMMING METODUNUN DEĞERLENDİRİLMESİ

Emin BORANDAĞ*, Fatih YÜCALAR*, Senol Zafer ERDOĞAN*, Fuat İNCE†

*Maltepe Üniversitesi, Mühendislik Fakültesi, 34857 Maltepe İstanbul, Türkiye
e-mail: {eminb, fatihy, senole}@maltepe.edu.tr

†Marmara Üniversitesi, Mühendislik Fakültesi, 34722 Kadıköy İstanbul, Türkiye
e-mail: fince@eng.marmara.edu.tr

Alınış: 6 Ekim 2008
Kabul Ediliş: 27 Şubat 2009

Özet: Günümüzde yazılım sistemlerinin iş alanlarının temel ve vazgeçilmez bir bileşeni olmasıyla yazılım geliştirme de kritik bir önem kazanmıştır. Son yirmi yılda yazılım geliştirmede kullanılan kalite sistemlerini ve süreçlerini değerlendirmek, yazılımda kalite sertifikasyonunu sağlamak, süreçleri iyileştirmek ve yetenek belirlemek için çeşitli modeller geliştirilmiştir. Son birkaç yıl içinde bunlar arasında en yaygın kullanılanı CMMI modeli olmuştur.

Geliştirilen bu süreç değerlendirme modelleri ile büyük ölçekli birçok yazılım firması yazılım ürünleri geliştirmek için süreç odaklı yaklaşımlar benimsemişlerdir. Ancak bu modeller, özellikle küçük ölçekli yazılım firmaları için yerine getirilmesi zor görünen şartlar içermektedirler. Bu nedenle 1990'lı yılların sonuna doğru, yazılım geliştirme süreçlerine alternatif bir bakış açısı olarak daha esnek ve uygulama kolaylığı var gibi görünen Çevik Metotlar geliştirilmiştir. En belli başlı Extreme Programming olan çevik metotlara Scrum, RUP, Crystal Family gibi diğer örnekler verilebilir.

Bu çalışmada, CMMI yazılım süreç değerlendirme modeli çerçevesinde bir yazılım geliştirme metodu olan Extreme Programming değerlendirilmiştir. Bu değerlendirme, CMMI 2. düzey süreç alanlarının Extreme Programming'in özel uygulamaları ve yaşam döngüsünün özellikleri tarafından karşılanmayacağı sorusuna cevap aramaktadır.

Anahtar Kelimeler: Çevik Metotlar, Extreme Programlama, XP, CMMI

Evaluation of Extreme Programming Method with CMMI Staged Model

Abstract: As a consequence of software systems becoming an integral and indispensable component of business operations, software development has attained critical importance. Several models have been developed in the last 20 years or so, to aid in the establishment and assessment of quality systems for software development, as well as improving processes and determining capability. CMMI is currently the most widely accepted and adopted one of these models.

Many large scale software development firms have based their development operations on well defined processes, derived from such models as CMMI. However the requirements of comprehensive models such as CMMI may be difficult to meet by smaller firms which prefer more informal ways. For this reason the so called agile methods have gained popularity among small scale firms since the late 1990s. Perhaps the most popular of the agile methods is Extreme Programming or XP, other examples being Scrum, RUP, and Crystal Family.

The purpose of this paper is to evaluate the XP, which is a software development method, in terms of CMMI which is a process assessment model. In other words, we will try to answer if and to what extent XP practices can meet the CMMI level 2 specifications, and what additional practices may or may not be necessary for full compliance.

Key Words: Agile Methods, Extreme Programming, XP, CMMI

1. Giriş

Yazılım krizi sonrasında geliştirilen yazılım kalite modellerinden biri olan CMM (Yetenek Olgunluk Modeli – Capability Maturity Model) [1], zaman içerisinde yapısını ve adını CMMI (Tümleşik Yetenek Olgunluk Modeli – Capability Maturity Model Integration) olarak değiştirerek daha güçlü bir şekil almıştır. CMMI, en iyi uygulamaların

özelliklerine bakılarak ortaya çıkan, yazılım şirketlerinin kaliteli yazılımlar üretmesini ilke edinen bir yazılım süreç değerlendirme modelidir.

Yazılım üreten büyük çaplı ve saygın firmaların büyük bir kısmı; CMM, CMMI, SPICE (Yazılım Süreç İyileştirme ve Yetenek Belirleme - Software Process Improvement and Capability dEtermination) [2] gibi süreç iyileştirme ve yetenek belirleme modellerini göz önünde bulundurularak yazılımlarını geliştirilmektedirler [3]. Bu modellerin kullanımı ile geliştirilen yazılımların başarı oranları artmıştır. Bununla beraber bazı sorunlar da ortaya çıkmıştır. Bunlar; yazılım sürümlerinin geç ortaya çıkması, değişiklik isteklerine çabuk cevap verilememesi, yazılım hatalarının geç fark edilmesi ve zaman içerisinde gelen isteklere göre sistemin kendi yapısını geliştirememesi gibi sorunlardır. Bu sorunların aşılmasına yönelik yapılan çalışmalar sonucu “Agile (çevik)” olarak isimlendirilen metotlar geliştirilmiştir. Extreme Programming, Scrum [4], RUP (Parçalı Birleştirilmiş Süreç - Rational Unified Process) [5], Crystal Family gibi çevik metotlar vardır [6]. Bu metotlar arasında en önemlisi kısaca XP olarak anılan “Extreme Programming” dir. XP, hızlı ve esnek yazılım geliştirmeyi amaçlayan bir yazılım geliştirme metodudur [7][8].

CMMI ve XP yazılım üretme işi ile ilgilenir, ama yazılım üretme işine bakış açıları ve işlevleri bir birinden farklıdır [9].

CMMI bir yazılım üretme metodu değildir, süreç değerlendirme modelidir. XP ise, bir yazılım üretme metodudur. Bir firma hangi metodu uygularsa uygulasin CMMI tarafından değerlendirilebilir. Hâlbuki XP bir firmanın seçebileceği çok sayıda metottan biridir. CMMI ise, metot ne olursa olsun, yapılan işlerin neyi amaçladığına, neyi elde ettiğine bakarak değerlendirmeyi gerçekleştirir. İşlerin nasıl yapılması gerektiğini ortaya koymaz. Şirketlerin yazılım geliştirme durumlarına bakan bir kontrol mekanizması gibidir.

XP ise, çalışma ortamının nasıl olacağı ve süreçlerin nasıl gerçekleştirileceği ile ilgilenir. Bir kontrol mekanizmasından ziyade, yazılım üretiminin nasıl yapılması gerektiğini söyler.

Çalışmanın 2. bölümünde kısaca CMMI ve yapısı, 3. bölümünde XP, 4. bölümünde CMMI 2. düzey süreçleri ile XP uygulamalarının karşılaştırılması, son bölümde ise elde edilen sonuçlar anlatılmaktadır.

2. CMMI ve Yapısı

CMMI süreç iyileştirme ve yetenek belirleme modeli olarak ABD Savunma Bakanlığı'nın isteği üzerine Carnegie Mellon Üniversitesi, Yazılım Mühendisliği Enstitüsü tarafından geliştirilmiştir. CMMI, organizasyonların insan kaynaklarını, süreçlerini ve teknolojilerini geliştirecek şekilde olgunlaşmasını sağlamaya yönelik bir modeldir [10].

CMMI bir süreç değerlendirme modelidir. Ancak bunun arkasında iki temel amaç vardır. Bunlardan ilki, yazılım geliştiren kuruluşlardaki süreç iyileştirme çabalarına rehberlik etmek; ikincisi ise, yazılım geliştirme işini gerçekleştirmek üzere sözleşme yapılacak nitelikli şirketlerin belirlenmesine yardımcı olmaktır [11].

CMMI modelinin yapısal temel elemanı süreç alanları (process areas) dir. CMMI'da 25 süreç alanı mevcuttur. Her süreç alanı için, birtakım hedefler (goals) tanımlanmaktadır. Bu tanımlamayı yaparken hedefleri, özel hedefler (specific goals) ve genel hedefler (generic goals) olmak üzere ikiye ayırmaktadır. Genel hedefler, her süreç alanı için geçerlidir. Buna karşılık her süreç alanının kendisine özgü özel hedefleri vardır. CMMI her bir hedefe ulaşılması için bir takım uygulamalar (practices) öngörür. Uygulamalar, hedefleri yerine getirmede beklenen bileşenlerdir. Uygulamalar, genel ve özel olmak üzere iki sınıfa ayrılır [11]. Özel hedeflerin özel uygulamaları, genel hedeflerin de genel uygulamaları vardır.

CMMI, sürekli ve basamaklı olmak üzere iki gösterim şekli kullanılmaktadır. Sürekli (continuous) gösterim süreçlerin yetenek düzeylerini tanımlarken, basamaklı (staged) gösterim ise kuruluşun olgunluk düzeyini tanımlar [11]. Basamaklı gösterim bir organizasyonun bir olgunluk seviyesini hedefleyerek, hangi süreç alanlarının iyileştirmesi gerektiğini belirler. Sürekli gösterim ise, bir organizasyonun belirli süreç alanlarını seçerek süreçlerini bu doğrultuda geliştirmesine imkân sağlar.

3. Extreme Programming (XP)

XP, yazılım geliştirmeye yeni bir bakış açıdır. Genel duyumundan bu yana yaklaşık dokuz yıllık bir geçmişi vardır ve içeriği gereği dinamik ve değişken projelerde uygulanabilen bir yapıya sahiptir. En önemli avantajı, müşterilerle kurduğu çok yakın ilişkidir. XP ile müşterilerin ihtiyaçları düzenli olarak algılanıp, gerektirdiği şekilde çözümlenmeye çalışılır [12].

XP'nin ortaya çıkışı 1990'lı yılların sonlarına doğru, yazılım dünyasında çeşitli eleştirilerin yüksek sesle dile getirilmesiyle başlamıştır [13]. Bu eleştiriler şunlardır:

- Sürümlerin çok geç ortaya çıkması.
- Sürümler çok geç ortaya çıktığı için hataların geç anlaşılması.
- Esnek olmaması nedeniyle, yeni gelen istekler doğrultusunda kendi yapısını değiştiremediği için, yeni istekleri karşılayamaması.
- Değişikliklerin geç ve zor yapılması.

Bu ve buna benzer eleştiriler doğrultusunda Kent Beck tarafından bir yazılım geliştirme disiplini olarak XP ortaya çıkarılmıştır. XP, grup içi iletişime çok önem veren bir yazılım geliştirme yöntemidir [14][15].

XP ile proje yaşam döngüsü (life cycle) çerçevesinde proje geliştirimi sağlanır. Proje yaşam döngüsü bazı uygulamalara (practices) bağlıdır. Yazılım geliştirmede kolaylığı ve esnekliği sağlamak için; XP, 12 farklı uygulamayı öngörür. Bunlar;

- Planlama Oyunu,
- Ekipte Müşteri,
- Önce Test,
- Basit Tasarım,
- Çiftli Programlama,
- Sürekli Tümlleştirme,
- Kısa Sürümler,
- Yeniden Yapılandırma,
- Ortak Sahiplenme,
- Metaphor,
- Kodlama Standardı,
- 40 Saat Haftada [16].

XP ile klasik yöntemler arasındaki karşılaştırma aşağıda yer alan Tablo 1'deki gibi özetlenebilir.

Tablo 1: XP ile klasik yöntemlerin karşılaştırmaları [17]

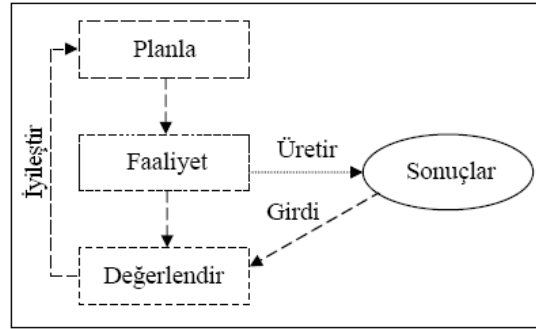
Extreme Programming	Klasik Yöntemler
Kalıcı Yazılım Ekibi	Ekip Üyeleri Değişebilir
Değişikliklere açıktır	Değişiklikler kontrol altındadır.
Küçük yazılım ekipleri için uygundur	Genelde büyük yazılım ekiplerince tercih edilir.
Uzman bir yazılım ekibi ister	XP kadar yüksek kalitede yazılımcıya ihtiyaç duymaz.
Belgeleme azdır	Belgelemeye çok önem verilir.
Projelere bakış açısı aşağıdan yukarıdır.	Projelere bakış açısı yukarıdan aşağıdır.
Küçükçe ekipleri öngörür.	Büyük ekipler için gerekli görülür
Kısa süreli (1 yılın altında) projeler için uygundur.	Uzun süreli projeler için uygundur.

4. CMMI 2. Düzey Süreçleri ile Extreme Programming Pratiklerinin Karşılaştırılması

CMMI 2. düzey süreçlerine ait isterlerin XP ile nasıl karşılaştırılacağına bakmadan önce CMMI 2. düzeyin ne anlama geldiğine bir bakarsak daha kolay şekilde bu karşılaştırmayı gerçekleştirebiliriz. CMMI 2. düzey bir yazılım projesinin başarılı olabilmesi için minimum koşulları belirler. Bu koşullar aşağıda verilen 7 süreç alanı içinde tanımlanmıştır:

- Gereksinim yönetimi,
- Proje planlama,
- Proje izleme ve takip,
- Tedarikçi sözleşme yönetimi,
- Ürün ve süreç kalite güvencesi,
- Konfigürasyon yönetimi,
- Ölçme ve değerlendirme

Şekil 1'de gösterildiği üzere, CMMI 2. düzey çalışmalarına başlamadan önce planlama yapılması gerekir. Plan içerisindeki işlemlerin sonuçlarına göre faaliyet değerlendirmesi yapılır ve değerlendirmeden sonra sonuçlara bakılarak temel testler yapılarak yeniden değerlendirilir [18].



Şekil 1. CMMI 2. Düzey Yapısı

4.1. Gereksinim Yönetimi

Gereksinim yönetimi süreç alanının içinde tek bir özel hedef vardır. Bu özel hedef “gereksinimleri yönet” mektir. Gereksinim yönetiminin amacı, projede ortaya çıkacak ürünleri ve ürünleri oluşturacak yapıları yönetmek ve proje çerçevesinde ihtiyaç duyulan bileşenler arasındaki tutarsızlıkları ortadan kaldırmaktır [8][18]. CMMI’ ın “Özel Hedefleri”, aşağıda Ö.H. olarak kısaltılmıştır.

Ö.H.1: Gereksinimleri Yönet: CMMI bu özel hedefi 5 özel uygulama ile tanımlar. XP ise, bu özel hedefi yerine getirebilmek için; kısa sürede ortaya çıkan küçük sürümleri kullanma, takımda müşteriden bir kişinin bulunması gibi özellikleri kullanarak gereksinim yöntemi sürecinin özel hedefini karşılamaya çalışır.

Bununla beraber, gereksinim yönetimi özel hedefinin bir özel uygulaması olan taahhütleri alarak gereksinimlerin karşılanması müşteri ve yazılımcı arasında kapsamlı bir şartnameyi ön gördüğünden XP bu özel hedefi tam olarak karşılamamaktadır. Bu da zaten XP’ nin baştan karşı çıktığı bir husustur.

4.2. Proje Planlama

Proje planlama süreç alanı içinde üç özel hedef vardır. Bu özel hedefler; kestirimlerin oluşturulması, proje planının geliştirilmesi, proje planı için taahhütlerin oluşturulmasıdır. Proje planlama süreç alanının amacı, proje faaliyetlerini tanımlayan planı gerçekleştirmek ve bu plan doğrultusunda hareket edilmesini sağlamaktır. Proje planlamanın temel faaliyetleri, proje planını oluşturmak, paydaşlarla gerekli iletişimin kurulmasını sağlamak, oluşturulan plan için taahhütlerin alınmasını gerçekleştirmektedir. Genel olarak planın güncel bir şekilde tutulmasını amaçlar [8][18].

Ö.H.1: Kestirimlerin Oluşturulması: CMMI bu özel hedefi 4 özel uygulama ile tanımlar. XP’ de ise, bu özel hedefi yerine getirebilmek için mevcut duruma göre bir plan yapılır. Yazılım takımı belirlenir. Yönetici, koç, test, ölçüm sorumluları ve danışmandan oluşan takım, planın güncelliğinin korumasından sorumludur. Ayrıca planlama oyunu kullanılarak proje çerçevesi tam olarak belirlenmeye çalışılır.

XP uygulamaları bu amaç doğrultusunda kestirimlerin oluşturulması özel hedefine ait 4 özel uygulamayı eksiksiz olarak yerine getirebilmektedir.

Ö.H.2: Proje Planının Geliştirilmesi: CMMI bu özel hedefi 7 özel uygulama ile tanımlar. XP ise, bu özel hedefi yerine getirebilmek için bir XP uygulaması olan planlama oyunu pratiğinden yararlanır. Plan, yazılımın geliştirilmesi süresince kontrolden geçirilir. Metaphor ile yapılacak projenin genelinde oluşabilecek aksaklıkların ortadan kaldırılması sağlanır. Ayrıca zamanlama kutuları (time box) kullanımı ile işlerin ne kadar sürede gerçekleşeceğini daha önceden hesaplanması sağlanır.

Bununla beraber, proje planının geliştirilmesi özel hedefine ait bir özel uygulama olan proje kaynaklarının planlanması, şirket yönetimi ile ilgili konuları ve insan kaynakları planlamasını da içerdiğinden XP tarafından tam anlamı ile karşılanmamaktadır.

Ö.H.3: Plan İçin Taahhütlerin Oluşturulması: CMMI bu özel hedefi 3 özel uygulama ile tanımlar. XP, bu özel hedefi yerine getirebilmek için proje kapsamının belirlenmesinde müşteri kartlarından yararlanır. Müşteri kartları sayesinde proje çerçevesini belirlemeye çalışılır. XP’deki çalışma ortamı proje çerçevesinin belirlenmesinde yardımcı olur. İş ve kaynak yeterlilik seviyelerinin belirlenmesi, teknik kartlarda istenilenlerin yapıp yapılmadığı ile doğru orantılıdır.

Ayrıca, plan için taahhütlerin oluşturulması özel hedefine ait bir özel uygulama olan proje taahhütlerinin oluşturulması, müşteri ve yazılımcı arasında planla ilgili kapsamlı bir şartnameyi ön gördüğünden kısmi olarak karşılanmaktadır.

4.3. Proje İzleme ve Takip

Proje izleme ve takip süreç alanı içinde iki özel hedef vardır. Bu özel hedefler; projeyi plana göre ayarlama, düzeltici aksiyonların tamamlanıncaya kadar yönetilmesidir. Proje izleme ve takip süreç alanı, proje durumunun tespiti ile ilgilidir [8][18].

Ö.H.1 Projeyi Plana Göre Ayarlama: XP proje takibi için müşterinin yazılım ekibi içerisinde olması prensibi ve günlük ayaküstü yapılan toplantıları temel alarak projenin gelişimini izler. Teknik kartlarda yazılan işlerin gerçekleştirilip gerçekleştirilmediğine bakılır. Başta oluşturulan planlama oyununa göre kaynaklar belirlenir. Müşteri ve yazılım geliştirme ekibi aynı odada ve tüm kaynaklar tek bir yerde olduğundan proje katılımları rahatlıkla takip edilebilir.

Bununla beraber, proje planına göre ayarlama özel hedefine ait özel uygulamalar olan veri yönetiminin takibi ve taahhütlerin takibi XP tarafından CMMI’ in istediği ölçüde ayrıntılı olarak yapılamamaktadır.

Ö.H.2: Düzeltici Aksiyonların Tamamlanıncaya kadar Yönetilmesi: XP, projenin tamamını görebilmek için büyük bir görsel grafikten yararlanır. Bu grafik üzerinde yapılanlar ve yapılmayanlar işaretlenerek projenin hızı, istenilenlerin ne derecede karşılandığı gibi bazı bilgiler ile proje izleme ve takibi hakkında tespitler yapılabilir. Müşteriden gelen bilgiler doğrultusunda oluşturulan planlama oyunu, bu işi gerçekleştirmek için çok önemlidir. Planlama oyunu sayesinde eksiklikler ve yanlışlıklar düzeltilmeye çalışılır. Bu görev ile ekibin koçu ve yazılım ekibi bizzat ilgilendirilir.

XP uygulamaları bu amaç doğrultusunda düzeltici aksiyonların tamamlanıncaya kadar yönetilmesine ait 3 özel uygulamayı eksiksiz olarak yerine getirebilmektedir.

4.4. Tedarikçi Sözleşme Yönetimi

Tedarikçi sözleşme yönetiminin amacı; tedarikçiden ürün satın alma işlemini, taraflar arasında imzalanmış bir sözleşme ile yönetmektir [8][18]. Bu süreç alanının temel faaliyetleri arasında; satın alma işlerinin belirlenmesi, sözleşmenin oluşturulması, satın alınan sözleşmede istenilenlerin yerine getirilmesi ve kabul süresi vardır. XP satın alma, tedarikçi sözleşmesi oluşturma gibi özel hedeflerle ilgilenmez. Tedarikçi sözleşme yönetimi, XP çalışma alanının dışında bir konudur.

4.5. Ürün ve Süreç Kalite Güvencesi

Ürün ve süreç kalite güvencesi süreç alanının iki özel hedefi vardır. Bunlar süreçlerin ve iş ürünlerinin tarafsız olarak değerlendirilmesi ve tarafsız olarak veri toplanmasıdır. Ürün ve süreç kalite güvencesi süreç alanının amacı; çalışanlara ve yöneticilere, süreçler ve iş ürünleri ile ilgili tarafsız gözlemler sunmaktır. Uygunsuzlukları tespit etmek, belgelemek, proje ekibini bilgilendirmek gibi faaliyetleri vardır [8][18].

Ö.H.1: Süreçlerin ve İş Ürünlerinin Tarafsız Olarak Değerlendirilmesi: XP içinde ürünlerin ve süreçlerin kalite güvencesi ve istenilenlerin karşılanması ile ilgilenen ayrı bir grup yoktur. XP bu özel hedefi çiftli programlama tekniği sayesinde karşılamaya çalışır. Doğrudan olmasa da dolaylı olarak bir kalite güvence aktivitesi XP içerisinde vardır. Ayrıca yazılım geliştirilmesi ile ilgilenen ekip üyeleri sık sık test ve entegrasyon yaparak bu aktiviteyi yerine getirmeye çalışır.

XP uygulamaları bu özel hedef için ürün ve süreç kalite güvencesini büyük ölçüde yerine getirebilmektedir.

Ö.H.2: Objektif Anlayışın Sağlanması: XP' de müşteriye sürekli ve düzenli bir ürün teslimi vardır. Ayrıca başta oluşturulan bir görsel grafik sayesinde süreçlerin bir arada düzenli bir yapıda olmasını sağlar.

Objektif anlayışın sağlanması özel hedefine ait olan uygunsuzluklarla ilgili konuların çözümü ve kayıtların oluşturulması özel uygulamaları XP tarafından CMMI' in beklediği ölçüde karşılamamaktadır.

4.6. Konfigürasyon Yönetimi

Konfigürasyon yönetimi süreç alanının 3 özel hedefi vardır. Bunlar temel sürümlerin oluşturulması, değişikliklerin takip ve kontrol edilmesi ve bütünlüğün sağlanmasıdır. Konfigürasyon yönetiminin amacı; konfigürasyon tanımlama, konfigürasyon kontrol, konfigürasyon durum değerlendirme ve konfigürasyon denetleme faaliyetlerini kullanarak iş ürünlerinin tutarlılığını ve bütünlüğünü oluşturmaktır [18][19].

Ö.H.1: Temel Sürümlerin Oluşturulması: XP küçük sürümlerin oluşturulması, sürekli entegrasyon ve kodun ortak sahiplenmesi gibi uygulamalar ile bu işleri gerçekleştirmeye çalışır.

XP, temel sürümlerin oluşturulması özel hedefine ait olan 3 özel uygulamayı büyük ölçüde yerine getirebilmektedir.

Ö.H.2: Değişikliklerin Takip ve Kontrol Edilmesi: Bu özel hedef XP' de sistem içerisindeki mevcut bütün parçaların kontrol edilmesi ile sağlanır. XP içinde bu özel hedef için gelecekte ne gibi isterler gelebilir sorusu üzerinde düşünülür ve buna göre sistem geliştirilmeye çalışılır. Sürekli entegrasyon yapılır.

XP, değişikliklerin takip ve kontrol edilmesi özel hedefini eksiksiz olarak yerine getirebilmektedir.

Ö.H.3: Bütünlüğün Sağlanması: XP' de yazılım bütünlüğünün sağlanması ve denetimi yazılım geliştirme ekibinin koçu tarafından yapılır. Ek olarak çiftli programcı desteği sayesinde, konfigürasyon denetimlerinin oluşturulması sağlanır.

Bununla beraber bütünlüğün sağlanması özel hedefine ait bir özel uygulama olan konfigürasyon yönetim kayıtlarının oluşturulması XP içinde CMMI' in beklediği ölçüde kapsamlı değildir.

4.7. Ölçme ve Değerlendirme

Ölçme ve analiz süreç alanının iki özel hedefi vardır. Bunlar ölçme ve analiz çalışmaları için uygun altyapının oluşturulması ve ölçüm sonuçlarının ilan edilmesidir. Ölçme ve değerlendirme süreç alanının amacı; yönetimin bilgi ihtiyacına destek olmak, uygun ölçme yeteneklerini geliştirmek ve devamlılığı sağlamaktır.

Ö.H.1: Ölçme ve Analiz Çalışmaları İçin Uygun Altyapının Oluşturulması: XP bu hedefi günlük yapılan toplantılar, kısa sürede çıkartılan sürümler ve mevcut kartların sayısının başlangıçtaki kart sayısına oranı gibi bazı temel ölçüm ve analiz sonuçları ile gerçekleştirmeye çalışır.

XP, ölçme ve analiz çalışmaları için uygun altyapının oluşturulması özel hedefini kısmen karşılamaktadır.

Ö.H.2: Ölçme Sonuçlarının İlan Edilmesi: XP'de proje ekibinin içinde olan ölçüm sorumlusu, elde ettiği değerler ile önceden tespit edilen değerlerin karşılaştırmasını yapar. Ölçüm sorumlusu, gidişatın tespit edilmesi ve ölçümlerin doğruluğundan sorumludur. Ekip dönüşümlü çalışır. Yazılımla ilgili her bilgi panolarda asılarak paylaşılır. Analiz sonuçları yönetici ve ekip koçları tarafından yorumlanır. Sonuçlar yönetici tarafından her tekrarda ilan edilir.

XP, ölçme sonuçlarının ilan edilmesi özel hedefini tam karşılamaktadır.

5. Sonuçlar

Bu çalışmada CMMI ile XP'nin uyumu araştırılmıştır. Daha belirgin olarak ifade edilirse, XP kullanan bir organizasyonun CMMI 2 düzeyini ne derece karşılayabileceği sorusu irdelenmiştir. Sonuç olarak XP uygulamalarını tam olarak uygulamaktan vazgeçmeden, ancak bazı ek uygulamalarla CMMI 2. düzeyin sağlanabileceği görülmüştür. Bu ek uygulamalar;

- Gereksinimler yönetimi için müşteri ve yazılımcı arasında esnek fakat kapsamlı bir anlaşma (veya şartname) yapmak.
- Proje kaynaklarının planlanmasında, şirket yönetimi ile ilgili konuları ve insan kaynakları planlamasını ilgilendiren bir yapı oluşturmak.
- Projenin plana göre takip edilmesi için ayrıntılı veri yönetiminin takibi ve taahhütleri takip etmek.
- Ölçme ve analiz çalışmalarında uygun altyapının oluşturulması için daha ayrıntılı veri analizi ve ölçüm kriterleri hazırlamak.

6. Kaynaklar

- [1] Çevrimiçi: <http://www.sei.cmu.edu/cmm/>, Eylül, 2008.
- [2] Çevrimiçi: <http://www.sqi.gu.edu.au/spice/what.html>, Eylül, 2008.
- [3] Sarıdoğan, E., “Yazılım Mühendisliği”, Papatya Yayıncılık, S. 110-112, 2004.
- [4] Rising, L., Janoff, N. S., “The Scrum Software Development Process for Small Teams”, IEEE Software, July/August 2000
- [5] “Rational Unified Process - Best Practices for Software Development Teams”, IBM, 2001
- [6] Çevrimiçi: http://en.wikipedia.org/wiki/Agile_Manifesto, Eylül, 2008.
- [7] Borandağ, E., “Basamaklı CMMI Modeli ile Extreme Programming Metodunun Değerlendirilmesi”, Maltepe Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 2006.
- [8] Paulk, M., C., “Extreme Programming from a CMM Perspective”, IEEE Software, November 2001.
- [9] Carnegie Mellon University, “History of CMMI”, Software Engineering Institute, p. 17, 2002.
- [10] Hjenling, J., “CMMI Integration”, Course Papers p. 5, 2004.
- [11] Chrissis, M. B., Konrad, M., Shrum, S., “CMMI: Guidelines for Process Integration and Product Improvement”, Addison Wesley, 2005.
- [12] Beck, K., “Extreme Programming”, Addison Wesley, 2002.
- [13] Brewer, J., “Extreme Programming FAQ”, çevrimiçi: www.java.com/FAQ.htm, 2004.
- [14] Çevrimiçi: www.agilemanifesto.com/12practice.html, Mayıs, 2006.
- [15] Çevrimiçi: www.xporganization.com/xp.html, Nisan, 2006.
- [16] Hurtado, J., A., “Agile Methods”, 2003.
- [17] İnce, F., “Yazılım Kalitesi ve Modelleri”, Ders Notları, Maltepe Üniversitesi, Fen Bilimleri Enstitüsü, 2004.
- [18] Kalaycı, O., “Uygulamalı CMMI/XP Eğitimi”, Nitelik Danışmanlık, 2006.
- [19] Layman, L., Williams, L., Daiman, D., Bures, H., “Essential communication practices for Extreme Programming in a global software development team”, Journal of Information and Software Technology, Elsevier, 2006