aurum

AURUM MÜHENDİSLİK SİSTEMLERİ VE MİMARLIK DERGİSİ
AURUM JOURNAL OF ENGINEERING SYSTEMS AND ARCHITECTURE

Cilt 3, Sayı 2 | Kış 2019
Volume 3, No 2 | Winter 2019, 143-152

ARAŞTIRMA MAKALESİ / RESEARCH ARTICLE

YAZILIM PROJELERİNDE KALİTENİN ARTTIRILMASI: TMMi

Fatih YÜCALAR[1]
Manisa Celal Bayar Üniversitesi, Yazılım Mühendisliği Bölümü
fatih.yucalar@cbu.edu.tr ORCID: 0000-0002-1006-2227

Emin BORANDAĞ[2]
Manisa Celal Bayar Üniversitesi, Yazılım Mühendisliği Bölümü
emin.borandag@cbu.edu.tr ORCID: 0000-0001-5553-2707

GELİŞ TARİHİ/RECEIVED DATE: 24.01.2019 KABUL TARİHİ/ACCEPTED DATE: 07.02.2019

Özet

Yazılım testi, yazılım geliştirme sürecinin en önemli safhalarından biridir. Özellikle yazılım geliştirme süreci sonunda ortaya çıkan ürünün kalitesinin belirlenmesi yazılım testi ile sağlanır. Bu önemine rağmen, yazılım sektöründe test safhası planlama ve kaynak eksikliğinden dolayı çoğu kez göz ardı edilmektedir. Son yıllarda, yazılımda kalite sertifikasyonunu sağlamak, yazılım geliştirme süreçlerini iyileştirmek ve yetenek belirlemek amaçlarıyla ilgili olarak CMM, CMMI, ISO 15504 gibi çeşitli modeller ortaya çıkmıştır. Sözü edilen bu modeller içerisinde yazılım test süreçleri ile ilgili olarak doğrulama, sağlama gibi süreç alanları mevcuttur. Ancak var olan bu süreç alanları, yazılım süreçlerindeki test işlemlerinin detaylarına yönelik yeterli bilgiyi içermemektedir. Yazılım kalite modellerinin test süreçleri ile ilgili eksikliğini tamamlayıcı olarak TMMi modeli ortaya çıkmıştır. Bu çalışmada, kaliteli yazılım geliştirme noktasında test süreçlerini iyileştirmek isteyen firmaların TMMi hakkında bilgilendirilmesi amaçlanmıştır. Ayrıca, yazılım firmalarının hali hazırda uyguladıkları test süreçlerinin olgunluk düzeylerini arttırma noktasında nelere dikkat etmeleri gerektiği konusu da çalışma içerisinde ele alınmıştır.

Anahtar Kelimeler: Yazılım testi, yazılım kalitesi, yazılım proje yönetimi, CMMI, TMMi

IMPROVING QUALITY IN SOFTWARE PROJECTS: TMMi

Abstract

Software testing is one of the most important steps in the software development process. In particular, the determination of the quality of the product at the end of the software development process is provided by software testing. Despite its importance, the testing process in the software industry is often ignored due to lack of planning and resources. In recent years, various models such as CMM, CMMI, ISO 15504 have emerged for the purpose of providing quality certification in software, improving software development processes and determining capability. There are process areas such as verification and validation related to software testing processes in these models. However, these existing process areas do not contain sufficient information about the details of the testing processes in the software process. The TMMi model has emerged to complement the lack of software quality models related to testing processes. In this study, it is aimed to inform the companies who want to improve their test processes in quality software development about TMMi. In addition, it is also discussed in the study what software companies should pay attention to improve the maturity levels of the test processes they have already applied.

Keywords*:* Software testing, software quality, software project management, CMMI, TMMi.

1. GİRİŞ

Son yıllarda, yazılım sektöründe geliştirilen yazılım ürünlerinin başarılı bir şekilde tamamlanması ve kalitesinin arttırılması için büyük çaba harcanmaktadır. Müşteri ve kullanıcı taleplerinin artmasıyla doğru orantılı olarak yazılım ürünlerinin büyüklüğü ve karmaşıklığı da artmaya başlamıştır. Bu durum yazılım ürünlerinin istenilen kalitede olmasını zorlaştırmaktadır. CMMI (Tümleşik Yetenek Olgunluk Modeli – Capability Maturity Model Integration) (Chrissis, 2011), ISO 15504 (Gökalp, 2015) gibi çeşitli kalite iyileştirme yaklaşımlarının kullanımı ile elde edilen başarılara rağmen, yazılım sektöründe sıfır hata ile yazılım ürünlerinin geliştirilmesi hala mümkün değildir (van Veenendaal, 2018). Yazılım sektöründe geliştirilen yazılım ürünlerinin istenilen kalitede olmasını sağlamak için, genellikle geliştirme süreçlerinin iyileştirilmesine odaklanılmaktadır. Yazılım geliştirme süreçlerinin iyileştirilmesinde yaygın olarak CMMI kullanılmaktadır. Yazılım sektöründe standart olarak kabul edilen CMMI, yazılım geliştirme süreçlerinin iyileştirilmesi için bir rehberdir (Yucalar, 2006). Yazılım geliştirme süreçlerinin etkin bir biçimde tanımlandığı CMMI standardının kullanımı ile yazılım projelerindeki değişkenlikler azaltılmaktadır. Yazılım kalitesinin arttırılması noktasında test süreci önemlidir. CMMI içerisinde test süreci ile ilgili olarak doğrulama (verification) ve sağlama (validation) süreç alanları yer almaktadır. Ancak CMMI içerisinde yer alan bu süreç alanları, yazılım süreçlerindeki test işlemlerinin detaylarına yönelik yeterli bilgiyi içermemektedir. Test süreçleri toplam proje maliyetlerinin en az %30-%40'ını oluşturmasına rağmen, CMMI gibi bir yazılım süreç iyileştirme modelinde bile test sürecine sınırlı düzeyde yer verilmediği görülmektedir (Yucalar, 2006). Bu nedenle CMMI'yı tamamlayıcı bir model olarak TMMi (Tümleşik Test Olgunluk Modeli – Test Maturity Model Integration) konumlandırılmıştır. TMMi, test süreçlerinin iyileştirilmesi amacıyla Chicago Illinois Institute of Technology tarafından geliştirilmiştir (Başar, 2015). TMMi, yazılım sektöründe yer alan firmaların yazılım test süreçlerine ilişkin olgunluk seviyelerinin belirlenmesi ve hedeflenen olgunluk seviyesine ulaşılabilmesi için kullanılan bir referans modeldir (Camargo, 2013).

Bu çalışmada, kaliteli yazılım geliştirme noktasında test süreçlerini iyileştirmek isteyen firmaların TMMi hakkında bilgilendirilmesi ve mevcut test süreçlerinin olgunluk düzeylerinin arttırılmasına yönelik nelere dikkat etmeleri gerektiği konusu ele alınmıştır. Çalışmanın ikinci bölümünde literatürde TMMi ile ilgili olarak yapılan çalışmalara yer verilmiştir. Üçüncü bölümde TTMi'ın yapısından ve olgunluk düzeylerinden bahsedilmiştir. Dördüncü bölümde, CMMI ile TMMi modelleri farklı bakış açılarından ele alınarak karşılaştırılmıştır. Çalışmanın son bölümünde ise sonuç ve önerilere yer verilmiştir.

2. LİTERATÜR ÖZETİ

Literatür taraması yapıldığında TMMi modeli ile ilgili olarak yapılan çalışmaların sınırlı olduğu gözlemlenmiştir. Yapılan çalışmaların ağırlıklı olarak yazılım test süreçlerinin iyileştirilmesinde TMMi modelinin kullanılabileceği yönündedir.

Farid ve ark. (Farid, 2015) yaptıkları çalışmada, TMMi ile Scrum uygulamaları arasında detaylı bir eşleştirme uygulayarak yazılım test sürecinin iyileştirilebileceğini savunmuşlardır. Deneysel çalışmalar sonucu elde ettikleri sonuçlar ile de yapmış oldukları bu eşleştirmeyi doğrulamışlardır.

Bris ve ark. (Bris, 2015) yaptıkları çalışmada, yazılım kalite güvence ve kontrolü ile ilgili olarak yazılım test süreçlerine odaklanmışlardır. Çalışma içerisinde yazılım geliştirme metodolojisi ile ilgili olan yazılım

test sürecini ve TMMi kullanımı ile test süreçlerinin iyileştirilebilmesine yönelik tanımlama ve önerilerde bulunmuşlardır. Ayrıca, TMMi modeline göre ikinci seviye yazılım test süreci olgunluğunu elde etmek amacıyla mevcut yazılım test sürecinin durumu üzerinde bu önerileri uygulamaya çalışmışlardır.

Kim ve ark. (Kim, 2014) ise yaptıkları çalışmada, yazılım mühendisliğinde hibrit analiz yöntemini temel alan TMMi ile değerlendirilen test organizasyonları için test sürecinin nasıl geliştirileceğini önermişlerdir.

Bose ve ark. (Bose, 2016) yaptıkları çalışmada, TMMi sertifikasını elde etmek için organizasyonların dönüşümünü ele almışlardır. Mevcut test uygulamalarının ve işlemlerinin istenen TMMi olgunluk seviyesine karşı ilk değerlendirilmesinin nasıl yapılacağını bir öneri yol haritası oluşturarak sunmuşlardır. Çalışma içerisinde TMMi'ye ulaşmak için taktiksel ve stratejik fırsatları göz önünde bulundurarak, mevcut ve hedef son durum arasındaki boşlukları ele almak için gereken örgütsel değişim yönetimine dikkat çekmişlerdir.

Araújo ve ark. (Araújo, 2013) yaptıkları çalışmada, TMMi modelini temel alan küçük ve orta ölçekli işletmelere yönelik yazılım olgunluk değerlendirmesi için bir çerçeve model önermişlerdir. Bunun için çerçeve model kapsamında TMMi alt uygulamalarını temel alan bir değerlendirme anketi hazırlamışlardır. Hazırlamış oldukları değerlendirme anketinin eksiksiz bir şekilde doldurulmasını sağlamak için gereken örneklerin yer aldığı otomatik bir araç ile de bunu desteklemişlerdir. Önermiş oldukları çerçeve modeli, küçük ve orta ölçekli dört işletme üzerinde test etmişlerdir.

Afzal ve ark. (Afzal, 2016) ise yaptıkları çalışmada, sistematik bir literatür taraması kullanarak mevcut yazılım test süreci iyileştirme yaklaşımlarını ve özelliklerini tespit etmeye çalışmışlardır. TPI (Test Process Improvement) NEXT ve TMMi olmak üzere seçmiş oldukları iki yaklaşımı, yazılım endüstrisindeki içerik ve analiz sonuçlarına göre değerlendirmişlerdir. Bu çalışma sonucunda 18 yazılım test süreci iyileştirme yaklaşımını ve özelliklerini belirlemişlerdir. TPI NEXT ve TMMi içeriğinin detaylı bir karşılaştırmasını yapmışlar ve yazılım test süreci iyileştirme yaklaşımlarının çoğunun yeterli bilgi sağlamadığını tespit etmişlerdir.

3. TMMi

Yazılım testi (software testing), bir yazılım ürünü içerisindeki hataları bulmak amacı ile yapılan işlemlerdir. Yazılım testi temel olarak elde edilen yani geliştirilen yazılım ürününün kalitesinin istenilen düzeyde olduğunu belirlemek, değilse de istenilen kaliteye ulaştırılmasını sağlama noktasında kullanılan bir süreçtir. Bu süreçte gereksinim dokümanında tanımlanmış işlevsel ve işlevsel-olmayan gereksinimlerin karşılanıp karşılanmadığının ve yazılımın istenildiği gibi çalışıp çalışmadığının kontrolü gerçekleştirilir. Test süreci için yazılım-geliştirme yaşam döngüsünün en önemli adımlarından biri olduğu söylenebilir. Toplam proje bütçesinin önemli bir kısmını oluşturmaktadır. Yazılım ürünlerindeki hatalardan dolayı işletmeler çok yüksek maliyetlere katlanmak zorunda kalabilmektedirler. Bu nedenle günümüzde yazılım test süreçlerinin iyileştirilmesi ve geliştirilmesi üzerine yapılan çalışmalar gittikçe önem kazanmaya başlamıştır.

TMMi, test süreçlerinin iyileştirilmesi amacıyla CMMI modeli referans alınarak Chicago Illinois Institute of Technology bünyesinde yer alan TMMi kurumu tarafından geliştirilmiştir. TMMi modelinin amacı; test süreçlerinin olgunluğunu belirlemek üzere bir çerçeve oluşturmaktır. TMMi, yazılım sektöründe yaygın

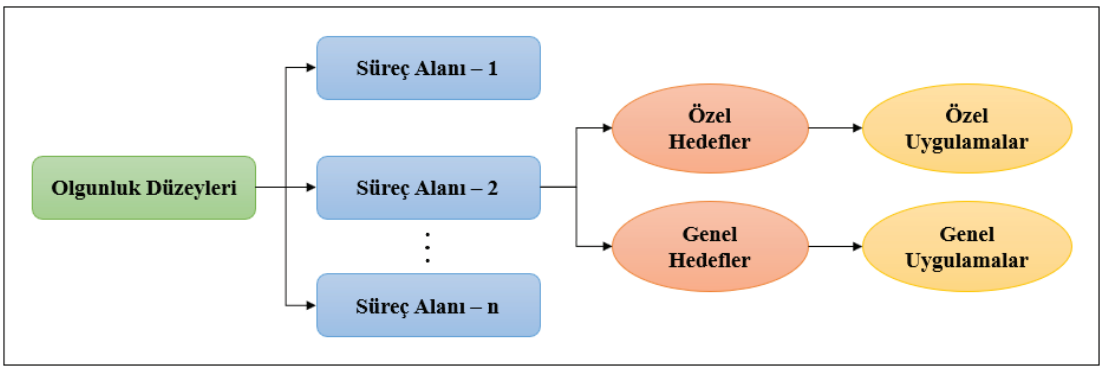
olarak desteklenmiş bir süreç iyileştirme modeli olan CMMI üzerinde yapılan çalışmalarla ortaya çıkmış bir modeldir.

CMMI, yazılım geliştirmede kullanılan kalite sistemlerini ve süreçlerini değerlendirmek, yazılımda kalite sertifikasyonunu sağlamak, süreçleri iyileştirmek ve yetenek belirlemek amacıyla geliştirilen bir modeldir (Chrissis, 2011). CMMI tek model olmakla birlikte, basamaklı ve sürekli olmak üzere iki gösterim şekli kullanmaktadır (Kalaycı, 2007). Sürekli gösterim (continuous) yetenek düzeylerini tanımlarken, basamaklı gösterim (staged representation) olgunluk düzeylerini tanımlar (Yucalar, 2006). CMMI sürekli gösterim, bir organizasyonun her bir süreç alanında süreç iyileştirmede gösterdiği başarı için uygulanır. CMMI basamaklı gösterim ise, organizasyonun ilgili basamaktaki tüm süreçlerinin olgunluğunun değerlendirilmesi için uygulanır (Yucalar, 2006). Her düzey için tanımlı süreç alanları vardır.

TMMi basamaklı bir model olarak geliştirilmiştir. Basamaklı model, bir yazılım organizasyonuna uygun bir gelişme yolunu tanımlamak üzere önceden tanımlanmış süreç alanları kümesini kullanır. Bu gelişme yolu, olgunluk düzeyi adı verilen bir model bileşeniyle tanımlanmaktadır. Olgunluk düzeyi, gelişmiş organizasyonel süreçlere ulaşmak için iyi tanımlanmış evrimsel bir seviyedir (van Veenendaal, 2018). TMMi'da test süreçlerinin olgunluğunu gösteren düzeyler vardır. Birinci düzey dışında, her bir düzey için olgunluk hedefleri tanımlanmıştır. Bu tanımlama ile yazılım organizasyonunun test sürecine ilişkin hangi aşamada olduğu bilgisi elde edilmektedir. CMMI'da olduğu gibi TMMi'ın her olgunluk düzeyinde, hedefler, alt hedefler, anahtar süreç adımları, aktiviteler, görev ve sorumluluklara ilişkin gerekli bilgiler tanımlanmaktadır. TMMi modeli ile yazılım test süreçlerini iyileştirmek isteyen yazılım organizasyonları, aynı CMMI'da olduğu gibi birinci olgunluk düzeyinden başlamak kaydıyla ilgili anahtar süreç adımlarını tamamlayarak daha üst olgunluk düzeyine ulaşabilmektedir (Yıldız, 2018).

3.1. TMMi'nin Yapısı

TMMi, CMMI'a benzer bir yapıya sahiptir. CMMI'da olduğu gibi TMMi da hedefler ve bu hedeflere ilişkin alt uygulamaların başarılmasıyla ulaşılan olgunluk düzeylerini içerir. TMMi için bir süreç, karmakarışık bir başlangıç durumundan, sürecin yönetildiği, kontrol edildiği ve optimize edildiği bir duruma doğru gelişir. TMMi modelinin yapısı Şekil 1'de görülmektedir.

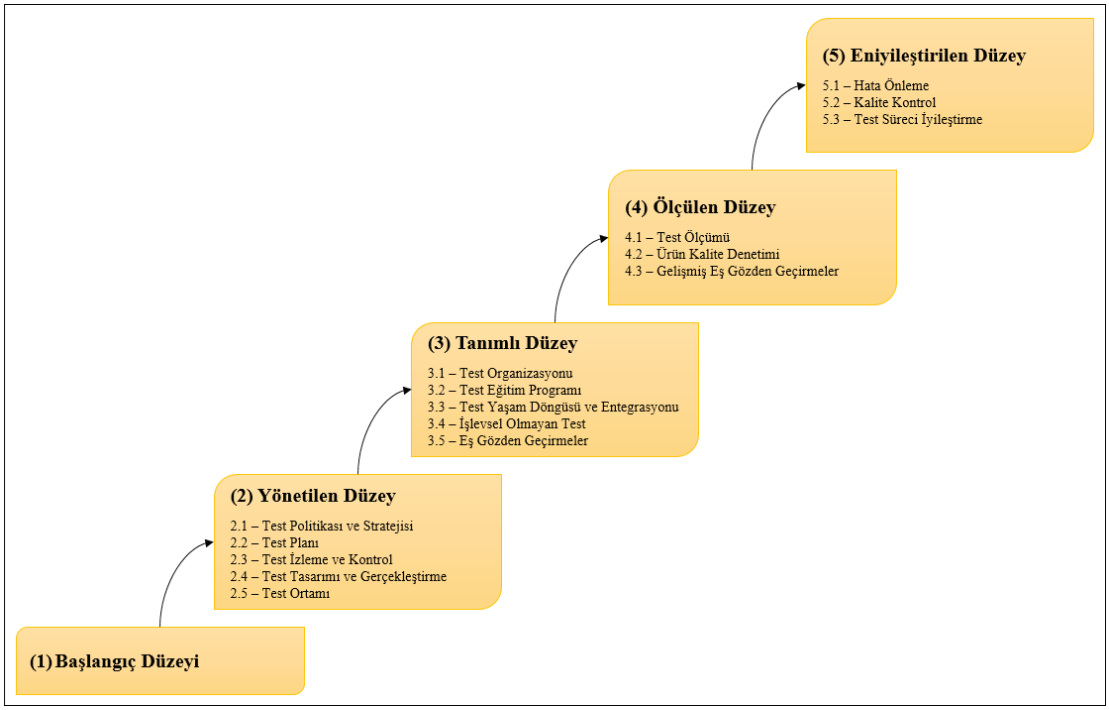


Şekil 1. TMMi Modelinin Yapısı

TMMi'da her bir süreç alanının yerine getirmesi gereken özel ve genel hedefler vardır. Her bir özel hedef, ilgili süreç alanını yerine getirmek için bulunması gereken tek bir özelliği gösterir. Bir özel hedef, hedefe ulaşmak için hangi faaliyetlerin önemli olduğunu ve gerçekleştirilebileceğini tanımlayan özel uygulamalara bölünmüştür. Genel hedefler birden fazla süreç alanıyla ilgilidir. Genel hedefler, test sürecini kurumsallaştırmak için kullanılabilecek özellikleri tanımlar.

3.2. TMMi Olgunluk Düzeyleri

TMMi, test süreçlerinin iyileştirilmesi için basamaklı bir mimariye sahiptir. TMMi, bir yazılım organizasyonunun test süreçlerini iyileştirmesine yönelik başlangıç (initial), yönetilen (managed), tanımlı (defined), ölçülen (measured) ve en iyileştirilen (optimization) olmak üzere beş olgunluk düzeyini içerir. Her olgunluk düzeyi, bir yazılım organizasyonunun o düzeyde olgunluğa ulaşabilmesi için uygulaması gereken bir dizi süreç alanına sahiptir. Şekil 2'de TMMi olgunluk düzeyleri ve bu olgunluk düzeyleri içinde yer alan süreç alanları görülmektedir.



Şekil 2. TMMi Olgunluk Düzeyleri ve Süreç Alanları (van Veenendaal, 2018)

Bir düzeyi başarma, bir sonraki düzey için temel olarak yeterli bir iyileştirmenin yapılmasını sağlar. Yazılım test süreçlerini iyileştirmeyi hedefleyen yazılım organizasyonları TMMi ile birinci düzeyden başlamak kaydıyla ilgili her düzeye ait anahtar süreç adımlarını tamamlayarak bir üst olgunluk düzeyine ulaşabilmektedir (Başar, 2015). TMMi'nin iç yapısı, sistematik bir şekilde öğrenilebilen ve uygulanabilen test uygulamaları bakımından zengindir.

3.2.1. Başlangıç Düzeyi

TMMi'nin başlangıç düzeyinde olan bir yazılım firmasında, test süreci tanımsız ve karmakarışıktır. Bu düzeyde olan firmalarda, test süreci genellikle hata ayıklamanın bir parçası olarak gerçekleştirilir. Yazılım firması içerisinde test süreçlerini desteklemek üzere istikrarlı bir ortam yoktur. Bu düzeydeki yazılım firmalarında başarı kanıtlanmış süreçlerin kullanımına değil, firma bünyesinde çalışan kişilerin yetkinliğine ve deneyimine bağlıdır. Testler, kodlama süreci tamamlandıktan sonra plansız bir şekilde gerçekleştirilir. Test ve hata ayıklama süreci, sistem üzerindeki hataları gidermek üzere yapılır.

Bu düzeydeki yazılım firmalarında yazılım ürünleri geç geliştirilir. Bunun yanında proje bütçesi aşılır ve yazılım ürünü istenilen kalitede ortaya çıkmaz. Bu düzeyde testin amacı, büyük bir arıza olmaksızın yazılımın çalıştığını göstermektir. TMMi'nin başlangıç düzeyi için tanımlanmış süreç alanları yoktur (van Veenendaal, 2018).

3.2.2. Yönetilen Düzey

TMMi 2. düzeyde, test süreci yönetilen bir süreçtir ve hata ayıklamadan açıkça ayrılmıştır. Yazılım firması içerisindeki test süreçlerini geliştirmek adına, test stratejileri belirlenmiştir ve test planları yapılmıştır. Test planı yapılırken, yazılım ürünü risk değerlendirme sonucuna göre test yaklaşımı belirlenir. Test planında, hangi testlerin gerekli olduğu, bu testlerin ne zaman, nasıl ve kim tarafından yapılacağı tanımlanır. Testlerin plana göre yapıldığından emin olmak için testler izlenir ve kontrol edilir. Test planında sapmalar meydana gelirse birtakım eylemler gerçekleştirilir.

Bu düzeyde yazılım ürünlerinin ve test süreçlerinin durumu üst yönetime bilgi olarak sunulmaktadır. Belirtim dokümanından test senaryolarının türetilmesi ve seçilmesi için test tasarım teknikleri uygulanır. TMMi 2. düzeyde bileşen, entegrasyon, sistem ve kabul testi olmak üzere test süreci çok seviyelidir.

Tanımlanan her test düzeyi için, kurum çapında veya program genelinde test stratejisinde tanımlanan özel test hedefleri vardır (van Veenendaal, 2018). Bu özel test hedefleri;

– test ve hata ayıklama işlemlerine ait hedeflerin tespit edilmesi,
– test planlama sürecinin önceliklendirilmesi,
– temel test teknik ve metotlarının kurumsallaştırılmasıdır.

TMMi'nin yönetilen düzey için tanımlanmış süreç alanları;

– test politikası ve stratejisi,
– test planı,
– test izleme ve kontrol,
– test tasarımı ve gerçekleştirme,
– test ortamıdır (van Veenendaal, 2018).

3.2.3. Tanımlı Düzey

TMMi 3. düzeyde, test süreci artık kodlamayı takip eden bir aşama ile sınırlı değildir. Test süreci, yazılım geliştirme yaşam-döngüsü ve ilgili tüm dönüm noktalarına tamamen entegre olmuş durumdadır. Test planı, yazılım geliştirme yaşam-döngüsünün en başından itibaren yapılır ve asıl test planında dokümante edilir. Asıl test planının geliştirilmesi, TMMi 2. düzeyde elde edilen test planlama becerileri ve taahhütlerine dayanmaktadır.

TMMi 2. düzeyde test tasarımları daha çok işlevsel testler ve test teknikleri üzerinedir. Ancak TMMi 3. düzeyde ise işlevsel testler ve test tekniklerine ilave olarak işlevsel-olmayan testler, güvenilirlik ve kullanılabilirlik gibi testler eklenir (van Veenendaal, 2018).

TMMi 3. düzey için de tanımlanan özel hedefler vardır. Bu özel hedefler;

– yazılım test grubunun oluşturulması,
– teknik eğitim programlarının oluşturulması,
– yazılım yaşam döngüsü içinde test işlemlerinin birleştirilmesi,
– test süreçlerinin kontrol edilmesi ve izlenmesidir.

TMMi'nin tanımlı düzey için tanımlanmış süreç alanları;

– test organizasyonu,
– test eğitim programı,
– test yaşam döngüsü ve entegrasyonu,
– işlevsel-olmayan test,
– eş gözden geçirmelerdir (van Veenendaal, 2018).

3.2.4. Ölçülen Düzey

TMMi 2. ve 3. düzey hedeflerinin başarılması, kapsamlı test yapabilecek ve test sürecinin iyileştirilmesi için destek sağlayabilecek teknik, yönetimsel ve personel altyapısının oluşturulmasını sağlar. Mevcut bu altyapı ile daha fazla büyüme ve başarı elde etmek için test ölçülebilen bir süreç haline gelir. TMMi 4. düzey olan yazılım firmalarında testler tamamen tanımlanmış, sağlam ve ölçülebilir bir süreç altyapısı üzerinde gerçekleştirilir.

Bu düzeyde test faaliyetleri yazılım geliştirme yaşam-döngüsünün bütün aşamalarında ele alınır. Yazılım firması tarafından kuruluş bazında bir test ölçüm programı uygulanarak test süreçlerinin kalitesi ölçülür. Yazılım geliştirme yaşam-döngüsü içerisinde ürünün kalitesini ölçmek üzere kullanılan gözden geçirmeler ve denetlemeler, test sürecinin parçası olarak düşünülmektedir. TMMi 4. düzey, statik testler ve dinamik testler arasında koordineli bir test yaklaşımı oluşturulmasını sağlar (van Veenendaal, 2018).

TMMi 4. düzey için de tanımlanan özel hedefler vardır. Bu özel hedefler;

– kurum çapında gözden geçirme programının oluşturulması,
– test ölçüm programının oluşturulması,
– yazılım kalite değerlendirmesidir.

TMMi'nin ölçülen düzey için tanımlanmış süreç alanları;

– test ölçümü,
– ürün kalite denetimi,
– gelişmiş eş gözden geçirmelerdir (van Veenendaal, 2018).

3.2.5. Eniyileştirilen Düzey

TMMi 1. düzey ile 4. düzey arasındaki bütün test iyileştirme hedeflerinin başarılması, test için tamamen tanımlanmış ve ölçülebilen bir süreci desteklemeye yönelik organizasyonel bir altyapının oluşturulmasını sağlar. TMMi olgunluk düzeyi 5 olan bir kuruluş, istatistiksel olarak kontrol edilen süreçlerin nicel anlaşılması nedeniyle süreçlerini sürekli olarak iyileştirme yeteneğine sahiptir. Test süreç performansının arttırılması, yenilikçi süreçler ve teknolojik gelişmeler ile sağlanır.

Bu düzeyde, test süreci altyapısının sürekli iyileştirilmesini desteklemek ve test iyileştirmelerini tanımlamak, planlamak ve uygulamak için özel eğitim almış üyelerden oluşan bir Test Süreç Grubu bulunmaktadır. Aslında Test Süreç Grubu, TMMi 3. düzeyde oluşturulmaktadır. Üst düzeylere çıktıkça test iyileştirme sürecini destekleyici faaliyetlere yönelik çalışmalarla bu grubun sorumlulukları artmaktadır (van Veenendaal, 2018).

TMMi 5. düzey için de tanımlanan özel hedefler vardır. Bu özel hedefler;

– hata önleme işlemlerine yönelik uygulamalar,
– istatiksel kalite kontrolü,
– test sürecinin eniyileştirilmesidir.

TMMi'nin eniyileştirilen düzey için tanımlanmış süreç alanları;

– hata önleme,
– kalite kontrol, ve
– test sürecini eniyileştirme'dir (van Veenendaal, 2018).

4. CMMI İLE TMMi'NİN KARŞILAŞTIRILMASI

TMMi, CMMI'yı tamamlayıcı olarak konumlandırılmış ve CMMI tarafından başlıklandırılan aynı yapıyı izlemesine rağmen, iki model arasında bazı önemli farklılıklar vardır. CMMI ile TMMi arasındaki farklılıklar Tablo 1'de listelenmiş halde görülmektedir.

Tablo 1. CMMI ile TMMi'nin Karşılaştırılması

CMMI	TMMi
• Carnegie Mellon Üniversitesi tarafından geliştirilmiştir.	• Chicago Illinois Institute of Technology tarafından geliştirilmiştir.
• Uçtan uca yazılım geliştirme yaşam-döngüsü uygulamaları üzerine odaklanmaktadır.	• Test sürçleri ve uygulamaları üzerine odaklanmaktadır.
• Test süreçleri ve test iyileştirme faaliyetleri üzerine sınırlı düzeyde odaklanmaktadır.	• Test dışı iyileştirmeler üzerine odaklanma sınırlı düzeydedir.
• Sürekli ve basamaklı olmak üzere iki gösterime sahiptir.	• Sadece basamaklı gösterime sahiptir.
• CMMI-DEV, CMMI-SVC, CMMI-ACQ olmak üzere üç çerçeve yapıya sahiptir.	• İlave bir TMMi çerçeve yapısı yoktur.

5. SONUÇ VE ÖNERİLER

Bu çalışmada, yazılım proje yönetimi içerisinde önemli bir safha olan yazılım testinin yönetimi ve test süreçlerinin ölçümü için TMMi modelinin kullanımı önerilmektedir. Modelin kullanımı ile yazılım kalitesini geliştirme noktasında test süreçlerini iyileştirmek isteyen firmaların TMMi hakkında bilgilendirilmesi amaçlanmıştır. Yazılım firmaları geliştirdikleri yazılım projelerindeki başarı yüzdelerini arttırabilmeleri için test faaliyetlerine yeterince zaman ve bütçe ayırmalıdırlar. Bunun yanında yazılım geliştirme grubu haricinde bir bağımsız bir test grubunun oluşturulması yazılım kalitesinin arttırılmasında pozitif katkı sağlayacaktır. Bu bağımsız test grubu içerinde yer alacak test mühendislerinin; yazılım geliştirme süreçleri hakkında deneyim sahibi olan, yazılım geliştirmeyi bilen, algoritmik düşünen, test planlamasını yapabilen, test senaryolarını yazabilen ve test araçlarını kullanabilen kişilerden oluşturulması gerekmektedir.

Yazılım geliştirme sürecinin kalitesi, test sürecinde elde edilen sonuçlara göre nicel olarak ölçülebilir. Test sürecinin iyileştirilmesi geliştirilen ürünün kalitesini doğrudan arttıracaktır. Test süreçlerinin iyileştirilmesi noktasında daha çok orta ölçekli yazılım firmaları için kullanılması tavsiye edilen TMMi modelinin önemi ortaya çıkmaktadır. TMMi, test süreçlerinin iyileştirilmesi için detaylandırılmış bir modeldir. Yazılım firmaları, test süreçlerini iyileştirmek ve etkinliğini arttırmak için en az TMMi 3. olgunluk düzeyini hedeflemelidirler. Yazılım firmaları kendi test süreçleri ile ilgili olarak hangi alanların iyileştirilmesi gerektiğine ve hedeflenen olgunluk düzeyine ulaşmak için nasıl harekete geçmesi gerektiğine iyi karar vermelidir. TMMi modelinin kullanımı, kurum bünyesindeki test süreçlerinin yazılım geliştirme süreçleri ile tamamen bir bütün haline gelmesini sağlayacaktır. Bu sayede, yazılım içerisindeki hatalar ortaya çıkmadan büyük oranda hata önleme faaliyetleri ile engellenmiş olacaktır.

Conflict of Interests/Çıkar Çatışması

Authors declare no conflict of interests/Yazarlar çıkar çatışması olmadığını belirtmişlerdir

KAYNAKLAR

Afzal, W., Alonec, S., Glocksienc, K., Torkar, R., (2016). Software test process improvement approaches: A systematic literature review and an industrial case study. The Journal of Systems and Software. Vol. 111. pp. 1-33.

Araújo, A. F., Rodrigues, C. L., Vincenzi, A. M. R., Camilo, C. G., Silva, A. F., (2013). A Framework for Maturity Assessment in Software Testing for Small and Medium-Sized Enterprises. *International Conference on Software Engineering Research and Practice (SERP'13)*.

Başar, A., (2015). Test Süreçlerinin Olgunluk Seviyesi Modeli ile İyileştirilmesi: Scrum ile Yazılım Geliştiren Bir İşletmede Uygulama. *XVII. Akademik Bilişim Konferansı (AB 2015)*. Anadolu Üniversitesi. Eskişehir.

Bose, S. C., Bose, G., (2016). Transforming organizations to achieve TMMi certification. *Thirty-Fourth Annual Pacific Northwest Software Quality Conference*. Portland. Oregon. USA.

Bris, P., Frantis, M., Kolkova, M., (2015). Software Quality Control with The Usage of Ideal and TMMi Models. *MM Science Journal*. pp. 799-807.

Camargo, K. G., Ferrari, F. C., Fabbri, S. C. P. F., (2013). Identifying a Subset of TMMi Practices to Establish a Streamlined Software Testing Process. *27th Brazilian Symposium on Software Engineering*. IEEE.

Chrissis, M. B., Konrad, M., Shrum, S., (2011). CMMI: Guidelines for Process Integration and Product Improvement. *Addison-Wesley Professional*; 3 Edition.

Farid, A. B., Fathy, E. M., Ellatif, M. A., (2015). Towards Agile Implementation of Test Maturity Model Integration (TMMI) Level 2 using Scrum Practices. *International Journal of Advanced Computer Science and Applications (IJACSA)*. 6(9).

Gökalp, E., Demirors, O., (2015). Proposing an ISO/IEC 15504 Based Process Improvement Method for the Government Domain, *Conference on Software Process Improvement and Capability Determination*. Gothenburg. Sweden. Vol. 526.

Kalaycı, O., (2007). "CMMI: Yöneticiler için Doğru Sorular". *Shamrock Process Improvement and Innovation*, ISBN: 978-0-9783530-0-1. Toronto. Kanada,

Kim, K., Kim, R. Y. C., (2014). Improving Test Process for Test Organization assessed with TMMi based on TPI NEXT. *International Journal of Software Engineering and Its Applications*. 8(2). pp.59-66.

van Veenendaal, E., (2018). Test Maturity Model integration (TMMi): Guidelines for Test Process Improvement, Release 1.2. TMMi Foundation. Ireland.

Yıldız, G., "Test Maturity Model integration (TMMi)", Çevrimiçi: https://gokyhome.com/2014/06/26/test-maturity-model-integration-tmmi/, Erişim Tarihi: (2018).

Yucalar, F., (2006). Evaluation of companies which are being in the software sector has an understanding process focused quality management with the CMMI staged model. *Master Thesis. Maltepe University. Institute of Science and Technology. Department of Computer Engineering*. Istanbul. Turkey.