

**T.C.
SAKARYA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ**

**YAZILIM MALİYETLERİNİN TÜRKİYE MUHASEBE
STANDARTLARINA GÖRE
MUHASEBELEŞTİRİLMESİNE İLİŞKİN MODEL
ÖNERİSİ**

DOKTORA TEZİ

İlker CALAYOĞLU

Enstitü Anabilim Dalı : İşletme

Enstitü Bilim Dalı : Muhasebe ve Finansman

Tez Danışmanı: Yrd. Doç. Dr. Recep YILMAZ

EYLÜL - 2016

T.C.
SAKARYA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ

YAZILIM MALİYETLERİNİN TÜRKİYE MUHASEBE
STANDARTLARINA GÖRE
MUHASEBELEŞTİRİLMESİNE İLİŞKİN MODEL
ÖNERİSİ

DOKTORA TEZİ

İlker CALAYOĞLU

Enstitü Anabilim Dalı : İşletme

Enstitü Bilim Dalı : Muhasebe ve Finansman

“Bu tez 23/09/2016 tarihinde aşağıdaki jüri tarafından Oybirliği / Öyçokluğu ile kabul edilmiştir.”

JÜRİ ÜYESİ	KANAATI	İMZA
Prof. Dr. Ahmet Ueadi Can	Kabul	
Prof. Dr. Vasfi Haftacı	Kabul	
Doç. Dr. Haluk Bengi	Kabul	
Yrd. Doç. Dr. Recep Yılmaz	Kabul	
Yrd. Doç. Dr. Kemal TAŞKIN	Kabul	

BEYAN

Bu tezin yazılmasında bilimsel ahlak kurallarına uyulduğunu, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, tezin herhangi bir kısmının bu üniversite veya başka bir üniversitedeki başka bir tez çalışması olarak sunulmadığını beyan ederim.



İlker CALAYOĞLU

23.09.2016

ÖNSÖZ

Tez çalışmam boyunca yoğun iş temposunda desteklerini ve önerilerini esirgemeyerek bana devamlı yol gösteren tez danışmanım Sayın Yrd. Doç. Dr. Recep YILMAZ'a, jüri dışında olmasına rağmen yaptığı katkılardan dolayı Sayın Prof. Dr. Selahattin KARABINAR'a, tezin geliştirilmesindeki desteklerinden dolayı izleme jürimde bulunan Sayın Prof. Dr. Ahmet Vecdi CAN ve Sayın Yrd. Doç. Dr. Kamil TAŞKIN'a, yapıcı eleştirileri sonucunda çalışmamın son haline gelmesine değerli katkılar yaptıklarından dolayı savunma sınavı jüri üyelerinden Prof. Dr. Vasfi HAFTACI ve Doç. Dr. Haluk BENGÜ'ye teşekkürlerimi arz ederim. Tez çalışmam sırasında devamlı olarak beni motive ederek konsantrasyonumu sağlayan sevgili eşim Ceyhan CALAYOĞLU'na ve son olarak beni büyütüp bu günlere getiren ve sayısız fedakârlıklarda bulunan annem Zindenur CALAYOĞLU ve babam Muzaffer CALAYOĞLU'na şükranlarımı sunarım.

İlker CALAYOĞLU

23.09.2016

İÇİNDEKİLER

KISALTMALARiv
TABLO LİSTESİ	vi
ŞEKİL LİSTESİ	ix
ÖZET	x
SUMMARY	xi
GİRİŞ	1
BÖLÜM 1: YAZILIM GELİŞTİRME SÜREÇLERİ, MODELLERİ ve PROJE YÖNETİMİ	5
1.1. Dünya’da ve Türkiye’de Yazılım	5
1.1.1. Yazılım ve Türleri	5
1.1.2. Yazılım Sektörünün Özellikleri	8
1.1.3. Dünya’da Yazılım Sektörü.....	9
1.1.4. Türkiye’de Yazılım Sektörü.....	14
1.1.5. Yazılım Sektöründen Genel Beklentiler	20
1.2. Yazılım Geliştirme Süreçleri	24
1.2.1. Proje Kabulü ve İlk Planlama	24
1.2.2. Analiz	27
1.2.3. Tasarım.....	28
1.2.4. Kodlama	30
1.2.5. Test.....	30
1.2.6. Devreye Alma	31
1.2.7. Bakım ve Destek	32
1.3. Yazılım Geliştirme Modelleri	34
1.3.1. Doğrusal Modeller.....	34
1.3.1.1. Şelale (Waterfall) Modeli.....	35
1.3.1.2. V-Model	36
1.3.2. Döngüsel Modeller.....	37
1.3.2.1. Artımlı (Incremental) Geliştirme Modeli.....	37
1.3.2.2. Spiral Model.....	38

1.3.2.3. Çevik (Agile) Model	40
1.4. Yazılım Proje Yönetimi	43
1.4.1. Proje Büyüklüğünü Ölçme ve Tahminleme.....	47
1.4.2. Proje Yönetim Yazılımları.....	50
BÖLÜM 2: TÜRKİYE MUHASEBE STANDARTLARI AÇISINDAN	
YAZILIMLARIN DEĞERLENDİRİLMESİ.....	55
2.1. TMS-38'e Göre Yazılımların Değerlendirilmesi	58
2.1.1. Maddi Olmayan Duran Varlıklar Kapsamı ve Yazılımlar	58
2.1.2. Satın Alınan Yazılımlar ve Değerlendirilmesi	60
2.1.3. İşletme İçi Oluşturulan Yazılımlar ve Değerlendirilmesi	61
2.1.4. Yazılımın Yararlı Ömrü ve Değerlendirilmesi.....	67
2.1.5. Yazılımın Yeniden Değerlenmesi ve Değerlendirilmesi	72
2.2. TMS 11'e Göre Yazılım Taahhütlerinin Değerlendirilmesi	73
2.2.1. Sözleşmenin Tanımı ve Değerlendirilmesi	74
2.2.2. Sözleşmelerin Birleştirilmesi ve Bölünmesinin Değerlendirilmesi	74
2.2.3. Sözleşme Gelirleri ve Değerlendirilmesi	76
2.2.4. Sözleşme Gelir ve Giderlerinin Muhasebeleştirilmesi.....	76
2.2.5. Sözleşmenin Tamamlanmasını Hesaplama Yöntemleri.....	77
2.2.6. Sözleşme Maliyetleri.....	80
2.2.7. Sözleşmeye Dair Yapılması Gereken Açıklamalar	81
2.3. TMS 18'e Göre Yazılım Hasılatının Değerlendirilmesi	82
2.3.1. Hasılatın Kapsamı	82
2.3.2. Hizmet Sunumu İle İlgili Hükümler	83
2.3.3. Hasılatın Ölçümü İle İlgili Hükümler	85
2.3.4. İşlemin Ayrıştırılması.....	87
BÖLÜM 3: YAZILIMLARIN MUHASEBELEŞTİRİLMESİ	88
3.1. Satın Alınan Yazılımların Muhasebeleştirilmesi	88
3.1.1. İlk Muhasebeleştirme	88
3.1.2. Yararlı Ömür ve Amortisman	91
3.1.3. Yeniden Değerleme.....	94
3.2. Kullanım Amaçlı Geliştirilen Yazılımların Muhasebeleştirilmesi	96
3.2.1. Modelin Kısıtları	97
3.2.2. Yazılım Geliştirme Süreci, Harcama Alanları ve TMS 38'in Yaklaşımı	98

3.2.3. Harcama Alanları, Tutarları ve Açıklamalar.....	100
3.2.4. Kredi Detayları ve Açıklamalar	103
3.2.5. Örneğin 2014 Yılı Gantt Şeması	105
3.2.6. Aylık Bazda Harcama Kalemleri, Sınıflandırma ve Muhasebeleştirme	106
3.2.7. Geliştirme Sürecindeki İşlerin Dağılımı ve Aylık Harcamalar Özeti	141
3.2.8. Yararlı Ömür ve Amortisman	144
3.2.9. Yeniden Değerleme	146
3.3. Sipariş Üzerine Geliştirilen Yazılımların Muhasebeleştirilmesi	146
3.3.1. Hesaplama Metodu ve Kısıtlar	147
3.3.2. Örnek Bir Uygulama ve Detayları	148
3.3.3. Tamamlanma Yöntemine Göre Hesaplama ve Raporlamalar.....	151
3.3.4. Tamamlanma Oranı Yöntemine Göre Hesaplama ve Raporlamalar.....	153
3.3.5. Vade Farkı Olması Durumunda Yapılması Gereken Kayıtlar	157
BÖLÜM 4: YAZILIM FİRMALARININ FİNANSAL DURUM TABLOLARI ve	
DİPNOTLARININ ANALİZİ.....	164
4.1. En Değerli Yazılım Firmalarının Analizi	164
4.2. BIST'deki Bilişim ve Teknoloji Endeksindeki Firmaların Analizi	168
SONUÇ ve DEĞERLENDİRME.....	172
KAYNAKÇA	177
ÖZGEÇMİŞ.....	191

KISALTMALAR

ABD	: Amerika Birleşik Devletleri
AICPA	: American Institute of Certified Public Accountants Amerikan Yeminli Mali Müşavirler Enstitüsü
AR-GE	: Araştırma ve Geliştirme
BİST	: Borsa İstanbul
BRIC	: Brazil, Russia, Indian, China, South Africa Republic Brezilya, Rusya, Hindistan, Çin, Güney Afrika Cumhuriyeti
BSMV	: Banka Sigorta Muamele Vergisi
CPM	: Critical Path Method Kritik Yol Metodu
DMODV	: Diğer Maddi Olmayan Duran Varlık
FASB	: Financial Accounting Standards Board Amerikan Finansal Muhasebe Standartları Kurulu
GSYH	: Gayri Safi Yurtiçi Hasıla
IAS	: International Accounting Standards Uluslararası Muhasebe Standartları
IASB	: International Accounting Standards Board Uluslararası Muhasebe Standartları Kurulu
IFPUG	: International Function Point Users Group Milletlerarası Fonksiyonel Nokta Kullanıcı Grubu
IFRS	: International Financial Reporting Standards Uluslararası Finansal Raporlama Standartları
KDV	: Katma Değer Vergisi
KKDF	: Kaynak Kullanımı Destekleme Fonu
KOBİ	: Küçük ve Orta Büyüklükteki İşletmeler
KOSGEB	: Küçük ve Orta Ölçekli Sanayiye Geliştirme ve Destekleme İdaresi Başkanlığı
MAM	: Marmara Araştırma Merkezi
MDV	: Maddi Duran Varlık
MODV	: Maddi Olmayan Duran Varlık
MS	: Microsoft
OECD	: Organization for Economic Co-operation and Development Ekonomik Kalkınma ve İşbirliği Örgütü

PAAS	: Platform as a Service Platform Hizmetleri
PERT	: Project Evolution and Review Technique Proje Değerlendirme ve Gözden Geçirme Tekniği
PWC	: Price House Water Coopers
SAAS	: Software as a Service Yazılım Hizmetleri
SFAS	: Statement of Financial Accounting Standards Finansal Muhasebe Standartları Açıklamaları
SOP	: Statements of Position Durum Açıklamaları
SSK	: Sosyal Sigortalar Kurumu
STK	: Sivil Toplum Kuruluşları
TDHP	: Tek Düzen Hesap Planı
TESMER	: Temel Eğitim ve Staj Merkezi
TEYDEP	: Teknoloji ve Yenilik Destek Programları Başkanlığı
TFRS	: Türkiye Finansal Raporlama Standartları
TMS	: Türkiye Muhasebe Standartları
TOBB	: Türkiye Odalar ve Borsalar Birliği
TÜBİTAK	: Türkiye Bilimsel ve Teknik Araştırma Kurumu
USD	: United States Dollars Birleşik Devletler Doları
US-GAAP	: United States Generally Accepted Accounting Principles Amerikan Genel Kabul Görmüş Muhasebe Prensipleri
VUK	: Vergi Usul Kanunu

TABLO LİSTESİ

Tablo 1: Piyasa Değeri En Fazla Olan Yazılım Firmaları	11
Tablo 2: Türkiye'nin Ükelere Göre Yazılım İhracatı (\$).....	17
Tablo 3: Türkiye'nin Ükelere Göre Yazılım İthalatı (\$).....	18
Tablo 4: Bilişim ve Yazılım Alanındaki Sivil Toplum Kuruluşları.....	19
Tablo 5: En Fazla Geliri Olan 100 Yazılım Firmasının Ükelere Göre Dağılımı.....	21
Tablo 6: Yerli Yazılım Üreticilerinin Yazılım Gelirleri	22
Tablo 7: Proje Yönetim Yazılımları.....	54
Tablo 8: Standartların Yazılım Geliştirme Aşamaları Hakkındaki Kabulleri.....	66
Tablo 9: Peşin ve Vadeli Hasılatın Muhasebeleştirilmesi.....	86
Tablo 10: Satın Alınan Yazılımların Listesi ve Fiyatları	89
Tablo 11: Örnek 1'in Muhasebe Kaydı.....	89
Tablo 12: Örnek 2'nin Muhasebe Kaydı.....	90
Tablo 13: Örnek 3'ün Muhasebe Kaydı.....	90
Tablo 14: Örnek 4'ün Muhasebe Kaydı.....	91
Tablo 15: Örnek 5'in Muhasebe Kaydı.....	92
Tablo 16: Örnek 6'nın Muhasebe Kaydı.....	94
Tablo 17: Yazılım Geliştirme Süreci, Harcama Alanları ve TMS 38'in Yaklaşımı.....	99
Tablo 18: Gider Çeşitleri, Tutarları ve Diğer Bilgiler.....	100
Tablo 19: Kredi Ödeme Tablosu.....	104
Tablo 20: Örneğin 2014 Yılı Gantt Şeması.....	105
Tablo 21: Gantt Şemasındaki Renklerin Anlamları	105
Tablo 22: Ocak Ayı Harcama Kalemleri ve Muhasebeleştirilmesi	110
Tablo 23: Şubat Ayı Harcama Kalemleri ve Muhasebeleştirilmesi.....	112
Tablo 24: Mart Ayı Harcama Kalemleri ve Muhasebeleştirilmesi	115
Tablo 25: Nisan Ayı Harcama Kalemleri ve Muhasebeleştirilmesi.....	118
Tablo 26: Mayıs Ayı Harcama Kalemleri ve Muhasebeleştirilmesi.....	121

Tablo 27: Haziran Ayı Harcama Kalemleri ve Muhasebeleştirilmesi	124
Tablo 28: Temmuz Ayı Harcama Kalemleri ve Muhasebeleştirilmesi.....	127
Tablo 29: Ağustos Ayı Harcama Kalemleri ve Muhasebeleştirilmesi.....	130
Tablo 30: Eylül Ayı Harcama Kalemleri ve Muhasebeleştirilmesi	133
Tablo 31: Ekim Ayı Harcama Kalemleri ve Muhasebeleştirilmesi	136
Tablo 32: Kasım Ayı Harcama Kalemleri ve Muhasebeleştirilmesi	139
Tablo 33: Aralık Ayı Harcama Kalemleri ve Muhasebeleştirilmesi.....	141
Tablo 34: Projedeki İşlerin Dağılımı ve Aylık Harcamalar Özeti	142
Tablo 35: Aktifleştirilen Geliştirme Maliyetleri	143
Tablo 36: Değeri Artan Varlığın Yeni İfta Payı Hesaplaması	145
Tablo 37: Tahmini Sözleşme Bilgileri	148
Tablo 38: Dönemler İçindeki Nakit Akışı Özeti	149
Tablo 39: Sözleşmeye Bağlı Yazılım Geliştirme Harcamalarının Özeti	150
Tablo 40: Tek Düzen Hesap Planına Göre Birinci Yılın Muhasebe Kayıtları.....	152
Tablo 41: Tek Düzen Hesap Planına Göre İkinci Yılın Muhasebe Kayıtları	152
Tablo 42: Tek Düzen Hesap Planına Göre Projenin Kar-Zarar Özeti	153
Tablo 43: Projenin Tamamlanma Aşamalarına Göre Kar-Zarar Özeti.....	154
Tablo 44: TFRS'ye Göre Birinci Yılın Muhasebe Kayıtları.....	155
Tablo 45: TFRS'ye Göre İkinci Yılın Muhasebe Kayıtları	156
Tablo 46: Hesaplanan Değerler ile Kayıtlarda Kullanılan Değerler Farkı	157
Tablo 47: Örnek 7'nin Hesaplanması	158
Tablo 48: Örnek 7'nin Muhasebe Kaydı	159
Tablo 49: Örnek 8'in Hesaplanması	159
Tablo 50: Örnek 8'in Muhasebe Kaydı	160
Tablo 51: Örnek 9'un Hesaplaması	161
Tablo 52: Örnek 9'un Muhasebe Kaydı	161
Tablo 53: Örnek 10'un Hesaplaması	162

Tablo 54: Örnek 10'un Muhasebe Kaydı	163
Tablo 55: En Değerli Yazılım Firmalarının Analizi	165
Tablo 56: Yerel Aktifleştirme Yapan Firmaların Analizi	169



ŞEKİL LİSTESİ

Şekil 1: Yazılım Tiplerinin Dünya Piyasasındaki Payları	12
Şekil 2: Dünyada Paket Yazılımların Gelirleri	12
Şekil 3: Şelale (Waterfall) Yazılım Geliştirme Modeli Yaklaşımı.....	35
Şekil 4: V-Yazılım Geliştirme Modeli Yaklaşımı	36
Şekil 5: Artırmalı (Incremental) Yazılım Geliştirme Modeli Yaklaşımı	38
Şekil 6: Spiral Yazılım Geliştirme Modeli Yaklaşımı.....	39
Şekil 7: Çevik (Agile) Yazılım Geliştirme Modeli Yaklaşımı	41
Şekil 8: Proje Yönetim Süreçlerinin Haritalandırılması	47
Şekil 9: Gantt Şeması Örneği.....	53
Şekil 10: Yazılımların Kullanım Amaçları ve TMS Yaklaşımı	56
Şekil 11: SFAS 86'ya Göre Aktifleştirme ve Gider Dönemleri.....	65
Şekil 12: Yazılımın İtfa Payının Hesaplanması.....	70
Şekil 13: Yazılımın Kullanım Yerine Göre İtfa Payının Kaydedilmesi.....	71
Şekil 14: Maddi Olan / Olmayan Varlıkların Yeniden Değerleme Modeli.....	95
Şekil 15: Taksitlerin Beklenme Süreleri.....	162

Tezin Başlığı: Yazılım Maliyetlerinin Türkiye Muhasebe Standartlarına Göre Muhasebeleştirilmesine İlişkin Model Önerisi

Tezin Yazarı: İlker CALAYOĞLU **Danışman:** Yrd. Doç. Dr. Recep YILMAZ

Kabul Tarihi: 23 Eylül 2016 **Sayfa Sayısı:** xi (ön kısım) + 191 (tez)

Anabilim Dalı: İşletme

Bilim Dalı: Muhasebe ve Finansman

Bilişim firmalarının önemli gelir ayağı yazılımlardır. Bu firmaların yazılımla olan ilişkileri iki ana başlıkta toplanır. Ana başlıkların da altında ikişer alt başlık vardır. Ana başlıklar şunlardır: Satış için üretim ve kullanım için edinimdir.

Satış amacı olduğunda yazılım geliştirme süreçlerine katlanılır ve ürün üretilir. Bu üretim ya özel bir sipariş için ya da genel ihtiyaçlara cevap vermek (paket program) için olabilir. Dolayısı ile aynı süreçlerden geçerek farklı iki durum ortaya çıkar. Bu sebeple muhasebe kayıtları da farklı olacaktır.

Kullanım amaçlı yazılım edinimi için işletmeler iki seçeneğe sahiptir. Birincisi hazır olarak piyasadan satın almak, ikincisi ise kendi üretimini yapmak. Burada aynı amaç hedeflenir ama farklı süreçlerden geçilir. Dolayısı ile muhasebe kayıtları da çok farklı olacaktır.

Bilişim firmaları yazılım üretimi yaparken başka yazılımları kullanmaya ihtiyaç duyar. Böylece bilişim firmaları yukarıdaki dört süreç için pozisyon sahibi olabilir.

Uluslararası ve Türkiye borsalarında işlem gören ve piyasa değeri açısından incelemeye değer yazılım firmaları bulunmaktadır. Bu firmalar dönemsel olarak finansal raporlama arz etmektedir. Bu sebeple TMS-11, TMS-18 ve TMS-38 standartları incelenmiştir. TMS’de, özellikle yazılım üretimi konusunda nasıl davranılması gerektiği net değildir. Bu sebeple farklı standartlar (US-GAAP) incelenmiştir. Yazılım satın alma, üretimi, aktifleştirilmesi, sözleşmeye bağlı satışı ve hasılatı ile ilgili konular incelenmiştir. Yukarıdaki işlemlerin kayıtları için ihtiyaç olan proje yönetim sisteminin çıktıları hakkında bir model geliştirmesi yapılmıştır. Buna göre detaylı senaryolar ile muhasebeleştirmeler yapılmıştır. Bu esnada Türkiye uygulaması için yeni hesap isim önerileri gerekçeleriyle ifade edilmiştir.

Uluslararası ve Türkiye borsalarındaki firma örnekleminin finansal raporları incelenmiştir. Çalışmada kullanılan model ile firmaların raporları ve dipnot açıklamaları karşılaştırılmıştır. Aynı zamanda farklı standartlara (US-GAAP) tabi olmanın sonuçları izlenebilmiştir.

Anahtar Kelimeler: Türkiye Muhasebe Standartları, Yazılım Üretim Amaçları, Yazılım Geliştirme Süreçleri, Araştırma ve Geliştirme, Proje Yönetimi Yazılımları

Title of the Thesis: An Accounting Model Proposal of Software Cost According to Turkish Accounting Standards	
Author: İlker CALAYOĞLU	Supervisor: Assist. Prof. Recep YILMAZ
Date: 23 September 2016	Nu. of pages: xi (pre text) + 191 (main body)
Department: Business Administration Subfield: Accounting and Finance	
<p>Software is the main source of incomes of IT companies. The relationship of these companies with the software collected in two main categories. There are two sub-groups in each main category. The main categories are: Production for sales and acquire for use.</p> <p>Software development processes occur and produce product when sale of software is purposed. This production can be for special order or responding to general needs (package program). Therefore there are two different situation although the processes are the same. Therefore accounting records will also be different.</p> <p>Business has two options for the acquisition of software. First is to buy from market; second is make own production. Here the same purpose is intended, but processes is different. Therefore accounting records will be very different.</p> <p>IT companies need to use other software while producing software. Thus, IT companies may take parts for above 4 processes.</p> <p>There are worth-analyzing companies in terms of market value which are producing software that are trading on Turkey and International Stock Markets. These companies present annual financial reports. Therefore TMS-11, TMS-18 and TMS-38 standard were analyzed. Especially TMS is not clear about how to report for producing software. Therefore different standards (US-GAAP) were examined. Related issues with software purchasing, producing, capitalizing, selling on contract and revenues were researched. A model development is made about outcomes of the project management system that needs to record the above transactions. Accordingly, the accounting is done with detailed scenarios. In the meantime, it is proposed some new account name with reasons for Turkey practice.</p> <p>Some companies in International and Turkey stock market have been examined. Model in this study was compared with reports and disclosure of companies. At the same time results which subject to different standards (US-GAAP) could be followed.</p>	
Keywords: Turkey Accounting Standards, Software Production Goals, Software Development Process, Research and Development, Project Management Software	

GİRİŞ

Bilişim firmalarının üretmiş oldukları yazılımlar her geçen gün işletmelerin ve bireysel kullanıcıların hayatını kolaylaştırmaktadır. Bu nedenle piyasa ihtiyacına cevap veren birçok yazılım firması benzer yazılımlar üreterek piyasaya sunmaktadır.

Yazılım, soyut bir varlık olduğundan üretim süreci de somut ürün üretiminden farklıdır. Yazılım firmaları çok az somut varlık kullanarak çokça maddi olmayan varlık yani entelektüel sermaye bileşenlerini kullanarak maddi olmayan bir ürün, yani yazılım üretirler.

Yazılım geliştirme, çeşitli yönler içeren, bir dizi sürecin olduğu biraz karmaşık bir konudur. Konuya dair hesaplamaları ve muhasebe yorumlarını açıklayabilmek için yazılım geliştirme süreçlerinin neler olduğu ve bu süreçlerde yapılan işler açıklanacaktır.

Yazılım geliştirme süreçleri farklı sıralarla kullanılabilir. Böyle olmasından dolayı farklı geliştirme modelleri bulunmaktadır. Her modelin özellikleri açıklanarak güçlü ve zayıf yönleri belirtilecektir.

Yazılım sektörü, Türkiye için yeni yeni gelişen ve ciddiye alınmaya başlanan bir sektördür. Profesyonelleşme de bu ivmeden nasibini almaktadır. Böylece yazılım sektörünün gelişmesi ve borsadaki işlem hacimleri artmaktadır. Borsada işlem gören firmaların Türkiye Finansal Raporlama Standartlarına uygun şekilde raporlama yapması beklendiğinden bu konudaki raporlamanın sunduğu bilgi ve ölçme tekniklerinin araştırılması gerekmektedir. Çünkü yazılım üretimi konusu çeşitli amaçlar içerebilir.

Yazılım edinmenin amaçları temelde iki ana başlıktadır. Bu ana başlıkların da ikişer alt başlıkları vardır.

- a) İşletme içinde kullanmak amacı aşağıdaki yollar ile sağlanabilir.
 - İşletme, kendi bünyesinde yazılım geliştirebilir.
 - Üretmeyip dışardan hazır olarak satın alabilir.
- b) Dışa servis ederek ekonomik fayda sağlama amacı aşağıdaki yollar ile sağlanabilir.
 - Sözleşmeye bağlı olarak yazılım üretimi taahhüt edilir.

- Sabit bir yazılım (paket program) üretilir.

Bilişim firmaları yazılım üretimi yaparken başka yazılımları kullanmaya ihtiyaç duyar. Böylece bilişim firmaları yukarıdaki dört süreç için pozisyon sahibi olabilir.

Çalışmanın Konusu ve Amacı;

Çalışma, “Yazılım Maliyetlerinin Türkiye Muhasebe Standartlarına Göre Muhasebeleştirilmesine İlişkin Model Önerisi” başlığı taşımaktadır. Bu başlık altında yukarıda bahsedilen yazılım geliştirme amaçlarının tamamına muhatap olabilen bilişim işletmelerinin konu hakkındaki finansal bilgi üretmeleri ve Türkiye Muhasebe Standartları kapsamında raporlama yapmanın dayanakları ile pratiğinin araştırılıp bir model olarak sunulmasıdır.

Çalışmanın Önemi;

Borsa İstanbul'da yer alan Bilişim ve Teknoloji endeksindeki işletmeler, Türkiye Finansal Raporlama Standartları çerçevesinde belirli dönemlerde denetlenerek mali tablolarını tüm paydaşlarının erişimine sunar. İşletme yöneticisi, ortaklar ve yatırımcılar bu mali tablolara göre karar verir. Doğru bilginin olmadığı durumlar manipülasyonlara sebebiyet verebilir. Türkiye Finansal Raporlama Standartları, önümüzdeki zaman içerisinde belli hacimde iş yapan firmaların geneline zorunlu olacak olmasından dolayı halka açık olmayan firmaları da ilgilendiren önemli bir durumdur. Birçok ilde, üniversitelerde ve merkezlerde Tekno-Parklar kurulmaktadır. Buradaki faaliyetlerin yarıya yakını yazılım geliştirmektir. Bunun dışındaki münferit işletmeler de hep bu kapsam dâhilinde olabilecek potansiyele sahiptir.

Çalışmanın Yöntemi;

Araştırma sürecinde yazılım geliştirme maliyetlerinin muhasebeleştirilmesine dair literatür taraması yapılmıştır. Bu alanda TFRS-TMS haricinde müstakil standartlar düzenlenmiştir. Bu standartları oluşturan kurullar şunlardır; Amerikan Yeminli Mali Müşavirler Enstitüsü (AICPA) ve Amerikan Finansal Muhasebe Standartları Kurulu (FASB)' dir. Türkiye Muhasebe Standartlarının açıklamadığı veya muğlak kaldığı noktalarda diğer standartlardaki açıklamalardan esinlenerek yorumlar yapılmıştır. Bu gibi noktalarda standartlar arasında bir yakınsamadan bahsedilebilir.

Bunun yanında kitaplar, süreli yayın makaleleri, tez ve raporlardan oldukça istifade edilmiştir. Konunun Türkiye için yeni bir konu olması Türkçe literatürdeki kaynakların

azlığına da yansımıştır. Türkçe literatürün azlığının temelinde Türkiye Muhasebe Standartlarında yazılım geliştirmeyle ilgili bir standardın bulunmayışıdır.

Suermann'ın doktora tezindeki yazılım edinme amaçlarını standartlar ile eşleştirmesini takip ederek aşağıdaki standartlar temel alınarak değerlendirmeler, konuyu açıklayan senaryolar ve muhasebe örnekleri verilmiştir.

Satın alınan ve işletme içi geliştirilen yazılımlar için TMS 38 – Maddi Olmayan Duran Varlıklar, Sözleşme çerçevesinde satılan yazılımlar için TMS 11 – İnşaat Sözleşmeleri, Satışların hasılatı için TMS 18 – Hasılat Standardı incelenmiştir.

Bu araştırmanın temelinde yazılım geliştirme harcamalarının hangisinin araştırma hangisinin geliştirme olduğuna karar verilmesi gerekmektedir. Çünkü yukarıdaki yazılım edinim amaçlarını sınıflandırdığımızda üç tanesi için üretim söz konusudur. Bu sebeple yazılım geliştirme süreçleri ile standartlardaki tanımlar eşleştirilmiştir. Böylece aktifleştirilecek olan ve giderleştirilecek olan harcamalar ortaya çıkmıştır.

Üretilen bilginin hangi hesaplarda raporlanacağına dair Türkiye uygulamasına dönük olarak TDHP'na bazı hesap isimleri önerilmiştir.

Sözleşmeye bağlı geliştirilen yazılımların Türkiye Muhasebe Standartlarına göre raporlanabilmesi için tamamlanma oranının tespiti gerekmektedir. Bunu sağlayacak şekilde üretimde fiili olarak gerçekleşen harcamaları dönemsel ve harcama kalemi bazında raporlamaya yarayan bir proje yönetim yazılımı modeli önerilmiştir.

Türkiye'deki Bilişim ve Teknoloji Endeksindeki firmalar ile Forbes dergisinin 2014 yılı en büyük 2000 firma içindeki 17 adet yazılım firmasının finansal raporları incelenerek raporlamada ve dipnotlardaki bilgiler ile çalışmada geliştirilmiş hesaplama modeli karşılaştırılmıştır. Türkiye'deki uygulayıcıların yetersiz bilgi sunmasına karşın bu konuda bir TFRS Yorumu'na ihtiyaç olduğu anlaşılmıştır.

Çalışmanın Kısıtları;

Yazılım geliştirme harcamalarını sınıflandırma ve raporlama modeli için; yazılım geliştirme modeli olarak “Şelale Modeli” tercih edilmiştir. Çünkü gerek geliştirme işlerini iş planında rahat gösterebilmek gerekse harcamaları takip edip, sınıflandırıp geliştirme faaliyetlerine dağıtabilmek daha kolay olacaktır.

Piyasadaki firmalar genellikle şelale modeline göre yazılım geliştirmez, çünkü çok fazla atıl kapasite oluşur ve harcamalar artar. Günümüzde geliştiricilerin tercih ettiği modeller çevik modellerdir. Ancak unutulmamalıdır ki çevik modeller döngüsel olsa da her döngü içinde şelale modelinin uygulanması gerekir. Yani, bir proje için birden çok şelale modelinin uygulanması şeklinde özetlenebilir. Burada esas olan geliştirme faaliyetlerini takip edebilmek ve harcamaların dağıtımını açıklayabilmektir. Bu sebeple izahı ve görseli en kolay olacak olan geliştirme modeli tercih edilmiştir.

İş planı, Gantt şemasından esinlenilerek Excel 2013'de oluşturulmuştur. Yazılım geliştirme faaliyetlerinin tükettiği harcamaları hesaplamak için faaliyet sürücüsü olarak zaman esası tavsiye edilmektedir. (Walker ve Oliver, 2005: 69) Hesaplaması ve görsel olarak gösterimi kolay olsun diye birim zaman olarak modelde hafta tercih edilmiştir. Yine gerçek bir yazılım geliştirme projesinde bir personel aynı gün içinde biraz A ve biraz B projesinde çalışabilir. Bu sebeple doğru bir hesaplama için çalışanın eforunun zaman cinsinden hangi projede ne kadar çalıştığının takip edilmesi gerekmektedir. (Gentle, 2011: 43) Yani, birim zaman daha küçük bir birim olan saat veya dakika cinsinden olmalıdır.

Sözleşme gelirinin hesaplanması için TMS-11 İnşaat İşlemleri standardının tavsiye ettiği “Tamamlanma Oranı” yöntemidir. Tamamlanma oranı yöntemi ile tamamlanma yöntemi arasındaki farkı açıklayabilmek için uygulama örneği her iki yöntemle göre hesaplanıp muhasebeleştirilmiştir.

Tamamlanma oranı yöntemi açıklanırken kullanılan örnekte tahsilatlarda vade farkına yer verilmemiştir. Tahsilatlar peşin olarak varsayılmıştır. Tamamlanma yönteminin anlatımı bittikten sonra çeşitli örnekler ile vade farkı durumunda yapılması gereken hesaplamalar ve muhasebe kayıtları açıklanmıştır. Bunun sebebi, her iki konunun (hasılat ve tamamlanma oranı) birbirinden farklı ancak iç içe olup, birbirini etkileyen olmasından dolayıdır. Böylece açıklamalar daha anlaşılabilir olmaktadır.

Örneklerde yapılan muhasebe kayıtlarında Katma Değer Vergisi dikkate alınmadığından hesaplanmamıştır.

TMS-2'nin kapsam dışında bırakılması: Çalışma, yazılım geliştirme harcamalarının sınıflandırılması, üretim maliyetinin hesaplanması ve muhasebeleştirilmesidir. Bu sebeple stoklar grubu teze dâhil edilmemiştir. Bu konu daha çok satılan ürünün maliyetinin nasıl hesaplanacağına dair modellere ihtiyaç duymaktadır.

BÖLÜM 1: YAZILIM GELİŞTİRME SÜREÇLERİ, MODELLERİ ve PROJE YÖNETİMİ

Bu bölümde yazılım sektörü ile ilgili bilgiler verilecektir. Öncelikle yazılımın tanımı yapılıp, türleri açıklanacaktır. Yazılım sektörüne ait genel bilgiler ile birlikte Dünya'daki ve Türkiye'deki yazılım sektörünün durumu hakkında istatistiki bilgiler sunulacaktır. Böylelikle karşılaştırma yapılıp, beklentiler arz edilecektir.

Konuya dair hesaplamaları ve muhasebe yorumlarını açıklayabilmek için yazılım geliştirme süreçlerinin neler olduğu ve bu süreçlerde yapılan işler açıklanacaktır.

Yazılım geliştirme süreçleri farklı sıralarla kullanılabilir. Böyle olmasından dolayı farklı geliştirme modelleri bulunmaktadır. Her modelin özellikleri açıklanarak güçlü ve zayıf yönleri belirtilecektir.

Son olarak, proje yönetimi hakkında tanımlayıcı bilgiler verilip Türkiye Muhasebe Standartları (TMS)'nin ihtiyaç duyduğu raporlama bilgileri açıklanacaktır.

1.1. Dünya'da ve Türkiye'de Yazılım

Yazılım sektörünün genel özellikleri açıklanarak sektörün ne denli katma değerli bir alan olduğu istatistiki bilgiler ile vurgulanacaktır. Ayrıca sektörün Dünya'daki ve Türkiye'deki durumu karşılaştırılarak Türkiye'nin konumunu iyileştirmesi için gerekli bir dizi reformlardan bahsedilecektir.

1.1.1. Yazılım ve Türleri

Yazılım, bir teknoloji ve bilişim terimidir. Yazılım, çalıştırıldığında bilgisayara belli fonksiyonları yaptırabilen bir dizi komutlar bütünüdür. (Bilişim Terimleri Sözlüğü, 2004: 354) Tanım dâhilinde geçen komut kavramı ise bilgisayara, klavye veya diğer giriş birimlerinden, bir takım işler yaptırabilmek amacıyla verilen emirler anlamına gelmektedir. (Arpacı, 2010: 4) Yazılım kavramı, ilk kez istatistikçi John Wilder Tukey tarafından 1958 yılında "American Mathematics Monthly" dergisinde yayınlanan bir makalede elektronik hesaplamalarda kullanılan programları tanımlamak amacıyla kullanmıştır. (<http://global.britannica.com>, 2014)

Yazılımın kendisini görmemiz ve dokunmamız mümkün değildir, çünkü yazılımlar bilgisayarın belleğine yüklenen komutlardan ibarettir. Yazılımı olmayan bir bilgisayar çalıştırıldığında ekranda sadece boş bir görüntü oluşur. Kısacası yazılımlar hep bir ihtiyacı gidermek için oluşturulan, tekrar tekrar aynı sorguyu veya işlemi yapmaktan bıkmayan, donanımların kullanılmasına da imkân veren komutlar bütünüdür.

Türk Dil Kurumunun sözlüğüne göre yazılım; bir bilgi işlem dizgesinin işleyişi ile ilgili bilgisayar izlencelerinin, yordamların, kuralların ve gerektiğinde belgelemenin tümü şeklinde tanımlanmaktadır. (Türk Dil Kurumu, 1981) Buna göre yazılım; belirli bir işlev odaklı olan ancak soyut özelliklere sahip bir varlıktır.

Yazılım türleri en genel anlamda incelendiğinde iki gruba ayrılır. Bunlar şunlardır;

- 1) İşletim Sistemi Yazılımları: Bu tür yazılımlar bilgisayarların tüm işlevlerini yöneten ve diğer yazılımların çalışmasına imkân sağlayan, platform yazılımlardır.
- 2) Uygulama Yazılımları: Çok çeşitli amaçlara hizmet eden yazılım türleridir. Her uygulama yazılımı bir işletim sistemi üzerinde çalışmak zorundadır. Word, Excel, Muhasebe paket programları, SPSS ve hatta oyunlar en iyi örneklerdendir.

Çevremizde işlemcisi olan her aygıt ki bu grup günümüzde oldukça çeşitlenmiş ve ebat olarak da küçülmüştür; masaüstü, laptop, netbook, ultrabook, tablet bilgisayarlar, akıllı telefonlar grubu ürünler sahip oldukları işlem yeteneklerini hep kullandıkları yazılımlara borçludur. Dolayısıyla uygulama yazılımları ile çeşitlenen işlevsellik ve görsellik asıl fayda gösteren kısımdır. Sorunlarına çözüm arayan ihtiyaç sahibi insanlar en çok uygulama yazılımlarına para öderler. Eğlence amaçlı geliştirilen oyunlar da yine eğlence ihtiyacını giderir. Yazılımları üretmek açısından sınıflandırmak da mümkündür. Buna göre beş gruba ayrılabilir. (Sarıdoğan, 2011: 18)

- 1) Hazır Ticari Ürünler: Piyasada hazır olarak bulunan, doğrudan satın alınarak kullanılan hazır ticari ürünler (Commercial Off-The-Shelf-COST) geliştirme maliyetini düşürürler. Bu yazılımların isterleri tam karşılayamaması, bakım gücülüğü olması, sık sık yeni sürümler çıkması ve uyarılama sorunları bulunması nedeniyle çok özel projelerde kullanımları pek tercih edilmez. Ancak, çok

dikkatle kullanıldıkları takdirde, çok büyük ölçüde zaman ve maliyet tasarrufu sağlarlar.

- 2) Var Olan Yazılımlar: Yeni sisteme daha önce geliştirilmiş bileşenlerin sağladığı yeteneklerin aynısını kazandırmak üzere var olan (legacy) yazılımlar tekrar kullanılabilir. Zaman ve maliyet tasarrufu yanında birçok açıdan büyük yarar sağlar. Ancak, tekrar kullanımdan dolayı çıkabilecek tümleştirme, beraber çalışabilirlik, nitelik ve başarımları sorunları mutlaka dikkate alınmalıdır.
- 3) Müşteri Tarafından Sağlanan Yazılımlar: Ana sistemi oluşturmak üzere bir kısım yazılım öğeleri müşteri tarafından verilir. Bunlar hazır veya müşterinin kendisi tarafından özel amaçlarla geliştirilmiş yazılımlar olabilir. Genellikle gizlilik dereceli bilgi işlemenin gerekli olduğu projelerde sözleşme tabanlı olarak kullanılır. Askeri projelerin bazılarında da güvenlik gerekçesiyle bir kısım yazılım öğelerinin geliştiriciye devlet tarafından hazır olarak verildiği bu tür uygulamalar olur.
- 4) Geliştirilen Yazılımlar: Sistem isterlerini karşılamak üzere özel ve eşsiz yetenekleri kazandırmak üzere geliştirilen yazılımlardır. Önceden var olan veya hazır olarak alınan yazılımlarla karşılanamayan isterleri karşılamak üzere, seçilen bir yöntemle göre geliştirilirler.
- 5) Destek Yazılımları: Yazılımların geliştirilmesi ve testi sırasında kullanılan yazılım araçlarıdır. Hazır olarak edinilebileceği gibi özel olarak da geliştirilebilirler.

Günümüzde bilgisayar teknolojilerinin ilerlemesi, ekonomik ve sosyal değişimler ile artık yazılım birkaç bilgisayar operatörünün bir araya gelerek meydana getirdiği basit programların çok ötesinde bir konumdadır. Programlama dilleri, işletim sistemleri, yazılım geliştirme araçları (platformları), veri tabanları, farklı geliştirme yaklaşımlarından oluşan karmaşık bir geliştirme sürecinin neticesinde ortaya çıkan ürünlerdir. Bu sebeptendir ki her yazılım ayrı bir projedir.

Proje, bir probleme çözüm bulma ya da beliren bir fırsatı değerlendirmeye yönelik, bir ekibin, başlangıcı ve bitişi belirli bir süre ve sınırlı bir finansman dâhilinde, birtakım kaynaklar kullanarak, müşteri memnuniyetini ve kaliteyi göz önünde bulundururken olası riskleri yönetmek şartıyla, tanımlanmış bir kapsama uygun amaç ve hedefler

doğrultusunda özgün bir planı başlatma, yürütme, kontrol etme ve sonuca bağlama sürecidir.

Tanımda belirtildiği gibi her projenin aşamaları ve yöntemi vardır. Bu sebeple her tamamlanmış yazılım projesi ister amatörce kabataslak bir yönetimle isterse profesyonel yöntemlerle olsun yönetilirler.

1.1.2. Yazılım Sektörünün Özellikleri

Yazılım sektörünün mamulü olan yazılımlar hep bir ihtiyacı karşılamak üzere geliştirildiğinden diğer sektörleri desteklemek veya tamamlamak üzere rol oynarlar. Kısacası yazılımlar hemen hemen tüm sektörlerde işleri kolaylaştırmakta ve mantıksal işlemlerin üstesinden gelerek karar vermede yardımcıdır. Kısacası yazılımlar; günlük hayatımıza dâhil olmak üzere, gerçekleştirilen tüm faaliyetler için, eksikleri tamamlayan ve mevcudu hızlandıran bir yapıya sahiptir. Zaman ile yarıştığımız günümüz yaşantısında konu ne olursa olsun kullanıcıya çözüm yaratabilen yazılımlar; gerek sosyal, gerekse iş hayatımızın bir parçası olmuştur.

Bugün bilinmektedir ki; bir ülkenin, diğer ülkeler üzerinde etkisi yalnızca teknolojik anlamdaki gücüyle değil, teknolojinin beraberinde ekonomik anlamda da güçlü olmasına bağlıdır. Bir ülkeyi ekonomik anlamda ileri götürecek yapıların, büyük yatırımlar ile yüksek karlılık hedefleyen şirketsel yapılar ve kazançları olduğu düşünülür. Buna karşılık, bilişim üzerine yapılan çalışmaların pahalı yatırımlar gerektirmemesi, hammaddesinin bilgi ve yaratıcılık olması, hitap edebileceği sektörlerin sınırsız olması, bilişim firmalarını yukarı taşır. Yazılım endüstrisinde tasarım ve geliştirme maliyetleri yüksek, çoğaltma ve dağıtma maliyetleri çok düşüktür. (Okur, 2007: 652)

Dünyada 3 I olarak adlandırılan Hindistan, İrlanda ve İsrail ülkeleri yazılım sektörünün önemini ve sektördeki boşluğu daha önce fark ederek, devlet desteği ile gerekli yatırımlar yapılarak yazılım ve hizmet ürünleri pazarında büyük paya sahip olmuşlardır. Gelişmekte olan ülkelerde bilişimin ülke ekonomisine sağladığı katma değer, bu üç ülkede açıkça görülebilir.

Yakın zamanda kurulmuş, geçmişi yedi yıllık olan bilişim firmaları, asırlık dev sanayi firmalarının değerlerinden birkaç kat değerlendirilerek bazılarının üstüne çıkmış ve ülke

ekonomisine sağladıkları katma değer tartışılmaz bir hâl almıştır. Peki, bilişim sektöründe geçmişi çok fazla olamayan bu firmalar nasıl oluyor da neredeyse bir asırlık sanayi devi firmaların önüne geçebiliyor? Çünkü artık “Sanayi Toplumundan”, “Bilgi Toplumuna” geçiş evresindeyiz. Avrupa Birliği tarafından başlatılan “Bilgi Toplumuna Dönüş” çalışmaları ve dünyada bilişim teknolojilerin önemi her geçen gün artmaktadır. Artık fabrika ve maddi üretim ülke ekonomilerinin katma değerli alanları olmaktan çıkmıştır. Bilişim çağında bilgi ve hizmet ürünleri önem kazanarak, maddi üretim, yerini düşünce üretimine bırakmıştır.

Bilişim sektörünün özellikle ABD ve diğer ileri ülkelerde hızla gelişmesi, bu alandaki bilgi birikimi, uzmanlık ve işgücü arzının da artmasına yol açmıştır. Sektörün bir özelliği, bu alana sonradan giren ülke ve firmaların da dünya pazarlarına açılma şanslarının yüksek olmasıdır. Bu nedenle, birçok gelişmekte olan ülke yazılım endüstrisinde yüksek potansiyel görmekte ve bu sektöre kendi ekonomilerinin geleceği açısından büyük ümit bağlamaktadır. Ancak, diğer bir gerçek, yazılım endüstrisinin 10-15 yıl önceye göre günümüzde daha düşük karlılıkla çalışan bir endüstri hâline gelmiş olmasıdır. (Arora ve Gambardella, 2004: 6) Bu durum, yazılım endüstrisini ekonomik açıdan kurtarıcı olarak gören gelişmekte olan ülkelerin beklentileri açısından olumsuzdur.

Gelişmekte olan ülkeler bakımından önemli diğer bir husus, yazılım sektörünün yoğun devlet desteğine ihtiyaç duymasıdır. Koruyucu ve gelişmeye yardımcı kurumsal destekler; sektörün rekabet gücünün artması, dış pazarlara açılma bakımlarından büyük önem taşımaktadır. Bu bağlamda; ülkenin genel Ar-Ge potansiyelinin artırılması, yenilikçiliğin yazılım alanında teşvik edilmesi, İrlanda ve İsrail örneklerinde olduğu gibi sektörün gelişmesine önemli katkılar sağlamaktadır. (Carmel, 2003: 4)

1.1.3. Dünya’da Yazılım Sektörü

Ülkeler; katma değeri yüksek olan bu alanda teşvikler yaparak bilgi teknolojilerinin gelişmesini ve gelecekte daha fazla milli geliri hedeflemektedirler. Öyle ki son yıllarda köklü sanayi firmalarının değerini, yeni kurulmuş bilişim firmaları geçmiştir. Yazılım sektörünün katma değeri ile sanayi sektörünün katma değeri arasındaki makas giderek açılmaktadır. Forbes dergisi tarafından 2014 yılında hazırlanmış dünyanın en büyük

2000 firmanın olduğu “Forbes Global 2000” listesine girebilen 17 yazılım firması tespit edilmiştir.

Tablo 1’de en büyük 2000 firma içinde bulunan firmaların piyasa değerine göre olan sıralamadaki yerleri ile firmaların varlık değeri açısından sıralamaları karşılaştırma yapılabilmesi için yan yana verilmiştir. Varlık değeri açısından sıralamada kaçınıcı olduğuna dair veri olmayan firmalar genel büyüklük açısından ilk 2000 firmanın içinde değildir. Dolayısıyla da varlıkları az olan yazılım firmalarından bazıları bu sıralamada 2000 firma arasında bile değilken piyasa değeri açısından sıralamada yerlerini alabilmişlerdir.

Yazılım sektörünün özelliği gereği işletmelerde oldukça fazla entelektüel sermaye birikimi oluşmaktadır. Bu yüzden işletmenin ürünleri markalaşma ile beraber piyasada daha değerli olmaktadır. Piyasa değeri üzerinden yapılan sıralamaya göre yazılım firmaları yukarılara doğru tırmanmaktadır. Öyle ki listedeki tüm firmaların varlık değerleri piyasa değerlerinin altındadır; bu durum yazılım firmalarının katma değerini açıkça ortaya koymaktadır. Fark edileceği üzere piyasa değerine göre firmaların sıralamadaki yerleri ciddi anlamda yükselmektedir. En çarpıcı farka sahip olan firma “Salesforce.com” dur. Eldeki verilere göre bu firma piyasa değeri ile varlık değeri arasında en fazla sıralamada fark yaratan işletmedir. Üstelik 1999 yılında kurulmasına rağmen bu başarıyı sağlamıştır. İşletme; müşteri ilişkileri yazılımları ile bulut bilişim alanında faaliyet göstermektedir.

Tablo 1’deki işletmelerin yaşları da sanayi firmalarının yanında çok genç kalmaktadır. Forbes’in araştırma listesinde sanayi firması olup çok yaşlı firmalar da bulunmaktadır. Buna rağmen bu gibi tablolarla karşılaşıyorsa ürettikleri entelektüel sermaye birikimi ve katma değer gerçekten yadsınamaz bir etki oluşturmaktadır. Yazılım sektörü, ülke ekonomilerine ihracat yolu ile ciddi katkılarda bulunan nitelikli ve katma değeri yüksek bir sektördür.

Bir başka dikkat çekici durum ise yazılım firmalarının ülkelere göre dağılımıdır. Tablo 1’deki 17 firma incelendiğinde 12 tanesinin ABD’ye ait olduğu görülmektedir. Bunun dışında Fransa, Almanya, Hindistan, İsrail ve İspanya’ya ait birer tane firma bulunmaktadır. Bu firmaların yazılım ihracı ile ülke ekonomisine olan katkıları da üst seviyelerdedir.

Tablo 1: Piyasa Değeri En Fazla Olan Yazılım Firmaları

Firma Adı	Ülkesi	Sıralama*	Varlık	Sıralama**	Piyasa Değeri	Kuruluş
Microsoft	ABD	176	\$153.5 M	4	\$343.8 M	1975
Oracle	ABD	289	\$86.6 M	25	\$185.0 M	1977
SAP	Almanya	634	\$37.3 M	72	\$97.1 M	1972
Vmware	ABD	1533	\$12.3 M	206	\$48.2 M	1998
Salesforce.com	ABD	1867	\$9.2 M	288	\$35.2 M	1999
Adobe Systems	ABD	1737	\$10.2 M	325	\$32.8 M	1982
Intuit	ABD		\$4.7 M	495	\$22.4 M	1983
Amadeus IT Holdings	İspanya		\$7.5 M	586	\$18.9 M	1987
HCL Technologies	Hindistan	819	\$4.2 M	677	\$16.6 M	1991
Dassault Systemes	Fransa		\$5.8 M	758	\$14.8 M	1981
Fiserv	ABD	1798	\$9.7 M	772	\$14.6 M	1984
CA	ABD	1592	\$11.8 M	795	\$14.1 M	1974
Symantec	ABD	1442	\$13.3 M	804	\$14.0 M	1982
Check Point Software	İsrail		\$4.9 M	857	\$13.1 M	1993
Citrix Systems	ABD		\$5.2 M	1007	\$10.7 M	1989
Amdocs	ABD		\$4.9 M	1406	\$7.4 M	1982
VeriSign	ABD		\$2.7 M	1443	\$7.2 M	1995

Kaynak: (Forbes, 2014, Mayıs)

* Toplam Varlık cinsinden sıralamadır.

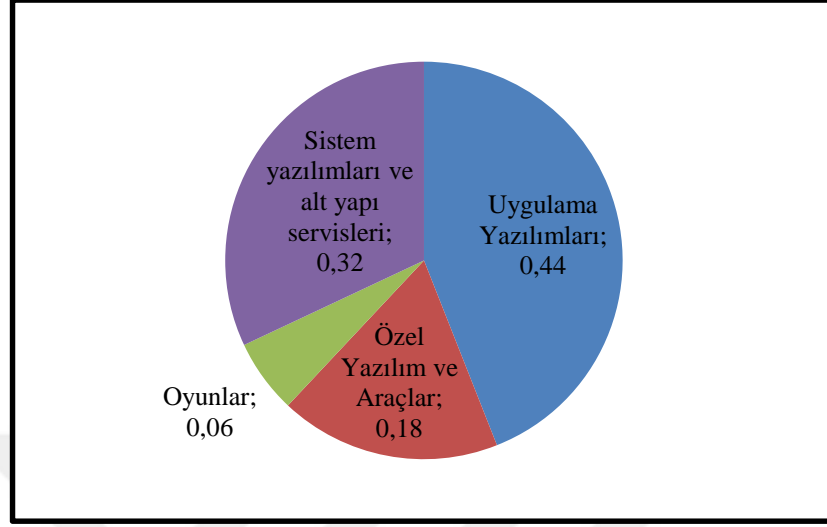
** Piyasa Değeri cinsinden sıralamadır.

Şekil 1’de görüldüğü üzere yazılım endüstrisini oluşturan yazılım tiplerinden satış geliri 2009 yılında en fazla olan %44 sektör payı ile “Uygulama yazılımları” olmaktadır. Daha sonra %32 sektör payı ile “Sistem yazılımları ve alt yapı servisleri”, %18 sektör payı ile “Özel Yazılım ve Araçlar” ve son olarak da %6 sektör payı ile “Oyunlar” gelmektedir. (Fois ve Lysonick, 2012: 3)

Uygulama yazılımları işletmelerin iş süreçlerini rahatça ve otomatik bir şekilde yerine getirebilmeyi amaçlar. Bu yazılımlar son kullanıcıya yönelik hazırlandığından kullanıcı dostu olarak tasarlanırlar ve kullanımları son derece kolaydır. Yönlendirmeler ve yardım menüleri bulunur.

Sistem yazılımları; donanım linklerini, ağ ve yazılımı yönetir. Bununla beraber alt yapı servislerinin temelidir. Özel yazılımlar ve araçlar; genelde sistem yazılımları ile uygulama yazılımları arasında iletişimi sağlamak üzere geliştirilirler. Oyunlar ise

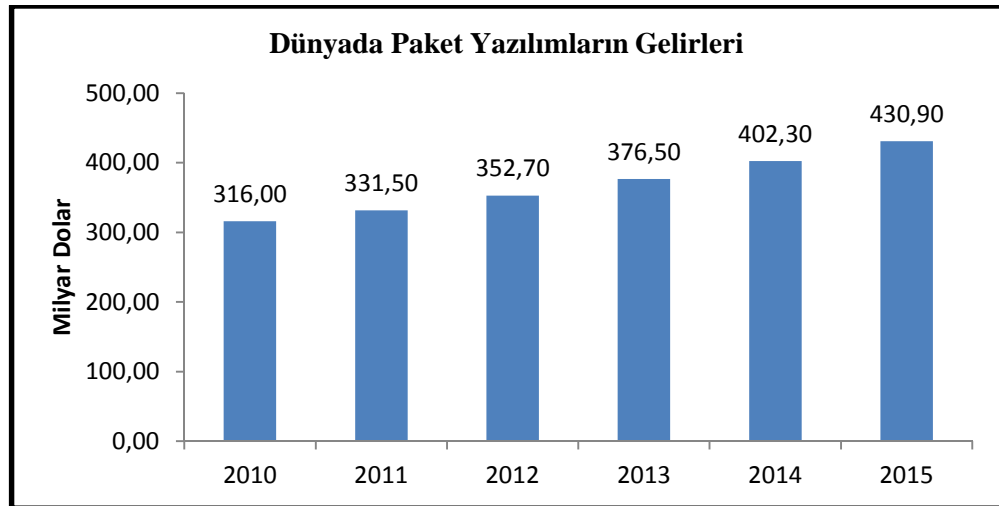
kullanıcı ara yüzü ile görsel geribildirim sunarak video gösterimi sağlayan yazılımlardır.
(Fois ve Lysonick, 2012: 3)



Şekil 1: Yazılım Tiplerinin Dünya Piyasasındaki Payları

Kaynak: (Fois ve Lysonick, 2012: 3)

Şekil 2’de yazılım endüstrisinin en çok gelirini oluşturan paket yazılımların yıllara göre gelirleri görülmektedir. Her yıl artarak devam eden trend, bu sektörün gelecekte daha da iyi performanslar göstereceği beklentisini doğurmaktadır.



Şekil 2: Dünyada Paket Yazılımların Gelirleri

Kaynak: (<http://www.statista.com/>, 20014)

OECD ülkeleri ekonomileri arasında yapılan araştırma sonuçlarına göre bu ülkelerde yapılan yazılım yatırımları, ülkelerin ortalama Gayri Safi Yurtiçi Hasıla'sının %0,5 ile %2,7 arasında yükselmesine neden olmaktadır.

Yazılım ihracatı konusunda başarı öyküsü hâline gelmiş, 3 I ülkeleri olarak adlandırılan İsrail, İrlanda ve Hindistan'ın yazılım ihracatında başarılı olmasının temel bir takım nedenleri bulunmaktadır. Uluslararası alanda yapılan akademik çalışmalara da konu olan bu başarı öykülerinin temelinde ulusal yazılım üretimi konusunda belirlenen ve kararlılıkla uygulanan stratejiler yatmaktadır. Stratejilerin ülke gerçeklerini, ülkelerin rekabet avantajlarını, ulusal yazılım sektörünün beklentilerini ve küresel trendleri çok iyi şekilde tahlil ettiği, yapılan çalışmalarda ortaya konulan bir gerçektir. (Arora ve Gambardella, 2005: 26-78) Özellikle stratejilerin siyasi sahipliğinin üst düzeyde olması ve stratejilerin uygulaması için kamu özel sektör karakterli yapılara yetki verilmesi de bu başarının kritik faktörleri olarak gösterilmektedir. (Alican, 2006: 150) Yazılım ihracatında 3 I olarak anılan Hindistan, İrlanda ve İsrail'in başarılı olmalarının bir diğer gerekçesi de ortaya konulan ulusal stratejinin bu stratejiyi bütünleyen aksiyon adımlarıyla desteklemesidir. Bu aksiyon adımları, yazılım sanayinin alt yapı ihtiyaçlarının karşılanması, yazılım sanayinin birlikte ve rekabetçi olarak üretim yapacağı ekonomik ortamın yaratılması ve yazılım sanayi tarafından üretilen yazılımların dış pazara sunulması, pazarlanması için gerekli olan koşulların ve talebin yaratılması olarak sıralanabilir. Bu başarıyı yakalamış ülkelere olan Hindistan'ın 2008 yılı yazılım ihracatının 47 milyar dolar olduğunu ve bu ihracat rakamının bir önceki yıla göre %36 artış gösterdiğini belirtiyor olmak; 3 I ülkelerinin yakaladığı başarıyı sayısal olarak kavrayabilmek bakımından da anlamlıdır.

Dünya genelinde ülkeler, yazılım ihraç gücü bakımından dört katmanlı bir sınıflandırmaya tabi tutulmuştur. (Carmel, 2003: 6) *“OECD ülkeleri ve 3 I' leri kapsayan birinci katman ülkeleri hâlen yazılım endüstrisinde de güçlü konumlarını muhafaza etmektedirler. İkinci katmanda sadece Rusya ve Çin bulunmaktadır. Üçüncü katmanı oluşturan ülkelerde ise yazılım endüstrileri gelişme aşamasındadır. Yazılım alanında ciddi atılımlar yapan veya bunun hazırlığı içinde olan Orta ve Güney Amerika, Uzakdoğu ve Doğu Avrupa ülkeleri bu katmandadır. Ayrıca Brezilya, Meksika, Romanya, Filipinler, Kore gibi ülkelerin yazılım ihraç edebilen başarılı firmaları ve firma kümelenmeleri bulunmaktadır. ”Başlangıç” aşamasındaki ülkelerin*

yer aldığı dördüncü katmanda ise Küba, Mısır, Vietnam, Endonezya bulunmakta ve bu ülkelerde firmalar küçük, yabancı yatırımlar ve pazarlar yetersiz durumdadır. Türkiye’de bu sınıfa dâhil edilmektedir. 2007 yılı itibariyle bu katmandaki ülkelerin durumunda belirgin bir gelişme olmamıştır. 3. ve 4. katmandaki ülkeler için bir üst katmana geçiş hâlâ zor görünmektedir. Bunun başlıca nedenleri şöyle ifade edilebilir”:

(Okur, 2007: 653-654)

- Yazılım endüstrisinin hedef / müşteri kitlesini oluşturan ülke ve firmalarla dil ve kültür uyumsuzluğu.
- Eğitilmiş insan gücünün nicelik ve nitelik olarak yetersizliği.
- Uzmanlaşmanın yetersiz olması nedeniyle yeni tasarım, marka ve ürünler oluşturulamaması.
- Rekabette en önemli dayanağın ucuz işgücü ve hizmet arzı olması.
- Yürütülmekte olan işlerin genellikle proje temelli olması ve bunların rakiplere kolaylıkla kaybedilebilmesi
- Ülke içinde yeterli devlet desteği ve kurumsal destek olmaması

Görüldüğü gibi bu faktörler esas olarak yapısaldır ve ancak dikkatli ve uzun vadeli politikalarla olumlu yönde değiştirilebilirler. Sektöre ilişkin son değerlendirmelere göre yazılım endüstrisi alanında varlık göstermeye başlayan, gelişme şansı olan bölgeler ve ülkeler şöyle sıralanabilir: (Okur, 2007: 654)

Asya: Çin, Güney Kore, Filipinler, Malezya, Vietnam, Endonezya, Tayland, Singapur

Doğu Avrupa: Macaristan, Romanya, Bulgaristan, Çek Cumhuriyeti, Polonya, Ukrayna.

Orta Doğu: Ürdün, İran, Türkiye

Afrika: Güney Afrika Cumhuriyeti, Mısır, Gana

Orta ve Güney Amerika: Meksika, Brezilya, Arjantin, Şili, Kosta Rika

1.1.4. Türkiye’de Yazılım Sektörü

Yazılım sektörü, ilk kez “6. Beş Yıllık Kalkınma Planı (1990-1994)” içerisinde öncelikli sektör olarak belirlenmiştir. 2000 yılında TBMM’de kabul edilen VIII. Beş

Yıllık Kalkınma Planına “Yazılım başta olmak üzere bilgi teknolojileri, Ar-Ge ve teknoloji geliştirme faaliyetleri desteklenecektir.” ifadesi ile bir yasa maddesi olarak girmiştir ve bu madde, ülkemizde yazılım sektörünün desteklenmesinin yasal zemininin meydana getirilmesinde atılan ilk adım olarak değerlendirilmektedir.

“Yazılım sektörünü besleyen ve insan kaynağını oluşturan genç nüfus ülkemiz için önemli bir avantajdır. Ülkemizde ortalama yaş 28,3 iken, 25 yaş altı nüfusun genel nüfusa oranı % 51’dir. Yazılım sektörünü satın almalarıyla besleyecek iç pazar dinamikleri bakımından konuya bakıldığında, ülkemizde bulunan 1,5 milyon KOBİ önemli bir pazar oluşturmaktadır.” (Batı Akdeniz Kalkınma Ajansı, 2012: 16)

“Türkiye İstatistik Kurumu’nun 2002 yılında yaptığı çalışmaya göre sektörde 1300 civarındaki yazılım firmasında yaklaşık 7500 kişinin çalıştığı belirlenmiştir. Sektörde çalışan sayısının 2006 yılında 35 bin kişi, firma sayısının ise 3200 olduğu tahmin edilmektedir.” (Türkoğlu, 2006: 10) Türkiye’deki genç nüfusun adeta doğar doğmaz teknoloji ile tanışması ve mevcut 35 yaş altı nüfus; bilişim teknolojisi ürünlerinin hedef kitlesini oluşturmakla beraber bu alandaki yeniliklerin geliştirilebilmesine sebep olmaktadır. Bu genç nüfus bilişim firmalarındaki çalışanların demografik özelliklerine de yansımaktadır.

“Yazılım sektörü diğer sektörlerle karşılaştırıldığında en fazla genç nüfusun istihdam edildiği sektör olarak da karşımıza çıkmaktadır. Yazılım sektöründeki gelişme aynı zamanda ülkenin işsizlik oranında da önemli bir azalma sağlamaktadır. Yazılım sektörü kadınların ve engellilerin istihdamında da önemli imkânlar sunmaktadır.” (Tiryakioğlu, 2012: 10)

Bilişim sektöründeki işveren-çalışan arasındaki problemin daha net anlaşabilmesi için webrazzi.com sitesi bir anket yayımladı. Yaklaşık 20 soru bulduran anket, sektöre en güncel verileri sunmaktadır. Buna göre; “% 3,4 oranla kadın yazılımcı veya tasarımcı bulunduğu sektöre % 96,5 oranla erkekler egemendir. Ankete katılan bilişim sektörü çalışanlarının % 58,1’i 25-32 yaş aralığında bulunuyor. İkinci sıradaysa % 30,9 ile 18-24 yaş grubu bulunuyor. Sektörde 33-40 yaş aralığında çalışanların oranı % 11,02 iken, 44 yaş üzerinde bulunanların oranı % 0,7 olarak sonuçlara yansıyor.” (<http://webrazzi.com/>, 2012)

“Dünya genelinde donanım harcamaları, tüm bilgi teknolojileri harcamasından % 39 pay alırken; bu oran Türkiye’de yaklaşık % 81 seviyesindedir. Bilgi teknolojileri

pazarının % 19'u ise yazılım ve hizmetlerden oluşmaktadır. Donanım ağırlıklı bu yapının, önümüzdeki yıllarda, yazılım ve hizmetler alanında daha fazla büyüyerek değişmesi gerekmektedir. TÜBİTAK-MAM verilerine göre; Türkiye'de yaklaşık 1.600 adet yazılım üreten yerli firma yer almaktadır, sektör şirketleri KOBİ yapısındadır ve ölçükleri küçüktür, sermaye yapıları güçlü değildir. Bu firmaların yaklaşık % 35'i, teknoloji geliştirme merkezlerinde yer almaktadır ve % 87,2'si KOBİ yapısındadır. Diğer bir ifadeyle; % 51'i 10 kişiden az, % 35,7'si 10 – 50 kişi arasında, % 9,8'i 50–250 kişi arasında ve % 3'ü ise 250 kişiden fazla işgücü istihdam etmektedir.” (Tiryakioğlu, 2012: 12) Bu gibi yapısal durumların getirdiği sorunlar nedeniyle Türkiye'nin yazılım alanındaki beklenen ihracat hacimlerine ulaşması pek gerçekçi olamaz. Bu sebeple son zamanlarda bu alanla ilgili bazı hamleler görülmektedir.

“Ulaştırma, Denizcilik ve Haberleşme Bakanlığı'nun 5809 sayılı Elektronik Haberleşme Kanunu'nun 5. maddesine göre, elektronik haberleşme sektöründe araştırma, geliştirme projelerinin teşvik edilmesi ve desteklenmesine ilişkin yönetmelik yakın zamanda uygulamaya geçecektir. Kalkınma Bakanlığı hazırlamakta olduğu 10. Kalkınma Planı ile 2023 yılına kadar olan hedefleri belirleyeceklerini ve 2. Bilgi Toplumu Stratejisi ile bazı sektörel hedefleri koyacağını belirtmektedir. Bilim, Sanayi ve Teknoloji Bakanlığı da teknoloji geliştirme bölgeleri kuracağını ve Ar-Ge faaliyetlerini destekleyeceğini söylemektedir. Ekonomi Bakanlığı da yazılım ihracatı ile ilgili çalışmalar yapmaktadır.” (Tiryakioğlu, 2012: 14)

Ekonomi Bakanlığı verilecek teşvikleri kategorize ederek ve Türkiye'yi bölgelere ayırarak hangi bölgedeki hangi kategoriye hangi alanlarda teşvik verileceğini ilan etmiştir. Buna göre genel anlamda teşvikte bulunulacak konular özetle şunlardır: KDV istisnası, gümrük vergisi muafiyeti, vergi indirimi, faiz desteği, yatırım yeri tahsisi, sigorta primi işveren hissesi desteği, sigorta primi desteği, gelir vergisi stopaj desteği ve KDV iadesi. (TOBB Türkiye Telekomünikasyon Meclisi, 2012: 7)

Tablo 2'de Türkiye'nin yazılım ihracatı yaptığı ülkeler ve dolar cinsinden büyüklükleri gösterilmiştir. Yazılım ihracatının toplam gelirleri durağan vaziyettedir. Bununla birlikte 70 civarında ülkeye ihracat gerçekleştiren Türk yazılım sektörünün ihracat gelirlerinin artarak devam edeceği beklenmektedir. Buna rağmen yazılım ihraç gelirlerinin yazılım ithalat giderleriyle (Tablo 3) karşılaştırıldığında arasında epey bir fark olduğu gözlemlenmektedir.

Tablo 2: Türkiye'nin Ükelere Göre Yazılım İhracatı (\$)

Ülke	2007	2008	2009
ALMANYA	3.735.689	2.431.523	1.549.775
ABD	1.317.092	277.638	1.541.124
BAE	225.489	59.190	1.098.833
LİBYA	37.562	646.468	981.631
İNGİLTERE	760.873	1.046.493	966.618
AZERBAJCAN	157.914	594.064	744.252
İSVİÇRE	9.875	374.527	524.284
TÜBİTAK SER.BÖL.	98.476	153.848	520.032
İST.DERİ SER.BÖL	275.717	207.051	503.599
ROMANYA	190.386	521.150	501.440

TÜRKMENİSTAN	14.490	99.174	480.114
İST.ATATÜRK HAVALİMANI SER. BÖL.	400.055	372.148	410.034
YUNANİSTAN	1.238.351	904.060	355.058
İSRAİL	39.306	117.254	343.888
KUZEY KIBRIS T.C.	317.171	210.815	315.658
BULGARİSTAN	37.168	2.006	313.909
ARNAVUTLUK	70.313	222.317	268.425
HOLLANDA	596.128	482.660	256.446
KAZAKİSTAN	917.285	1.036.474	154.784
EGE SERBEST BÖL.	128.899	137.545	141.187
DİĞER	3.753.973	2.993.579	930.290
TOPLAM	14.322.212	12.889.984	12.901.381

Kaynak : Dış Ticaret Müsteşarlığı

Tablo 3: Türkiye'nin Ükelere Göre Yazılım İthalatı (\$)

Ülke	2007	2008	2009
İRLANDA	21.155.756	26.266.744	22.372.737
ALMANYA	15.123.090	15.511.604	14.448.598
A.B.D.	13.592.340	15.896.238	12.118.732
AVUSTURYA	2.299.611	4.463.492	10.405.503
GÜNEY KORE	203.013	167.608	8.152.851
ÇİN	1.797.112	10.073.683	6.855.636
İNGİLTERE	7.985.323	10.016.455	5.470.210
HOLLANDA	2.242.848	2.200.141	3.551.743
ÇEK CUMHURİYETİ	1.169.195	948.351	3.144.140
JAPONYA	2.555.125	2.090.300	2.976.373
İST.ATATÜRK HAVALİMANI SER- BEST BÖLGESİ	7.504.550	8.613.290	2.444.675
FRANSA	3.458.542	2.852.330	2.350.220
KANADA	842.320	1.121.563	1.869.538
TÜBİTAK SER.BÖL.	526.069	568.237	1.712.496
BELÇİKA	516.181	336.296	1.647.720
İSTANBUL DERİ SERBEST BÖLGESİ	1.463.360	1.933.661	1.544.561
POLONYA	1.536.473	1.454.366	1.407.918
İTALYA	836.134	1.390.197	1.360.179
FİNLANDİYA	530.143	970.602	1.064.752
MEKSİKA	161.341	411.623	845.444
İSVİÇRE	946.027	586.473	829.220
MACARİSTAN	61.947	167.462	753.275
İSRAİL	1.418.955	3.373.192	619.333
DİĞER	20.550.562	6.448.430	2.914.747
TOPLAM	108.476.017	117.862.338	110.860.601

Kaynak : Dış Ticaret Müsteşarlığı

Türk yazılım sektörü, diğer yüksek teknoloji sektörlerinde olduğu gibi, net ithalatçı konumundadır. Sektörün ithalatı 2009 yılında 110 milyon doları aşmıştır. İhracat ve ithalat yapılan ülke grupları incelendiğinde ihracatımızın ülke çeşitliliğinin arttığı ancak, ihracat gelirlerinin küçük kaldığı görülmektedir. Türkiye, en fazla AB üyesi ülkelerden yazılım ithal etmektedir. (Batı Akdeniz Kalkınma Ajansı, 2012: 14)

Ülkemizde yazılım firmaları tek bir devlet kurumuna bağlı olmadıkları için ortada gelecek dönemler için planlanmış hedefler ve koordinasyon süreçleri bulunmamaktadır. Hâli hazırda piyasadaki işletmeler çeşitli dernek ve vakıf çatısı altında organize olmaya çalışmaktadır. Sektörel katkı ile ayakta duran bu sivil toplum kuruluşlarının sektörün gelişmesi ve büyümesi için gösterdiği çabalar dağınık ve yetersiz kalmaktadır. Bu dernek ve vakıfların büyük çoğunluğu ise bilişim sektörünü bir bütün olarak ele almaktadır. Türkiye’de belli başlı öne çıkan sivil toplum kuruluşları ve dernekler tablo 4’de gösterilmiştir.

Tablo 4: Bilişim ve Yazılım Alanındaki Sivil Toplum Kuruluşları

Sivil Toplum Kuruluşu	Web Adresi
Yazılım Sanayicileri Derneği	www.yasad.org.tr
Bilişim ve Yazılım Eser Sahipleri Meslek Birliği	www.biyesam.org.tr
Bilişim Sektörü Derneği	www.tubider.org.tr
Bilişim Sanayicileri Derneği	www.tubisad.org.tr
Bilişim Derneği	www.tbd.org.tr
Bilişim Vakfı	www.tbv.org.tr
Türk Teknoloji Geliştirme Vakfı	www.ttgiv.org.tr

Kaynak: (Arpacı, 2010: 31)

Bir diğer konu ise yazılım ürünlerinin telif hakları ile ilgilidir. Patent ve fikri mülkiyet haklarının tanımı, uygulaması ve adil kararlara bağlanması yönünde ciddi eksiklikler bulunmaktadır. Mevcut mevzuatta bilişim ürün ya da hizmetlerinin anlamlı biçimde lisanslanmasına olanak verecek iyileştirmeler yapılmalıdır.

Telif hakları ve tescil kanun ve yönetmelikleri genellikle mamul hâle gelmiş yazılım ürünlerine hitap etmektedir. Yazılım ürünlerini kapsayan, yeni yürürlüğe sokulmuş olan İsteğe Bağlı Tescil Yönetmeliği, fiziki koşulları ve kaynak kodunu korumak konusunda atılmış ilk önemli adım olmakla birlikte ihtiyaçları karşılamaktan uzaktır. Yazılım ürünü, TOBB kanununda sanayi ürünü kabul edilmekteyken, ihracat mevzuatında hâlâ

hizmet olarak kabul edilmektedir. Bu durum yazılım sektörünün ihracat teşviklerinden faydalanmasını engellemektedir. (Türkiye Bilişim Derneği, 2013: 17)

1.1.5. Yazılım Sektöründen Genel Beklentiler

Türkiye’deki yazılım sektörünün bağlı olduğu otoriter bir makam olmadığından sektörün yönetilmesi istendiği gibi olmamaktadır. Hükümet tarafında İnternet ve Bilişim Araştırma Komisyonu, Ulusal Siber Güvenlik Koordinasyon Kurulu ve Ulusal Bilgisayar Olaylarına Müdahale Organizasyonu gibi yapılanmalar mevcuttur. Ancak istenilen etkin yönetim için sektörün beklentisi, Bilişim Bakanlığı’nın kurulmasıdır. Bununla birlikte aşağıdaki talepler yapılmaktadır; (Türkiye Bilişim Derneği, 2013: 21)

- *Bilişim Teknolojisi stratejisi, büyüme stratejisinin ayrılmaz bir parçası olarak kabul edilmeli*
- *Küresel rekabetçi Bilişim Teknolojileri için hizmet ve yazılım ihracatı desteklenmeli*
- *Rekabetçi, yaygın ve ucuz iletişim altyapı ve hizmetleri sağlanmalı*
- *Bilişim Teknolojilerinin iş dünyasına nüfuzu artırılmalı*
- *Girişimci kültürün güçlendirilebilmesi için girişim sermayesinin oluşumu hızlandırılmalı*
- *Bilişim Teknolojilerinden alınan vergiler büyümeyi teşvik edecek, derinleşmeyi sağlayacak ve ihracatı güçlendirecek şekilde, değiştirilmeli ve vergi yükü azaltılmalı*
- *Görsel ve işitsel politikalarla ilgili yasal düzenlemelerde uyuma devam edilmeli ve “Sınırsız Televizyon Yönergesi ”ne uyum sağlanmalı*
- *İnternet içeriği ve İnternet hizmet sağlayıcılarının bağlı olduğu kurallar ifade özgürlüğünün korunmasıyla ilgili uluslararası standartlara uygun hâle getirilmeli.*

Türkiye’nin 21. yüzyılda bir dünya gücü olarak var olabilmesi ancak ve ancak kendi teknolojilerini üreterek uluslararası rekabet edebilen konumdaki bilişim sektörüne sahip olmasıyla mümkündür. Bunun için devletimizin bilişim sektörünü stratejik sektör olarak tanımlaması ve ölçülebilir hedefleri belirlemesi gerekmektedir. Türkiye’nin 2023 hedeflerinde, bilişim sektöründe % 50 yerli ürün kullanılması, bilişim sektörünün

GSYH'nın % 8'ine ulaşması, küresel pazarda söz sahibi en az bir firma olması gibi hedefleri vardır. 2023 yılında 500 Milyar dolar ihracat hedefi ve 10 Milyar dolarlık bölümünün yazılım sektörüne ait olmasının hedeflendiği noktada, 2012 yılından itibaren, sektörün her yıl ihracatını ve boyutunu % 31 kümülatif artırması gerekecektir. Diğer sektörlerle göre daha hızlı büyüyen bir sektör olsa da, bu hedeflere ulaşmak için yazılım sektöründe köklü ve radikal değişikliklere ihtiyaç olduğu açıktır.

“Yazılım üretim sektörünün gelişmesini desteklemek üzere TÜBİTAK/TEYDEB, KOSGEB gibi kuruluşlar eliyle verilen Teşvikler ve Teknoparklar, bugünkü Türk yazılım üretim gücünün oluşmasında önemli katkı sağlamıştır. Ancak bu teşviklerle üretilen üstün nitelikli yazılım ürünlerinin kamu alımlarında tercih edilmemesi, bu birikimin sürekliliğine ket vurmaktadır. Oysa ne ABD, Almanya gibi ileri ülkelerde, ne Brezilya, Rusya, Hindistan, Çin (BRIC) ya da Güney Kore gibi, bu alanda atılım yapan rakiplerimizde, ne de Türkiye’de, yazılım ürünlerinin, ilkin o ülkenin kamu kesiminde yaygın biçimde kullanılmadan küresel pazarlara açılabilirdiği bir örnek yaşanmamıştır.” (TBD Genel Kurul Kararı, 2014: 58)

Tablo 5, Global Software Top 100 Listing (2012) raporundan özetlenmiştir. Yazılım gelirleri açısından en büyük 100 yazılım firmasının ülkelere göre sıralanışıdır.

Tablo 5’de en alt satırda yer alan “diğer” başlığı; Fransa, Rusya, Brezilya, Belçika, Norveç ve İsviçre ülkelerinin birer tane firmasını ifade eder. 2012 yılı verileri incelendiğinde Türkiye’den herhangi bir firmanın olmadığı anlaşılmaktadır.

Tablo 5: En Fazla Geliri Olan 100 Yazılım Firmasının Ülkelere Göre Dağılımı

Ülke	Firma Sayısı
Amerika	67
İngiltere	7
Almanya	6
Japonya	4
Kanada	2
Çin	2
İsrail	2
Hollanda	2
İsveç	2
Diğer	6

Kaynak: (Price of Water House Coopers, 2012)

Bunun yanında en az yazılım geliri ile tabloya girmeyi başaran firma; 335 milyon dolar ile İsviçreli bir firma olmuştur. Türkiye'nin 2023 yılı hedefleri arasında küresel piyasada en az bir firmanın olması hedefi bulunduğuna göre en fazla yazılım geliri olan yerli firmaları gerekmektedir. Her yıl Interpromedya firmasının “Bilişim 500” anketi ile Türkiye’de çok çeşitli kategorilerde firmaların pazar içindeki durumları ortaya konmaktadır. 2012 yılı anket sonucuna göre tablo 6 hazırlanmıştır.

Tablo 6’ya göre birinci olan Turkcell Teknoloji, 127.672.851 TL / 1,78 (2012 yılsonu \$/TL kuru) = 71.726.320 \$ etmektedir. Global Software Top 100 Listing (2012) raporuna göre en sonuncu olan firmanın yazılım geliri 335 milyon \$’dır. Böylece Türkiye’nin en fazla yazılım geliri olan firmanın 4,5 katından fazla olduğu sonucuna varılmaktadır. Tablo 6 dikkatlice incelendiğinde ilk 5 firmanın yazılım gelirleri de birbirlerine oldukça uzaktır. Bu durum da göz önüne alındığında küresel bir yazılım firmasının Türkiye’den çıkması için yapılması gereken daha çok iş vardır.

Tablo 6: Yerli Yazılım Üreticilerinin Yazılım Gelirleri

Sıra	Firma Adı	Gelir (TL)
1	Turkcel Teknoloji	127.672.851
2	Sofitech	88.592.451
3	Fintek	46.325.438
4	Arvento	35.849.286
5	Fonet	27.158.267

Kaynak: 2012 Yılı Bilişim 500 Anketi

“Yazılım sektörünün gelişebilmesi için özel sektör, STK’lar, üniversiteler ve kamu arasındaki işbirliği son derece önemlidir ve bu paydaşlara önemli görevler düşmektedir. Bu bağlamda devlete de görevler düşmektedir. Dünya örnekleri incelendiğinde, yazılım sektörüne devlet tarafından sağlanan katkılar Devlet Doğrudan Destekleri, Devlet Teşvikleri ve Devlet Dolaylı Destekleri olarak üç ana başlıkta gruplanmaktadır.” (Tiryakioğlu, 2012: 14)

- 1) *Devlet Doğrudan Desteklerini; devletin, yazılım sektörü şirketlerine sağlayacağı geri ödemesiz (hibe) veya geri ödemeli (faizsiz-düşük faizli, belli bir dönem geri ödemesiz uzun vadeli kredi) nakdi katkılar olarak,*
- 2) *Devlet Teşviklerini; kurumlar vergisi, gelir vergisi, stopaj, SSK primi, SSK işveren katkısı ve KDV’de uygulanacak istisnalar/ertelemeler/azaltmalar olarak,*

3) *Devlet Dolaylı Desteklerini; yazılım satışlarının artırılmasını sağlayacak pozitif ayrımcılık da dâhil olmak üzere uygulamaya konacak enstrümanlar; KOBİ'lere yazılım satın almalarına ilişkin sağlanacak KOSGEB destekleri, kamu satın almalarında yerel yazılım üreticilerinin ürünlerine öncelik/avantaj sağlanması vb. olarak, tanımlayabiliriz. Bu tür destek ve teşvikleri İrlanda, İsrail, Hindistan, Çin, Malezya, Tayvan ve Brezilya gibi ülkeler uygulamaktadırlar.*

Sektöre uygun özel yasa ve düzenlemeler getirilmesi; hızla gelişen teknolojiye göre hukuki düzenlemelerin, örneğin kişisel verilerin korunması yasa tasarısının kanunlaşması; bilişim hizmetlerinde standardizasyon ve sertifikasyon uygulanması gibi konular da sektörümüzün gelişiminin hızla önünü açmasını sağlayacaktır.

“ABD, dünya genelinde yazılım sektörüne en fazla AR-GE bütçesi ayıran ülke dolayısıyla da yazılım pazarında lider ülke konumundadır. ABD, teknoloji üreten ülkeler konumunda ilk sıradadır. ABD, Türkiye için bu konuda çok güzel bir model. Türkiye'de sektörün 2023 hedefi, pazarı 160 milyar dolar büyüklüğe ulaştırmaktır. Bunun için önemli adımlar atılıyor ancak kamunun da destek olması gerekir. Kamuda yerli yazılımın kullanımının desteklenmesi sektör açısından çok önemlidir. Sektörü stratejik sektörler kapsamına alıp bir strateji vizyonu geliştirilmelidir.” (Anık, 2014: www.bloomberght.com)

“İstihdam açısından bakacak olursanız sanayi sektöründe bir kişi istihdam etmek için 70 bin dolar yatırım yapmanız gerekiyor. Hâlbuki yazılım sektöründe işgücünün maliyeti ise çok düşük sadece 2 bin dolar. Buna karşın katma değeri en yüksek sektörlerden biridir.” (Güneş, 2014: www.bloomberght.com) Sektörün dinamik ve esnek yapısı kadınlara, engellilere ve gençlere istihdam imkânı sağlayabilmektedir. Bununla beraber sektörün aradığı nitelikli ara elemanın temini konusunda ciddi sıkıntılar çekilmektedir. Meslek lisesi ve üniversitelerin bu konuda kendilerini yenilemeleri gerekmektedir.

Yazılım sektörü özelinde bilişim endüstrisinde çalışanların mutlu olduğu, mutluluğun ortak bir paydada verimli çıktılara dönüştüğü, bu dönüşümden kazanılan maddi ve manevi katma değerlerin ise bilinçli bir şekilde paylaşıldığı bir iş örgütlenmesinin sağlanması gerekmektedir.

1.2. Yazılım Geliştirme Süreçleri

Bu bölümde yazılımların geliştirme süreçlerindeki işlemler açıklanacaktır. Böylece yazılıma katkı yapan süreçlerdeki harcamalar, proje gideri veya maliyeti olarak ileride sınıflandırılabilir.

1.2.1. Proje Kabulü ve İlk Planlama

Planlama; her işletme için elzem olduğu gibi yazılım geliştiren firmalar için de elzemdir. Hatta çok daha gereklidir. Çünkü yazılım firmalarının üretim kaynakları ve ürünleri soyuttur. Takip edilmeye daha çok muhtaçtır. Planlama, yönetimin fonksiyonlarından ilki olarak kabul edilmiştir. Çünkü bir iş yapılmadan evvel o işin nasıl, ne miktarda, nerde, ne zaman, ne kalitede yapılacağı belirlenmesi gerekir ki ondan sonra diğer yapılacaklar belirlenebilsin. Planlama geleceğe dair yapılan çalışmalardır. Hem olması istenen hem de beklenti ve kestirimlere göre kararlar verilir. Daima gerçekleşenler ile beklentilerin güncellenerek ileriye doğru tekrar oluşturulması (dinamik planlama) ile en etkin kararlar alınabilir. Etkin kararlar verimli bir yönetim demektir.

Planlama; her ne kadar işin en başında yapılsa da zaman ilerledikçe içerden (hatalı üretim, zaman kaybı, vs.) veya dışardan (müşteri isteklerinin değişmesi, vs.) kaynaklı olarak değişimler söz konusu olacağından belirli aralıklarla tekrarlanmalıdır.

Yazılım firmalarında proje yönetimi için planlama işlemleri çok hayatidir. “*Planlama, proje hedeflerine ulaşmak ve müşteri ihtiyaçlarını karşılamak için izlenecek yöntemi, yapılacak işleri, kullanılacak kaynakları ve iş takvimlerini proje kısıtlarını da dikkate alarak belirlemektir. Planda kimin neyi, ne zaman, nasıl ve ne maliyetle yapacağı yer alır.*” (Saridoğan, 2008)

Planlı, programlı, iyi belgelendirilmelerine rağmen bazı projeler beklenmeyen olay ve riskler yüzünden başarısız olabilir. Ancak plansız ve programsız başarılı olan neredeyse hiçbir proje mevcut değildir.

Yazılım projeleri, herhangi bir işlevi veya işlevler silsilesini yerine getirerek ihtiyaçları karşılayan ve sorunları çözmek için oluşturulurlar. Buna göre bir yazılım projesinin ele alınması için herhangi bir konuda ihtiyaca binaen bir talebin olması gerekmektedir.

Bunun akabinde de ihtiyacın detaylarının tanımlanması ve gerekli planlama işlemlerinin yapılması gerekmektedir. Bu tür proje başlangıçlarına “planlı istek” denir.

Diğer bir proje başlangıcı ise “plansız istek” olarak tanımlanan şekildir; ki talep edenin küçük istekleri devam eden bir başka proje içindir, ancak istekleri başka bir proje gerektirir. İsteklerin tanımlanmaması veya geç tanımlanmasından dolayı yazılım geliştirme süreçlerinin başlangıç aşamalarının gerçekleştirilmemiş olması proje maliyetini yükselten etki yaparak bir kaosa sebep olmaktadır.

Proje kabulü, bir yazılım geliştirme sürecinin ilk aşamasıdır. Müşteri isteklerinin dinlendiği ve yazılım ekibi ile müzakerelerin yapılarak genel anlamda anlaşmaların sağlandığı aşamadır. Buna uygun olarak proje başlangıçlarında müşteri isteklerini olabildiğince tanımlayıp kayıt altına almak gereklidir; ki planlama aşamasında rasyonel hesaplar yapılabilir. “İyi bir gereksinim dokümanı şu özelliklere sahip olmalıdır:” (Atbaş, 2012: 20)

- Doğru: Doğruluk önemlidir ancak güncel tutularak uzun süre doğru kalması gerekmektedir.
- Tartışmasız: Bir gereksinim herkes tarafından aynı şekilde yorumlanmalıdır. Taraflar (müşteri ve yazılım ekibi) farklı anlamlar çıkartmamalıdır.
- Tam: Yazılımın gerçekleşmesi için gerekli olan tüm konulara değinilmelidir.
- Tutarlı: Gereksinim dokümanı içinde her maddenin diğer maddeler ile uyumlu olması gerekmektedir. Gerek mantık gerekse yazım biçimi olarak uyum olmalıdır.
- Öncelik Sıralı: Gereksinimler gerçekleşme önceliklerine göre gruplandırılır ve sıralanırsa proje takvimi açısından çok değerli bir iş yapılmış olur. Önemli gereksinimler önce, daha az önemliler daha sonra gerçekleşir. Gereksinim dokümanında öncelik bilgisi de vermelidir.
- Doğrulanabilir: Gereksinim olarak doğrulanması güç, muğlak ifadeler kullanılmamalıdır.
- Değişikliğe uygun: Gereksinimler ve açıklamalar basit, ufak ve mümkün olduğunca birbirlerinden bağımsız olmalıdır. Ayrıca bir gereksinim birçok yerde

tekrarlanmamalıdır. Böylece gereksinim dokümanı değiştirilmek istendiğinde kolayca değiştirilebilmelidir.

- İzlenebilir: Gerek sistem gereksinimlerine gerekse testlere olan ilgileri açıkça belirtilmelidir. Bir yazılım gereksiniminin hangi sistem gereksiniminden doğduğu bilinirse gerçekleştirme daha tutarlı ve hatasız yapılabilir. Öte yandan hangi testin bu gereksinimi test ettiği bir yerde yazılı olursa testler esnasında gözden kaçan yani test edilmemiş gereksinim kalmaz.
- Gerçeklenebilir: Gereksinimlerin gerçekten Gerçeklenebilir olduğunun önceden düşünülmesi faydalıdır. Gereksinimler yazılırken bile bir ön tartışma yapıp bu gereksinimlerin eldeki imkânlarla başarılıp başarılamayacağı düşünülmelidir.

Eğer test adımlarını yazmakta zorlanıyorsanız büyük ihtimalle kalitesiz gereksinimleriniz vardır. Ayrıca gereksinim dokümanının aşağıdaki konularda bilgi vermesi gerekmektedir: (Atbaş, 2012: 22)

- İşlev: Yazılımın ne yapması isteniyor? Amacı nedir?
- Kullanım senaryoları: Modern gereksinim dokümanları insan-makine arasındaki kullanım senaryolarını da içerir.
- İş akışı: Hem modern hem de klasik gereksinim dokümanlarında akış senaryoları, cihazların iletişim protokolleri, mesajlaşma akışına dair bilgiler yer alır.
- Dış arayüzler: Yazılımın insanlarla olan ve diğer cihaz, sistem donanımı, diğer yazılımlarla olan arayüzü
- Performans: Hız, düzgün çalışma oranı, cevap verme süresi
- Yazılımın Özellikleri: Taşınabilirlik, doğruluk, bakımının yapılabilmesi, güvenlik vb. kriterler.
- Gerçekleme üzerindeki kriter ve kısıtlamalar: Uygulanan standartlar, uygulama yazılım dilinin adı, veri tabanı politikası, kaynak sınırlamaları, işletim sistemi ortamı
- İşe özel kurallar: Bu kısımda işin karakteristikleri, kendine has özellikleri anlatılır.

- *Diyagramlar: Gereksinim dokümanında sadece gereksinim maddeleri bulunmaz, ayrıca teknik detayları veya büyük resmi daha iyi anlamaya yarayan şemalar, şekiller, akış diyagramları da bulunmalıdır.*

“Planlama işleminde genel olarak proje hedeflerine ulaşmak için, belirlenmiş süre ve mali kısıtları dikkate alarak, izlenecek yöntem, yapılacaklar, kullanılacak kaynaklar ve süre takvimlerinin belirlenmesidir. İstekler ve ürün özellikleri ayrıntılara bölünerek yapılacak görevler ortaya çıkar. İhtiyaç duyulan kaynakları belirlemek, tedarik etmek ve görevlendirmek bunu takip eder.” (Nizam, 2014: 39)

Planlamanın hedeflediği alanlar ve buna bağlı yararlar aşağıda sıralanmıştır. (West, 2002: 9)

- Ürünün ortaya çıkacağı zaman ve aşamaları özet olarak tarif eder.
- Proje hedeflerini net şekilde ortaya koyar.
- Ekip, yönetici ve diğer paydaşlar arasında bir iletişim köprüsü oluşturur.
- Görevleri, görevler arası ilişkileri ve kaynak ihtiyacını gösterir.
- Proje yöneticisinin yapılanları kontrol edebilmesini sağlar.
- Kaynaklara ne zaman ihtiyaç duyulacağını gösterir.
- Kontrol noktalarını ve yapısını tanımlar.
- Projenin maliyeti hesaplanabilir.

Planlama aşamasında geçirilen zamanın maliyeti çok tartışmalara sebep olmuştur ancak projenin ilerleyen aşamalarında gözden kaçan bir talep veya uyarılma işleminin proje maliyetine olan etkisi daha fazla olmaktadır. Bu sebeple planlama aşaması projenin başlangıcı olmasına rağmen asıl yönetsel kararların alındığı çok önemli bir aşamadır.

1.2.2. Analiz

Analizden kasıt ihtiyaç analizidir. İhtiyaç veya beklenti; müşteri tarafından sistemin gerçekleştirmesini istediği temel görev veya fonksiyondur.

“Yazılım tasarımı yapmadan ve geliştirmeye başlamadan önce müşterinin ne istediği büyük ölçüde anlaşılmalıdır. Ancak genellikle bu ilk anda oldukça güçtür. Nasıl bir

yazılım geliştirileceğinin prototiplerle müşteriye gösterilmesi bu konuda yardımcı olabilir. Sürece dâhil olan herkes karşılıklı olarak birbirini anlamalıdır. Yazılımı geliştirenlerin müşteriye anlamaları önemli olduğu gibi müşterinin de yazılımı geliştirenlerin projeden ne anladığını ve ne yapacaklarını anlaması gereklidir. Neticede geliştirilecek projede ortak mutabakat gereklidir.” (Nizam, 2014: 39)

İhtiyaçlar, kabaca fonksiyonel ve fonksiyonel olmayan şekilde ikiye ayrılabilir. (International Institute of Business Analysis, 2006: 43) “Fonksiyonel ihtiyaç, kullanıcının sistemden doğrudan gerçekleştirmesini beklediği bir işlemdir. Fonksiyonel olmayan ihtiyaç ise projenin istenilen performans, güvenlik, süreklilik gibi kriterleri sağlaması için gerekli veritabanı, donanım, işletim sistemi gibi altyapıların seçimi ve konfigürasyonları tanımlar. Fonksiyonel olmayan bir ihtiyaç da sistemin temel amaçlarına hizmet eder, ancak kullanıcıyla doğrudan değil dolaylı ilişkidir.” (Nizam, 2014: 39)

1.2.3. Tasarım

“Analiz aşaması sonucunda belirlenen gereksinimlere yanıt verecek yazılımın temel yapısının oluşturulduğu aşamadır. Yazılım tasarımı, bir bileşen veya sistemin nasıl gerçekleştirileceğini belirlemek için kullanılan teknikler, stratejiler, gösterimler ve desenlerle ilgilidir. Bu aşama yazılım bileşenleri arasındaki içsel ara yüzler, mimari tasarım, veri tasarımı, kullanıcı ara yüzü tasarımı, tasarım araçları ve tasarımın değerlendirilmesi alt süreçlerini de kapsamaktadır. Tasarım aşaması, yazılımın hem kullanıcı ara yüzünü hem de programın omurgasını ortaya koymaktadır. Yapılacak tasarım, yazılımın işlevsel gereksinimlere uygun olmasının yanı sıra kaynaklar, performans ve güvenlik gibi kavramları da göz önüne alınarak gerçekleştirilmelidir.” (Özbilgin, 2010: www.bilgiguvenligi.gov.tr)

“Analiz ve tasarım süreçleri uygulamada karıştırılmakta ve sık sık tek adımda gerçekleştirilmektedir. Ancak aralarındaki kavramsal farklılıktan dolayı bu çok hatalı sonuçlara yol açar. Analizde kullanıcı isteklerinin alınmasını ve problemi öğrenmek amaçlanır. Tasarımda ise analiz sonucu ortaya çıkan ihtiyaçların, yazılım geliştirme araçlarına uygun çeşitli seviyelerde modelleri çıkartılır. Analizde yön anlayışı anlamaya, tasarımda ise çözüme dönüktür.” (Nizam, 2014: 42) Bu düşüncelerle

hazırlanmış çalışmalar ve belgelenmeler, daha planlı ve profesyonel yazılım geliştirme süreci yaşatır.

Tasarım sürecinde çeşitli etmenlere göre sistem yapısı kurgulanabilir. Tasarım yaparken en çok göz önüne alınan faktörler şunlardır; (Saridoğan, 2011: 55-56)

- Maliyet: Sistemi gerçekleştirebilmek için kullanılacak bilgisayar veya işlemci sayısı ve cinsi, sistem maliyetini çok önemli ölçüde etkiler. Var olan bir donanım üzerinde sistemi geliştirmek ayrı bir bilgisayar grubu oluşturmaktan daha hesaplı olabilir. Bazen de basit ve ayırık bir donanım ile sonuca ulaşmak daha pratik ve ucuz olabilir.
- Etkinlik: Sistemin tepkilerinin önem derecesi, yani işlev ve başarımı yazılım olduğu kadar donanımdan da etkilenir. En uygun başarımı sağlayacak donanım ve ona uygun yazılım mimarisinin seçilmesi paralel, çevrimiçi ya da gerçek zamanlı çalışma isterlerini etkin bir şekilde karşılayabilmek için gereklidir.
- Güvenilirlik: Sistemin hatasız ve arıza yapmadan uzun süre çalışabilmesi, sürekli devrede kalma zorunluluğu, bakım için dahi devreden çıkarılmaması gibi gereksinimler uygun donanım türünün ve yazılım mimarisinin seçilmesini gerektirir. Güvenilirliği arttırmak üzere, çalışırken parçaları değiştirilebilen (hot swap), yedekte paralel çalışan başka bir donanım veya yazılım ögesi olarak kullanabilen (hot standby) veya arıza anında otomatik devreye giren (cold standby) öğelerden yararlanan sistemler tasarlanabilir. Kritik veri depoları farklı düğümler hatta farklı coğrafi konumlardaki bilgisayarlar üzerinde çoğaltılabilir.
- Güvenlik: Sistemin güvenlik gereksinimleri, donanımın bir kısmının veya tamamının özel güvenlik önlemleri alınmış bölgelere yerleştirilmesini gerektirebilir. İşlemciler arası iletişimin güvenlik içinde yapılması, sistem donanım ve yazılımlarına erişimin yalnızca yetkili kişilerce yapılabilmesi, hem uygulama yazılımının hem de işletim sisteminin özel güvenlik yöntemlerini desteklemesi gerekmektedir.
- Özel kısıtlamalar: Müşteri tarafından kullanılması özellikle istenen donanım veya yazılımlar tasarıma etki eder. Sistemin kullanılacağı ortam koşulları (ısı, nem, toz, şok, titreşim, elektromanyetizma gibi) belirli bir donanım türünün kullanılmasını zorunlu kılabilir.

1.2.4. Kodlama

Kodlama aşaması, tasarım sürecinde ortaya konan veriler doğrultusunda yazılımın gerçekleştirilmesi aşamasıdır. (Yılmaz, 2007: <http://web.iku.edu.tr>) Bu süreç programlama çalışmalarının yanı sıra yazılımın geliştirilmesi ve kullanıcıya ulaştırılması sürecindeki bütün çalışmaları kapsar. Tasarım sonucu üretilen süreç ve veri tabanının fiziksel yapısını içeren fiziksel modelin bilgisayar ortamında çalışan yazılım biçimine dönüştürülmesi çalışması olarak da nitelendirilebilir.

Tasarımı tamamlanmış bir uygulama, programcıya ulaştığında, programcının yapacağı ilk iş, modülü iyi anlamak olmalıdır. Yazılacak modülü zihninde canlandıramayan programcı kodlamayı yapamaz. Tasarımda hazırlanan, kodlama detayları, iş akış diyagramları, tablolar ve tetikleyici kodların, programcıya yazılı olarak verilmesi ve yapılan işin anlatılması gereklidir.

Kaliteli kodlamanın taşınması gereken özellikler şunlardır;

- Basit ve anlaşılır olması
- Şekil (Yazım) kurallarına uyması
- Yardımcı fonksiyonlar ve özel kitaplıklarla sadeleştirilmesi
- Desteklenebilir olması
- Esnek ve değiştirilebilir olması
- Karmaşadan uzak ve sade olması

1.2.5. Test

Test aşaması, yazılım kodlanması sürecinin ardından gerçekleştirilen sınama ve doğrulama aşamasıdır. Elde edilen uygulama yazılımının hem belirlenen gereksinimleri sağlayıp sağlamadığı hem de gerçekleştirimin planlanana uygun olup olmadığını kontrol etmek için yapılan çalışmalardır. Bu kontroller sürecinde aslında iki bakış açısı ile duruma yaklaşılır, birincisi işlem “doğrulama” (verification) diğeri “onaylama” (validation) işlemidir. Doğrulama işlemi; düşünüp planlananın yapılıp yapılmadığının testi iken onaylama işlemi ise müşteri isteklerinin karşılanıp karşılanmadığının testidir. (Nizam, 2014: 42) (Yılmaz, 2007: <http://web.iku.edu.tr>) Test çalışmaları ve yaptıkları işlemler aşağıdaki şekilde özetlenebilir.

- Programcı testleri: Bu çalışmalar doğrulama işlemi gereği kodlama yapan ekibin kendi içinde yaptığı testlerdir. Daha çok teknik testlerdir. Bunlar arasında metot veya nesne seviyesinde yapılan birim testi, bileşenler arası entegrasyon testi, bütünleşik sistem testi, performans ve yük testi sayılabilir. Test neticelerine göre yazılımın tasarım ve kodunda değişiklikler yapılabilir.
- Test ekibinin testleri: Kodlama yapan ekibin dışındaki çalışanlardan oluşan bir takımdır ve yapılanların müşteri isteklerini karşılayıp karşılamadığının kontrolüdür ki böylelikle yazılıma onay verilir.
- Kullanıcı grubu testleri: Test sürecinde, en faydalı geri bildirimler, son kullanıcı test gruplarından gelir. Yazılımın beta testlerinde mutlaka müşteriden test grupları oluşturması istenmelidir. Bu sayede müşterinin yeni yazılıma adaptasyonu da sağlanmış olur.

Testlerin sonunda ilgili birimlerle birlikte değerlendirme toplantıları yapılmalıdır. Bazen hataların kaynağı analizde ya da tasarımda olabilir. Ya da kullanıcıdan gelen yeni bir istek var ise, bu talep doğrudan analiz ekibine iletilmelidir. Çünkü yazılımın mimarisinin temelleri analizciler tarafından hazırlanmıştır.

1.2.6. Devreye Alma

Test aşaması bittikten sonra kullanıcı eğitimi, pilot kurulum ve nihai proje kurulumu yapılarak proje devreye alınır. Yani piyasanın tabiri ile canlı sisteme geçilir. Devreye aldıktan sonra proje bakım ve destek çalışmaları başlar.

Bu çalışmalar kapsamında donanımla ilgili bakım ve onarım konularını açıklayan belgeler hazırlanır, bunun yanında yazılımın kullanıma hazır hâle getirilmesi, yazılımın kurulumu ve kullanımla ilgili bir dizi belgenin hazırlanması gerekir. Bazen hazırlamak yetmez, kâğıda bastırılması da gerekir.

Anlaşıldığı üzere test aşamasından sonra yazılım geliştirme süreci devam etmektedir. Yazılımın kullanımı ve entegrasyonu için gereken belgelerin hazırlanması için planlamaya yeterince zaman ayırarak maliyet tahminleme çalışmalarına dâhil edilmesi gereken bir süreçtir.

Bu sürecin planlı olması, ihtiyaç duyulan belgelerin anlaşılır ve çözüm sunan yaklaşımı, müşteride kurulum için geçen süredeki tutum ve davranışlar müşteri gözünde en çok iz

bırakan ve firma imajını derinden etkileyen faktörlerdir. Çünkü müşteriler yazılımın geliştirme süreçlerinde ne kadar mesai harcadığını ne kadar kaliteli kodlamalar yapıldığının farkında olamaz, onun için gerekli olan ekran düzeni ve kullanımı ile kurulum ve devreye alma aşamalarının kalitesidir. (Atbaş, 2012: 59) Kısacası müşteriler, içinde oldukları süreçleri puanlayabilir ve fikir sahibi olabilirler.

1.2.7. Bakım ve Destek

Devreye alma sürecinden sonra yazılım için bir sonraki süreç bakım ve destektir. Bu kapsamda yapılan çalışmaların çoğu kısmı yazılım geliştirmeye etkisi olmayan çalışmalar olduğundan yazılım geliştirme süreçlerinin içinde sayılmamaktadır. Ancak yazılım üreticisi olan firmaların yaşadıkları süreçlerden biridir. Dolayısı ile ilerleyen bölümlerde ürün maliyetlerinin hesaplanması konusunda maliyet ve gider kayıtlarının ayrıştırılmasında yazılım geliştirmeye katkısı olan faaliyetler ile katkısı olmayan faaliyetlere göre hesaplamalar yapılacaktır.

“Yazılım bakım işlemleri yalnızca hataları düzeltmek olmayıp ürünün tesliminden sonra ortaya çıkabilen, aşağıdaki gruplarda belirtilen işleri kapsar.” (Saridoğan, 2011: 182)

- *Düzeltilici Bakım: Geliştirici tarafından, kusurun kaynaklanma sebebini araştırmaya ve gidermeye yönelik yürütülen işlere düzeltilici (corrective) bakım denir. Yazılım teslim edildikten sonra kullanıcı, kullanım sırasında karşılaştığı yanlış hesaplama, hatalı davranış, eksik veri gösterimi gibi yazılım kusurlarını bildirerek düzeltilmesini isteyebilir. Bu gibi durumlarda hatanın önemine bağlı olarak ya hemen yapılır ya da birkaç tanesi beraberce ele alınıp gerekli düzeltmeler sonrasında yeni bir sürüm çıkartılır.*
- *Uyarlayıcı Bakım: Yazılım kullanan kurum veya işletmeler zaman içinde çeşitli nedenlerle işlerin yürütülme şeklinde, kullanılan yöntemlerde (örneğin yazıcıdan alınan kâğıt baskının biçiminde değişiklik gerekmesi) değişiklikler yapma gereksinimi duyulabilirler. Bilgi işleme dünyasındaki hızlı değişimler ve teknolojik gelişmeler nedeniyle, yazılımın yeni donanıma, işletim sistemlerine, bunların yeni sürümlerine, yeni uçbirimlere göre uyarlanması, sürümlerin yükseltilmesi ve güncelleştirilmesi işlemlerine uyarlayıcı (adaptive) bakım denmektedir.*

- *İyileştirici Bakım: Yazılım geliştirilip test edilerek başarılı bir şekilde kullanıcıya sunulduktan sonra, yeni işlevler eklemek, var olanlara başarıyı ve verimi artırıcı düzenlemeler yapmak iyileştirici (perfective) bakım işlemleridir. Bu şekilde yazılımın yeni sürümleri ortaya çıkar ve kullanıcının hizmetine sunulur. Örneğin, veritabanı erişimi 5 saniye olan bir yazılımda yeni bir arama algoritmasının kullanımı ile arama süresi 3 saniyeye indirilebiliyorsa bu değişikliğin uygulanması bir iyileştirici bakımdır. Yine kullanımdaki bir yazılıma, yeni kullanıcı isterlerine göre yeni işlevler kazandırmak iyileştirici bakım kapsamına girer. Bu gibi durumlarda ayrı bir sözleşme yapılır ve maliyetler hesaplanır. Onaylanan isterler için yazılım geliştirme süreci yeniden başlar.*
- *Önleyici Bakım: Yazılımın gelecekte uygulanabilecek değişikliklere daha iyi bir temel oluşturması, bakılabilirliğinin ve güvenilirliğinin artırılması için ön tedbir niteliğindeki işlemler önleyici (preventive) bakım kapsamına girmektedir. Örneğin, sık sık değişiklik yapılması gereken bir modülün tasarımını daha esnek bir hâle getirmek üzere yeni bir geliştirme süreci uygulanması, bunun sonunda oluşan yeni sürümün müşteriye verilmesi ve sonraki değişikliklerin daha kolay yapılmasının sağlanması önleyici bir bakımdır.*

Bakım işlemlerini takip etmek ve karara bağlamak için “Değişiklik Kontrol Kurulu” oluşturulmalıdır. Herhangi bir değişiklik; analiz, tasarım, kodlama, test ve devreye alma aşamalarından geçirilerek uygulanır. Değişiklik isterleri müşteriden veya geliştiriciden gelebilmektedir. Yazılım, bakım kolaylığı sağlayacak şekilde geliştirilmeli, bakım sırasında niteliksel özellikler korunmalı ve niceliksel ölçümler yapılmalıdır.

Bakım işlemleri, yazılım geliştirme maliyetleri hesaplanırken hiç de göz ardı edilmemesi gereken oranda gider oluşturmaktadır. Toplam geliştirme maliyetlerinin yaklaşık %10 ila %15'i kadar her yıl bakım gideri oluşmaktadır. (Atbaş, 2012: 63) Geliştirmeden sonra bu düzeydeki harcamalara hazırlıklı olunması gerekir.

Bakım işlemlerinin yanında destek işlemleri de vardır ki bu kapsamdaki çalışmalar yazılıma bir müdahalede bulunmadan kullanıcıların yaşadığı sorunları çözmek amacıyla çeşitli kanallardan yapılan yardımları ve eğitim faaliyetlerini kapsamaktadır. Örneğin; telefon, uzak erişim, yerinde destek ve yazılım eğitim işlemleridir. Bu faaliyetler sürekli

olarak verilen hizmetlerdir ve yazılım üreticisine devamlı olarak gider oluşturan işlemlerdir.

1.3. Yazılım Geliştirme Modelleri

Yazılım geliştirme modelleri, yazılım geliştirme süreçlerinin kendi aralarındaki ilişkilerinden ve kriterlerden kaynaklı farklılıklara dayanmaktadır. Her bir model, geliştirme ihtiyacını karşılamaya yönelik olup farklı yollardan sonuca ulaşmaktadır. Bu farklılıkların esasında süreçlerin sırası ve aralarındaki geçiş kriterleri bulunmaktadır.

Bir geliştirme modelinin seçilmesiyle ilgili modelin standartlarına uyulmasından dolayı önemli noktaların atlanmaması ve planlamanın kolaylığı gibi faydalar elde edilir. Seçilen modele göre işlemlerin nasıl birbirine bağlanacağı, planların nasıl yapılacağı ve yürütüleceği ortaya çıkmaktadır. *“Proje yöneticisi; ürün, yapılacak proje, kurum ve ekibin yapısına en uygun modeli seçilmelidir.”* (Nizam, 2014: 44)

Yazılım geliştirme modelleri en temelde ikiye ayrılır. Bu kategoriler doğrusal; tek yönlü ve döngüsel; yinelemeli, artımlı ve tekrarlı modeller şeklindedir.

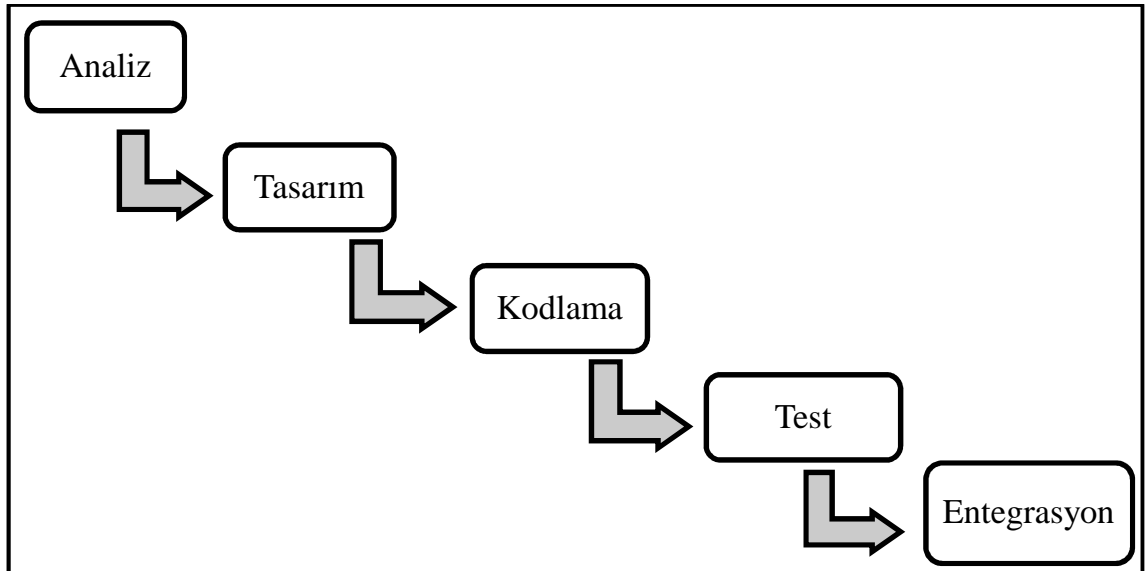
1.3.1. Doğrusal Modeller

Doğrusal yazılım geliştirme modellerinde en baştaki süreçten (ihtiyaç belirleme) en sonuncu sürece (devreye alma) kadarki aşamalarda hep bir aşama bittikten sonra diğerine geçilmesi düşünülmektedir. Yani analiz işlemleri bitip de belgelenmeden yapılmadan tasarım sürecine geçilmez. Süreç daima ileriye doğru bir yön içerisindedir. Değişen şartlara iyi adapte olamazlar çünkü değişen şartlar demek tekrar en başa dönerek planlama yapmak demektir. Doğrusal yazılım geliştirme modelleri en çok kullanılmış yazılım geliştirme modelleridir. Ancak değişen şartlara çok uygun olmamaları ve günümüzde insan ihtiyaçlarının da devamlı değişkenlik göstermesi projeleri oldukça zora sokmaktadır. Bu sebeple döngüsel yazılım geliştirme modelleri geliştirilmiştir. Döngüsel modellerin temelini de yine doğrusal modeller oluşturmaktadır.

1.3.1.1. Şelale (Waterfall) Modeli

Geleneksel yazılım geliştirme modeli olarak bilinen bu model yakın zamanlara kadar en popüler modeldir. *“Bu süreç modeli için doğrusal ve ardışık bir yaşam döngüsü olarak bahsedilir. Bu modelde bir sonraki aşama başlamadan önceki her aşamanın tam olarak tamamlanması gerekir. Bu modelde projenin her safhasının sonunda doküman oluşturulur ve projenin devam edip, edilmemesi yorumlanır. Basit, kısa süreli ve kolay yazılım programları için uygun bir modeldir.”* (Temur, 2013: <http://salyangoz.com.tr>) Bu model esneklik sağlamadığından uzun süreli projeler için uygun olmamaktadır.

Şekil 3’ten de anlaşılacağı üzere bir süreç bitmeden diğer bir sürece geçilmez. Her süreç sonu belgeleme yapılır ve sonraki sürece geçilir. Ayrıca analiz, tasarım gibi süreçlerde iyi planlanmayan veya değişen şartlardan dolayı yeni talepler geldiğinde tekrar bu süreçler gündeme gelir ve kendisinden sonraki süreçlerde ilerleme olmaz. Bu daha uzun bir teslim süresi doğurur. Bu yüzden müşteri çok fazla beklemek zorunda kalır. (Pressman, 2009: 41) Uzun süreli yazılım projelerinde zaman kötü yönetilmiş olur ve geliştirme ekibindeki birçok çalışan kendilerine sıra gelene kadar atıl bekler. (Nizam, 2014: 46) *“Projenin kalitesinin, projenin süreç ve maliyetinden daha önemli olduğu koşullarda tercih edilmelidir.”* (Temur, 2013: <http://salyangoz.com.tr>)



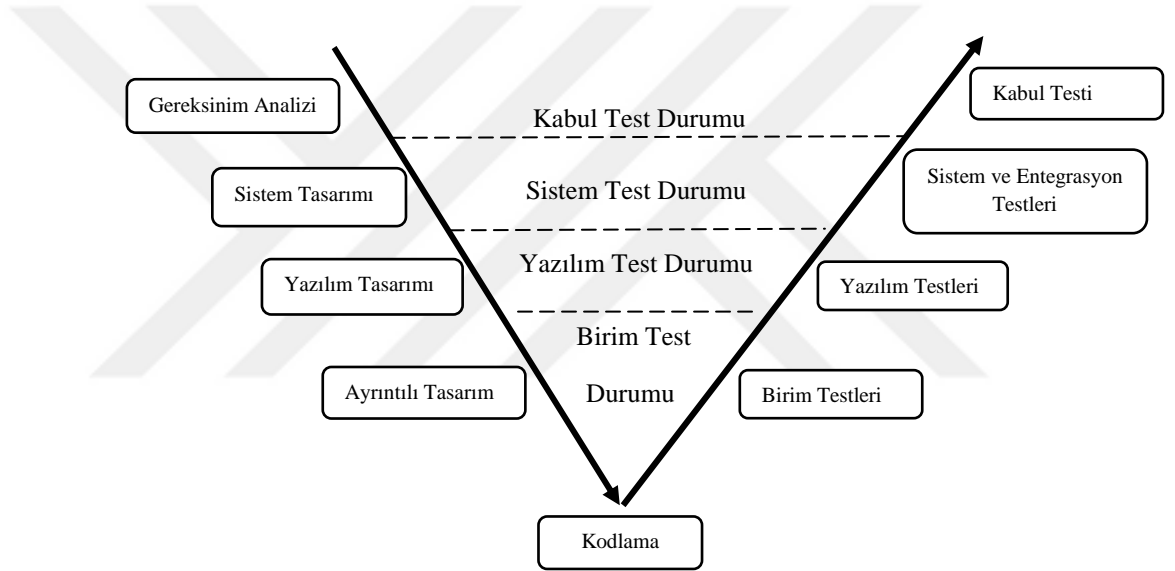
Şekil 3: Şelale (Waterfall) Yazılım Geliştirme Modeli Yaklaşımı

Kaynak: (Temur, 2013: <http://salyangoz.com.tr>)

En önemli avantajı ise sürecin planlaması daha sabit olduğundan maliyet tahminlemeleri daha keskin hesaplanabilmektedir ve müşteriye daha rahat fiyat verilebilmektedir.

1.3.1.2. V-Model

“Bu model şelale modelinin genişletilmiş bir yaklaşımıdır. Bu yaklaşımda daha güvenilir ve doğru yazılımlar geliştirmek için her bir yazılım geliştirme süreç adımının karşılığı olan test eylemleri model içerisinde gösterilmiştir. V- Modeli aynı zamanda doğrulama eylemlerini tanımlama ile birlikte başlattığından proje zamanında büyük kazanımlar elde edilir.” (Gürbüz, 2010: 18) Şekil 4’te gösterildiği gibidir.



Şekil 4: V-Yazılım Geliştirme Modeli Yaklaşımı

Kaynak: (Gürbüz, 2010: 18)

Modelin grafik gösterimi “V” harfine benzediği için modele bu isim verilmiştir. Şekilde sol taraf tasarım, sağ taraf ise doğrulama aktivitelerini içerir. Modelin ana ilkeleri şunlardır:

- Her tasarım işlemine denk gelen bir doğrulama işlemi olmalıdır.
- Kodlama için lazım olan gereksinimler listesi aşama aşama detaylandırılarak hazır hâle getirilmeli ve bu aşamalar teker teker doğrulanarak birleştirilerek ürün elde edilmelidir.

V-Modelinin, şelale modelinin test kısmı standartlaşmış bir türü olduğunu ve yeni bir şey getirmedığı ileri sürülmektedir. (Pressman, 2009: 39) Modelin avantajı test edilmedik bir süreç bırakmamaktadır. Aynı zamanda eksik kalan veya atlanan bir aşama da bırakmaz. Dezavantajı ise tekrarlı bir yapı sunmadığı için değişikliklere cevap verebilecek yapıda olmayışıdır. Belirsizlikleri az olan projelerde kullanılabilir.

1.3.2. Döngüsel Modeller

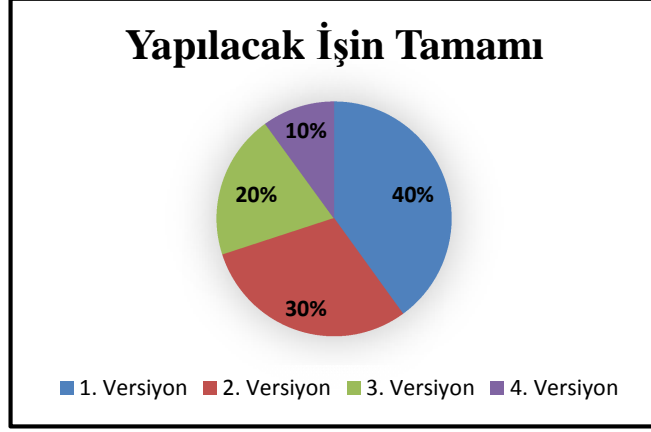
Bu gruba ait olan modeller doğrusal modellerin geliştirme esnasında içerden veya dışardan gelen değişiklik taleplerine cevap verememenin dezavantajının üstesinden gelebilmek için ortaya çıkmıştır. Bu yüzden daima başa dönebilen ve süreç tekrarlarından oluşan bir yapıları vardır.

1.3.2.1. Artımlı (Incremental) Geliştirme Modeli

“Bu model yazılım geliştirme süreçlerinden tasarım, geliştirme ve onaylama süreçlerini dönüşümlü olarak çalıştırır.” (Sommerville, 2011: 33) Yani, müşteri isteklerini önceliklendirerek planlama, tasarlama, geliştirme, test ve teslim ile birinci sürümü oluşturmuş olur. Daha sonra diğer bir istek için yine aynı döngüyü tekrarlar. Böylece sürümler şeklinde müşteriye teslimat yapılmış olur. İlk sürüm en temel isteklerin karşılanması için geliştirilir.

Şelale modelinin dezavantajları göz önüne alınca tekrar işlemeli ve artırımsal bir model düşünülmüştür. Böylece daha esnek geliştirme imkânı doğmuş olmaktadır. Bazı sürümler proje ekibini hedef alırken, bazı sürümler kurum üst yöneticilerinin kısa süreli beklentileri için, bazıları da son kullanıcının ayrıntılı ihtiyaçlarına yönelik geliştirilir. (Nizam, 2014: 49)

Her sürüm bir öncekine göre daha işlevseldir. Şekil 5'ten anlaşılacağı üzere artırım kelimesi bütüne olan katkıdan dolayı işlevlerin artmasından gelmektedir. *“Bu model evrimsel geliştirmeye benzemekle beraber en büyük farkı, sürümlerin (uyarlamaların) tam işlevsellik içermemesidir. Yani, sürümlerde eksiklikler bulunduğundan ürünün tam kullanımı mümkün olmaz. Sistem, henüz gerçekleştirilmemiş yeteneklerinin ve işlevlerinin kullanılmak istenmesi durumunda kullanıcıya uyarı iletileri verir.”* (Saridoğan, 2011: 74)



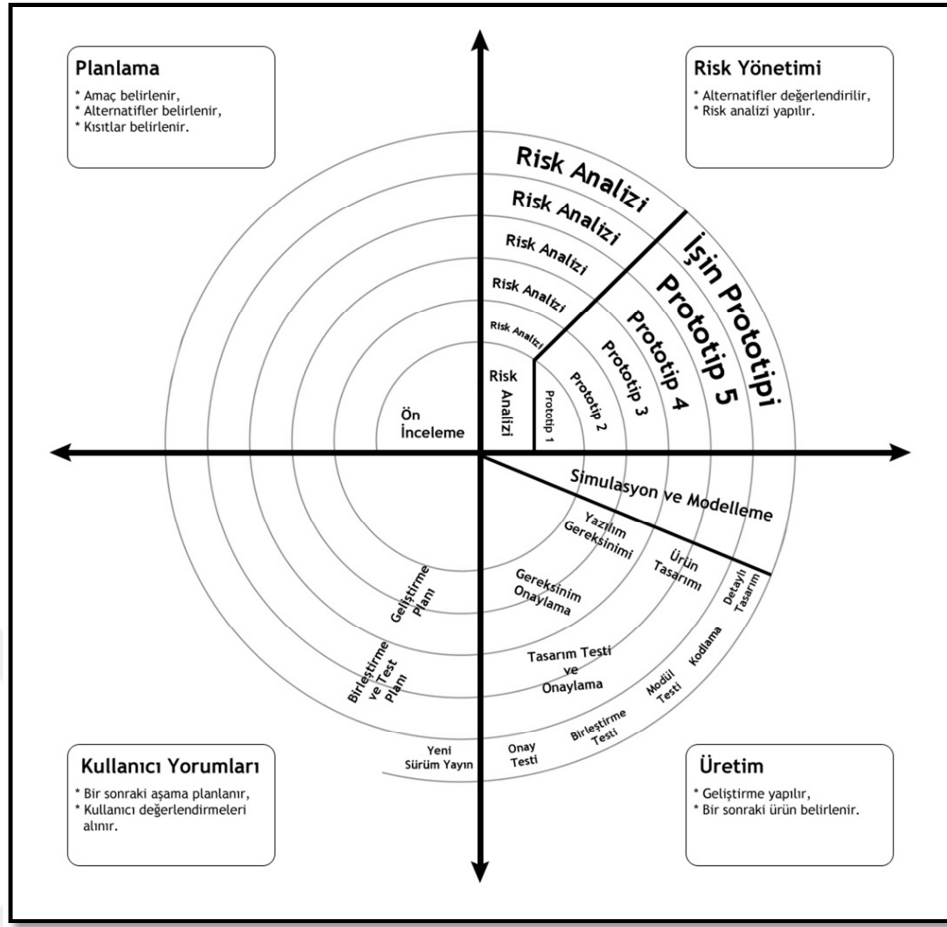
Şekil 5: Artırımlı (Incremental) Yazılım Geliştirme Modeli Yaklaşımı

Kaynak: (Saridoğan, 2011: 73)

Daha kısa sürede müşteriye teslim edilen ürün, daha az maliyet demektir. Ayrıca değişen müşteri taleplerine yeni sürümler ile cevap verebilmesi yine maliyetleri azaltmaktadır. Çünkü şelale modeline göre yeniden yapılanma ve belgelendirme işlemleri azdır. *“Bitmemiş de olsa müşterilere teslim edilen sürümlerin müşteri testleri de gerçekleştirildiğinden daha kısa sürede düzeltme işlemleri de yapılabilir.”* (Sommerville, 2011: 33)

1.3.2.2. Spiral Model

Risk çerçevesinde yazılım geliştirmeye odaklı olan bu modeli 1988 yılında Barry W. Boehm önermiştir. Yinelemeli ve artırımsal bir yaklaşım vardır. Her döngü bir fazı ifade eder. Her fazda şelale modelinin geliştirme süreçleri uygulanır. Ayrıca prototip yaklaşımı da vardır. Şelale ve prototip yöntemlerinin iyi yanlarının alınarak birleştirilmesinden oluşmuştur. Temel amaç her sürümün daha güvenli ve hızlı yapılmasını sağlamaktır. Bu modelde süreç 4 gruba ayrılır. Bu süreçler şekil 6’da görüldüğü gibi planlama, risk yönetimi, üretim ve kullanıcı yorumlarıdır.



Şekil 6: Spiral Yazılım Geliştirme Modeli Yaklaşımı

Kaynak: (Boehm, 1988: 7)

Her aşamada yapılan işlemler şunlardır;

Planlama: Yapılacak işlerin eldeki kaynaklar ve zaman gibi sınırlılıkların dikkate alınarak amaçların belirlendiği ve diğer seçeneklerin hesaplandığı işlemleri içerir.

Risk Yönetimi: “Olası alternatif çözümler geliştirici tarafından incelenir ve riskler tespit edilir. Projenin devamı açısından riskler dikkate alınır. Bu aşamanın sonunda bir prototip üretilir.” (Temur, 2013: <http://salyangoz.com.tr>)

Üretim: Ürünün gerçekleştirildiği mühendislik aşamasıdır. Bu aşamada şelale modelinin tüm süreçleri kullanılır.

Kullanıcı Yorumları: Kullanıcı geri bildirimlerinin alınması ve yapının geçerliliğinin ve gerektiği gibi çalıştığına ortaya konulması işlemleridir.

Bu modele adının spiral konmasının sebebi; her versiyonun 4 aşamadan oluşarak artırımlı ve tekrar eden biçimde olmasından dolayıdır.

“Spiral (sarmal) model çok karmaşık büyük projelerde uygulanabilecek gerçekçi bir model olarak görülmektedir. Çünkü süreç ilerledikçe her bir evrimleşmede geliştiriciler ve müşteri birbirini daha iyi anlamaya başlar ve riskleri paylaşmayı öğrenir. Böylece hem müşteri hem de geliştirenler tarafından risklerin önceden öngörülmesi ile risklerin önlenmesi sağlanabilir.” (Gürbüz, 2010: 23)

1.3.2.3. Çevik (Agile) Model

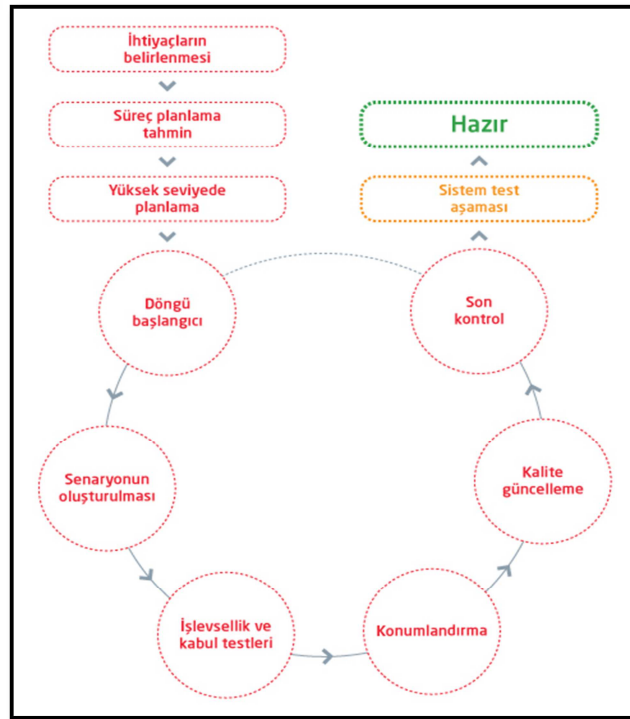
Bu teknikler süreç ve belgeleme yerine doğrudan yazılımın kendisine yoğunlaşan ve yenilemeli geliştirmeye dayanır. Teknoloji ve kullanıcı beklentilerindeki sürekli değişime cevap verebilmek için doğmuştur. 2001 yılında Kent Beck çevik yazılım geliştirme prensiplerini on iki maddede toplamıştır. Bu prensipler aşağıdaki gibidir: (Beck, 2001: www.agilemanifesto.org)

- *En önemli önceliğimiz değerli yazılımın erken ve devamlı teslimini sağlayarak müşterileri memnun etmektir.*
- *Değişen gereksinimler yazılım sürecinin son aşamalarında bile kabul edilmelidir. Çevik süreçler değişimi müşterinin rekabet avantajı için kullanır.*
- *Çalışan yazılım, tercihen kısa zaman aralıkları belirlenerek birkaç haftada ya da birkaç ayda bir düzenli olarak müşteriye sunulmalıdır.*
- *İş süreçlerinin sahipleri ve yazılımcılar proje boyunca her gün birlikte çalışmalıdırlar.*
- *Projelerin temelinde motive olmuş bireyler yer almalıdır. Onlara ihtiyaçları olan ortam ve destek sağlanmalı, işi başaracakları konusunda güven duyulmalıdır.*
- *Yazılım ekibinde iletişimin en verimli ve etkin yöntemi yüz yüze olandır.*
- *Çalışan yazılım ilerlemenin birincil ölçüsüdür.*
- *Çevik süreçler sürdürülebilir geliştirmeyi teşvik etmektedir. Sponsorlar, yazılımcılar ve kullanıcılar sabit tempoyu sürekli devam ettirebilmelidir.*

- Teknik mükemmeliyet ve iyi tasarım konusundaki sürekli özen çevikliği artırır.
- Sadelik, yapılmasına gerek olmayan işlerin mümkün olduğunca arttırılması sanatı, olmazsa olmazlardandır.
- En iyi mimariler, gereksinimler ve tasarımlar kendi kendini örgütleyen takımlardan ortaya çıkar.
- Takım, düzenli aralıklarla nasıl daha etkili ve verimli olabileceğinin üzerinde düşünür ve davranışlarını buna göre ayarlar ve düzenler.

Kent Beck, bu yazılım geliştirme modelinin özetini ise şu sözlerle yapmaktadır. “Süreçler ve araçlardan ziyade **bireyler ve etkileşimlere**; kapsamlı dokümantasyondan ziyade **çalışan yazılıma**; sözleşme pazarlıklarından ziyade **müşteri ile işbirliğine**; Bir plana bağlı kalmaktan ziyade **değişime karşılık vermeye** kanaat getirdik. Özetle, sol taraftaki maddelerin değerini kabul etmekle birlikte, sağ taraftaki maddeleri daha değerli bulmaktayız.”

Görüldüğü üzere daha dinamik çalışma anlayışı ve ortamı ile daha çok iş bitiricilik ve müşteri memnuniyeti ön plana çıkartılmak istenmektedir. Çevik model yaklaşımı şekil 7’de özetlenmiştir.



Şekil 7: Çevik (Agile) Yazılım Geliştirme Modeli Yaklaşımı

Kaynak: (<http://www.innova.com.tr/>, 2015)

Bu gruba giren diğerk alt geliştirme yöntemleri şunlardır; (Gürbüz, 2010: 20)

- *Uç (Sınırsal) Programlama (Extreme Programming-XP)*
- *Çevik Birleştirilmiş Süreç (Afile Unified Process)*
- *Scrum*
- *Test Güdümlü Geliştirme (Test-Driven Development)*
- *Çevik Bilgi Yöntemi (Agile Data Method)*
- *Özellik Güdümlü Geliştirme (Feature-Driven Programming)*

Tüm çevik modeller geliştirilecek olan yazılımı farklı önceliklendirmeyeyle geliştirmeye çalışsalar da temelde kullanılan prensipler yukarıda belirtilenlerdir.

Spiral modelin iyi bir örneğidir. Kısa vadeli planlar ve küçük geliştirmelerle yazılım ilerler. Her geliştirmeyle birlikte yeni bir versiyon çıkar ve her versiyonda toplam istenen bütüne bir adım daha yaklaşmış olunur. Belgeleme zayıftır ve zaman kaybı olarak görülmektedir.

Bu modelin de bazı dezavantajları bulunmaktadır; (Atbaş, 2012: 95)

- *Çeviklik kavramı yanlış anlaşıldığında personelin tükenmesi denilen yan etki görülür.*
- *Ekip liderine çok fazla sorumluluk ve güç verildiğinde, hem ekip hem de ekip üyelerini yıpratır.*
- *Her türlü yazılım problemini çözeceği düşünülduğünden hayal kırıklığına sebep olabilir.*
- *Takımın disiplinli, düzenli ve tecrübeli olması tercih edildiğinden buna hazır olmayan organizasyonlar, takım ve geliştiriciler için uygun değildir.*
- *Dokümantasyon tamamen ihlal edilebilir ve bunun sonuçları çok kötü olabilir.*

Spiral ve Çevik yazılım geliştirme modelleri birbirleriyle karıştırılabilirler çünkü her ikisi de artımlı ve tekrar işlemeli modellerin alt grubudur. Aralarındaki farklar şunlardır: (Atbaş, 2012: 97)

- a) *Spiral modelde ön planlama daha ağırlıklıdır, tekrarlar önceden planlanır. Oysa çevik yazılım geliştirmede sadece o anki tekrar için detaylı planlama yapılır, diğer tekrarlar için planlama yapılmaz.*
- b) *Spiral modelde “planla-gerçekle (kodla) – test et – düzelt” tipi bir yaklaşım varken çevik modelde testlerin mümkün olan en kısa zamanda hatta bazen tekrarın en başında hazırlanması düşünülür. Çevik modelde test, kodlamadan daha önceliklidir.*
- c) *Çevik modelde her tekrar sonunda, öncelikli gereksinimlerin gerçekleştiği, çalışabilir, tamamen test edilmiş, az öncelikli ama hazır bir program üretilir. Spiral modelde bu şart değildir, ilk tekrarda sadece bir prototip mahiyetindeki bir ürün ortaya çıkabilir.*
- d) *Spiral model teknik üretkenlik yaklaşımı ile iş yaparken; çevik model insan psikolojisini de göz önünde bulundurarak sosyal içerikli bir yaklaşımla iş yapar.*
- e) *Spiral model daha klasik bir yaklaşım olduğundan dokümantasyondan, kod içi yorumlardan, planlardan ve gereksinimlerden taviz vermez; çevik modelde ise bu konular suistimale biraz daha açıktır.*

Yazılım firmaları değişik modellere uyarak geliştirme yapsalar da temelde her firma eksiksiz olarak Şelale Model’indeki süreçlerden geçer. Kimi seçtiği modele göre bir kere geçer, kimi defalarca geçer.

1.4. Yazılım Proje Yönetimi

Proje yönetimi; “benzersiz ve bir kez uygulamaya konulan projelerin süre, maliyet, kapsam ve bunların odağındaki kalite hedefleri dâhilinde tamamlanması ve bir miktar önüne geçerek aşılması için, eldeki var olan veya yaratılan kaynak ve kapasitenin en rasyonel şekilde belirli metotlar yardımı ile kullanılmasıdır.” (Günaydın ve Bolposta, 2002: 27)

“Proje yönetimi, proje hedeflerine ulaşmak için maliyet, zamanlama ve kalite kriterlerinin göz önünde bulundurularak, mühendislik aktivitelerinin doğru olarak planlanması ve denetimi işlemleridir.” (Borandağ, 2012a, <http://slideplayer.biz.tr>)

“Küçük ya da büyük her türlü yazılım projesinde yönetim son derece önemlidir. Gerek maddi kaynakların gerekse insan kaynaklarının en uygun şekilde kullanımı her kurum

ya da işletme için yüksek önem taşır. Bireysel olarak çok başarılı ve yetenekli insanları bir araya getirmekle projelerin iyi bir şekilde sonlanması beklenmemelidir. Belirli kurallara göre bir örgüt yapısı kurulup uygun bir yönetimle başlamayan proje karmaşa ile son bulmaya mahkûmdur.” (Saridoğan, 2011: 229)

Yazılım geliştirme projelerinin yönetimi, kaynakların ve ürünün soyut olması anlamında sanayi sektörlerinin üretim yönetiminden daha karmaşık olduğu söylenebilir. Bu yüzden de daha çok yönetilmeye ihtiyaç duyar. Bu farklılıklar şöyle özetlenebilir:

- a) Sanayi üretimlerinde otomasyon ve robotlaşma en üst düzeye çıkabilirken, yazılım geliştirenler insanlardır. İnsanlar, robotlara göre daha fazla ihtiyaç sahibidir.
- b) İnsanların moral, motivasyon, hastalık gibi iş yapma kapasitelerini etkileyen unsurlar varken robotların bu türden kapasitelerini etkileyen unsurlar bulunmamaktadır.
- c) Robotlu sanayi üretimi neredeyse sıfır hata ile üretim yapabilirken, yazılım geliştirme sürecindeki hatalar için uygun sapma paylarının hesaplanması gereklidir.
- d) Sanayi üretimlerinde ürün özellikleri uzun süre aynı kalır, ancak yazılımın sipariş edilen özellikleri çok kısa zamanda değişiklik gösterebildiğinden üretim çok daha değişken ve esnektir.

Tüm bu farklılıkların odağında insan vardır. İnsanın olduğu her türlü organizasyonun dinamik yönetilmeye ihtiyacı vardır. Yoksa işlerin karmaşıklığı kaosa neden olabilir.

Sanayi firmalarının ürettikleri ürünler somut ve üretim süreci rahatça takip edilerek maliyetleri kaydedilebilirken yazılımın üretimini takip etmek (ürün yaşam döngüsü) daha zordur ve bundan dolayı maliyetinin hesaplanması da daha karmaşıktır.

Bir projenin başarılı olarak kabul edilmesi için; (Borandağ, 2012a, <http://slideplayer.biz.tr>)

- a) Tahmin edilen ve taahhüt edilen zamanda tamamlanması,
- b) Tahmin edilen bütçe dâhilinde tamamlanması,
- c) İstenen performans gereklerini yerine getirmesi gerekmektedir.

Proje yöneticiliğinin günümüzde uluslararası normları belirlenmiş bir meslek olarak, Dünya’da kabul görmüş olması, en çok kazanan meslekler sıralamasında avukatlıktan sonra 2. sırada (PWC’nin çalışması) olması, bu ihtiyacın en belirgin göstergesidir. Proje yönetimi ve uygulamaları, taahhüt ve beklentiler arttıkça, kaynaklara ulaşım zorlaştıkça önemi artan bir alan olarak ilerliyor. (<http://www.projeegitimmerkezi.com/>, 2015)

Proje yönetiminin son yıllarda popülaritesinin artmasının bazı nedenleri şunlardır: (Borandağ, 2012a, <http://slideplayer.biz.tr>)

- *Firmalarda çalışan insan kaynağında azalma olmasına rağmen, ortaya çıkarılan işlerin niteliğinde ve kalitesindeki artış (teknolojik gelişmelerin etkisi),*
- *Projelerin daha kapsamlı olmaya başlaması,*
- *Küreselleşmenin getirdiği rekabet ortamı,*
- *İletişimin daha kolaylaşması,*
- *Pazarın genelini devamlı takip eden ve üreticilerin istekleri doğrultusunda yönlendirilebilen müşterilerin etkisi,*
- *Çok uluslu iş yapma olanaklarındaki artış,*
- *Gelişmeleri merkezden düzenli olarak izleyebilme ve gerekirse müdahale edebilme isteği*

Proje yönetim sürecinin aşamaları şunlardır; (Project Management Institute, 2000: 38)

- 1) **Tanımlama (Başlatma):** Proje başlatma belgesinin geliştirilmesi
- 2) **Planlama:** Proje yönetim planının geliştirilmesi
- 3) **Uygulama:** Projenin yürütülmesinin yönlendirilmesi ve yönetilmesi
- 4) **İzleme ve Kontrol:** Proje çalışmalarının izlenmesi ve kontrolü entegre değişiklik kontrolünün gerçekleştirilmesi
- 5) **Kapatma (Bitirme):** Proje ya da fazın kapatılması

Yukarıdaki aşamaların her birinde ihtiyaç duyulan bilgi alanları vardır. Bilgi alanlarından kasıt farklı konulardaki bilgilerdir. Proje Yönetim Enstitüsü’ne göre bilgi alanları şunlardır;

- 1) **Kapsam Yönetimi:** Projenin başarıyla tamamlanabilmesi için gerekli tüm çalışmaların ele alındığı bilgi alanıdır.
- 2) **Zaman Yönetimi:** Projenin hedeflenen zamanda tamamlanabilmesi için proje yöneticisi ve takımının bilmesi gereken temel konuların ele alındığı bilgi alanıdır.
- 3) **Maliyet Yönetimi:** Projenin hedeflenen bütçe içerisinde tamamlanabilmesi için proje yöneticisinin ve takımının bilmesi gereken temel konuları kapsar.
- 4) **Kalite Yönetimi:** Hem proje yönetiminin hem de proje çıktılarının kalite beklentilerinin belirlenmesi ve kalite kriterlerinin oluşturulmasını tarif eden bilgi alanıdır.
- 5) **İnsan Kaynakları Yönetimi:** Projeye dahil olacak bireylerin seçiminden, etkin kullanımına, eğitiminden, motivasyon konularına kadar geniş yelpazede bir proje yöneticisinin bilmesi gereken konuları kapsar.
- 6) **İletişim Yönetimi:** Proje boyunca toplanması, oluşturulması, dağıtılması ve depolanması gereken tüm bilgilerin hangi süreçlerden geçmesi gerektiğini belirleyen bilgi alanıdır. Bu alanda projeye taraf olan kişiler arasında, departmanlar arasında veya kurumlar arasındaki resmi veya resmi olmayan iletişim kuralları detaylı olarak tanımlanır.
- 7) **Risk Yönetimi:** Proje içerisinde karşılaşılan risklerin tanımlanması, analiz edilmesi ve nasıl yanıt verileceğini içerir.
- 8) **Tedarik Yönetimi:** Proje süresince ihtiyaç duyulacak her türlü ürünün veya hizmetin tedarik etme adımlarını detaylı olarak tanımlar.
- 9) **Entegrasyon (Tümleştirme) Yönetimi:** Tüm bilgi alanları arasında proje yönetimi döngüsünde tüm safhalarda yer alan tek bilgi alanıdır. Entegrasyon yönetimi, süreç grupları içindeki aktiviteleri bütünleştirmek adına, farklı proje yönetim elemanlarını belirlemek, tanımlamak, birleştirmek ve koordine etme faaliyetlerini tanımlar.

Yukarıda ifade edilen proje yönetim süreçleri ve bu süreçlerde ihtiyaç duyulan bilgi alanları Şekil 8’de haritalandırılmıştır.

İstanbul Kurumsal Gelişim											
PY Süreçlerini Haritalandırma		Proje Kapsam Yönetimi	Proje Zaman Yönetimi	Proje Maliyet Yönetimi	Proje Kalite Yönetimi	Proje İnsan Kaynakları Yönetimi	Proje İletişim Yönetimi	Proje Risk Yönetimi	Proje Satınalma Yönetimi	Proje Entegrasyon Yönetimi	
	Başlangıç										* Proje Duyurusu Oluşturma * Ön Proje Kapsam Tanımlama Dokümanı Oluşturma
	Planlama	* Kapsam Planlama * Kapsam Tanımlama * İş Ayrışım Yapısı Oluşturma	* Aktivite Tanımlama * Aktivite Sıralama * Aktivite Kaynakları Tahminleme * Aktivite Süresi Tahminleme * Zaman Programı Oluşturma	* Maliyet Tahminleme * Maliyet Bütçeleme	* Kalite Planlama	* İnsan Kaynakları Planlama		* İletişim Planlama	* Risk Yönetimi Planlama * Risk Belirleme * Nitel Risk Analizi * Nicel Risk Analizi * Risk Yanıtlama Planı Oluşturma	* Satınalma ve Tedarik Planlama * Sözleşme Planlama	* Proje Yönetim Planı Oluşturma
	Uygulama				* Kalite Güvence Sağlama	* Proje Ekibi Oluşturma * Proje Ekibi Geliştirme		* Bilgi Dağıtım		* Satıcılardan Teklif İsteme * Satıcıların Seçimi	* Proje Uygulamasını Yönlendirme ve Yönetme
	İzleme ve Kontrol	* Kapsam Doğrulama * Kapsam Kontrolü	* Zaman Programı Kontrolü	* Maliyet Kontrolü	* Kalite Kontrol	* Proje Ekibi Yönetme	* Performans Raporlama * Proje Taraflarının Yönetimi	* Risk İzleme ve Kontrolü	* Sözleşme Yönetimi		* Proje İşlerinin İzlenmesi ve Kontrolü * Genel Değişiklik Kontrolü
	Kapanış								* Sözleşme Kapanışı		* Proje Kapanışı

Şekil 8: Proje Yönetim Süreçlerinin Haritalandırılması

Kaynak: (Kansu, 2013: <http://canatufkansu.com>)

Projenin kapsam, zaman, kaynak ve insan kaynakları yönetimi en temel yönetilmesi gereken ve en kritik aşamalardır. Bu süreçlerin yönetiminden önce projenin gereksinimlerine göre bazı tahminlerin yapılması gerekir ki bunu da proje yönetim yazılımlarıyla elde etmek daha kolay, çabuk ve kaydedilebilir. Bu çalışmalar ile işletmenin ve projelerin sürdürülebilirliği, nakit akışı ve başarılı bir şekilde yönetilerek tamamlanması mümkün olabilir. Bu sebeple proje büyüklüğünün tahminlemesi ve proje yönetim yazılımları açıklanmaya çalışılacaktır.

1.4.1. Proje Büyüklüğünü Ölçme ve Tahminleme

Proje kapsamını yönetmek, öncelikle, projeye nelerin dâhil edileceğini ve nelerin dâhil edilmeyeceğini tanımlama ve kontrol etme işidir. (Kansu, 2013)

Kapsam yönetimi çalışmaları içinde proje kapsam büyüklüğü de tahmin edilmelidir. Çünkü yazılımın büyüklüğü arttıkça, gerçekleştirmek için harcanan süre ve kaynak kullanımı dolayısı ile maliyetler artar. “100 bin satırlık bir proje 1 ayda bitiyorsa 200

bin satırlık bir proje 2 aydan daha uzun sürer. Yazılım bu yönüyle diğer ürünlerden ayrılır. Ekonomideki diğer ürünlerde genellikle büyüklük arttıkça birim maliyet düşmektedir. Bu durum negatif ölçek ekonomisi (diseconomy of scale) olarak isimlendirilir.” (Nizam, 2014: 114)

“Küçük ölçekli yazılım projelerinde ayrıntılı tasarım, kodlama ve birim test; büyük projelerde ise planlama, ihtiyaç analizi, tasarım ve bütünleşme daha çok zaman alır. Bunun temel sebebi yazılımın birbiriyle bütünleşik çalışan birçok parçadan oluşmasıdır. Parça sayısındaki artış, proje karmaşıklığını da üstel olarak arttırır. Karmaşıklık, planlama ihtiyacının artmasına sebep olur. Projenin büyümesiyle artan ekip sayısı da kişiler arasındaki aktif iletişim ihtiyacı ve karmaşayı arttırır.” (Nizam, 2014: 114) Bu sebeple proje büyüklüğünün öngörülebildiği kadarı ile tahmin edilmesi profesyonel proje yönetimi ve planlaması için gereklidir.

Tahminler proje hedeflerine göre yönlendirilmiş olmamalıdır. Analitik ve objektif olarak elde edilmelidir ki gerçek durum görülebilsin. Daha sonra farklı değişkenler kullanarak tahminde bulunmak daha tutarlı karar alınmasını sağlayacaktır. Yoksa tahmin modelinin değişkenleri sabitken üst yönetimin farklı teslim taahhütlerindeki tamamlanma oranı, kalite, işlevsellik gibi faktörlere yetişebilmek mümkün olmayacaktır. Tahmin, öncelikle doğrudan sayılacak büyüklüklere, daha sonra dolaylı hesaplamalara ve en son olarak da kişisel yargılara dayanmalıdır. (McConnell, 2006: 79)

Doğrudan (Aritmetik olan) sayılacak büyüklüklerden kasıt proje büyüklüğünü ifade edebilmek için en yaygın şekilde kullanılan değişkenlerden olan Kod Satır Sayısı ve Fonksiyonel Nokta Sayısıdır.

- a) Kod Satır Sayısı; sayılabilir bir büyüklüktür ancak maalesef kesin sayı proje sonlarına doğru ortaya çıkar. *“Kod Satır Sayısı bilgilerinin faydalı olması için ya eski bir projeye ait bilgilerin bulunması gerekir ya da projenin birkaç ayının geçmiş olması gerekmektedir ki bir ölçüm yapabilesiniz ve bunu ilerisi için kıyaslama değeri olarak kullanabilesiniz.” (Atbaş, 2012: 7) “İlk aşamalarda tespiti için ihtiyaç, ürün özelliği, kullanım senaryosu, ekran, rapor ve bileşen gibi büyüklüklerden yola çıkan dolaylı yöntemler kullanmak gerekir.” (Nizam, 2014: 168)*

b) Fonksiyonel Nokta Sayısı: Yaygın kullanılan bu yöntem 1984 yılından itibaren Milletlerarası Fonksiyonel Nokta Kullanıcı Grubu (IFPUG: www.ifpug.org) tarafından geliştirilmektedir. Bu yöntemle aşağıdaki özelliklere sahip bir ölçme sistemi önerilir: (Albrecht, 1979: 83-92)

- Kullanılan geliştirme teknolojisinden bağımsız
- Kolayca uyarlanabilen
- İhtiyaç analizinden yola çıkarak tahmin edilebilir
- Son kullanıcı için manalı

“Bu işlevsel öğeler sisteme girişler, çıkışlar, dış arabirimler, sistemde yapılan veritabanı sorguları ve bunun gibi öğelerdir. Sonuçta işlevsel noktaların toplamı sistemin yeteneklerinin sayısının toplamıdır, sistem özelliklerinin toplamıdır.”
(Atbaş, 2012: 7)

Bu yöntemde sistemin sağlayacağı yeteneklerin listesi yapıldığı için bu yeteneklerin önemine ve zorluk derecesine, ekibin bu konulardaki daha önceki tecrübelerine göre süre, efor tahmini yapmak daha kolaylaşır. Oysaki toplam satır sayısı bize işin zorluk derecesi hakkında bilgi vermez. Ancak fonksiyonel nokta sayısı ile tahmin yapmanın hazırlığı da daha fazla zaman gerektirmektedir.

Yukarıda bahsedilen, iki sayılabilen öğenin kayıtları proje yönetim yazılımlarıyla entegre olarak çalışabilmektedir. Böylece tahmin modeli ne olursa olsun bu birimlerin kaydedilmesi ve raporlanabilmesiyle veri elde etmek, karar almayı kolaylaştırmaktadır.

Sayısal olmayan daha çok tecrübeye dayanan tahmin modelleri de vardır. Bunlar şunlardır:

a) Uzman Kararı: Yazılım endüstrisinde çaba tahmini için en çok kullanılan yöntemdir. Yıllardan beri proje yöneticileri kendi deneyimlerine güvenmişlerdir. Pek çok uzman önerilen yazılımın uygulama alanı ve geliştirme tekniklerine göre proje emeğini tahmin etmektedir. Yeni proje daha önce tamamlanan projelerden çok farklı değilse ve deneyimli tahminciler mevcutsa, emek kestirimi için en uygun yöntem uzman kararıdır. (Borandağ, 2012b, <http://slideplayer.biz.tr>)

- b) Benzerlik ile Kestirim: Bu yöntemde daha önceden tamamlanmış projelere ait verilerin kaydedildiği bir ortamda kullanmak mümkün olmaktadır. Daha önce tamamlanan projelerin işlev ve ihtiyaçlarındaki benzerliklerden yola çıkarak yeni projenin tahminlemeleri yapılır.

1.4.2. Proje Yönetim Yazılımları

Proje yönetim aşamasında ihtiyaç duyulan bilgiler çok farklı alanlardan olabilmektedir. Bu alanların yaptıkları işlemleri ve ihtiyaç duydukları tüm bilgileri tek bir havuzda toplayabilmekle en etkin proje yönetimi sağlanacaktır. Teknolojinin ilerlemesi ile beraber birçok veriyi anlamlı hâle getiren ve karmaşık formüllerin sonucunu hesaplayan yazılımlar geliştirildi. Bu yazılımlara “Proje Yönetim Yazılımı” denmektedir. Bu yazılımlar sayesinde proje yönetimi için gerekli olan süreçleri ve bilgi alanlarına ait verileri yönetebilmekle beraber proje büyüğüne dair tahminler yapılabilip, geliştirme süreçleri izlenebilmektedir.

“Projeyi başarılı bir şekilde yönetmek için planı, geliştirme sürecini ve geliştirilen ürünlerin yapısını birbirleriyle bütünlük yönetmek gerekir. Yazılımların çeşitlenmesi ile proje yapısı, proje içi ve projeler arası ilişkilerin karmaşık hâle gelmesi bunu güçleştirir. Binlerce parçadan meydana gelen, farklı birimlere dağılmış ekiplerle geliştirilen, büyük ve karmaşık bir yazılım projesini bir sistem kullanmadan yönetmek, neredeyse imkânsızdır. Aksi hâlde yönetmek için harcanan çaba, geliştirmek için harcanan çabayı geçebilir.” (Nizam, 2014: 455)

“Proje yönetim sistemi, projeleri merkezi bir ortamda planlama, planı yürütme ve izleme amacıyla kullanılan yazılım altyapısıdır. Proje planının geliştirilmesi, iş bölümlene yapısının görsel olarak tasarlanması, tahmin araçları, görev atama ve takibi, kaynak gereksinimlerinin belirlenmesi, planın kaynak çakışmalarını önleyecek şekilde yapılması, yapılan planların merkezi veri tabanında saklanması ve paylaşılması gibi temel fonksiyonlar proje yönetim sistemleriyle gerçekleştirilir.” (Nizam, 2014: 457)

Proje yönetim yazılımı kullanmak proje başarısı için önemli kazançlar sağlar. Araştırmalar göstermiştir ki planlama ve tahmin işlemlerini yazılım araçları yardımıyla yapan yöneticilerin, elle yapanlara göre daha az hata yaptığını ve bu projelerin daha kısa sürede tamamlandığını göstermektedir. (Jones, 2010: 353) Proje geliştirme sürecinde çok fazla bilgi açığa çıkmaktadır. Doğru bilgiyi doğru kişiye doğru zamanda verebilmek

önemli bir meseledir. Çok fazla bilgi karar vermek için acele etmeye sebep olabilir. (Kliem, 2008: 7) Süreçleri algılamak, fonksiyonellik, bilgi kalitesi, proje yöneticisinin eğitimi ve performansı gibi faktörler proje yönetim yazılımlarının kullanımı ile pozitif yönde ilişkilidir. (Ali, Anbari ve Money, 2008: 14)

Proje yönetim yazılımları işlemleri hızlandıran, otomatik hâle getiren, standartlaştıran ve yönetimi kolaylaştıran imkânlar sunarlar. Ancak proje yönetim yazılımlarını kullanmak kesin başarıyı getiremez. Çünkü kullanıcıların yazılımı etkin kullanımları, yönetsel becerileri ve tecrübeleri diğer başarıyı destekleyen faktörlerdir. “*Proje yönetim sistemlerinin genel faydaları şunlardır.*”: (<http://www.projeegitimmerkezi.com/>, 2012)

- *Üst yönetimi karar verme sürecinde destekler!*
- *Tek bir sorumluluk merkezi belirler!*
- *Net hedefler tanımlar!*
- *Planlama, kontrol ve koordinasyon süreçlerini güçlendirir!*
- *Güçlü iletişim altyapısını destekler!*
- *Dinamik ve teşvik edici iş ortamı sağlar!*
- *Takım elemanlarına yetki ve sorumluluk verir!*
- *Öğrenmek ve iletişimi arttırmak için imkân yaratır!*
- *Daha çok işin, daha az zaman ve kaynakla yapılmasını sağlar!*
- *Karlılığı artırır!*
- *Organizasyonu daha etkin ve verimli yapar!*
- *İç ve dış müşteriler ile daha yakın bir çalışma ortamı sağlar!*
- *Problemlerin çözümü için yöntemler sağlar!*
- *Kaliteyi geliştirir!*
- *Kurumsal kararlar vermek için çalışanlara olanak sağlar!*
- *Çözümler üretir, iş geliştirir!*

Proje yönetim yazılımlarının seçimi ayrı bir araştırma gerektirir. Bu araştırmada özellikle işletmenin iş yapma biçimi, süreçleri hatta kurum kültürüne cevap verebilen

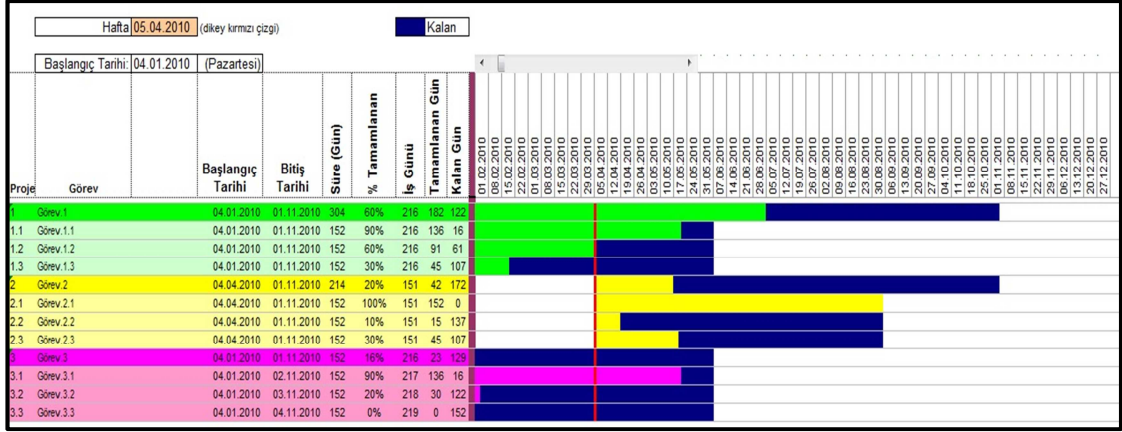
yapısının olup olmadığı incelenmelidir. Piyasada üretilmiş olan proje yönetim yazılımları, yazılım geliştirme süreçlerinin tamamını kapsadıklarını iddia ederken hangisinin hangi yazılım geliştirme modeline göre süreç takip yapabildiği, hangi modüllerin olduğu, raporlama ve belgeleme fonksiyonlarının derinliğinin incelenmesi gerekmektedir.

Projede, yapılacak işlerin detaylandırılması, işlerin önceliklendirilmesi, bazı işlerin mihenk taşı olarak belirlenmesi ve işlerin zorluklarının tanımlanması gibi aşamalar projenin zaman ve maliyet ihtiyaçlarını ortaya çıkarmanın ilk aşamasıdır. Detaylandırılan işlerin hangi personelin ne kadar süre ile bu işi yapması gerektiğinin de tahmini olarak belirtilmesi gereklidir. Yani, detaylandırılan işlere önce kaynak ataması sonra da zaman ilişkilendirilmesi yapılarak projenin zaman ve kaynak yönetimi sağlanmaya çalışılır.

Proje planlanmasında ve kontrolünde Gantt, CPM (Critical Path Method) ve PERT (Project Evolution and Review Technique) gibi ileri programlama yöntemlerinden yararlanır. (Şahin, 1996: 21) (Kır, 2007: 41) Yapılacak görevin hangi kaynaklar ile ne kadar sürede yapılması gerektiğini, görevlerin tamamlanma yüzdelerini ve sapmaları gösteren bir tablo yardımı ile takip edilmeye çalışılır.

“Sık kullanılan proje programlama yaklaşımlarından biri Gantt Modeli’dir. Gantt seması, bir zaman çizgisi boyunca projenin tüm safhalarının planlanan ve gerçekleşen sürelerinin, başlangıç ve bitiş zamanlarının belirtilmesi ile oluşturulur. Faaliyetler üstten alta, zamanlar ise soldan sağa doğru sıralanmıştır. Gantt semalarında zaman çizgisi boyunca her proje aktivitesi için yatay barlar çizilir.” (Kır, 2007: 42) (Evans, 1997: 748)

Şekil 9’dan anlaşılacağı üzere basit düzeyde görevler tanımlanıp gruplandırıldıktan sonra her bir iş için işe başlama ve bitiş süreleri atanmıştır ve arka planda bu görevleri gerçekleştirecek personel eşleştirmesi mevcuttur. Gantt şeması ile proje yönetimi ve süreç takibi kolaylaşır. (Şekil 9, bir excel görüntüsüdür.)



Şekil 9: Gantt Şeması Örneği

Bilişim firmaları da yazılım geliştirme projelerinde “Proje Yönetim Yazılımları” kullanmalıdır. Bu yazılımlar ile projenin başlangıcından bitişine kadarki süreçler tahminlenerek planlanır, uygulamalar gerçekleşirken planlar izlenebilir ve kontroller ile projenin bitişini tamamlanır. Sistemden elde edilebilecek raporlarla ki bunlar sadece Gantt şemasından ibaret değildir. Karar verme süreleri kısalmış ve rasyonel kararlar doğar. Böylelikle projedeki insan kaynakları, taşeron, malzeme (donanım ve yazılım altyapısı), zaman ve para gibi unsurların en rasyonel şekilde yönetimi gerçekleşir.

Proje yönetim yazılımları daha çok projenin süreçlerini izlemek ve yönetmek amacıyla zaman esaslı bir takip sağlar. Ancak muhasebe sisteminin ihtiyaç duyduğu verileri toplamak üzere geliştirilmemişlerdir. Piyasadakiler, daha çok tahmine dayalı gelecek harcamaları üzerinden proje likiditesine odaklı raporlar oluşturur.

Piyasada satılan bazı proje yönetim yazılımları tablo 7’de gösterilmiştir.

Tablo 7: Proje Yönetim Yazılımları

Yazılımın Adı	Açıklama	İnternet Sitesi
MS Project	Pazar payının en büyüğüne sahiptir.	https://products.office.com/en-us/project/project-and-portfolio-management-software
Open Workbench	Ms Project'ten öncekidir. Açık kaynak sınıfında alternatiftir.	http://www.ca.com
Primavera	Oracle tabanlı yazılımdır. Son teknoloji ürünleri içerir.	https://www.oracle.com/applications/primavera/index.html
Daptiv	İşletmeler için kolayca özelleştirilebilir. Esnekler.	http://www.daptiv.com/products.htm
Innotas	Bulut Teknolojisini kullanır ve esnekler.	https://www.innotas.com/solutions-project-portfolio-management
Power Steering	Finans modülünün kuvvetiyle öne çıkmaktadır.	http://powersteeringsoftware.com/ppm-software/project-portfolio-management/
Mile Stone	Şirketin kendi profesyonel ürünüdür.	http://www.kidasa.com/kidasa-products.html
AMS Real Time	SAP, Oracle, SQLve ODBC veri tabanları ile bütünleşik yazılımdır.	http://www.amsrealtime.com/
Artemis 7	Nükleer ve savunma sanayii projeleri için geliştirilmiştir.	http://www.aisc.com/a7
Fasttrack Schedule	ACE Yazılım, iyi grafik çıkışı ve Mac desteği vardır.	http://www.aecsoftware.com/
UMT	Proje yönetimi yazılımı ve uzman danışmanlık sunar.	http://www.umt.com/
Augeo6	Avrupa merkezli bir yazılımdır.	http://www.augeo.com/page/augeo6
QSM	Yazılım geliştirme için tasarlanmış özel uygulama serisi	http://www.qsm.com/
Clockware	Süreç izlemeyi web tabanlı gerçekleştiren ve MS Project ile bütünleşik	http://www.clockware.com/default.htm
Project Kick Start	Kolay kullanıma yönelik geliştirilmiştir.	http://www.projectkickstart.com/
Turbo Project	Kolay, ucuz ve aşamaların yönetilmesiyle kolay planlama sağlar.	http://www.turboproject.com/
Worktube	İşletmenin farklı süreçleri için çözüm sunan bir firma	http://www.workcube.com/urunler/pms-proje-yonetim-sistemi/
Okyanus Teknoloji	Masraf Yönetimi ile öne çıkıyor.	http://www.okyanusteknoloji.com/Cozumler/Proje-Yonetim-Sistemleri
Etkasoft	Bulut üzerine konuşlanmış web tabanlı bütünleşik mobil uygulamalar kümesidir.	http://www.etkasoft.com/proje-yonetim-sistemi.aspx

BÖLÜM 2: TÜRKİYE MUHASEBE STANDARTLARI AÇISINDAN YAZILIMLARIN DEĞERLENDİRİLMESİ

Yazılım geliştirme, çeşitli yönler içeren, bir dizi sürecin olduğu biraz karmaşık bir konudur. Bir önceki bölümde yazılım geliştirme süreçleri ve modellerinden bahsedilmiştir. Bu bölümde yazılım edinme ve geliştirme amaçlarına göre kategorize etmek ve her bir amaca göre ilgili TMS standardının görüşlerini açıklayıp yorumlamak yer alacaktır.

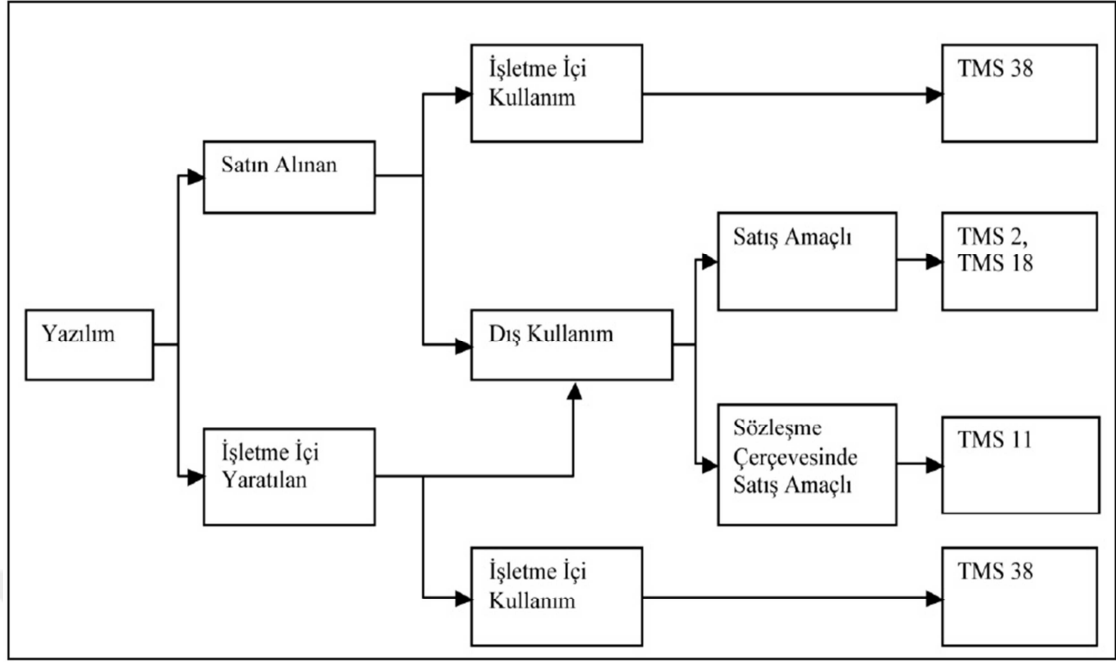
Yazılımların muhasebeleştirilerek raporlanması için yazılımın kullanım alanının ve geliştirme amacının bilinmesi gerekmektedir. Bu konuda ne yapılmasını kendi açlarından izah eden Uluslararası Muhasebe Standartları (IAS), Amerikan Yeminli Mali Müşavirler Enstitüsü (AICPA) ve Amerikan Finansal Muhasebe Standartları Kurulu (FASB) 'da çeşitli standartlarda hükümler bulunmaktadır. Ancak Türkiye Muhasebe Standartları (TMS), IAS'ın Türkçe tercümesi olduğundan ağırlıklı olarak IAS hükümleri, yani TMS eşliğinde ilerlenecektir.

Literatürdeki genel araştırmalardan sonra yazılımların TMS çerçevesinde raporlanabilmesi için yukarıda bahsedilen kategorileşmeyi özetleyen şekil 10 yararlı olacaktır.

Şekil 10'da yazılım üretim ve temini açıklanmıştır. Buna göre her sonucun farklı bir standart ile alakalı olduğu ifade edilmektedir. Şekil 10'a göre yazılım temin etmede iki amaç vardır. Ya işletme içinde kullanmak ya da dışa servis ederek ekonomik fayda sağlamaya çalışmaktır.

İşletme içinde kullanma amacı da iki şekilde gerçekleştirilebilir; ya işletme kendi bünyesinde yazılım geliştirir, ya da üretmeyip dışardan hazır olarak yazılım satın alır.

Dışa servis ederek (satış) ekonomik fayda sağlamak da ikiye ayrılmaktadır. Ya bir sözleşmeye bağlı olarak yazılım üretimi taahhüt edilir ya da sabit bir yazılım üretilir. Bunu ihtiyaçlara cevap verme açısından izah etmek konunun anlaşılmasını kolaylaştıracaktır.



Şekil 10: Yazılımların Kullanım Amaçları ve TMS Yaklaşımı

Kaynak: (Suermann, 2006: 134)

İhtiyaçlar bazen müşteriden bazen de yazılım firması tarafından belirlenerek yazılım üretimi yapılmaktadır. Üretim, ihtiyacın belirlendiği tarafa göre ikiye ayrılmaktadır. Bunlar şunlardır:

- Yazılım firmasının piyasadaki **genel** bir ihtiyaca odaklanarak ürün geliştirmesi
- Müşterinin kendi **özel** ihtiyacı için bir yazılım firmasına sipariş vermesi

Birinci tip üretimde, yazılım firması piyasada birçok kişinin (genel) ihtiyaç duyduğu bir soruna çözüm üretmek için yazılım geliştirir ve piyasada satışa sunar. Bu gibi ihtiyaçların nasıl karşılanacağı genelde belirli kurum ve kuruluşlarca standartları önceden belirlenmiştir veya piyasadaki lider firmanın ürününe göre geliştirme yapılır. Kısacası ne yapılması gerektiği belirgindir. Müşteri de kendi pazar araştırmasından sonra alternatifler arasından kendisine göre en rasyonel olan yazılımı satın alır. Örneğin; tüm işletmelerin iktisadi olaylarını muhasebe ilkelerine uygun şekilde kayıt altına almaları ve gerektiğinde raporlar üreterek ilgili kurumlara beyanlarını yapmaları gerekmektedir. Bunun gibi genel ihtiyaçlar için bazı yazılım firmaları da Muhasebe Paket Programları yazarak piyasada satarlar. Örneğin; Logo, Luca, Eta, Micro, Link, vs. İkinci tip üretimde, geliştirilecek yazılımın neye çözüm sağlayacağı, hangi ihtiyaçları karşılayacağı ve hangi niteliklerinin olacağı müşteri talepleri doğrultusunda belirlenir.

Yani, müşteri kendi özel ihtiyacı için bir çözüm talep eder ve yazılım firması da söz konusu işi yapmayı yüklenerek taahhüt eder. Bu tip bir yazılım üretimi sürecinde ne yapılacağı birinci kategorideki gibi basit ve anlaşılır olmayabilir. İstenen işin müşteri tarafından anlatılması ile yazılım ekibi tarafından algılanması farklı olabilir. Sonuç olarak üretilen yazılım, ihtiyacı karşılamayabilir. Bu tip bir ürün geliştirmenin kendine has dinamikleri olduğundan dikkat edilmesi gerekenler vardır. Bu konular yazılım mühendisliği, yazılımcılara tavsiyeler gibi kaynaklarda detaylıca anlatılmaktadır.

Siparişe dayalı üretime örnek vermek gerekirse, bir üniversitede kullanılan öğrenci bilgi sisteminde devamlılıkların otomatik olarak kayıt altına alınması, takibi, hesaplamaların yapılabilmesi ve notlara dönüşebilmesi için öğrenci kartlarının bir okuyucu tarafından okunarak öğrenci bilgi sistemine aktarılması sipariş edilmiş olsun.

Bu ihtiyacı karşılamak için mevcut öğrenci bilgi sisteminin algılayabileceği bir arayüz yazılarak kart okuyuculara temas ettirilen çipli öğrenci kartlarından veriler okunabilmeli ve öğrenci bilgi sistemine aktarılabilmelidir.

Mevcut yapının üzerine (öğrenci bilgi sistemi) yeni taleplerde bulunmak özel talep olduğu gibi ortada hiçbir yapının olmadığı durumlarda istenenlerin hepsi özel taleptir ve sipariş üzeri geliştirilir.

Yazılım satışlarında muhakkak tarafların hakları güvence altına alınmalı ve bunun için satış ve taahhüt sözleşmeleri yapılmalıdır Bilişim sektöründe yazılım satışı ile ilgili sözleşme türleri şunlardır;

- a) Lisans Sözleşmesi: “*Bilgisayar yazılım evleri ile kullanıcılar arasında geçerli olan ve yazılım paketinin satın alınması ile kabul edildiği varsayılan yasal anlaşmadır.*” (www.bilisimsozlugu.net, 2015)
- b) Siparişe Dayalı Üretim Sözleşmesi: Alıcının taleplerine uygun yazılım geliştirmelerinin yapılabilmesi için işin konusu ve detayları, işin teslim süreleri, tarafların yükümlülükleri ve ihlallerde yapılması gereken cezai işlemlerin açıklandığı sözleşmelerdir.

Bu projelerde, toplam yapının parçalara ayrılarak geliştirilmesi ve teslim sürelerine uyarak teslim edilmesi en çok rastlanan geliştirme modelidir. Bu sebeple devamlı olarak müşteri onayına sunulan üretim parçaları bulunmaktadır.

Kısaca, geliştirilen yazılımların satışı ikiye ayrılmaktadır. Bunlar; paket olarak yazılanlar (genel amaç) ve müşterinin taleplerine göre özel geliştirilenlerdir.

Yazılım geliştirme amaçlarına göre şekil 10'dan de takip edilebileceği üzere farklı TMS standartları hükümlerince muhasebeleştirilmesi gerektiği anlaşılmaktadır. Buna göre; satın alınan ve işletme içi geliştirilen yazılımlar TMS 38 – Maddi Olmayan Duran Varlıklar, sözleşme çerçevesinde satılan yazılımlar TMS 11 – İnşaat Sözleşmeleri, paket hâlinde üretilen genel amaçlı yazılımlar ise TMS 2 – Stoklar ve satışların hasılatı da TMS 18 – Hasılatlar standartlarının hükümlerince muhasebeleşmesi gerekmektedir.

Konumuz yazılım geliştirme harcamalarının sınıflandırılması, üretim maliyetinin hesaplanması ve muhasebeleştirilmesi olduğundan genel ihtiyaca cevap veren satış amaçlı geliştirilen yazılımlara dair olan kısım araştırma dışında tutulmuştur. Çünkü daha çok satılan ürünün maliyetinin nasıl değerlendirilmesi gerektiğine dair modellere ihtiyaç duyulan bir alandır.

2.1. TMS-38'e Göre Yazılımların Değerlendirilmesi

38 numaralı Türkiye Muhasebe Standartları, Maddi Olmayan Duran Varlıklar ile ilgilidir. Bu standartta yer alan konumuzla alakalı başlıklar ve maddeler ortaya konup açıklanacaktır. Net açıklanmayan durumlar için Amerikan Yeminli Mali Müşavirler Enstitüsü (American Institute of Certified Public Accountants - AICPA) ve Amerikan Finansal Muhasebe Standartları Kurulu'ndaki (Financial Accounting Standards Board – SFAS) ilgili maddeler incelenmiştir. Böylece tartışmaya açık olan kısımlara destek bulunmuş ve hesaplama modeli oluşturulmuştur.

2.1.1. Maddi Olmayan Duran Varlıklar Kapsamı ve Yazılımlar

Standart kapsamında maddi olmayan kalemler ve varlık adında iki kavramdan bahsedilmektedir. Öncelikle yazılımların bu standarda göre değerlendirilip değerlendirilmeyeceğine dair bazı tanımlamaları ve özellikleri incelemek gerekir. Standarda göre Maddi Olmayan Kalemler ve Maddi Olmayan Varlıklar şunlardır;

- a) Maddi Olmayan Kalemler: *“İşletmeler sıklıkla kaynak tüketir veya bilimsel ya da teknik bilgi, yeni süreç veya sistemlerin tasarım ve uygulanması, lisans, fikri mülkiyet hakları, piyasa bilgisi ve markalar (marka isimleri ve yayın hakları*

dâhil) gibi maddi olmayan kaynakların elde etme, geliştirme, bakım veya iyileştirilmesi sırasında çeşitli borçlar yüklenirler. Bu geniş kapsamlı başlıklar altındaki kalemlerin yaygın örnekleri; bilgisayar yazılımı, patentler, telif hakları, sinema filmleri, müşteri listeleri, ipotek hizmeti sunma hakları, balıkçılık lisansları, ithalat kotaları, isim hakları, müşteri ve tedarikçi ilişkileri, müşteri sadakati, pazar payı ve pazarlama haklarıdır.” (TMS-38, 2015: md.9)

“Yukarıda tanımlanan bütün kalemler, bir maddi olmayan duran varlık, örneğin belirlenebilirlik, bir kaynak üzerindeki kontrol ve gelecekteki ekonomik yararının varlığı gibi, tanımını karşılamaz.” (TMS-38, 2015: md.10)

b) Maddi Olmayan Varlık: Maddi olmayan kalemler arasında sayılan varlıkların, belirlenebilirlik, kontrol ve gelecekteki ekonomik yararların varlığı şartlarını sağlayanlardır. Bu şartlar şu şekilde açıklanmıştır.

- Belirlenebilirlik şartı: Maddi olmayan duran varlıklar şerefiyeden ayrılabilir olmalıdır. Bunun için ayrı bir şekilde ölçülebilir ve raporlanabilir olmalıdır ki şerefiyeden ayrılabilir olsun. Bu manadasatılabilir, devredilebilir, lisans altına alınabilir, kiralanabilir ya da takas edilebilir olmalıdır. (TMS-38, 2015: md.12)
- Kontrol şartı: “İşletmenin ilgili varlıktan ortaya çıkan gelecekteki ekonomik yararları kullanabilme ve başkalarının bu yararları erişimini kısıtlama gücü olması durumunda, bu varlıktan ekonomik yarar sağlama gücü var demektir.” (TMS-38, 2015: md.13)
- Gelecekteki ekonomik yararlar şartı: “Maddi olmayan duran varlıktan beklenen gelecekteki ekonomik yararlar; ürün ve hizmet satışından sağlanan geliri, maliyet tasarruflarını ya da işletme tarafından varlıkların kullanımından kaynaklanan diğer yararları içerebilir.” (TMS-38, 2015: md.17)

Yukarıdaki kriterler incelendiğinde yazılımlar gelecekteki ekonomik yarar şartını sağlamaktadır. Zaten yazılımı temin etmenin amacı budur. İşletme, temin ettiği yazılım üzerinde kontrol gücüne sahiptir, çünkü yazılımın kullanılmasını istediği zaman başlatabilir veya istediği zaman durdurabilir. Satın alınan veya geliştirilerek temin edilen yazılımların elde etme bedelleri tespit edildiğinde işletme şerefiyesinden ayrı

olarak kaydetmek, izlemek ve raporlamak mümkündür. Satın alınan yazılımların fatura bedelleri belli olduğundan problem olmamaktadır. Ancak geliştirilen yazılımların bedellerinin tespiti açıklık getirilmesi gereken bir konudur. Geliştirme harcamalarının tespit edilip kayıt altına alınabildiği durumlarda TMS 38 kapsamınca varlık olarak kabul edilir.

2.1.2. Satın Alınan Yazılımlar ve Değerlendirilmesi

Satın alınan yazılımlar her zaman yukarıdaki şartları sağladığı kabul edilir. (TMS-38, 2015: md.25) Bu nedenle satın alınan yazılımlar TMS 38 standardına uygun olarak muhasebeleştirilmelidir

“Ek olarak, ayrı olarak elde edilen bir maddi olmayan duran varlığın maliyeti, genellikle güvenilir bir şekilde ölçülebilir. Bu, özellikle, satın alma bedelinin nakit veya diğer parasal varlıklarla ödenmiş olması durumunda geçerlidir.” (TMS-38, 2015: md.26) Ayrı olarak elde edilen ifadesi alan taraf için hem sipariş usulü tedariki hem de paket yazılım tedarikini kapsar. Alan taraf olmak yeterlidir.

Bir yazılımın satın alma fiyatı işletmenin elde etme maliyetini oluşturur. Buna ilave olarak eklenebilecek bazı harcamalar daha olabilir. Bunlar şunlardır: (TMS-38, 2015: md.27)

1. *“İthalat vergileri ve iade edilmeleri mümkün olmayan satın alma vergileri de dâhil, ticari iskontolar ve indirimler düşüldükten sonraki satın alma fiyatı,*
2. *Varlığı, amaçlanan kullanımına hazır hâle getirmeye yönelik, doğrudan varlıkla ilişkilendirilebilen herhangi bir maliyet.”*

Varlıkla doğrudan ilişkilendirilebilen maliyetlere ilişkin örneklere aşağıda yer verilmiştir: (TMS-38, 2015: md.28)

1. *Doğrudan, varlığın çalışabilir duruma getirilmesi sırasında oluşan (“TMS 19 Çalışanlara Sağlanan Faydalar” Standardında tanımlandığı gibi) çalışanlara sağlanan fayda maliyetleri;*
2. *Doğrudan, varlığın çalışabilir duruma getirilmesi sırasında oluşan mesleki ücretler,*
3. *Varlığın düzgün çalışıp çalışmadığının testine yönelik maliyetler.*

Bu duruma göre, yazılımın çalışır hâle gelinceye kadar yapılan işlerden direkt ilişkili olan işlemler için yapılan harcamalar da yazılımın maliyetine eklenmelidir.

“Maliyetlerin, bir maddi olmayan duran varlığın defter değerinde muhasebeleştirilmesi işlemi, ilgili varlığın yönetimce planlanan şekilde çalışabilir duruma gelmesi durumunda biter. Bu nedenle, bir maddi olmayan duran varlığın kullanımında ya da tekrar düzenlenmesi sırasında oluşan maliyetler, söz konusu varlığın defter değerine dâhil edilmez.” (TMS-38, 2015: md.30)

Yazılımlar, fiziksel olmadıkları için işletmeler somut olarak bir şey satın almazlar ancak satın alınan yazılımdan ileriki dönemlerde faydalanmak mümkündür. Bu sebeple yazılım için ödenen ücret, kullanıma hazır hâle getirmeye yönelik ve değerini arttırıcı harcamalar aktifleştirilmelidir. İşlerliğinin sürdürülmesine yönelik harcamalar ise aktifleştirilmez.

2.1.3. İşletme İçi Oluşturulan Yazılımlar ve Değerlendirilmesi

İşletmelerin yazılım üretimi, iki temel amaçtan birisi içindir. Ya kendi kullanımları için, ya da dışa servis ederek (satış) ekonomik yarar sağlamak içindir.

İşletmeler bazen kendi ihtiyaçlarını karşılamak üzere dışarıdan ne hazır yazılım satın alır, ne de sipariş vererek tedarik eder. Bunun yerine işletme bünyesinde kurulan bir yazılım ekibi tarafından kendi iş süreçlerine cevap verebilen bir yazılım geliştirmeyi tercih eder. Bu aynı zamanda işletmenin uzun dönemli bir stratejik yönetsel, ekonomik ve finansal kararıdır. Muhtemelen daha fazla maliyete katlanarak elde edilecek varlıktan, kısa vadede ziyade uzun vadede yararlanma beklentisi olacaktır. (Dulaney, 2000: 34) Buna kararlılık göstermek oldukça zor birkaç dönem yaşatacaktır. Buna rağmen ihtiyaçlara anında müdahale ve tam kontrol gibi avantajlarla birlikte tüm sistemin sorumluluğunun da üstlenilmesi gerekecektir.

TMS 38 standardı, işletme içinde oluşturulan yazılımları yukarıda anlatılan şekilde satış veya kullanım amacına göre bir ayırım yaparak ele almamıştır. Sadece işletme içinde üretiminden bahsetmektedir. Ancak standardın isminde geçen “duran varlık” tanımına uygun olan işletme içi üretim şekli iç kullanıma yöneliktir. Çünkü iç kullanım için geliştirilen yazılım duran varlık olarak aktifte yer alabilir. Satış amaçlı geliştirilen yazılımlar stok özelliği taşıyacaktır ancak dönen varlık özelliği yani tükenmesi

olmadığından tartışmaya açıktır. Bu sebeple standardın ifadeleri iç kullanım amacı ile geliştirilen yazılımlara yöneliktir.

Satış amaçlı geliştirilen yazılımların muhasebeleştirilmesi için TMS 38 hükümleri sadece yeterli değildir. Zaten, şekil 10 incelendiğinde dış kullanım (satış) amaçlı geliştirme olduğunda TMS 38'e değil, diğer standartlara dikkat etmek gerektiği anlaşılmaktadır.

TMS 38'in 51. maddesine göre kısaca, işletme içinde oluşturulan yazılımların kayıt altına alınabilmesi için üretim maliyetinin güvenilir bir şekilde ölçülebilmesi gerekir. Standart bu amaçla maliyet ve gider kavramlarını ayırabilmek için üretim safhaları tanımlamıştır ve bu safhalar ile harcamaların türünü eşleştirerek bir düzen sağlamak istemiştir. Buna göre varlık üretimini ikiye ayırmıştır. Bunlar şunlardır:

a) Araştırma Safhası: “Yeni bir bilimsel ya da teknik bir bilgi ve anlayış kazanma amacıyla üstlenilen özgün ve planlı incelemedir.” (TMS-38, 2015: md.8)
Araştırma faaliyetleri ile ilgili örnekler şunlardır: (TMS-38, 2015: md.56)

- *Yeni bilgi elde edilmesine yönelik faaliyetler;*
- *Araştırma bulguları ve diğer bilgilerin uygulanmasına yönelik olarak değerlendirme ve nihai seçim araştırması;*
- *Malzeme, aygıt, ürün, süreç, sistem veya hizmetler için alternatif araştırması ve*
- *Yeni veya geliştirilmiş malzeme, aygıt, ürün, süreç, sistem veya hizmetlerin olası alternatiflerinin oluşturulması, tasarlanması, değerlendirilmesi ve nihai seçimi.*

Araştırma safhasındaki harcamalar gider olarak muhasebeleştirilir. (TMS-38, 2015: md.54)

b) Geliştirme Safhası: “Ticari üretim ya da kullanıma başlamadan önce, yeni veya önemli ölçüde geliştirilmiş malzeme, aygıt, ürün, süreç, sistem ya da hizmetlerin üretim planı veya tasarımında araştırma sonuçları ya da diğer bilgilerin uygulanmasıdır.” (TMS-38, 2015: md.8) *Geliştirme faaliyetleri ile ilgili örnekler şunlardır: (TMS-38, 2015: md.59)*

- *Üretim ve kullanım öncesi prototip ve modellerin tasarımı, yapımı ve test edilmesi;*
- *Yeni teknoloji içeren alet, model ve kalıpların tasarımı;*
- *Ticari üretim açısından ekonomik olmayan bir ölçekteki pilot tesisin tasarlanması, inşası ve çalıştırılması,*
- *Yeni veya geliştirilmiş malzeme, aygıt, ürün, süreç, sistem veya hizmetler için karar verilmiş bir alternatifin tasarımı, inşası ve denenmesi.*

Geliştirme safhasına ait harcamalar aktifleştirilir. Ancak aşağıdaki şartların tamamının sağlanıyor olması gerekmektedir. (TMS-38, 2015: md.57)

- *Maddi olmayan duran varlığın kullanıma veya satışa hazır hâle gelebilmesi için tamamlanmasının teknik olarak mümkün olması.*
- *İşletmenin maddi olmayan duran varlığı tamamlama ve bu varlığı kullanma veya satma niyetinin bulunması.*
- *Maddi olmayan duran varlığı kullanma veya satma imkânının bulunması.*
- *Maddi olmayan duran varlığın muhtemel gelecek ekonomik faydayı nasıl sağlayacağını belirli olması. Ayrıca, maddi olmayan duran varlığın ürününün veya kendisinin bir piyasasının olması ya da işletme bünyesinde kullanılacak olması durumunda buna elverişli olması.*
- *Geliştirme safhasını tamamlamak ve maddi olmayan duran varlığı kullanmak veya satmak için yeterli teknik, mali ve diğer kaynakların mevcut olması.*
- *Geliştirme sürecinde maddi olmayan duran varlıkla ilgili yapılan harcamaların güvenilir bir biçimde ölçülebilir olması.*

Yazılım üretiminde yukarıdaki şartlardan sonuncusu hariç diğerleri sağlanmaktadır. “Yazılımlarla ilgili olarak araştırma safhasının ne zaman bitip geliştirme safhasının ne zaman başladığının belirlenmesi büyük ölçüde bilanço düzenleyenin yargısına kalmıştır.” (Sumer ve Erer, 2010: 39)

Çalışmamızın en kritik noktası, araştırma ve geliştirme safhalarının yazılım geliştirme süreçleri ile eşleştirebilmektir. Çünkü ancak bundan sonra harcamalar sınıflandırılabilmiş olacak ve muhasebe kayıtlarına geçilebilecektir. (Munter, Moores ve

Sanders, 2006: 159) Aşağıdaki paragrafta altı çizili olan ifadeler yazılım geliştirme süreçleri ile eşleştirmede yardımcı olacak ifadelerdir.

“İşletme içi yaratılan maddi olmayan duran varlığın maliyeti, yönetim tarafından amaçlanan şekilde çalışabilmesi için ilgili varlığın yaratılması, üretilmesi ve hazırlanmasında gerekli olan ve varlıkla doğrudan ilişkilendirilebilen maliyetlerin tümünü içerir. Varlıkla doğrudan ilişkilendirilebilen maliyetlere ilişkin örnekler aşağıdaki gibidir”: (TMS-38, 2015: md.66)

- *Maddi olmayan duran varlığın oluşturulmasında kullanılan veya tüketilen malzeme ve hizmet maliyetleri;*
- *Maddi olmayan duran varlığın oluşturulmasından kaynaklanan çalışanlara sağlanan faydalara ilişkin maliyetler (TMS 19 Standardında belirtildiği biçimde);*
- *Yasal hakkın tesciline yönelik ödemeler,*
- *Maddi olmayan duran varlığın oluşturulmasında kullanılan patent ve lisansların itfa payları.*
- *TMS 23 Standardı, faizin, işletme içi yaratılan bir maddi olmayan duran varlığın maliyetinin bir unsuru olarak muhasebeleştirilmesine ilişkin ölçütleri belirlemiştir.*

Yukarıdaki maddelerden geliştirme maliyeti sayılabilecek olan harcama konularını somutlaştırırsak sırasıyla aşağıdakiler gibi olacaktır. (Putra, 2009: <http://accounting-financial-tax.com>)

- Danışmanlık hizmeti ödemeleri,
- Yönetim giderleri,
- Yazılım ekibinin maaşları,
- Yasal hakkın tesciline yönelik ödemeler,
- Yazılım ve donanım alt yapısı harcamalarının amortisman payları,
- Faiz ödemeleri,

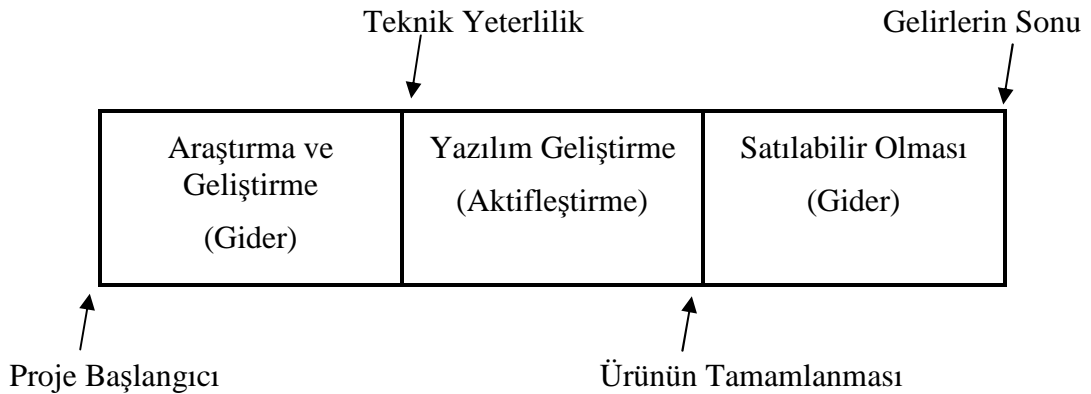
Araştırma ve geliştirme safhalarının yazılım geliştirme süreçleri ile eşleştirmede TMS 38’in 66. maddesinde geçen “yaratılması, üretilmesi, hazırlanması” ifadelerinden yaratılması kısmı tasarıma, üretilmesi kısmı kodlama ve test aşamalarına ve

hazırlanması kısmı da devreye almaya denk düştüğü yorumu yapılabilmektedir. Bu yorumun başka standartlardaki durumu nedir diye araştırma yapılmıştır.

Amerikan Yeminli Mali Müşavirler Enstitüsü (American Institute of Certified Public Accountants - AICPA) ve Amerikan Finansal Muhasebe Standartları Kurulu (Financial Accounting Standards Board – FASB) ilgili maddeleri incelendiğinde kodlama ve test safhalarının kesinlikle geliştirmeden sayıldığı fark edilmiştir. “Tasarım safhası” ile ilgili olarak iki farklı tanımlama ve sınıflandırma çıkmaktadır.

AICPA SOP 98-1 standardının 17. paragrafında geliştirme safhasına dâhil olan yazılım geliştirme süreçlerini sıralarken “tasarım” açıkça belirtilmiştir. Tanıma göre tasarım faaliyetleri yazılım konfigürasyonu ve yazılım arayüzü şeklinde ifade edilmiştir. Geliştirme safhasında bulunan süreçlerdeki harcamaların dâhili ve harici olanları aktifleştirilir. (SOP-98-1, 1998: md. 17)

SFAS 86 4. maddesinde ise teknik yeterlilik anlamında “technological feasibility” kavramı ile ayrıştırma yapılmıştır. Kavrama göre; üretimin tüm planlama, tasarım, kodlama ve test işlemlerine dair fonksiyonları, özellikleri ve teknik performans gereksinimlerinin belirlendiği aşamadır. Bunun için detaylı program listesinin yapılması gerekmektedir. Bunun anlamı da yazılımı üretmek için gerekli olan donanım, yazılım teknolojisi ve becerilerinin tasarlanması ve belgelenmesidir. (SFAS-86, 1985: md.4) Zaten bahsedilen bu nokta tasarım sürecinin sonunda elde edilir olduğundan bu standart tasarım sürecini geliştirme safhası olarak görmemektedir. Şekil 11’de görüleceği üzere bu standart, geliştirmenin teknik yeterlilik ile başlayacağını ve diğer kavramları özetlemektedir.



Şekil 11: SFAS 86’ya Göre Aktifleştirme ve Gider Dönemleri

Kaynak: (Mulford ve Roberts, 2006: 4)

Müşteri ihtiyaçlarının anlaşılıp buna uygun çözümleri üretebilmek için arayüz ve veritabanı tasarımlarının oluşturulması, ortaya satılabilecek bir değer çıkarmaktadır. Örneğin; <http://www.wix.com> gibi sitelerden hazır tasarlanmış şablon üzerinden internet sitesi geliştirme imkânı satmaktadır. Buna ilave olarak, gerek TMS-38 8. maddesindeki geliştirme tanımı, gerekse SOP 98-1 17. maddesi incelendiğinde tasarım aşamasının geliştirmeden olduğu anlaşılmaktadır. (Mirza ve Holt, 2011: 564)

İç içe geçmiş bir başka konu ise “Devreye alma” aşamasıdır. Bu aşamanın içindeki “kullanıcı eğitimi” kalemi sözleşme taahhütlerinin içindeyse sözleşme kapsamında değerlendirilerek geliştirme sayılması gerekmektedir. Yazılım işletmelerinin eğitim sözleşmeleri diye ayrı bir gelir türü vardır. Eğitim hizmeti için mümkünse ayrı sözleşme düzenlenmesi, işleri birbirinden ayırabilmek için en idealidir. Böyle olduğunda eğitim ve bakım hizmetlerinin harcamaları gider olarak kayıt edilmez. (Suermann, 2006: 137)

Kavramları tablo 8’de özetlemek, konunun anlaşılması açısından faydalı olacaktır.

Tablo 8: Standartların Yazılım Geliştirme Aşamaları Hakkındaki Kabulleri

Yazılım Geliştirme Aşamaları	TMS 38	SOP 98-1	SFAS-86
İsterlerin Belirlenmesi ve Analiz	Araştırma	Araştırma	Araştırma
Tasarım	Geliştirme	Geliştirme	Araştırma
Kodlama	Geliştirme	Geliştirme	Geliştirme
Test	Geliştirme	Geliştirme	Geliştirme
Devreye Alma	Geliştirme	Geliştirme	Geliştirme

Kısaca hangi standarda göre raporlama yapılacaksa ilgili standardın “geliştirme safhası” diye kabul ettiği aşamalar için yapılan harcamalar, işletme içi üretilen yazılımın maliyetini oluşturacaktır. Maliyet olarak hesaplanan değerler, 263 Araştırma ve Geliştirme Hesabına “Borç” olarak kayıt edilirken bir varlık veya kaynak hesabı da “Alacak” olarak kayıt edilir.

TMS 38’e göre raporlama yapmak, US-GAAP’a göre raporlama yapmaktan daha fazla aktifleştirme sonucunu doğuracaktır. Aynı zamanda beklenen ekonomik faydanın ortadan kalkması durumlarıyla da daha fazla karşılaşılacaktır. Çünkü geliştirmenin daha önceki aşamalarında aktifleştirmeye başlanmaktadır ve yolunda gitmeyen işler olabilir. Bu gibi durumlarda aktifleştirilmiş olan varlığın tamamı gider olarak yazılır ve bilanço dışına alınır. (Ergün, 2001: 75)

Harcamaları, yazılım geliştirme süreçlerine ve araştırma-geliştirme safhaları cinsinden gruplandıran ve rapor üreten bir yazılım yoktur. “1.4.2. Yazılım Proje Yönetimi” kısmında Proje Yönetim Yazılımlarından bahsedilmiştir. Bu yazılımlar, birçok projenin süreçlerinin rahatlıkla takip edilebildiği ve proje yöneticisinin işini kolaylaştıran, çok çeşitli süreç raporları üretebilen yazılımlardır. En iyi yaptıkları işler arasında projenin zaman, iş gücü ve kaynak planlamalarını yapmaları gelir. Bu üç kritik ögenin birbiriyle ilişkisini Gantt şemaları ile raporlamak ve istendiğinde maliyet raporlarını çıkarabilmektedirler. Yazılım üretmek için kullanılan diğer yazılımlar da kaynak olduğundan yapılacak işler ile süreçlerde geçecek süreler bu tür proje yönetim yazılımları kullanılarak takip edilebilir ve raporlar üretilebilir. (Yılmaz ve Calayoğlu, 2015a: 190)

Türkiye Muhasebe Standartları gerçeğe uygun raporlama istediğinden yazılım üretimi yapan işletmelerin araştırma ve geliştirme safhalarını birbirinden ayırarak muhasebe kayıtlarını tutmaları gerekmektedir. Zira “İşletme, dönem içinde gider olarak muhasebeleştirilen araştırma ve geliştirme harcamalarının toplam tutarını kamuoyuna açıklar.” (TMS-38, 2015: md.126) ifadesi gereği safhaların ayrıştırılarak muhasebeleştirilmesi ve raporlanması gerekmektedir. Eğer işletme, araştırma ve geliştirme harcamalarını ayırtamazsa tüm harcamaları araştırma safhasında yapmış sayılır ve dönemin giderine kaydetmesi gerekir. (TMS-38, 2015: md.53)

2.1.4. Yazılımın Yararlı Ömrü ve Değerlendirilmesi

Yazılımların ilk kayıtlarından sonra bu varlıkların itfa paylarının hesaplanması ve ilgili hesaplara kayıt edilmesi gerekmektedir. Bu konuda TMS 38 standardı iki kavramdan bahsetmektedir. Bunlar şunlardır:

- a) Sınırlı Yararlı Ömür: Varlığın işletmeye olan faydasının önceden belirlenebilen biçimde olduğu durumlardır.
- b) Sınırsız Yararlı Ömür: Varlığın işletmeye olan faydasının ne zaman biteceğinin belirlenemediği durumlardır.

Maddi olmayan duran varlığın muhasebeleştirilmesinde yararlı ömür dikkate alınır. Sınırlı yararlı ömre sahip maddi olmayan duran varlık itfaya tabi iken, sınırsız yararlı ömürlü maddi olmayan duran varlık itfaya tabi olmaz. (TMS-38, 2015: md.89)

Yukarıdaki hüküm gereğince sınırlı yararlı ömrün nasıl belirleneceği, neyin temel alınması gerektiği araştırılmalıdır. Standardın 90. maddesinde ifade aynen şöyledir. “*Bir maddi olmayan duran varlığın yararlı ömrünün belirlenmesinde, aşağıdakiler de dâhil olmak üzere birçok unsur dikkate alınır: (TMS-38, 2015: md 90)*

- a) *Varlığın işletme tarafından beklenen kullanım süresi ve başka bir yönetim ekibi tarafından etkin olarak kullanılıp kullanılmayacağı;*
- b) *Varlığa özgü ürün yaşam süresi ve aynı şekilde kullanılan benzer varlıklara ilişkin yararlı ömür tahminlerine ilişkin kamuoyu bilgisi;*
- c) *Teknik, teknolojik, ticari veya diğer kullanımdan kaldırılma nedenleri;*
- d) *Varlığın içinde bulunduğu sektörün istikrarı ve varlıktan sağlanan ürün veya hizmetlere ilişkin pazar talebindeki değişiklikler;*
- e) *Rakiplerden veya potansiyel rakiplerden beklenen eylemler;*
- f) *Varlıktan beklenen gelecekteki ekonomik yararları sağlamak için gerekli bakım harcamalarının düzeyi ile işletmenin bu düzeye ulaşma kapasite ve niyeti;*
- g) *Varlık üzerindeki kontrol süresi ve varlığın kullanımı ile ilgili, buna ilişkin kiralama bitiş tarihi gibi, yasal ve benzeri sınırlamalar;*
- h) *Varlığın yararlı ömrünün, işletmenin sahip olduğu diğer varlıkların yararlı ömürlerine bağlı olup olmadığı.”*

Satın alınan yazılımların yararlı ömrü için TMS-38 md. 90’ın “g” şikkını incelemek gerekir. Varlığın kullanım süresini sınırlayan lisans sözleşmesindeki süre işaret edildiği yönünde yorumlanabilir. Yani, lisans sözleşmesi kaç yıllık yapılırsa satın alınan yazılımın yararlı ömrünün de o kadar olduğu kabul edilebilir. En somut gösterge budur.

Kullanım amaçlı üretilen yazılımların yararlı ömrü için TMS-38 md. 90’ın “a” şikkını incelemek gerekmektedir. İfade oldukça nettir ancak rasyonel bir beklenti için de diğer maddeler fikir vermektedir. Örneğin; TMS-38 md. 90’ın “f” ve “h” şıkları ile beraber düşünüldüğünde işletme yönetimi, geliştirilen varlığın ömrüne bir sınır koyabilir. Bunu da yönetim kurulunda karara bağlayabilir.

“*Bir maddi olmayan duran varlığın yararlı ömrünü etkileyen hem ekonomik hem de yasal etkenler olabilir. Ekonomik etkenler, işletmenin gelecekteki ekonomik yararlarının süresini belirler. Yasal etkenler ise, işletmenin söz konusu yararlarla olan erişimini*

sürdürebildiği süreyi sınırlayabilir. Yararlı ömür, bu etkenler tarafından belirlenen sürelerden kısa olanıdır.” (TMS-38, 2015: md.95)

Aksi olarak, geliştirilen varlığa dair ilk etapta, faydalanmanın sonu olarak bir tarih kararlaştırılmaz ve sınırsız yararlı ömürlü olarak da karara bağlanabilir. *“Bir maddi olmayan duran varlığın yararlı ömrü çok uzun, hatta sınırsız olabilir. Belirsizlik, bir maddi olmayan duran varlığın yararlı ömrünün tahmininde ihtiyatlılık ilkesinin uygulanmasını gerektirir; ancak gerçekçi olmayacak kadar kısa bir ömrün seçilmesini gerektirmez.” (TMS-38, 2015: md. 93)* Bu belirsizlik sınırsız yararlı ömür olarak tanımlanır. Belirlilik olduğu durumda sınırlı yararlı ömür üzerinden işlem görmeye devam eder.

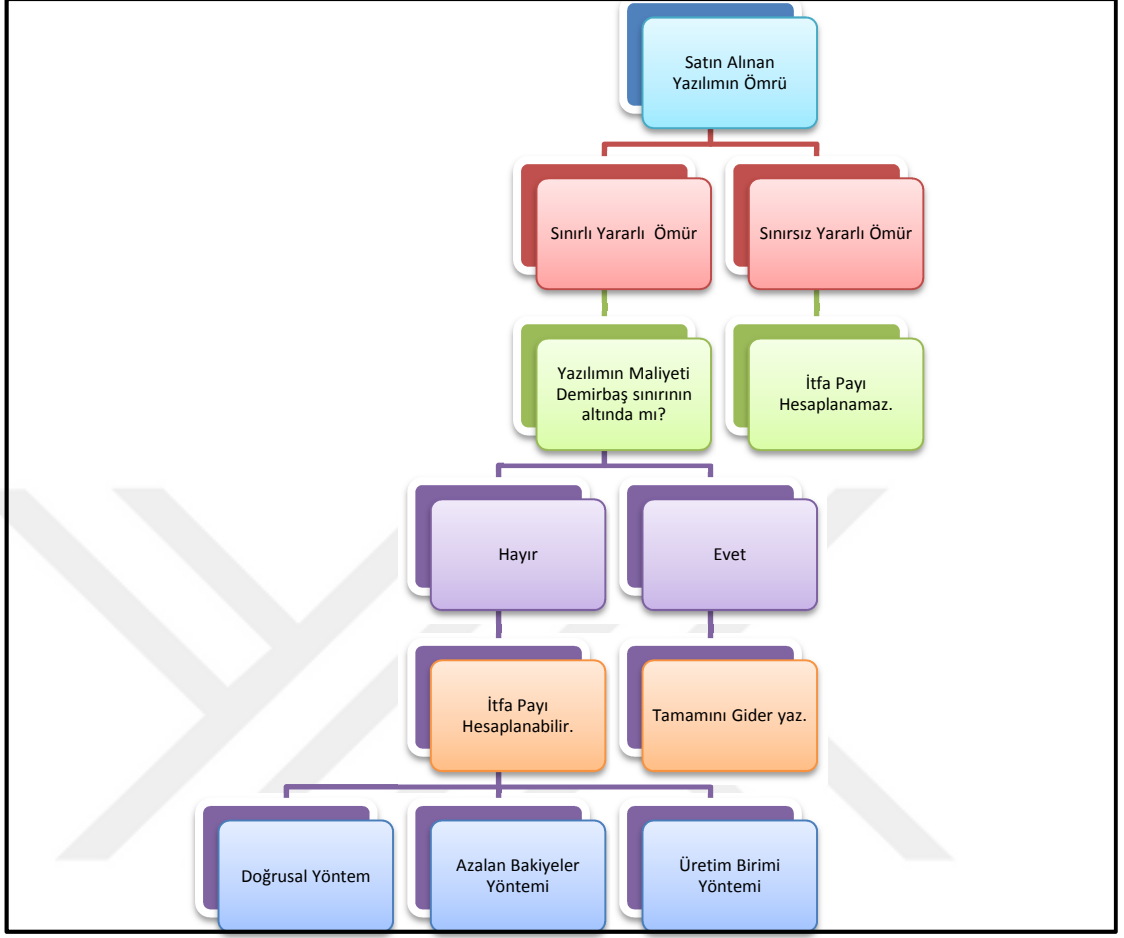
Sınırlı yararlı ömürlü maddi olmayan varlıkların itfa yöntemleri standardın 98. maddesinde belirtildiğine göre doğrusal, azalan bakiyeler ve üretim birimi yöntemlerinden biri olabilir. Bu yöntemlerden hangisinin seçileceği konusunda, varlığın tahmini olarak gelecekte yaratacağı ekonomik faydanın durumuna göre belirlenmesi esasına dayanmalıdır.

Satın alınan yazılımların yararlı ömrü için lisans sözleşmesinin süresine bakmak gerekmektedir. Genelde lisans sözleşmesinin süresi kadar da bakım sözleşmesi yapılmaktadır. Bu süre genelde 1 yıldır. (Munter, 2002: 27) Çünkü genelde tek sözleşmede lisans ve bakım anlaşması yapılır. (Sonnellitter ve Pacter, 1994: 56) Esasen satıcı için her iki işlem için ayrı ayrı sözleşme düzenlemek; gelir ve gider ayrımlarının tanımlanıp takip edilebilmesini kolaylaştıracağından daha uygun olur.

1 yıl kullanım süresi olan yazılımda İtfa yöntemi fark etmeyecektir. Ancak daha uzun süreli kullanım hakkı sağlayan sözleşmelerin muhasebeleştirilmesinde itfa yöntemleri arasındaki farktan dolayı işletmenin bazı avantajları ve dezavantajları olabilir.

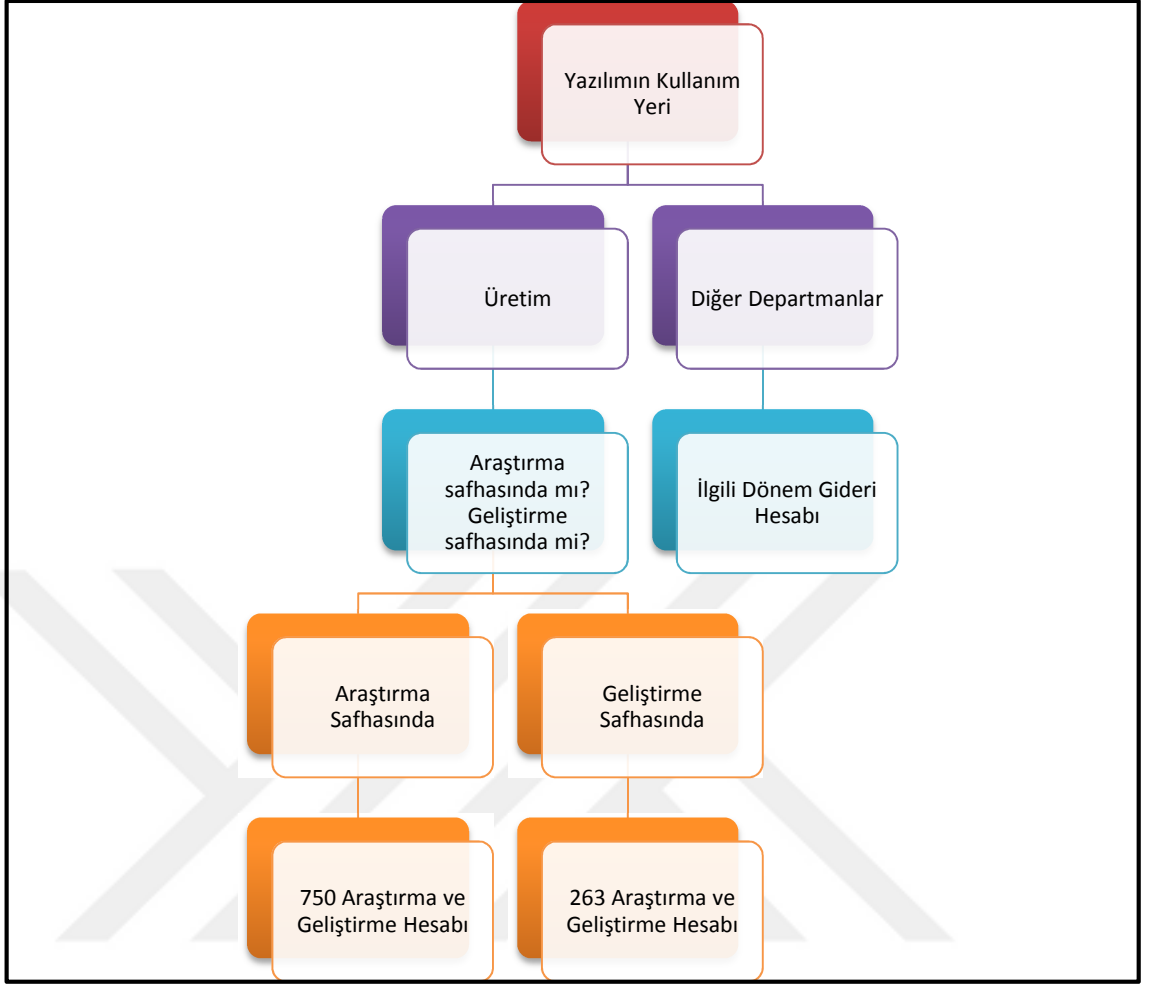
Kullanım amaçlı geliştirilen yazılımlar için sınırsız yararlı ömür tanımlaması yapılabilir. Sınırsız yararlı ömrü olan maddi olmayan duran varlıklar için itfa payı hesaplanamaz ve hesaplanmaz. (TMS-38, 2015: md.107) Çünkü gelecekte yazılımdan ne zamana kadar fayda görüleceği tahmin edilememiştir. Bu durumda aktifteki varlık, amorti edilmez. İlerleyen dönemlerde varlığın faydasında sınırlılık olabileceğinden her dönem sonunda sınırlılığın olup olmadığı gözden geçirilmelidir. Makul gerekçelerle sınırlılık söz konusu ise o seneden itibaren itfa payı hesaplanarak giderleştirilmeye başlanabilir.

Yazılımın itfa payının hesaplanıp hesaplanmaması gerektiğini şekil 12 özetlemektedir.



Şekil 12: Yazılımın İtfa Payının Hesaplanması

Amortisman kaydı, 268 Birikmiş Amortisman Hesabını “alacak” ve yazılımın kullanım amacı veya departmanına göre bir gider hesabını da “borç” yazarak kayıt altına almak gerekmektedir. Ancak, yazılım geliştirmek için kullanılan diğer yazılımların itfa payları, geliştirilen yazılımın defter değerine eklenir. (TMS-38, 2015: md.99) (Akdoğan, 2000: 16) Yazılımın kullanım yerine göre itfa payının kaydedileceği hesaplar şekil 13’de belirtilmiştir.



Şekil 13: Yazılımın Kullanım Yerine Göre İtfa Payının Kaydedilmesi

İşletme yönetimi, satın alınan bir yazılıma dair “artık kullanılmayacak” yönünde karar vermesi durumunda varlığın defter değerinin tamamı zarar yazılır. Bu durum, işletme içinde üretilen yazılımlar için de geçerlidir.

TMS 38’in 100. maddesinde kalıntı değerden bahsedilmektedir. Amortisman hesaplaması yapılırken varlığın maliyetinden kalıntı değerinin çıkarılarak kalan bakiye üzerinden itfa edilmesi gerekmektedir. Ancak dışarıdan satın alınan ve kullanım süresi sınırlı olan bu tür varlıkların satışı ile el değiştirmesi mümkün olmamaktadır. Çünkü işletme, 3. kişilere yazılımın kullanım yetkisini verme, şifre üretme gibi bir imkâna sahip değildir. Dolayısı ile yasal olarak 3. kişilere satış mümkün olmadığından kullanım süresi bittiğinde ekonomik olarak geri kazanılabilecek bir değer olmamaktadır.

Kalıntı değer ancak şu durumda ortaya çıkabilir: yazılım satın alma sözleşmesinde “Kullanım süresinin uzatılmasında belirli bir oranda indirim uygulanır”; gibi bir ifade var ise yazılımın vade sonunda ne kadar ederi olduğu önceden bilinebilir. Ancak bu

durumda kalıntı değerden bahsedilebilir ve itfa hesaplamasına dâhil edilerek muhasebeleştirilebilir.

Kullanım amaçlı geliştirilen yazılımlar, kullanılmaya başlandığı dönemden itibaren itfa payları hesaplanarak muhasebeleştirilir. Ancak yararlı ömrünün belirli bir dönemi tüketildikten sonra geliştirici bakım sayesinde varlığın yararlı ömrü uzar ve değerinde artış meydana gelebilir. Bu gibi durumlarda varlığın net değeri ile yeni yararlı ömür değerleri dikkate alınarak itfa payı hesaplamaya devam edilmelidir. (Sonnellitter ve Pacter, 1994: 49)

2.1.5. Yazılımın Yeniden Değerlenmesi ve Değerlendirilmesi

Yeniden değerlendirme yapabilmek için ürünün aktif bir piyasasının olması gerekmektedir. Varlığın fiyatında anlamlı bir artış veya azalışın olması gerekir. Şartlar sağlanıyorsa ürünün defter değeri ile piyasa değeri arasındaki farka göre gerekli muhasebe kayıtları yapılır. Bu kontrol her sene sonunda yapılmalıdır.

Satın alınan yazılımların temin yollarını hatırlatmak faydalı olacaktır. Daha önceden bahsedildiği üzere işletmeler, sipariş vererek yazılım yaptırabilir, işletme içinde kendi yazılımını geliştirebilir veya hazır-paket olarak satılan yazılımları tercih edebilir. Bu durum aktif piyasa koşulu açısından önemlidir. Çünkü paket olarak satın alınan yazılımlara dair bir liste fiyatı vardır ve bu kamuoyunca bilinebilmektedir. Ancak siparişe dayalı geliştirilmiş yazılım temin eden işletmeler ile yazılım firması özel-birebir sözleşme yapar. Fiyat bilgisinden kamuoyunun haberi olmamaktadır. Bu sebeple bu tür teminlerde aktif bir piyasadan bahsetmek mümkün olmaz. Keza kendi bünyesinde yazılım geliştirmiş olan firmalar için de aktif piyasadan bahsedilemeyeceği için yeniden değerlendirme yolu kapanmıştır.

Ancak, teknolojinin gelişimi, ihtiyaçların çeşitlenmesi, pazarlama teknikleri gibi nedenlerle yazılımlar devamlı olarak yeni versiyonlar şeklinde piyasa sürülmektedir. Dolayısı ile ürünlerin yaşam döngüleri kısa olmaktadır. Satın alınan yazılımların yararlı ömürlerinin kullanım sözleşmesinin bitişi ile sonlanıyor olması ve çoğu lisans kullanım sözleşmesinin süresi 1 yıl ile sınırlı olması durumu, zaten yeniden değerlendirme yapmaya imkân verecek kadar değişimi ortaya çıkartmayacaktır.

Bununla beraber, sözleşme süresi 1 yıldan daha uzun olduğunda yeniden değerlendirme yapmak gerekebilir. Yine de itfa süresi bitmeden yeniden değerlendirme işlemine tabi olacak kadar değer değişikliğine uğraması oldukça zor bir durumdur. Şayet bu şart sağlanmışsa, satın alınan yazılımların üzerine yeni ürünlerin gelmesinden ve ürün yaşam sürelerinin kısa olmasından dolayı değer artışının aksine genelde değer düşüklüğünden bahsedilebilir.

2.2. TMS 11'e Göre Yazılım Taahhütlerinin Değerlendirilmesi

Yazılım endüstrisinin bir başka yönü, işletmelerin dışa servis ettikleri (satış amaçlı) yazılımlardır. (Bakınız: Şekil 10) Yazılım satışı için yazılımın işletme içinde üretimi gereklidir. Ticari firmalar gibi yazılımı belli bir fiyattan alıp kar marjı ekleyerek satmak gibi bir durum söz konusu değildir.

Satış amaçlı geliştirilen yazılımları; sipariş üzerine (özel) geliştirilenler ve hazır-paket (genel) geliştirilenler şeklinde ikiye ayırmak mümkündür.

Sipariş üzerine geliştirilen yazılımlar için müşteri ile yüklenici (yazılım firması) özel bir sözleşme yapar. Böylelikle yazılım firması, detayları tanımlanan işi, vaat edilen zamanda ve özelliklerde-fonksiyonlarda teslim etmeyi taahhüt etmiş olur. Taahhüdün yerine gelmesi bir dönemden daha fazla bir süreyi kapsıyorsa, geliştirme sürecinde katlanılan maliyetler ile tahsilatların dönemsel ilkesine uygun şekilde muhasebeleştirilmesi gerekir.

Bu standardın asıl amacı, sözleşme gelir ve maliyetlerinin gelir tablosuna ne zaman gelir ve gider olarak gösterileceklerini kapsamlı olarak açıklayarak, dönemsel esasına göre gelir ve maliyetleri ilgili dönemlere yansıtılmakta ve mali tabloların karşılaştırılabilirliğini sağlamaktır. (Çatıkkaş ve Şuekinçi, 2013: 136)

TMS 11 standardı her ne kadar inşaat sözleşmelerini zikrederek anlatımını yapsa da tüm sözleşme ile birden fazla mali döneme denk gelecek üretim yapan işletmeler için uyarlanabilir.

Sözleşme çerçevesinde geliştirilen yazılımlar için yapılan harcamalarının hangisinin araştırma hangisinin geliştirme olduğunun kararı yine TMS 38 hükümlerince belirlenir. Buna ilave olarak sözleşme gelirlerinin ve maliyetlerinin dönemlere dağıtılabilmesi için

gerekli düzenlemeler de TMS 11 hükümlerince belirlenir. (Sonneltter ve Pacter, 1994: 54)

Birebir standart maddelerinin kullanımından dolayı metin içlerinde inşaat kelimesi geçecektir. Bundan sonraki her inşaat kelimesinin yerine yazılım kelimesini koyarak okumak, konuyu anlamaya daha yardımcı olur.

2.2.1. Sözleşmenin Tanımı ve Değerlendirilmesi

“İnşaat sözleşmesi; bir varlığın veya tasarım, teknoloji ve fonksiyon ya da nihai amaç veya kullanım açısından birbiriyle yakından ilişkili ya da birbirine bağımlı bir grup varlığın inşası için özel olarak yapılmış bir sözleşmedir.

TMS 11’de sözleşme türü olarak aşağıda tanımlanmış 2 çeşitten bahsedilir. Sözleşme türüne göre sözleşme sonucunun güvenilir tahmin edilme kriterleri değişmektedir.

Sabit fiyatlı sözleşme; yüklenicinin sabit bir sözleşme fiyatını (ihâle bedeli) veya üretim birimi başına sabit bir tutarı (birim fiyat) kabul ettiği ancak belli koşullarda maliyet güncelleştirmesine (eskalasyon) konu olan inşaat sözleşmesidir.

Maliyet artı kâr sözleşmesi; yükleniciye kabul edilebilir ya da başka bir şekilde tanımlanmış maliyetler üzerine bu maliyetlerin bir yüzdesi veya sabit bir tutar eklenerek ödeme yapılan inşaat sözleşmesidir.” (TMS-11, 2015: md.3)

Sabit fiyatlı bir sözleşmeyle başlayan bir projede tahmin edilen harcamalardan daha fazla bir meblağ ile proje teslim edilirse müşteriye ek bir ödeme fatura edilmez. Projenin karlılığı azalmış olur. Maliyet artı kar sözleşmelerinde proje başlangıcına göre eklenen sonraki maliyetler fatura edilirken dikkate alınır ve müşteriye daha fazla tutarda fatura kesilir.

2.2.2. Sözleşmelerin Birleştirilmesi ve Bölünmesinin Değerlendirilmesi

Siparişe göre yazılım üretirken "birden çok varlığı kapsayan bir sözleşmede aşağıdaki durumlar mevcutsa her varlığın inşaatı ayrı bir inşaat sözleşmesi olarak değerlendirilir: (TMS-11, 2015: md.8)

a) Her bir varlık için ayrı teklif verilmiş olması;

- b) Her bir varlığın ayrı bir müzakereye konu olması ve yüklenici ile müşterinin sözleşmenin her bir varlığa ilişkin kısmını kabul veya reddetme hakkının olması,
- c) Her bir varlığın maliyetleri ve gelirlerinin ayrı ayrı belirlenebilmesi.”

Örneğin; alınan bir siparişe göre bir paket programın bir veya birden çok (tamamı değil) modülünün geliştirilmesi durumunda yukarıdaki maddeye göre her bir modül için ayrı ayrı sözleşme yapılması gerekmektedir. Çünkü oluşturulan varlıklar birbirinden bağımsız şekilde teslim edilebilecek şekildedir.

“Bir veya birden çok müşteri ile yapılmış olmasına bakılmaksızın, aşağıdaki durumlar mevcutsa bir sözleşmeler grubu tek sözleşme olarak nitelenir: (TMS-11, 2015: md.9)

- a) Sözleşmeler grubunun tek bir paket olarak birlikte müzakere edilmesi,
- b) Sözleşmelerin birbiriyle aslında genel bir kâr marjına sahip tek bir projenin parçası olarak, çok yakın ilişkili olması,
- c) Sözleşmeler kapsamındaki işlerin aynı anda veya birbirini izleyen bir sırada yapılması.”

Yukarıdaki açıklamalara göre sipariş üzerine geliştirilen bir paket programın içinde her ne kadar modüllerden oluşan alt parçalar olsa da sözleşmedeki ifadeler ve şartlar paketin tamamı üzerine ise bu durumda tek bir sözleşme yapılır ve tek bir üründen bahsedilir.

“Bir sözleşme müşterinin seçimine bağlı olarak ek bir varlığın inşaatını içerebileceği gibi, ek bir varlık inşaatını içermek üzere değiştirilebilir. Bu durumda, ek varlığın inşaatı aşağıdaki koşullardan birinin varlığı durumunda ayrı bir inşaat sözleşmesi olarak değerlendirilir: (TMS-11, 2015: md.10)

- a) Varlığın orijinal sözleşmenin kapsadığı varlık veya varlıklardan tasarım, teknoloji veya fonksiyon açısından önemli farklılıklar arz etmesi veya
- b) Varlığın fiyatının orijinal sözleşme bedeli dikkate alınmaksızın müzakere edilmesi.”

Yukarıdaki maddeye göre geliştirme sözleşmesinin kapsamadığı bir iş için yapılan her türlü yeni müzakereler yeni bir sözleşme gerektirir. Yeni bir fiyat söz konusu olur. Önceki sözleşmenin fiyatına dâhil edilmez. Yeni bir hasılat doğar. Örneğin; geliştirilen yazılımın kullanıcı dili Türkçe olarak kararlaştırılıp sözleşme yapıldıktan sonra

istenildiğinde yazılımın kullanımı İngilizce de olabilsin isteniyorsa bu işlem için ayrı bir sözleşme yapılmalıdır.

2.2.3. Sözleşme Gelirleri ve Değerlendirilmesi

“Sözleşme geliri aşağıdakileri içerir: (TMS-11, 2015: md.11)

- a) Sözleşmede başlangıçta üzerinde anlaşmaya varılmış bedel ve
- b) Sözleşmeye konu işteki değişiklikler, ek ödeme talepleri ve teşvik ödemelerinden:
 - i. Gelir olarak sonuçlanması muhtemel olanlar,
 - ii. Güvenilir biçimde ölçülebilenler.“

Yazılım geliştirme sözleşmelerinde bir bedel üzerinden anlaşıldığından belirsiz bir durum yoktur. Ancak güvenilir biçimde ölçülmesinin açıklanması gerekmektedir.

“Sözleşme geliri alınan veya alınacak olan hakedişlerin gerçeğe uygun değeri ile ölçülür. Sözleşme gelirinin ölçülmesi gelecekteki olayların sonuçlarına bağlı çeşitli belirsizliklerden etkilenir. Olaylar meydana geldikçe ve belirsizlikler çözümlendikçe tahminlerin sık sık gözden geçirilmesi gerekir. Bu nedenle sözleşme geliri tutarı dönemden döneme artabilir veya azalabilir.” (TMS-11, 2015: md.12)

2.2.4. Sözleşme Gelir ve Giderlerinin Muhasebeleştirilmesi

“Bir inşaat sözleşmesinin sonucu güvenilir biçimde öngörülebiliyorsa, inşaata ilişkin gelir ve maliyetler raporlama dönemi sonu itibariyle sözleşmeye konu işin tamamlanma aşaması esas alınarak, gelir ve giderler olarak finansal tablolara yansıtılır.” (TMS-11, 2015: md.22) Standart özetle “Bir yüklenici işletme aşağıdakileri içeren bir sözleşme yapmışsa, güvenilir öngörülerde bulunabileceği kabul edilir: (TMS-11, 2015: md.29)

- a) Tarafların inşa edilecek varlığa ilişkin yaptırma bağlanmış hakları;
- b) Alışveriş konusu bedeller,
- c) Ödeme şekil ve koşulları.

Ayrıca yüklenici işletmenin etkin bir iç finansal bütçeleme ve raporlama sistemine sahip olması gerekir.” şeklinde ifade ederek üç şartın da sözleşmede belirtilmesini istemektedir. Her biri de sözleşmelerde genelde bulunan ifadelerdir. Sözleşme tiplerine

göre sözleşmenin güvenilir biçimde ölçülebilmesi için aşağıdaki maddelerin sağlanması gerekmektedir.

“Sabit fiyatlı sözleşmede aşağıdaki koşulların tamamı mevcutsa inşaat sözleşmesi sonucu güvenilir biçimde tahmin edilebilir: (TMS-11, 2015: md.23)

- a) *Toplam sözleşme gelirinin güvenilir biçimde ölçülebilmesi;*
- b) *Sözleşmeye ilişkin ekonomik yararların işletmeye akışının muhtemel olması;*
- c) *Sözleşme konusu işin bitirilmesi için gereken inşaat maliyetleri ile işin tamamlanma aşamasının raporlama dönemi sonunda güvenilir biçimde belirlenebilmesi,*
- d) *Sözleşmeye yüklenebilecek inşaat maliyetlerinin açıkça belirlenebilmesi ve güvenilir biçimde ölçülebilir olması sonucu, katlanılan fiili inşaat maliyetlerinin önceki tahminlerle karşılaştırılabilir olması.”*

Tamamlanma aşamasından kasıt tamamlanma yüzdesi yöntemidir. “Bu yöntemde sözleşme geliri, ulaşılan tamamlanma aşamasına kadar katlanılan inşaat maliyetiyle eşleştirilerek, bitirilen işle orantılı gelir, gider ve kârın raporlanması sağlanır. Bu yöntem, ilgili dönemde sözleşme kapsamındaki işin aşaması ve işteki ilerleme konusunda yararlı bilgi sağlar.” (TMS-11, 2015: md.25) Yani işin tamamlanma oranını hesaplayabilecek bir oranın bulunmasıyla dönemin geliri hesaplanmalıdır.

“Maliyet artı kâr sözleşmesinde, aşağıdaki koşulların tamamı mevcutsa inşaat sözleşmesi sonucu güvenilir biçimde tahmin edilebilir: (TMS-11, 2015: md.24)

- a) *Sözleşmeye ilişkin ekonomik yararların işletme tarafından elde edilmesinin muhtemel olması;*
- b) *Sözleşmeye yüklenebilecek inşaat maliyetlerinin, geri tahsil edilebilir nitelikte olsun veya olmasın, açıkça belirlenebilmesi ve güvenilir biçimde ölçülebilmesi.”*

Taahhüdün tamamlanma oranını hesap edebilmek için birkaç yol bulunmaktadır. Bu yollar sıradaki başlıkta açıklanacaktır.

2.2.5. Sözleşmenin Tamamlanmasını Hesaplama Yöntemleri

TMS-11 standardı tamamlanma aşamasının ölçülebilmesi için aşağıdaki yöntemlerden bahsetmektedir. “Bir sözleşmenin tamamlanma aşaması çeşitli yollarla belirlenebilir.

Yüklenici işletme, yapılan işi güvenilir biçimde ölçen yöntemi kullanır. Sözleşmenin niteliğine bağlı olarak, bu yöntemler aşağıdakileri içerebilir: (TMS-11, 2015: md.30)

- a) Bugüne kadar yapılan işle ilgili katlanılan sözleşme maliyetlerinin öngörülen toplam inşaat maliyetlerine oranı;*
- b) Yapılan işe ilişkin incelemeler veya*
- c) Sözleşmeye konu işin fiziki tamamlanma oranı.*

Müşteriden alınan avanslar ve hak edişler genellikle yapılan işi yansıtmaz.”

“Bilindiği gibi, ülkemizdeki cari uygulamada tamamlanma yöntemi kullanılmaktadır. Bu yöntemle göre, bilgi alıcılarına sunulacak olan finansal tablo (gelir tablosu) dönemi işin başlangıcı ile bitirilmesini kapsamaktadır. İş bitimine kadar (genel kabule kadar) gelir tablosunun düzenlenmesi söz konusu olmamaktadır. Bu uygulamanın dönemsellik ilkesine, gerçeğe uygun bilgi vermeye ve muhasebe ilke ve standartlarına uygun olmadığı açıktır. Bu yöntemin yararları ile sakıncaları karşılaştırıldığında sakıncaları daha fazladır. Bu nedenle 11 no’lu standart bu yöntemi önermemektedir. Standartta önerilen yöntem tamamlanma yüzdeleri yöntemidir.” (Örten, Kaval ve Karapınar, 2012: 120) Bu oranlara 30. maddenin “a” şikkından ya da “c” şikkından ulaşılabilir.

Tamamlanmanın hesap edilebilmesi için önce yazılım büyüklüğünün (yüzde yüz olan nedir?) belirlenmesi gerekir ki bunda da birden fazla yöntem bulunmaktadır. Bu yöntemler şu şekilde sıralanabilir: (Yücalar, 2014, s. 6)

1. Satır Sayısı
2. İşlev Puanı
3. Geçmiş Proje Verileri

Her yöntemin diğerine göre avantajlı ve dezavantajlı olduğu durumlar vardır. Projenin maliyet kestirimi, yaşam döngüsünün daha başında, genel yazılım isterleri tanımlanır tanımlanmaz yapılmalıdır. Maliyet, genellikle işgücü cinsinden adam-gün biçiminde ifade edilerek belirlenir. Daha sonra çalışan ücretlerine dayalı maliyete dönüştürülür. Kestirimler en az iki yöntem kullanılarak ve en az iki kişiyle yapılmalıdır. Kullanılan farklı yöntemlerle elde edilen farklı sonuçların arasında kabul edilebilir olan seviye yönetime bildirilir. Bu şekilde yazılım projesinin ilk toplam maliyeti tahmin edilmiş olur ve sözleşme fiyatı belirlenebilir.

Maliyet oranı yöntemi, a şıkında anlaşılabilir biçimde açıklanmıştır. Yine de basit bir örnek gerekirse; yazılım sözleşmesi geliri 3.000 TL, beklenen toplam maliyet 2.500 TL iken dönemde gerçekleşen fiili maliyet 250 TL olsun. Bu durumda beklenen ve gerçekleşen maliyet oranından $250/2.500=10\%$ bulunur ve bu oran dönem gelirin e uygulanır. $3.000 \times 10\% = 300$ TL gelir; $300-250=50$ TL dönem karı olarak hesaplanır.

“Tamamlanma aşamasının hesaplanma gününe kadar katlanmış olan inşaat maliyetleri esas alınarak belirlenmesi durumunda, hesaplanma gününe kadar katlanan maliyetler içine yalnızca yapılan işi yansıtan sözleşme maliyetleri dahil edilir.” (TMS-11, 2015: md.31) *“Bu tür sözleşme maliyetleri, geri alınabileceklerinin muhtemel olması koşuluyla, bir varlık olarak finansal tablolara alınır.”* (Akbulut, 2012: 1164) Sözleşme maliyetlerine bir sonraki başlıkta değinilecektir.

Fiziki tamamlanma yöntemi (c şikkı) için söz konusu yazılımın kendisinin soyut ancak yöntemin ismi fiziki olunca somutlaştırmak için ek çabalar gerekmektedir.

İşin tamamlanma oranını fiziki verilerden elde etmek için, yapılan geliştirmenin belirli parametreleri karşılayıp karşılamadığı ile kıyaslayarak elde etmek mümkündür. Yukarıda sayılan her üç yöntem için hesaplanmış olan proje büyüklüğü ile geliştirmeye gelinen noktanın verileri oranlanarak bir hesaplama yapılabilir. Ancak, yapılan işlerin müşteri tarafından onaylanmadığı kısımlar hâlâ tamamlanmamış olarak düşünölmelidir. Çünkü her ne kadar çaba sarf edilmiş olursa olsun, müşteri onaylamadığı sürece teslimat gerçekleştirilemez ve maliyetlerde artış söz konusu olur.

Örneğin; işlev puanı yöntemine göre önce yapılacak işlerin miktarının bilinmesi ve ağırlıklandırılması (işin zorluğuna göre puan) gerekmektedir. Her çalışanın günde yapabileceği maksimum puan da tanımlanır. İşlerin dağıtılmasında kişilerin kapasite puanlarını aşmamaya özen gösterilir. Böylece toplam puan ile tamamlanan puan arasındaki oran sayesinde tamamlanma oranı elde edilebilir. Puanlamanın bir standardı olmadığından objektifliği her zaman sorgulanabilir.

Maliyet oranı yöntemini uygulamak pratikte daha kolaydır. Çünkü fiili maliyetleri takip etmek fiziki tamamlanmayı takip etmekten daha basittir.

2.2.6. Sözleşme Maliyetleri

Sözleşme maliyetleri aşağıdakilerden oluşur: (TMS-11, 2015: md.16)

- a) Belli bir sözleşmeyle doğrudan ilişkili maliyetler,
- b) Genel olarak sözleşmeye konu işle ilişkisi kurulabilen ve sözleşmeye yüklenebilecek olan maliyetler,
- c) Sözleşme hükümlerine göre özellikle müşteriye yüklenebilecek olan diğer maliyetler.

Yukarıdaki ifadeleri biraz açmak gerekirse “a” şıkkı aşağıdakileri kapsar: (TMS-11, 2015: md.17)

- a) Gözetim dâhil inşaat alanı işçilik maliyetleri;
- b) İnşaatta kullanılan malzeme maliyetleri;
- c) İnşaatta kullanılan tesis ve makinelerin amortismanı;
- d) Tesis, makine ve malzemelerin inşaat alanına getirilmesi ve buradan götürülmesi ile ilgili taşıma maliyetleri;
- e) Tesis ve makine kiralama maliyetleri;
- f) Sözleşmeyle doğrudan ilişkili tasarım ve teknik destek hizmeti maliyetleri;
- g) Tahmini garanti maliyetleri dâhil olmak üzere, garanti kapsamında yapılan işler ve büyük onarımlara ilişkin öngörülen maliyetler;
- h) Üçüncü kişilerin ödeme talepleri.

“b” şıkkı ise aşağıdakileri kapsar: (TMS-11, 2015: md.18)

- a) Sigorta,
- b) Belli bir sözleşmeyle doğrudan ilişkisi kurulamayan tasarım ve teknik destek hizmeti maliyetleri,
- c) İnşaat genel giderleri.

Genel olarak sözleşme kapsamındaki işle ilişkisi kurulabilen ve belli sözleşmelere yüklenebilecek olan maliyetler borçlanma maliyetlerini de içerir.

TMS-11 standardında sözleşme maliyeti olarak sayılabilecek olan harcamalar yukarıdaki gibidir. Bu maddelerde altı çizili olan kısımlar TMS-38 standardında da

maddi olmayan duran varlığın maliyetini oluşturan harcamalar arasında sıralanmıştır. Kısaca sözleşme ile ilgili olan ve geri kazanılması muhtemel olan harcamalar sözleşmenin maliyetine eklenir. Bu noktada şunu belirtmek gerekir ki bir yazılım geliştirme taahhüdü yapıldığında sözleşmenin gerektirdiği hizmeti sağlamak için yapılan araştırma ve geliştirmenin tamamını ilgili sözleşmenin maliyetine eklenir. Çünkü araştırma safhasının gideri geri kazanılması muhtemeldir. Sözleşme, gelir beklentisini desteklemektedir.

2.2.7. Sözleşmeye Dair Yapılması Gereken Açıklamalar

TMS 11 standardı, işi yapmayı taahhüt eden işletmeye aşağıdaki konularda sorumluluk yüklemektedir.

“Bir yüklenici işletme aşağıdaki hususları açıklar: (TMS-11, 2015: md.39)

- a) *Dönem geliri olarak finansal tablolara yansıtılan sözleşme geliri tutarı;*
- b) *Dönem içinde kaydedilen sözleşme gelirin saptanmasında kullanılan yöntemler;*
- c) *Devam eden sözleşmelere konu işlerin tamamlanma aşamasının saptanmasında kullanılan yöntemler.*”

“Bir işletme raporlama dönemi sonunda devam eden sözleşmelere ilişkin olarak aşağıdakilerden her birini açıklamak durumundadır: (TMS-11, 2015: md.40)

- a) *Raporlama dönemi sonuna kadar katlanılan maliyetler ile finansal tablolara yansıtılan kârların (kaydedilmiş zararlar düşüldükten sonra) toplam tutarı;*
- b) *Alınan avansların tutarı;*
- c) *Hak edişler üzerinden teminat olarak alıkonulan tutarları.*”

Yazılım projeleri birden fazla mali döneme sirayet eden çalışmalar olduğundan dönem gelirinin dönemsellik ilkesi gereği her döneme isabet eden tutarların raporlanması gerekmektedir.

Döneme isabet eden gelirin hesaplanması için tamamlanma oranı yöntemlerinden biri kullanılır. Tamamlanma oranının hesaplanabilmesi için tahmin edilen bütçe değerleri ile işin tamamlanan (fiili) kısmı karşılaştırılabilmesi gerekir ki projenin kaçta kaç

tamamlandığı hesaplanabilsin. Ayrıca hangi tamamlanma oranının kullanıldığının da açıklanması istenmektedir.

TMS-11 standardına uygun raporlama yapılabilmesi için önceki kısımlarda bahsedilen proje yönetimi yazılımları ile geliştirme süreçlerindeki harcamaların takip edilerek dönemler bazında raporlanabiliyor olması gerekir. Finansal bilgilerin rahatça muhasebeleştirilebilmesi için proje yönetim yazılımının muhasebe bilgi sistemine entegre edilmesi uygun olacaktır. Şu aşamada yukarıdaki anlatılanları tam anlamıyla gerçekleştiren yönetim araçları olduğuna rastlanılmamış olmakla beraber var olan yazılımlara iyileştirici bakım uygulanarak ilgili raporların üretilmesi sağlanabilir. Böylece karmaşık gibi gözükken ve zamana yayılmış işlerin muhasebe kayıtları TMS'ye uygun şekilde rahatlıkla tutulabilir.

Son bölümde, siparişe dayalı yazılım geliştirmek için yapılan işlere dair olan harcamalar, maliyet oranı yöntemiyle hesaplanıp; TMS-11 gereğince muhasebe örnekleri detaylıca açıklanacaktır.

Sözleşmeye bağlı üretilen yazılımlardan doğan hasılatların TMS'ye uygun biçimde kayıt edilebilmesi için açıklanması gereken bir konu daha vardır. Bu da hasılatdaki vade farkıdır. Sözleşme geliri, TMS-11 standardında daha çok hasılatın tamamlanma yöntemine göre (dönemsellik) hak edildiği miktarın hesaplanması üzerine odaklanmıştır. Bu sebeple TMS 18 standardının incelenmesi gerekmektedir.

2.3. TMS 18'e Göre Yazılım Hasılatının Değerlendirilmesi

Siparişe dayalı geliştirilen yazılım projelerinin hasılatlarının muhasebeleştirilebilmesi için TMS 18 Hasılat Standardı hükümleri dikkate alınacaktır. Bunun için standardın konumuzla ilgili olan tanımlamalarına değinilecek ve ilgili yorumlar yapılacaktır.

2.3.1. Hasılatın Kapsamı

Bu standart aşağıdaki işlem ve olaylardan kaynaklanan hasılatın muhasebeleştirilmesinde uygulanır. (TMS-18, 2015: md.1)

- a) *Mal satışları;*
- b) *Hizmet sunumları;*

- c) İşletme varlıklarının başkaları tarafından kullanılmasından sağlanan faiz, isim hakkı ve temettüleri.

Hizmet sunumları standartta şu şekilde tanımlanmıştır: “Hizmet sunumu; üzerinde anlaşmaya varılmış sözleşmeye bağlı bir işin taraflarca belirlenmiş sürede işletme tarafından yapılmasını içerir. Hizmetler bir veya birden çok dönem içinde sunulabilir. Hizmetlerin sunumuyla ilgili, proje yöneticileri ve mimarların sunduğu hizmetler gibi, bazı sözleşmeler doğrudan inşaat sözleşmeleri ile ilgilidir. Bu sözleşmelere bağlı olarak ortaya çıkan hasılat bu standart kapsamında olmayıp, “TMS 11 İnşaat Sözleşmeleri” Standardında belirlenmiş olan inşaat sözleşmelerine yönelik hükümler kapsamında ele alınır.” (TMS-18, 2015: md.4)

Sözleşmeye bağlı geliştirilen yazılımların satışları hizmet sunumu olarak değerlendirildiğinden TMS-11 İnşaat Sözleşmeleri standardı göz önünde bulundurularak açıklanmalıdır. Yukarıdaki tanımda yapılan atfın gereği projenin birden çok döneme sirayet etmesidir. Bu durumdan dolayı projenin her dönem içinde müşteriye teslim edilen miktarına göre hasılatın hesaplanmasını gerektirir. Bu sebeple TMS-11 İnşaat Sözleşmeleri standardı incelenmiştir. TMS-11 standardında, sözleşme geliri bölümü vardır ancak tanımlaması yeterince net değildir. Bunun için TMS-18 Hasılat standardındaki tanımlamaların açıklanmasına devam edilecektir.

2.3.2. Hizmet Sunumu İle İlgili Hükümler

Hizmet sunumlarının muhasebeleştirilebilmesi için aşağıdaki koşulların tamamının güvenilir şekilde tahmin edilebilmesi gerekir. Bu koşullar şunlardır: (TMS-18, 2015: md.20)

- a) Hasılat tutarının güvenilir biçimde ölçülebilmesi;
- b) İşleme ilişkin ekonomik yararların işletme tarafından elde edileceğinin muhtemel olması;
- c) Raporlama dönemi sonu itibariyle işlemin tamamlanma düzeyinin güvenilir biçimde ölçülebilmesi;
- d) İşlem için katlanılan maliyetler ile işlemin tamamlanması için gereken maliyetlerin güvenilir biçimde ölçülebilmesi.

Siparişe dayalı yazılım üretmiş olan işletmelerin lisans sözleşmeleri ile satış yapmaları a şikkının güvenilir kılmaktadır. Çünkü bir satış sözleşmesi ve fiyatı bulunmaktadır. Alacağının riske girmediği durumlarda da b şikkı sağlanmış olur. C ve d şıkları için dönem sonunda yazılımın ne kadarının tamamlandığı bilgisinin üretiliyor olması gerekmektedir. Bu konuda TMS 18 Hasılat standardı aşağıdaki gibi yorum yaparak hesaplamının TMS 11 İnşaat Sözleşmeleri standardına göre hesaplanıp, hasılat kabul edilmesini tavsiye etmektedir.

“İşlemin tamamlanma düzeyi dikkate alınarak hasılatın kayda alınması tamamlanma yüzdesi yöntemi olarak adlandırılır. Bu yöntemle göre hasılat, hizmetlerin sunulduğu dönemlerde finansal tablolara yansıtılır. Hasılatın bu yöntemle göre muhasebeleştirilmesi bir dönemde yapılan hizmet ve performansı hakkında yararlı bilgiler sağlar. ”TMS 11 İnşaat Sözleşmeleri” Standardı da hasılatın bu esasa göre muhasebeleştirilmesini gerekli kılmaktadır. Bu Standardın hükümlerinin genellikle hizmet sunumu kapsamında hasılatın ve ilgili giderlerin muhasebeleştirilmesine uygulanması mümkündür.” (TMS-18, 2015: md.21)

“Hasılat, ancak işleme ilişkin ekonomik yararların işletme tarafından elde edilmesi muhtemel olduğunda finansal tablolara yansıtılır. Ancak daha önce muhasebeleştirilmiş olan hasılat tutarının tahsil edilebilirliği konusunda bir belirsizlik ortaya çıkarsa, tahsil edilemeyen veya tahsil edilebilmesi muhtemel olmaktan çıkan tutar başlangıçta kayda alınmış hasılatın düzeltilmesi yerine gider olarak finansal tablolara yansıtılır.” (TMS-18, 2015: md.22)

“İşletme tarafından işlemin karşı taraflarıyla aşağıdakiler üzerinde anlaşmaya vardıktan sonra güvenilir tahminlerde bulunabileceği kabul edilir: (TMS-18, 2015: md.23)

- a) *Taraflarca sunulacak ve alınacak hizmetle ilgili olarak her iki tarafın yaptırımı bağlanmış hakları,*
- b) *Hizmet bedeli,*
- c) *Ödeme şekli ve koşulları.*

İşletmede etkin bir dâhili finansal bütçeleme ve raporlama sisteminin mevcut olması da gereklidir. İşletme, hizmet sunumu devam ederken, verilen hizmete ilişkin hasılat tahminini gözden geçirip gerektiğinde değiştirir. Tahminlerin bu şekilde düzeltmeye

tabi tutulması, işlemlerin sonuçlarının güvenilir biçimde tahmin edilemediği anlamına gelmez.”

Siparişe dayalı geliştirilen yazılım projelerinde taraflar arasında bir sözleşme yapılır ve bu sözleşmede yukarıdaki her üç koşul ifade edilir. Böylece üreticinin güvenilir tahminlerde bulunabileceği kabul edilebilir. Ancak devamında etkin bir finansal bütçeleme ve raporlama sisteminin var olması şartı da getirilmiştir. Çünkü toplam maliyeti tahmin etmek ve devam eden üretim sürecinde iken toplam üretimin neresindeyiz sorularına cevap üretilmesi gerekecektir.

“Kişiyeye özel yazılım geliştirilmesine ilişkin ücretler, teslimat sonrası hizmetler desteği için sağlanan hizmetlerin tamamlanma aşaması da dâhil olmak üzere, geliştirme işlemlerinin tamamlanma aşaması dikkate alınmak suretiyle, hasılat olarak muhasebeleştirilir.” (KOBİ-TFRS, 2013: md.23A.33) Yani geliştirme işlemlerinin takip edildiği bir proje yönetim yazılımı ve muhasebe bilgi sistemi ile entegre edilmiş bir yapıya ihtiyaç vardır. Ancak böyle bir ortamda standartların tavsiye ettiği şekilde gerçeğe ve dönemselliğe uygun raporlama yapılabilir.

2.3.3. Hasılatın Ölçümü İle İlgili Hükümler

“Hasılat alınan veya alınacak olan bedelin gerçeğe uygun değeri ile ölçülür.” (TMS-18, 2015: md.9) *“Bir işlemde doğan hasılat tutarı, genellikle işletme ile varlığın alıcısı veya kullanıcısı arasındaki anlaşma ile belirlenir. Hasılat, işletme tarafından uygulanan ticari iskontolar ve miktar indirimleri de göz önünde tutularak, alınan veya alınacak olan bedelin gerçeğe uygun değeri ile ölçülür.”* (TMS-18, 2015: md.10)

“Gerçeğe uygun değer: Piyasa katılımcıları arasında ölçüm tarihinde gerçekleşecek olağan bir işlemde bir varlığın satışında elde edilecek veya bir borcun devrinde ödenecek fiyattır.” (TMS-18, 2015: md.7) Yani, satış fiyatıdır. Onuncu maddedeki alıcı veya kullanıcı ile olan anlaşma ifadesi, lisans sözleşmesini ifade etmektedir. Dolayısı ile yazılım satışlarında hasılatı belirleyen unsur, lisans sözleşmesindeki satış fiyatıdır. *“Mal ve hizmet satışının peşin olarak yapılması durumunda fatura tutarı hasılat olarak kaydedilebilir. Ancak nakit ve nakit benzerleri girişlerinin ertelendiği durumlarda (Örneğin; vadeli satışlarda); satış bedelinin gerçeğe uygun değeri, alınacak olan nakdin nominal tutarından daha düşük olabilir. Çünkü alınacak olan nakdin gerçeğe uygun değeri her zaman daha düşük olacaktır.”* (Karabınar, 2011: 21)

Tahsilin zaman farkı, muhasebeleştirmede de farklılıklar oluşturur. “İşlemin vadeli veya vadesiz olması, vadeli satışlardaki gelirin içerisinde yer alan örtük durumdaki faiz gelirini ayırıştırıp gerçek satış bedelinin bulunması bakımından önemlidir.” (Özerçen, 2012: www.muhasabetr.com)

Ayrıştırılan faiz gelir olarak kaydedilmelidir. Bu durumda işletmeler, vadeli satışları için standardın gerektirdiği şekilde hesaplama yapabilecek ve gerekli entegrasyonlar ile doğru kayıtları oluşturabilecek bir muhasebe kayıt sistemine ihtiyaç duymaktadır. (Örten, Kaval ve Karapınar, 2012: 269)

Tablo 9’da peşin ve vadeli satışın muhasebeleştirmedeki etkisi özetlenmiştir. Buna uygun şekilde hesaplamalar yapılmalı ve kayıtlar tutulmalıdır. Vadeli satışlarda hesaplanacak faiz miktarı 382/482 Ertelenmiş gelirler, vadeli alışlarda ise 182/282 ertelenmiş giderler isminde hesaplar oluşturularak kayıt edilebilir. (Örten, Kaval ve Karapınar, 2012: 269-270) Diğer bir kaynak ise hesap numarası vermeyerek “Ertelenmiş Vade Farkı Gelirleri” ve “Ertelenmiş Vade Farkı Giderleri” isminde hesap ismi önermiştir. (Gençoğlu, Özerhan ve Karapınar, 2013: 254)

Tablo 9: Peşin ve Vadeli Hasılatın Muhasebeleştirilmesi

Hasılatın Ölçümü	Gerçeğe Uygun Değer	Muhasebeleştirme
Vadesiz Satışlar	Satışın Kendisi	Satışın Tamamı Hasılat
Vadeli Satışlar	Vadeye kadar İskonto Edilmiş Tutarı	Vade Farkı Faiz Hasılatı Kalamı Satış Hasılatı

Kaynak: (Özerçen, 2012)

Standardın tavsiye ettiği bu durum, dönem sonlarında işletme değerlerini gerçekçi bir şekilde yansıtabilmenin çabasıdır. Bunun için gelecekte tahsil edilecek hasılatlar bugünkü değerleri ile kayıt edilmelidir. Bugünkü değer formülünde kullanılacak faiz oranları şunlar olabilir: (Karapınar, 2011: 23)

- Özellikle taraflar arasında belirlenebilecek bir vade farkı oranı,
- Şirketin uyguladığı vade farkı oranı (yazarın önerisi),
- TC Merkez Bankası gösterge faiz oranı,
- TC Merkez Bankası kısa vadeli reeskont faiz oranı veya vadeli mevduat faiz oranı.”

Pratikte kullanılacak faiz oranının en uygun olanı şirketin genelde uyguladığı faiz oranı olmalıdır. (Örten, Kaval ve Karapınar, 2012: 269)

2.3.4. İşlemin Ayırıştırılması

Her satış konusunun hasılatı ayrı olarak kaydedilmesi gerekmektedir. Bazen bir satış işleminin içinde ayrıştırılabilir olan ve olmayan unsurlar olabilir. Bu gibi durumlarda ayrıştırılabilir olan kısım için işlemleri ayrı ayrı kaydetmek standardın gereğidir. “Örneğin; bir ürünün satış fiyatı ayrıştırılabilir nitelikteki satış sonrası servis tutarını da içeriyorsa, bu tutarın hasılat olarak muhasebeleştirilmesi ertelenerek, servis hizmetinin verildiği dönem boyunca hasılat olarak finansal tablolara yansıtılır.” (TMS-18, 2015: md.13) Örneğin; bir yazılım firması yazılımlarını yıllık 20.000 TL’ye satmaktadır. Bunun yanında yıllık 3.000 TL’ye de bakım sözleşmesi yapmaktadır. Tahsil edilecek 23.000 TL’nin standardın hükümleri gereği sadece 20.000 TL’si hasılattır. Çünkü bakım hizmeti tamamlanmamıştır, zamana dayalı olduğundan ancak ilgili sürenin bitimiyle ve gerekli hizmetin ifası ile beraber tamamlanmış bir satış olduğundan bugün hasılat olarak kabul edilmez. Peşin tahsilat olduğunda sözleşme süresi sonuna kadar bir avans hesabından bekletilmesi uygun olur. Hizmet tamamlanıp müşteriye teslim edilince tamamlanma oranına göre hasılat kaydedilir. (Bakım sözleşmesi süresince kaç kere rutin bakım yapılacağı belli olabilir.)

Hasılat kavramının tanımlanması için malın veya hizmetin satıcıdan müşteriye transfer edilerek, müşterinin kontrolüne bırakılması gerekmektedir. Müşterinin kontrolüne bırakıldığının işaretleri şunlardır: (Olsen ve Weirich, 2010: 59) “Koşulsuz borçlanma, kanuni mülkiyet, fiili zilyetlik, mal veya hizmetin tasarımı veya işlevlerinin müşteriye göre özelleştirilmesi.” Bu şartlardan en az biri sağlandığında hasılattan bahsedilebilir.

Hasılatın tanımı için Türkiye Finansal Raporlama Standartlarında iki temel standart bulunmaktadır. Bunlar TMS-11 İnşaat Sözleşmeleri ve TMS-18 Hasılat standartlarıdır. Bu standartlar farklı uygulamalara benzer işlemler sunar. Çünkü hasılatı doğuran şey birinde riskin ve getirinin transferi iken diğeri faaliyet (dönemsellik) temellidir. (Olsen ve Weirich, 2010: 55) Yani iç içe geçmiş kavramları açıklayan iki standart bulunmaktadır. Bu sebeple son bölümde detaylı örnek ile hesaplamalar yapılırken önce işin tamamlanma oranı TMS-11’e göre hesaplanacak, daha sonra örneğe vade farkı eklenerek TMS-18’e göre hesaplanacak ve muhasebeleştirilecektir.

BÖLÜM 3: YAZILIMLARIN MUHASEBELEŞTİRİLMESİ

Önceki bölümün başında yazılımların kullanım amaçlarının farklılığı açıklanmış olup her bir durumun bir veya birkaç standardı ilgilendirdiği ifade edilmiştir. Standartlar özelinde yazılımları muhasebeleştirebilmek için tanım, kapsam ve uygulamaya dönük maddeler aynen nakledilmiş ve yorumlanması gereken kısımlara açıklamalar yapılmıştır. Böylece yazılımların muhasebeleşebileceğinin izahı yapılmaya çalışılmıştır.

Bu bölümde, yapılan izahların örnekler ile güçlendirilmiş şekilde tekrar ele alınması sağlanacaktır. Örnekler bir senaryoya bağlı olacağından daha iyi anlaşılabilmesi için ilgili konunun başlangıcından itibaren takip edilmesi gerekmektedir.

3.1. Satın Alınan Yazılımların Muhasebeleştirilmesi

Satın alınan yazılımlar düşünüldüğünde gerçekten satın alınan şey ilgili yazılımın kullanım hakkıdır. Dolayısı ile satın alınan yazılımlar Tek Düzen Hesap Planı 260 Haklar Hesabında izlenmelidir. Bu hesabın tanımına göre “*İmtiyaz, patent, lisans, ticari marka ve unvan gibi bir bedel ödenerek elde edilen bazı hukuki tasarruflar ile kamu otoritelerinin işletmeye belirli alanlarda tanıdığı kullanma, yararlanma gibi yetkiler dolayısıyla yapılan harcamaları kapsar.*” işleyişine göre “*Edinilen haklar, maliyet bedelleri ile bu hesaba borç kaydedilir. Yararlanma süreleri içerisinde, yararlanma sürelerinin belli olmaması durumunda, 5 yıllık sürede eşit taksitlerle itfa olunarak yok edilir.*” (Gelirler İdaresi Başkanlığı, 2015: www.gib.gov.tr)

Buna göre ilk muhasebeleştirme ve sonrasında alınabilecek kararlar ve yapılabilecek harcamaların nasıl muhasebeleşmesi gerektiğine dair birbiri ile ilişkili aşağıdaki örnekleri dikkatle incelemek gerekir. Örnekler, daha önceden yayınlanan makalemizden alınmıştır. (Yılmaz ve Calayoğlu, 2015a: 185-190)

3.1.1. İlk Muhasebeleştirme

Örnek 1: Ahmet A.Ş. işletmesinde geliştireceği yazılımlar için bir alt yapı oluşturmaktadır. Bu kapsamda birden fazla yazılım 1 yıllık süre ile 5 kişilik kullanım hakkı için sözleşmeler yapılmış ve satın alınmıştır. Satın alınma tarihi 01.01.2015’tir. Yazılımların listesi tablo 10’da gösterilmiştir.

Tablo 10: Satın Alınan Yazılımların Listesi ve Fiyatları

No	Yazılım İsmi	Fiyatı
1	Visual Studio 2013	3.000 TL
2	MS SQL Server Standard Edition 2012	7.500 TL
3	Crystal Report	1.500 TL
4	MS Team Foundation Server	7.500 TL

Daha sonra işletmede kurulması ve işler olabilmesi amacıyla 4 numaralı varlık için 1.500 TL kurulum ücreti ve kurulum ile test aşamasında geçen süre için de servis sağlayıcının çalışanına 1.000 TL ödemiştir. Tablo 11'deki gibi muhasebeleşmesi gerekmektedir.

İlk kurulum için yapılan harcamalardan kurulum ücreti, kurulumda ve test süreçlerinde çalışan görevli çalışana ödenen ücretler, yazılımın işler olabilmesi için yapıldığından elde etme harcamaları kapsamındadır ve varlığın maliyetine eklenir. 4 numaralı varlık için de böyle olmuştur. (Sumer ve Erer, 2010: 37)

Tablo 11: Örnek 1'in Muhasebe Kaydı

Hesap Adı	Borç	Alacak
260 Haklar Hesabı	22.000	
260.01 Visual Studio 2013	3.000	
260.02 MS SQL Server Standard Edition 2012	7.500	
260.03 Crystal Report	1.500	
260.04 MS Team Foundation Server (7.500+1.500+1.000)	10.000	
100 Kasa Hesabı		22.000

Örnek 2: aynı varlık için 1 yıllık bakım ve danışmanlık sözleşmesi yapılmıştır. Bedeli 2.400 TL'dir.

Yazılımın işler vaziyette olması için yapılan harcamalar varlığın maliyetine eklenmez ve gider olarak yazılır. Çünkü varlığın değerini arttırıcı veya varlığa değer katacak bir durum oluşmamaktadır. (Sumer ve Erer, 2010: 37) Buna göre tablo 12'deki gibi muhasebeleşmesi gerekmektedir.

Tablo 12: Örnek 2'nin Muhasebe Kaydı

Hesap Adı	Borç	Alacak
180 Gelecek Aylara Ait Giderler Hesabı	2.400	
180.04 MS Team Foundation Server 1 Yıllık Bakımı	2.400	
100 Kasa Hesabı		2.400
770 Genel Yönetim Giderleri Hesabı	200	
770.04 MS Team Foundation Server Bakım Gideri (2.400/12=200)	200	
180. Gelecek Aylara Ait Giderler Hesabı		200
180.04 MS Team Foundation Server 1 Yıllık Bakımı		200

Örnek 3: 01.02.2015 tarihinde 3 numaralı varlık için İspanyolca yabancı dil yaması satın alınmıştır. Bedeli 1.000 TL'dir.

İlk işlerliğin ardından varlığın değerinde artış yapacak olan yukarıdaki işlem için harcanan tutar, varlığın maliyetine eklenerek aktifleştirilir. Ancak rutin işlerden olan bakım harcamaları giderleştirilir. Buna göre tablo 13'deki gibi muhasebeleşmesi gerekmektedir.

Tablo 13: Örnek 3'ün Muhasebe Kaydı

Hesap Adı	Borç	Alacak
260 Haklar Hesabı	1.000	
260.03 Crystal Report	1.000	
100 Kasa Hesabı		1.000

Örnek 4: 01.03.2015 tarihinde 1 numaralı yazılım için versiyon değişikliğine gidileceği yönetim kurulunda kararlaştırılmıştır. Visual Studio **2015**, 5.000 TL'ye 1 yıl kullanım hakkı ile peşin olarak satın alınmıştır.

Yönetim kurulu kararı ile yeni bir yazılım alımı sonucu bir öncekinin kullanılmayacak olması durumunda yazılımın defter değerinin zarar olarak yazılması gerekmektedir. (TMS-16, 2015: md.67) (Yazılımın aylık itfa payı 250 TL idi.) Böylece bilanço dışına çıkartılır. Yeni alınan yazılım bilançoya eklenir. (Çankır, 2010: www.burhaneray.com) Buna göre tablo 14'deki gibi muhasebeleşmesi gerekmektedir.

Tablo 14: Örnek 4'ün Muhasebe Kaydı

Hesap Adı	Borç	Alacak
260 Haklar Hesabı	5.000	
260.05 Visual Studio 2015	5.000	
268 Birikmiş Amortismanlar Hesabı	500	
268.01 Visual Studio 2013	500	
659 Diğer Olağan Gider ve Zararlar Hesabı	2.500	
100 Kasa Hesabı		5.000
260 Haklar Hesabı		3.000
260.01 Visual Studio 2013		3.000

3.1.2. Yararlı Ömür ve Amortisman

Yazılımlarda yararlı ömür tespiti sonrasında amortisman ayırmak mümkün olur. Yararlı ömür konusu sınırlı ve sınırsız yararlı ömür şeklinde tanımlanmaktadır. Satın alınan yazılımların yararlı ömürleri incelendiğinde sınırsız olması mümkün değildir, çünkü yazılımdan göreceği fayda (kullanılabilir süre) sınırlıdır. “2.1.4 Yazılımın Yararlı Ömrü İle İlgili Hükümler ve Değerlendirilmesi” kısmında detaylıca açıklamalar yapılmıştır.

Kısaca satın alınan yazılımlar, kullanıcı lisans sözleşmesinin süresi kadar faydalıdır. Bu her yazılım için farklı olabilir. Bununla beraber piyasada yazılım üreticileri genelde 1 yıllığına yazılımların kullanıcı haklarını satmaktadır.

Amortisman kaydı, 268 Birikmiş Amortismanlar Hesabını “alacak” ve yazılımın kullanım amacı veya departmanına göre bir gider hesabını da “borç” yazarak kayıt altına almak gerekmektedir. Ancak, yazılım geliştirmek için kullanılan diğer yazılımların itfa payları, geliştirilen yazılımın defter değerine eklenir. (TMS-38, 2015: md.99)

Örnek 5: Mevcut satın alınan yazılımların Ocak ayı sonundaki doğrusal yöntem benimsenerek, üretim ile ilişkilendirilmeyen biçimde itfa payları hesaplanıp, sunulmuştur. (Kullanım hakları 1 yıl idi.) (Maliyet / 12) (Kullanım süresi 1 yıl olan ve üretimle ilişkilendirilmeyen varlıkları aylık dönemde itfa payı ayırmaya gerek yoktur. Sonraki örnekle karşılaştırılabilirlik olsun diye devam edilecektir) Buna göre aktifteki yazılımların amortismanı Tablo 15'teki gibi muhasebeleşmesi gerekmektedir.

Tablo 15: Örnek 5'in Muhasebe Kaydı

Hesap Adı	Borç	Alacak
770 Genel Yönetim Giderleri Hesabı	1.833	
770.01 Visual Studio 2013	250	
770.02 MS SQL Server Standard Edition 2012	625	
770.03 Crystal Report	125	
770.04 MS Team foundation server	833	
268 Birikmiş Amortismanlar Hesabı		1.833
268.01 Visual Studio 2013		250
268.02 MS SQL Server Standard Edition 2012		625
268.03 Crystal Report		125
268.04 MS Team Foundation Server		833

Kullanılan yazılımların itfa paylarından üretim süreçleriyle eşleştirilebilen kısımları 770 numaralı hesaba kayıt edilmemelidir. Bunun yerine oluşturulmakta olan varlığın maliyetine eklenmelidir.

Yazılım geliştirme sürecindeki tasarım, kodlama, test, devreye alma aşamalarında üretimle ilişkilendirilebilen yazılımların itfa payları yeni oluşturulan yazılımın maliyetine eklenir. (TMS-38, 2015: md.66-d) Konunun detayı “2.1.3. İşletme İçi Oluşturulan Yazılımlara Dair Hükümler ve Değerlendirilmesi” kısmında detaylıca açıklanmıştır.

Eğer geliştirme sürecindeki harcamalar ile araştırma sürecindeki harcamalar birbirinden ayrıştırılabilirse geliştirme sürecine denk düşen kullanılan yazılımların itfa payları da standardın 99. maddesi gereği aktifleştirilir. Bunun için şu anda kullanılması gereken hesap 263 Araştırma ve Geliştirme Giderleri Hesabıdır.

Tek düzen hesap planının TMS'nin tavsiye ettiği şekilde kayıtlar tutulabilmesi için bazı düzenlemelerin yapılması gerektiğine dair birçok eser kaleme alınmıştır. İncelenen konu hakkında bazı hesap isimleri üstünde eleştirisel düşünülmüştür. Her şeyden önce “araştırma harcamaları” aktifleştirilemez olduğundan 263 numaralı hesabın adından bu ifadenin kaldırılması gerekmektedir. Yine “gider” ifadesi de aktifleştirilemez olduğundan hesap isminden “araştırma ve gider” ifadeleri çıkarılmalıdır. Bunun yerine “Geliştirme Harcamaları Hesabı” ismiyle kayıtlar tutulmalıdır. Benzer bir yaklaşımla “Geliştirme Giderleri Hesabı” önerilmiştir. (Akdoğan ve Sevilengül, 2007: 60) Geliştirilen yazılımın amacının (satış veya kullanım) belirtilerek kaydedilmesi alt

hesaplar ile sağlanabilir. Örneğin; Satış Amaçlı Geliştirilen Varlıklar ve Kullanım Amaçlı Geliştirilen Varlıklar gibi.

Yazılım geliştirme sürecindeki bazı aşamaların harcamaları da araştırma kapsamına gireceğinden bu kategorideki itfa payları 750 Araştırma ve Geliştirme Giderleri Hesabına kayıt edilmelidir. Söz konusu aşamalar şunlardır: Yazılım geliştirme süreçleri arasında geçen “İsteklerin belirlenmesi ve Analiz” ile geliştirme sonrası ortaya çıkan “Bakım” süreçleridir.

Geliştirme faaliyetlerinin harcamaları gider yazılamayacağından hesabın isminin sadece “Araştırma Giderleri” olarak düzenlenmelidir. Bu şekilde düzenlemeler yapılırsa hesap isimleri ve işleyişleri ile ifadelerin kavramsal bütünlüğünün çelişmemesi sağlanmış olur.

Örnek 6: Önceki örnekteki değerler kullanılarak geliştirilen bu örnekte; Ocak ayının itfa paylarının işletme faaliyetlerine göre hangi süreçte ve ne kadar zaman geçirdiği esasına göre ayrıştırılmıştır. TMS-38’in 57. maddesindeki geliştirme safhasına ait işlemlerin gerçekleştiği kabul edilmiş ve yazılımların oransal olarak kullanımı, %50 geliştirme, %25 Araştırma ve %25 Üretimle ilgisi olmayan faaliyetlerde bulunduğu varsayılmıştır. (Saban ve Genç, 2005: 132)

TMS 38 kapsamında yazılım geliştirme süreçlerinde yapılan harcamaların muhasebesel açıdan nasıl ele alınması gerektiğinin özeti tablo 8’de özetlenmiştir. Bu kapsamda yazılım geliştirmek için kullanılan yazılımların itfa payları da hangi süreçte ne kadar kullanıldığına göre hesaplanıp muhasebeleştirilmelidir.

Varsayımdaki gibi yazılımların hangi işlerde ne kadar süreyle kullanıldıkları tespit edilebiliyorsa tablo 8’e göre ayrıştırarak muhasebeleştirmek en doğrusudur. İtfa payları faaliyetlerle eşleştirilebiliyor ve üretim ile ilişkisi kurulabiliyorsa yazılımların amortismanı tablo 16’daki gibi muhasebeleşmesi gerekmektedir.

Tablo 16: Örnek 6'nın Muhasebe Kaydı

Hesap Adı	Borç	Alacak
263 Araştırma ve Geliştirme Giderleri Hesabı	917	
263.01 Öğrenci İşleri Yazılımı (Satış Amaçlı)	917	
750 Araştırma ve Geliştirme Giderleri Hesabı	458	
750.01 Öğrenci İşleri Yazılımı (Satış Amaçlı)	458	
770 Genel Yönetim Giderleri Hesabı	458	
770.01 Visual Studio 2013	63	
770.02 MS SQL Server Standard Edition 2012	156	
770.03 Crystal Report	31	
770.04 MS Team foundation server	208	
268 Birikmiş Amortismanlar Hesabı		1.833
268.01 Visual Studio 2013		250
268.02 MS SQL Server Standard Edition 2012		625
268.03 Crystal Report		125
268.04 MS Team foundation server		833

3.1.3. Yeniden Değerleme

Yeniden değerlendirme hükümleri ve yaklaşımları 2.1.5 kısmında açıklanmıştır. Satın alınan yazılımların yeniden değerlendirilebilmesi için aktif bir piyasanın varlığından bahsedilebilse de yeniden değerlendirme yapabilecek kadar uzun ömürlü varlıklar değildir. Ekseriyetinin yararlı ömrü, kullanıcı lisans sözleşmesinin 1 yıl olmasından dolayı yılsonunda aktifin defter değeri sıfırı bulacaktır. Ancak daha uzun süreli kullanım hakkı tanıyan sözleşmeler ile hak satın alındıysa fiyat dalgalanmaları yaşanabilir.

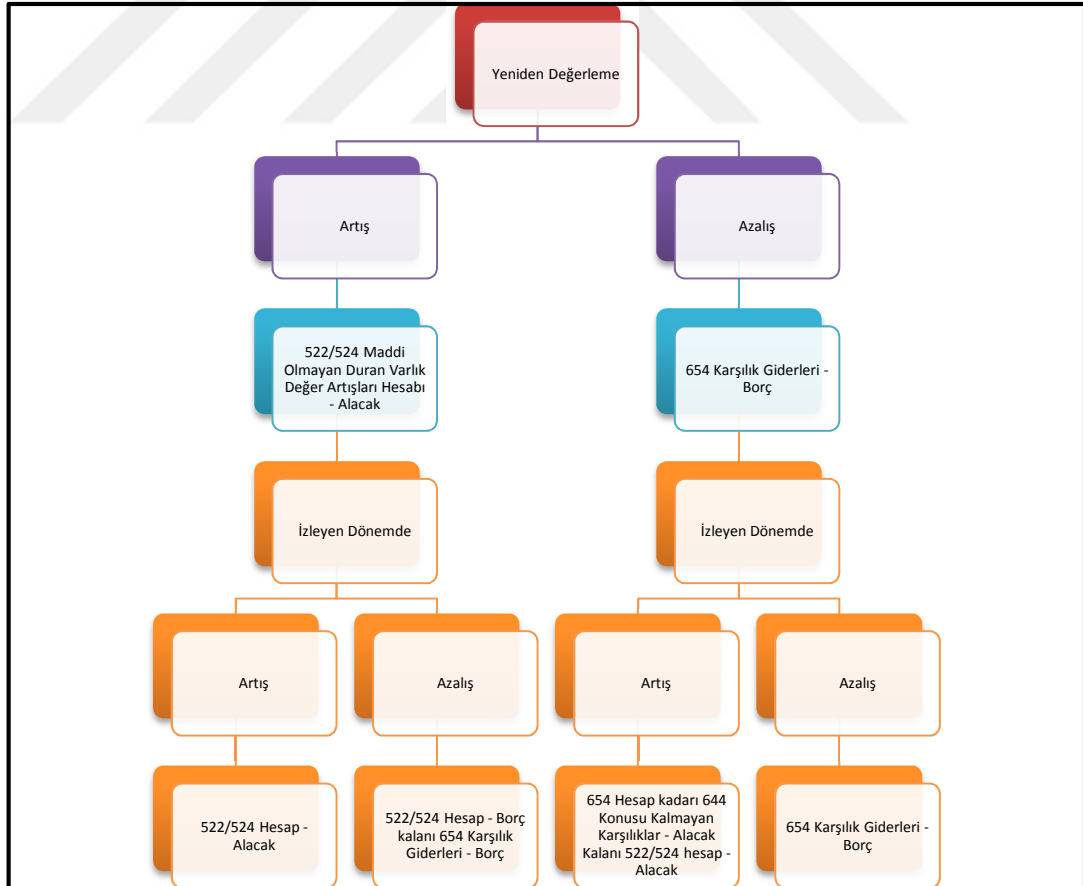
Bu sefer de teknolojiye yaşanan hızlı gelişmeler neticesinde genellikle satın alınan yazılımın piyasa değerinin giderek düşmesi muhtemeldir. Çünkü üzerine yeni yazılımlar gelecektir.

Maddi olmayan duran varlıklar, ilk muhasebeleştirilmesinin ardından, muhasebe politikası olarak “maliyet modeli” ve “yeniden değerlendirme modeli” seçeneklerinden birisini seçerek değerlemeye tabi tutulur. (TMS-38, 2015: md.72) Yeniden değerlendirme modeli uygulandığı takdirde; maddi olmayan duran varlığın değerinde değer artışı varsa, bu artış öz kaynaklar grubunda, diğer kapsamlı gelir unsuru olarak yeniden değerlendirme

fazlası adı altında muhasebeleştirilir. Eğer, değer azalışı söz konusu ise, bu azalma gider olarak kayıtlara alınır. (TMS-38, 2015: md.85-86)

“Daha önceki dönemde değer azalışı olmuş bir maddi olmayan duran varlığın yeniden değerlendirme sonucu değerinde artış olmuşsa, önceki dönemde kâr veya zararda muhasebeleştirilen yeniden değerlendirme azalışı kadar kar veya zarar, kalan kısmı diğer kapsamlı gelir olarak muhasebeleştirilir.” (Gökgöz, 2013: 27) Aksi durumda, daha önceki dönemlerde değer artışı olmuş bir maddi olmayan duran varlığın yeniden değerlendirme sonucunda değer azalmışsa, önceki dönemlerdeki diğer kapsamlı gelirden muhasebeleştirilen yeniden değerlendirme artışı ile içinde bulunulan dönemdeki değer azalışına karşılık mahsup edilir. Değer azalışından kalan tutar olursa gider yazılarak kar veya zarar arasında muhasebeleştirilir. (TMS-38, 2015: md.85-86)

Şekil 14’deki maddi duran varlıkların yeniden değerlemesi ile ilgili hiyerarşi maddi olmayan duran varlıklar için de geçerlidir. Hangi şart (değer artışı veya azalışı) sağlanıyorsa maddi duran varlıklar gibi muhasebeleştirilebilir.



Şekil 14: Maddi Olan / Olmayan Varlıkların Yeniden Değerleme Modeli

Kaynak: (Akbulut ve Marşap, 2006: 92)

Örneğin; ilk değerlemede değer artışı söz konusu olduğunda 522 M.D.V. Yeniden Değerleme Artışları Hesabının “Alacak” tarafına yazılır. İlerleyen dönemlerde değer azalışı olursa önce bu hesap sıfırlanıncaya kadar “Borç” yazılır. Hâlâ daha değer düşüklüğü var ise 654 Karşılık Giderleri Hesabının borcuna yazılmalıdır. (Sarıay, 2012: 117)

Örneğin; ilk değerlemede değer düşüklüğü varsa, önce 654 Karşılık Giderleri Hesabı borçlandırılır, gelecek dönemlerde bir değer artışı söz konusu olursa (654’teki bakiye aşılmaya kadar) 644 Konusu Kalmayan Karşılıklar Hesabının alacak tarafına yazılır. Daha sonra 522 numaralı hesabın alacak tarafına yazılır. (Fırat, 2008: 112) Değer artışı ve azalışlarında birikmiş amortisman ve varlığın hesabı ilgili oranla genişletilir veya daraltılır.

Alıntılarda ifade edilen 522 numaralı hesap maddi duran varlıklar ile ilgilidir. Alıntıya müdahale etmemek için değiştirilmemiştir. Maddi olmayan duran varlıkları ifade edebilecek hesap 524 numara ile kullanılabilir. (Usul ve Özdemir, 2007: 131)

3.2. Kullanım Amaçlı Geliştirilen Yazılımların Muhasebeleştirilmesi

İşletmeler, bazen ihtiyaç duydukları yazılımları satın almayı, kendi bünyelerinde geliştirmek isterler. Bunun için bir yazılım ekibi gerekir ve oldukça harcama yapılır. En sonunda maddi olmayan duran varlık açığa çıkar. TMS 38 standardı gereği aktifleştirilecek olan harcama kalemlerinin nelerden olabileceği açıklanmıştır. Ancak bu süreçlerin muhasebeleştirilerek bir varlık olarak kayıt edilebilmesi için öncelikle harcamaların üretimle ilişkisinin takip edilebildiği bir sisteme ihtiyaç vardır. Muhasebe bilgi sisteminin ihtiyaçlarına göre uyarlanmış proje yönetim yazılımlarıyla istenen bilgiler rahatlıkla elde edilebilir; ve muhasebe bilgi sistemiyle entegre edilebilir.

TMS 38’e uygun şekilde muhasebe bilgisi raporlayacak olan proje yönetim yazılımlarının takip etmesi gereken noktalar şunlardır: Harcama kalemleri, harcamaların dağıtımı, dönemsellik, proje geliştirme süreçleriyle ilişkilendirme. Tüm bu kriterlere dikkat ederek senaryolaştırılmış bir örnek yardımıyla geliştirme sürecinin başından sonuna kadar detaylıca açıklamalar 3.2.3 konulu başlıkta yapılacaktır.

Öyleyse senaryo şu şekildedir: İşletme, kendi iş süreçlerini, kaynaklarını takip edebilmek ve raporlar üretebilmek için yazılım satın almayı, kendi bünyesinde

geliştirmeye karar vermiştir; ve İnsan Kaynakları, Stok, Satın Alma, Satış Takibi, Finansal İşlemler, Muhasebe, Üretim, Raporlar gibi ana işlemlerin takip edildiği bir sistem istemektedir. Bunu elde etmek için de işletme bünyesinde geliştirme kararlaştırılmıştır. Bu senaryo ve senaryodaki değerler ve tablolar önceki çalışmamızda kullanılmıştır. (Calayoğlu ve Yılmaz, 2015: 130-153)

Bu stratejik ve yönetsel bir karardır. Çünkü bu amaç için bir dizi yatırımda bulunulması gerekecektir. Yatırımın başında ve özellikle devamında yapılacak olan harcamalar, işletmenin likiditesini önemli ölçüde etkileyecektir.

3.2.1. Modelin Kısıtları

Konunun uygulaması gerçek veriler ile çalışılmamıştır. Çünkü başvuru firmalar kendi iç işlerinin öğrenilmesi konusunda tedirgin olmuştur. Araştırma ve gözlemlere izin verilmemiştir. Bu sebeple bir senaryo gereği oluşturulan değer eşliğinde bir hesaplama modeli üzerinde çalışılmıştır.

Yazılım geliştirme modeli olarak “Şelale Modeli” tercih edilmiştir. Çünkü gerek geliştirme işlerini iş planında rahat gösterebilmek gerekse harcamaları takip edip, sınıflandırıp geliştirme faaliyetlerine dağıtabilmek daha kolay olacaktır.

Piyasadaki firmalar genellikle şelale modeline göre yazılım geliştirmez, çünkü çok fazla atıl kapasite oluşur ve harcamalar artar. Günümüzde geliştiricilerin tercih ettiği modeller çevik modellerdir. Ancak unutulmamalıdır ki çevik modeller döngüsel olsa da her döngü içinde şelale modelinin uygulanması gerekir. Yani, bir proje için birden çok şelale modelinin uygulanması şeklinde özetlenebilir. Burada esas olan geliştirme faaliyetlerini takip edebilmek ve harcamaların dağıtımını açıklayabilmektir. Bu sebeple izahı ve görseli en kolay olacak olan geliştirme modeli tercih edilmiştir.

İş planı, Gantt şemasından esinlenilerek excel 2013’de oluşturulmuştur. Yazılım geliştirme faaliyetlerinin tükettiği harcamaları hesaplamak için faaliyet sürücüsü olarak zaman esaslı tavsiye edilmektedir. (Walker ve Oliver, 2005: 69) Hesaplaması ve görsel olarak gösteriminin kolay olabilmesi için birim zaman olarak modelde hafta tercih edilmiştir. Yine gerçek bir yazılım geliştirme projesinde bir personel aynı gün içinde biraz A ve biraz B projesinde çalışabilir. Bu sebeple doğru bir hesaplama için çalışanın eforunun zaman cinsinden hangi projede ne kadar çalıştığının takip edilmesi

gerekmektedir. (Gentle, 2011: 43) Yani, birim zaman daha küçük bir birim olan saat veya dakika cinsinden olmalıdır.

3.2.2. Yazılım Geliştirme Süreci, Harcama Alanları ve TMS 38'in Yaklaşımı

Söz konusu yazılımı geliştirirken süreçler boyu yapılacak harcamaların araştırma ve geliştirme safhaları cinsinden ayrıştırılmasının yapılması gerekir. Bu sebeple tablo 17 dikkatle incelenmelidir.

Tablo 17'de görüldüğü üzere sol tarafta yazılım geliştirme safhaları ve satılabilir olduktan sonraki aşamalar (kırmızı çizginin altı) gösterilmiştir. Yan sütunda ise TMS 38'in araştırma ve geliştirme tanımlarına göre yazılım geliştirme süreçleriyle olan eşleştirme yapılmıştır. 2.1.3 başlığında detaylıca anlatıldığı gibi standartlar arasında küçük farklılıklar vardır. TMS 38'e göre isterlerin belirlenmesi ve analiz aşamalarında yapılan harcamalar, araştırma safhasına aittir ve giderleştirilir. Ürün, satılabilir oluncaya kadarki diğer geliştirme aşamaları geliştirme safhasına aittir ve aktifleştirilmesi gerekmektedir.

Yazılım geliştirirken ürünle direkt alakalı olan maliyete eklenebilecek harcama kalemleri yan sütunlarda belirtilmiştir. Böylece hangi geliştirme sürecindeki, hangi harcamanın nasıl muhasebeleşmesi gerektiği açıklanmaya çalışılmıştır. Bu tablo yardımı ile muhasebe kaydının yönü ortaya çıkmaktadır.

En alt satırda yer alan iyileştirici bakımın açıklaması, 1.2.7 başlığında ifade edilmiştir. Kısaca, yazılıma değer katan harcamalar yazılımın maliyetine eklenmesi gerektiği vurgulanmaktadır. (Stanford University, 2015: <http://www.stanford.edu/>) Diğer bakım türleri ise daima dönem gideri olarak kaydedilir. (Munter, 1997: 33)

Tablo 17: Yazılım Geliştirme Süreci, Harcama Alanları ve TMS 38'in Yaklaşımı

YAZILIM GELİŞTİRME SÜRECİ		TMS 38'e Göre		Yazılım Geliştirme Harcamaları ve TMS 38'e Göre Muhasebeleştirme Kriterleri					
Faaliyet	Yapılan İş	Araştırma	Geliştirme	Yazılım Alt Yapısı	Donanım Alt Yapısı	Genel Yönetim Giderleri	Faiz	Personel Harcamaları	Danışmanlık Hizmeti
İsterlerin Belirlenmesi - Analiz	İsterlerin Öğrenilmesi	✓		Gider	Gider	Gider	Gider	Gider	Gider
	İsterlerin Belirlenmesi	✓		Gider	Gider	Gider	Gider	Gider	Gider
	Yapısal Çözümleme	✓		Gider	Gider	Gider	Gider	Gider	Gider
	Nesneye Dayalı Çözümleme	✓		Gider	Gider	Gider	Gider	Gider	Gider
	Veri Yapılarına Yönelik Çözümleme	✓		Gider	Gider	Gider	Gider	Gider	Gider
	Belgelendirme	✓		Gider	Gider	Gider	Gider	Gider	Gider
Tasarım	Veri Tasarımı		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Mimari Tasarım		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Yordamsal Tasarım		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Arayüz Tasarımı		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Kullanıcı Arayüzü Tasarımı		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Veri Tabanı Tasarımı		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Belgelendirme		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Veri Bildirimi		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Kodlama		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Hata Ayıklama		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
Test	Belgelendirme		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Birim Testi		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Tümleştirme Testi		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Yeterlilik Testi		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Sistem Testi		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Kabul Testi		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Belgelendirme		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Phiyasaya Sürme veya Müşteriye Teslim		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Kurulum ve Onarım Belgelerinin Hazırlanması		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet
	Telefon Desteği		✓	Gider	Gider	Gider	Gider	Gider	Gider
Destek	Uzak Yardım Desteği		✓	Gider	Gider	Gider	Gider	Gider	Gider
	Yerinde Destek		✓	Gider	Gider	Gider	Gider	Gider	Gider
	Eğitim Desteği		✓	Gider	Gider	Gider	Gider	Gider	Gider
Bakım	Düzeltilici Bakım		✓	Gider	Gider	Gider	Gider	Gider	Gider
	Uyarılayıcı Bakım		✓	Gider	Gider	Gider	Gider	Gider	Gider
	Önleyici Bakım		✓	Gider	Gider	Gider	Gider	Gider	Gider
	Yıllıleştirici Bakım		✓	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet	Maliyet

3.2.3. Harcama Alanları, Tutarları ve Açıklamalar

Tablo 18’de yazılım geliştirme süreciyle direkt alakalı olan harcama kalemleri, muhasebe kayıtları için hesaplamaların yapıldığı matrah değerler, açıklamalar ve amortisman ayrılacak olanların itfa süreleri görülmektedir. Değerler, sonraki hesaplamaları somutlaştırmak için verilmiştir.

Tablo 18: Gider Çeşitleri, Tutarları ve Diğer Bilgiler

		Değerler	Açıklama	İtfa Süresi
Yazılım Alt Yapısı	Visual Studio 2013	2.760 TL	Geliştirme ortamı	1 Yıl
	MS SQL Server Standard Edition 2012	7.241 TL	Veritabanı	1 Yıl
	Crystal Report	1.124 TL	Raporlama Aracı	1 Yıl
	MS Team Foundation Server	6.810 TL	Çoklu çalışmaya olanak tanır ve dosya sürümü oluşturur.	1 Yıl
	Windows 7	2.100 TL	İşletim Sistemi	3 Yıl
	MS Office	780 TL	Dokümantasyon için	3 Yıl
Donanım Alt Yapısı	Ana Bilgisayar (Server)	10.000 TL	İ7 işlemci, 16 GB DDR3 1866 Mhz Ram, 2 GB 512 Mbit Ekran Kartı, Buna uygun anakart, 10 TB hard disk, DVD rom, 22 inç ekran	3 Yıl
	Kullanıcı Bilgisayarları	21.000 TL	İ3 işlemci, 4 GB DDR3 1866 Mhz Ram, 1 GB 256 Mbit Ekran Kartı, Buna uygun anakart, 1 TB hard disk, DVD rom, 22 inç ekran	3 Yıl
	Veri tabanı sunucusu	3.300 TL		3 Yıl
	Web Sunucusu	2.550 TL		3 Yıl
	Yerel Ağ Altyapısı	750 TL		3 Yıl
	Geniş Ağ Altyapısı	250 TL		3 Yıl
Yönetim Giderleri	Faturalar	700 TL		
	Büro Giderleri (Yiyecek-İçecek)	1.000 TL		
	Kırtasiye Giderleri	200 TL		
Faiz	24 ay vadeli kredi çekilmiştir.	50.000 TL	Ödeme Planı için tıklayınız.	
Yazılım Ekibi	Ahmet	6.000 TL	Proje Yöneticisi	
	Mehmet	5.000 TL	Deneyimli	
	Hasan	4.000 TL	Deneyimli	
	Hüseyin	3.000 TL	Deneyimli	
	Ömer	2.500 TL	Başlangıç Seviyesi	
	Ali	2.500 TL	Başlangıç Seviyesi	
	Veli	2.250 TL	Testçi	
Danışmanlık Hizmeti	Danışman Ücreti	1.500 TL		

Tablo 18’de üretimle alakalı olan her bir gider kaleminin detayları verilmiştir. Bazı gider grupları (yazılım ve donanım alt yapısı) sadece proje başlangıcında gider oluştururken diğer grupların giderleri her ay olmaktadır. Bu sebeple yazılım geliştirme ortamının başlangıç maliyetleri diğer sanayi üretim işletmelerine nazaran çok düşüktür. Gider gruplarını aşağıdaki şekilde kısaca açıklamak mümkündür;

- a) Yazılım Alt Yapısı: Yazılım üretebilmek için başka yazılımlara ihtiyaç vardır. Söz konusu duruma örnek olsun diye birkaç yazılım listeye eklenmiştir. Satın alınan yazılımların muhasebeleştirilmesine dair hükümler bu alana uygulanır ve 260 Haklar Hesabında ilk olarak kayıtlara geçirilir. Lisans kullanım sözleşmesinin süresi 1 yıl olduğu varsayılmıştır ve bu süre üzerinden itfa edilecektir.

Windows 7 ve MS Office yazılımları bilgisayarlar ile birlikte satın alındığında fiyatını ayrıştırılması mümkün olmadığında bilgisayarlar (demirbaşlar) ile birlikte muhasebeleşebilir. Örnekteki gibi fiyatının ayrı olarak belirlenebildiği durumda ise satın alınan yazılımlar (Haklar) statüsünde muhasebeleşmelidir.

İtfa payları aylık dönemlerde hesaplanacağından hesaplaması karmaşık değildir. İtfa payları araştırma veya geliştirme safhasının belirlenmesine göre aktif veya gider hesaplarında muhasebeleşecektir.

Mavi renklendirilmiş olanlar sadece kodlama aşamasında kullanılacak olmaları sebebiyle diğer geliştirme süreçlerine pay verilmeyecektir. Kodlama sürecinin toplam zamanı ise çalışan 4 yazılımcının toplam zamanı ($4*4=16$) üzerinden oranlanarak hesap edilir.

Mavi Renklendirilmemiş olan kalemlerin matrahları ilgili aydaki geliştirme süreçlerinin toplam zaman içindeki payına göre hesaplanır.

- b) Donanım Alt Yapısı: Sadece yazılım bir şey ifade etmez, yazılımların çalışacağı ortam gereklidir. Bunlar bilgisayarlar gibi fiziki olan araç ve cihazlardır. Bunun yanında bilginin oluşturulması, depolanması ve iletimi için gerekli diğer parçalar da bu grupta yer almaktadır. Muhasebe açısından 255 Demirbaşlar Hesabına kayıt edilir.

Yararlı ömürleri 3 yıldır. İtfa payları aylık dönemlerde hesaplanacağından hesaplaması karmaşık değildir. İtfa payları araştırma veya geliştirme safhasının belirlenmesine göre aktif veya gider hesaplarında muhasebeleşir.

Mavi renklendirilmiş olanlar sadece kodlama aşamasında kullanılacak olmaları sebebiyle diğer geliştirme süreçlerine pay verilmeyecektir. Kodlama sürecinin toplam zamanı ise çalışan 4 yazılımcının toplam zamanı ($4*4=16$) üzerinden oranlanarak hesap edilir.

Mavi Renklendirilmemiş olan kalemlerin matrahları ilgili aydaki geliştirme süreçlerinin toplam zaman içindeki payına göre hesaplanır.

- c) Yönetim Giderleri: Üretimle ilişkili olan dışarıdan sağlanan faydalar için yapılan harcamaların ayrıştırılabilmesi durumunda maliyete eklenmesinde bir sakınca bulunmamaktadır. Bu harcamalar yazılım ekibinin bulunduğu kısımla alakalıdır. Bu tür giderler zaten aylık dönemlerde fatura edildiğinden dönemsellik ilkesine uygunluk için ek çaba gerekmemektedir. Fatura tutarının ne kadarının araştırma ne kadarının geliştirme safhasının belirlenebilmesi için ilgili aydaki geliştirme süreçlerinin toplam zaman içindeki payına göre hesaplanır.

Örnekteki yönetim giderleri sabit olup yılsonuna kadar değişmeyeceği varsayılmıştır.

- d) Faiz: Yazılım üretimi özellikli bir varlık olduğundan bu süreç için kullanılan kredinin faiz kısmı projenin geliştirme dönemlerinde yazılımın maliyetine eklenmesi gerekir. (Usul ve Özdemir, 2007: 125) Aktifleştirmeye başlamanın tarihi kredinin çekildiği zamandan itibaren değil, projeye başlangıç tarihinden itibaren olmalıdır. Örnekteki kredi, proje başlangıç harcamalarına yetebilmek için çekilmiştir. Tamamen proje ile alakalıdır. Dağıtım yöntemi, ilgili aydaki geliştirme sürecinin toplam zamanı içindeki ilgili safhanın payına göre hesaplanmaktadır. Bu kaynağın başka projelere dağıtımı söz konusu olmadığından çalışılan süreçler arasında tamamen dağıtılması gereklidir. Böylece dağıtılmayan pay kalmaz.
- e) Yazılım Ekibi: Bir yazılım projesinde en fazla harcamayı oluşturan kalem personel giderleridir. Çünkü diğer harcama gruplarının çoğu bir kere yapılır ve diğer aylar yapılmaz. Personel ücretleri her ay ödenir, bu tip projelerin süresinin

de uzunluđu düşünöldüğünde yazılımın toplam maliyetinin içindeki en fazla pay, personele yapılan ödemelerdir. Örnekteki personel ücretleri sabit olup yılsonuna kadar değışmeyeceđi varsayılmıştır.

Ödenen personel ücretlerinin ne kadarının araştırma ne kadarının geliştirme safhasına ait olduđunun belirlenebilmesi için ilgili ayda personelin toplam zamanını hangi süreçlerde geçirdiđine göre oranlaması yapılır ve hesaplanan tutarlar, aktif veya gider hesaplarında muhasebeleşir.

- f) Danışmanlık Ücreti: İyi bir yazılım ekibinizin olması, mevzuatlara uygun, gelecekteki sıkıntıları bugünden çözüme kavuşturan mükemmel bir yazılım üretilmesine imkân vermeyecektir. Çünkü yazılım ekibi, yazılımın üretilmesindeki teknik kısımları iyi yönetmekle sorumludur. Zaman zaman işlerin nereye doğru yönleneceği veya mevzuata uygunluklar gibi durumların danışılması gerekir ki yazılımın gidişatı olumlu şekilde planlanabilsin. Bu ve benzeri konularda piyasa deneyimi olan veya mesleki yeterlilikleri olan kişilerden fikir alınır.

Danışmanlık hizmet alımının geliştirme süreci boyunca devam edip, her süreç için alınacağı ve tutarın değışmeyeceđi varsayılmıştır. Tamamen proje ile alakalıdır. Dağıtım yöntemi, ilgili aydaki geliştirme sürecinin toplam zamanı içindeki ilgili safhanın payına göre hesaplanmaktadır. Bu kaynağın başka projelere dağıtımını söz konusu olmadığından çalışılan süreçler arasında tamamen dağıtılması gereklidir. Böylece dağıtılmayan pay kalmaz.

Yukarıda açıklanan harcama gruplarını muhasebeleştirebilmek için aylık dönemlerde hesaplamalar yapılmalı ve araştırma, geliştirme veya hiçbir cinsinde kategorize etmek gereklidir.

Tablo 18’de detaylıca gösterilmemiş olan harcama kalemi çekilen kredidir. Geliştirme süreçlerine dağıtılması için takip edilmesi gerekir.

3.2.4. Kredi Detayları ve Açıklamalar

Projenin finansmanı için çekilen kredinin faizi, TMS 38 geređi aktifleştirilecek ve giderleştirilecek bir harcama kalemidir. Bu sebeple kredinin detayları bilinmelidir.

Tablo 19 yardımı ile aylık taksit tutarını oluşturan bileşenler rahatlıkla takip edilerek muhasebe kayıtları için sınıflandırmaya imkân tanımaktadır.

Tablo 19: Kredi Ödeme Tablosu

Vade	TAKSİT TUTARI	AYLIK ANAPARA	FAİZ	KKDF	BSMV	KALAN BORÇ
1	2.444,33 TL	1.784,33 TL	550,00 TL	82,50 TL	27,50 TL	48.215,67 TL
2	2.444,33 TL	1.807,89 TL	530,37 TL	79,56 TL	26,52 TL	46.407,78 TL
3	2.444,33 TL	1.831,75 TL	510,49 TL	76,57 TL	25,52 TL	44.576,03 TL
4	2.444,33 TL	1.855,93 TL	490,34 TL	73,55 TL	24,52 TL	42.720,09 TL
5	2.444,33 TL	1.880,43 TL	469,92 TL	70,49 TL	23,50 TL	40.839,66 TL
6	2.444,33 TL	1.905,25 TL	449,24 TL	67,39 TL	22,46 TL	38.934,41 TL
7	2.444,33 TL	1.930,40 TL	428,28 TL	64,24 TL	21,41 TL	37.004,01 TL
8	2.444,33 TL	1.955,88 TL	407,04 TL	61,06 TL	20,35 TL	35.048,13 TL
9	2.444,33 TL	1.981,70 TL	385,53 TL	57,83 TL	19,28 TL	33.066,43 TL
10	2.444,33 TL	2.007,86 TL	363,73 TL	54,56 TL	18,19 TL	31.058,57 TL
11	2.444,33 TL	2.034,36 TL	341,64 TL	51,25 TL	17,08 TL	29.024,21 TL
12	2.444,33 TL	2.061,22 TL	319,27 TL	47,89 TL	15,96 TL	26.963,00 TL
13	2.444,33 TL	2.088,42 TL	296,59 TL	44,49 TL	14,83 TL	24.874,57 TL
14	2.444,33 TL	2.115,99 TL	273,62 TL	41,04 TL	13,68 TL	22.758,58 TL
15	2.444,33 TL	2.143,92 TL	250,34 TL	37,55 TL	12,52 TL	20.614,66 TL
16	2.444,33 TL	2.172,22 TL	226,76 TL	34,01 TL	11,34 TL	18.442,44 TL
17	2.444,33 TL	2.200,89 TL	202,87 TL	30,43 TL	10,14 TL	16.241,55 TL
18	2.444,33 TL	2.229,95 TL	178,66 TL	26,80 TL	8,93 TL	14.011,60 TL
19	2.444,33 TL	2.259,38 TL	154,13 TL	23,12 TL	7,71 TL	11.752,22 TL
20	2.444,33 TL	2.289,21 TL	129,27 TL	19,39 TL	6,46 TL	9.463,01 TL
21	2.444,33 TL	2.319,42 TL	104,09 TL	15,61 TL	5,20 TL	7.143,59 TL
22	2.444,33 TL	2.350,04 TL	78,58 TL	11,79 TL	3,93 TL	4.793,55 TL
23	2.444,33 TL	2.381,06 TL	52,73 TL	7,91 TL	2,64 TL	2.412,49 TL
24	2.444,33 TL	2.412,49 TL	26,54 TL	3,98 TL	1,33 TL	0,00 TL
Toplam	58.663,92 TL	49.999,99 TL	7.220,03 TL	1.083,01 TL	361,00 TL	

Buraya kadar açıklananlar, yazılım geliştirme süreçlerinde yapılacak harcama kalemlerinin tespiti ve hesaplamalarda kullanılacak olan değerlerdir. Bir sonraki aşama, yapılacak harcamaların ayrıştırılması (araştırma, geliştirme, hiçbiri) için geliştirme süreçleri ile harcamaların dönemlerini eşleştirmektir. Bunun için de hangi işin ne zaman yapıldığının bilinmesi gerekmektedir.

3.2.5. Örneğin 2014 Yılı Gantt Şeması

Şekil 9’da bir Gantt şeması örneği verilmiştir. Hatırlanacağı üzere yazılım proje sorumlusu yapılacak işlerin planını yapar. Böylece yazılım bütçesi ve yapılacak işlerin detayları, kimin üstleneceği, ne kadar süre geçeceği gibi birçok bilgi açığa çıkar. Bunun grafik şeklindeki sunumuna Gantt şeması denmektedir. Bu şema şuanda proje yönetim yazılımları ile kolaylıkla elde edilmektedir, ancak muhasebe ihtiyaçlarına cevap verebilecek şekilde değildir. Olması gereken dönemselliğe ve TMS 38’e göre kategorize edilmiş raporlamalardır.

Örneğin Gantt şeması ise 6 şar aylık şeklinde tablo 20’deki gibidir.

Tablo 20: Örneğin 2014 Yılı Gantt Şeması

Aylar	Ocak '14				Şubat '14				Mart '14				Nisan '14				Mayıs '14				Haziran '14			
Haftalar	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Ahmet (Proje Yöneticisi)	İsterlerin Öğrenilmesi				Yapısal Çözümleme				Veri Yapılarına Yönelik Çözümleme															
Mehmet (Analist)	İhtiyaçların Belirlenmesi				Nesneye Dayalı Çözümleme				Belgelendirme															
Hasan (Deneyimli Yazılımcı)									Veri Tasarımı		Yordamsal T.		Arayüz Tasarımı		Veri Tabanı T.		Veri B.							
Hüseyin (Deneyimli Yazılımcı)									Mimari Tasarım		Yordamsal T.		Kullanıcı Arayüzü T.		Veri Tabanı T.		Veri B.							
Ömer (Başlangıç Seviyesi)											Yordamsal Tas.		Arayüz Tasarımı		Belgelendirme		Veri B.							
Ali (Başlangıç Seviyesi)											Yordamsal Tas.		Kullanıcı Arayüzü T.		Belgelendirme		Veri B.							
Veli (Testçi)																								

Aylar	Temmuz '14				Ağustos '14				Eylül '14				Ekim '14				Kasım '14				Aralık '14			
Haftalar	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
Ahmet (Proje Yöneticisi)																								
Mehmet (Analist)									Kurulum ve Onarım Belgelerinin Hazırlanması															
Hasan (Deneyimli Yazılımcı)									Kodlama - Hata Ayıklama								Piyasaya Sürme veya Müşteriye Teslim							
Hüseyin (Deneyimli Yazılımcı)									Kodlama - Hata Ayıklama								Piyasaya Sürme veya Müşteriye Teslim							
Ömer (Başlangıç Seviyesi)									Kodlama - Hata Ayıklama								Piyasaya Sürme veya Müşteriye Teslim							
Ali (Başlangıç Seviyesi)									Kodlama - Hata Ayıklama								Piyasaya Sürme veya Müşteriye Teslim							
Veli (Testçi)	Birim Testi - Tümlleştirme Testi - Yeterlilik Testi - Sistem Testi								Kabul Testi				Belgelendirme											

Tablo 20’de kullanılan her rengin bir anlamı vardır. Renkler, yazılım geliştirme süreçlerinden hangisine ait olduğunu ifade etmektedir. Her geliştirme sürecine ait birkaç iş, tablo 17’de detaylıca verilmiştir. Gantt şemalarında renklerin içinde yazan işler tablo 17’den seçilmiş birer iştir. Tablo 21’de renklerin anlamları verilmiştir.

Tablo 21: Gantt Şemasındaki Renklerin Anlamları

	İsterlerin Belirlenmesi - Analiz
	Tasarım
	Kodlama
	Test
	Devreye Alma

Tablo 21'den de anlaşılacağı üzere geliştirme modelimiz doğrusaldır. Görülen beyaz karelerin anlamı ise o vakitte ve o kişinin örnekte belirtilen yazılımı üretmek için çaba sarf etmediğidir. Anlatımı kargaşadan arındırmak için bu tercihler yapılmıştır. Oysa piyasada aylığı ödenen bir çalışanın bu kadar boş kalması işletmenin oldukça fazla atıl kapasiteyle çalışması anlamına gelmektedir. İstenmeyen bir durumdur. Bir projede işi biten çalışan başka bir projeye yönlendirilir. Keza daha verimli yazılım geliştirme modellerinden döngüsel modeller piyasada daha çok tercih edilmektedir.

Gelecekte, ihtiyaç olunan muhasebe bilgisinin üretimi için proje yönetim yazılımlarıyla çalışacak olan yazılımcılar, tabii ki döngüsel geliştirme modellerinde sonuç verecek olan algoritmik tanımlamaları yapmaları gerekecektir.

Daha rahat anlatım ve gösterim yapılabilmesi için tüm ayların 4 hafta olduğu ve işlerin de haftalık düzeyde bittiği ve başladığı varsayımı yapılmıştır.

Yukarıdaki hesaplamayı daha gerçekçi kılabilmek için hafta üzerinden değil, gün; hatta saat üzerinden hesaplamak gereklidir. Eğer bu seviyelerde veri kayıt edilebilirse gerisi kolaylaşır. Proje yönetim yazılımları bu seviyede veri kayıt edebilmektedir; ancak kaynaklar ile ilişkilendirilip muhasebe bilgi sistemine entegre edilmedikçe, hâlâ muhasebe açısından teorinin uygulanabilirliği zor olmaktadır.

3.2.6. Aylık Bazda Harcama Kalemleri, Sınıflandırma ve Muhasebeleştirme

İşletmenin sürdürdüğü yazılım geliştirme projesi için sadece personel ücretlerinin olmadığı bunun yanında daha başka harcama kalemlerinin olduğundan bahsedilmişti. Tablo 20 esastır çünkü fiili olarak yapılan işi ortaya koyar ve hesaplamalar bunun üzerinden gelişir. Bu itibarla faaliyetleri aylık dönemlerde incelemek gerekir ki; yapılan harcamaları araştırma, geliştirme veya hiçbiri cinsinden kategorize ederken dönemselliğe dikkat edilmiş olsun. Harcamaların dağıtımı için dağıtım anahtarı kullanılacaktır.

Dağıtım anahtarı; dağıtılacak matrahın (aya düşen payın) geliştirme süreçlerindeki kullanım oranına göre (rengine) göre ayırır. Ancak bunu gerçekleştirirken kimi zaman çalışanların başka projelerde de çalışabilmesi mümkün olduğundan toplam kapasite ile oranlanır kimi zaman projeye ait süreçlerin aylık toplam zamanına

oranlanır. Yani, toplam zamandaki veya toplam proje içindeki pay şeklinde özetlenebilir.

Unutmamak gerekir ki; gerçekte bir kişi birden fazla projede çalışabilir, dolayısı ile kişinin çabasının ne kadarını hangi projede veya geliştirme sürecinde olduğunu anlayabilmek için projelerde veya aşamalarda geçen sürelerle göre dağıtım yapılmalıdır. Bu sebeple de gün sonunda yazılım ekibi personelinin gün içinde hangi projede ne iş yaptığını ve ne kadar süre harcadığını rapor etmesi gereklidir. Bunu otomatik sağlayabilmek amacıyla bazı yazılımların kullanımları için oluşturulan kullanıcı hesapları kullanılabilir. Kullanıcı hesaplarında geçen süre üzerinden rahatlıkla personelin hangi işler ile ne kadar süre geçirdiğinin verisi elde edilebilir. Dağıtım anahtarı olarak gerçekçi veriler kullanılmış olur.

Tablo 20’de gösterilen Gantt şemasının aylık bazda incelenmesiyle projeye ait işlemler ve Ocak ayındaki harcama grupları tablo 22’de ortaya konmuştur. Ardından işlemler kategorize edilerek muhasebe kayıtları açıklanmaya çalışılmıştır.

İlk muhasebe kayıtları tablo 22’nin ortasında görülebilir. Bu kayıtlara, sonraki kayıtların anlaşılmasını kolaylaştırması bakımından yer verilmiştir. Yazılım ve donanım alt yapısına dair yapılan ilk kayıtlar için şunlar söylenebilir; her yazılım ve donanım için ana muhasebe hesabının altına ilgili varlık için bir alt hesap açılmalı ve bu hesapta bakiyeler takip edilmelidir.

Faiz ile ilgili hesaplamada finansman gideri olan faiz, KKDF ve BSMV tutarları anaparadan ayrı olarak kısa ve uzun vade cinsinden ayrıştırılarak kayıtlara geçirilmiştir. Toplam borç miktarı da uzun vadeli banka kredileri hesabına kayıt edilmiştir.

Ocak ayında yapılan işler (Tablo 22), isterlerin belirlenmesi ve analiz aşamasına aittir. Sadece 2 kişi bu projede çalışmış ve bu aşamada görev almıştır. Diğer kişilerin başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların ocak ayına ait payları kodlama ile alakalı olmadığından herhangi bir geliştirme sürecinde raporlanmamıştır ve sonuç olarak “Ar-Ge Değil” sınıfına pay edilmiştir. Mavi ile işaretlenmemişler ise; (Örneğin; Windows 7) şu şekilde hesaplanmıştır. Toplamda 7 kişi 4 haftadan ilgili kaynağı kullanabilecekken sadece 2 kişi 4 hafta boyunca kullanmıştır. Dolayısı ile dağıtılacak

rakam $(58)/28*8=17$ olarak bulunur. Dağıtılamayan pay Ar-Ge Değil kısmına gider, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta=28 haftadır. Örneğin; faturalar şöyle hesaplanmıştır: İsterlerin belirlenmesi ve analiz süreci için dağıtılacak rakam $(700)/28*8=200$ TL. Dağıtılamayan pay Ar-Ge Değil kısmına gider, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 8 hafta ve bunun tamamı tek bir süreçte geçtiğinden matrahın tamamı aktarılır. Dağıtılacak rakam $(660)/8*8=660$ TL isterlerin belirlenmesi ve analiz sürecine yazılmıştır.

Yazılım ekibinin ücretlerinin dağıtılması konusunda çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Ahmet ve Mehmet, maaşları karşılığında tamamen isterlerin belirlenmesi ve analiz aşamalarında çalışmışlardır. Bu sebeple her birinin maaşı söz konusu faaliyete bölünmeden aktarılmıştır. Dağıtılamayan pay, Ar-Ge Değil kısmına gider, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımı da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 8 hafta ve bunun tamamı tek bir süreçte geçtiğinden matrahın tamamı aktarılır. Dağıtılacak rakam $(1500)/8*8=1500$ TL isterlerin belirlenmesi ve analiz sürecine yazılmıştır.

Diğer meblağlar da faaliyet sürücüsü kullanılarak yazılım geliştirme süreçlerine dağıtılmıştır. Böylece ocak ayında (Tablo 22) yapılan işler incelendiğinde toplamda 14.000 TL araştırma, 0 TL geliştirme ve 17.936 TL Ar-Ge olmayan şeklinde sınıflandırılmıştır. Ar-Ge olmayan rakamın çok büyük kısmının başka projelerin araştırma veya geliştirme safhalarına dağıtılabileceğini unutmamak gerekir.

Bu sonuçlara göre tablonun en sağ kısmında araştırma safhasına ait olanlar 750 hesapta, geliştirme safhasına ait olanlar 263 hesapta ve ar-ge olmayan kategorisindekiler de 770 veya 780 hesapta muhasebeleşmiştir. Yine her bir varlık için bir alt hesap açmak gerektiğini unutmamak gereklidir. Aşağıya doğru inildikçe safhalara göre borçlu

hesapların deęiřtięini alacak hesapların deęiřmedięi grlecektir. ünkü ama aktif olanı giderden ayırmaktır.

263 numaralı hesabın kırmızı olarak yazılmıř olmasının sebebi bir noktaya dikkat ekmek iindir. O da řudur; 750 ve 263 numaralı hesapların isimleri “Arařtırma ve Geliřtirme Giderleri” ifadesi řeklinde aynıdır. Biri gider dięeri varlık cinsi bir hesaptır. TMS 38 tanımlamasına gre arařtırma giderleri aktifleřtirilemezdir. Bu sebeple 263 numaralı hesabın adından bu ifadenin kaldırılması gerekmektedir. “Geliřtirilen Varlık” bařlıęı altında geliřtirilen yazılımın amacının (satıř veya kullanım) belirtilerek kaydedilmesini saęlayan biimde olması gerekir. rneęin; Satıř Amalı Geliřtirilen Varlıklar ve Kullanım Amalı Geliřtirilen Varlıklar gibi.

TMS 38 tanımlamasına gre geliřtirme harcamaları giderleřtirilemez. Bu sebeple 750 numaralı hesabın isminin sadece “Arařtırma Giderleri” olarak dzenlenmesi daha doęru olacaktır. Bu řekilde dzenlemeler yapılırsa hesap isimleri ve iřleyiřleri ile ifadelerin kavramsal btnlęnn eliřmemesi saęlanmış olur.

Tablo 22: Ocak Ayı Harcama Kalemleri ve Muhasebeleştirilmesi

Ayarlar	Ocak '14				Varlık / Kredi Alımı	Faliyet Sürecüsü	Yazılım Geliştirme Süreçleri				Satış Sonrası				Araştırma Harcamasının Muhasebe Kaydı				
	1	2	3	4			İhtisap	Test	Devreye Alma	Bakım	Destek	Araştırma Harcaması	Geliştirme Harcaması	Ar-Ge Değil	Ar-Ge Değil	H. Kodu	Hesap Adı	Boş	Alacak
Ahmet (Proje Yöneticisi)																			
Mehmet (Analist)																			
Hasan (Deneyimli Yazılımcı)																			
Hüseyin (Deneyimsiz Yazılımcı)																			
Ömer (Başlangıç Seviyesi)																			
Ali (Başlangıç Seviyesi)																			
Visual Studio 2013	230 TL	2.760 TL			2.760 TL														
MS SQL Server Standard Edition 2012	603 TL	7.241 TL			7.241 TL														
Crystal Report	94 TL	1.124 TL			1.124 TL														
MS Team foundation server	568 TL	6.810 TL			6.810 TL														
Windows 7	58 TL	2.100 TL			2.100 TL														
MS Office	22 TL	780 TL			780 TL														
Ana Bilgisayar (Server)	278 TL	10.000 TL			10.000 TL														
Kullanıcı Bilgisayarları	583 TL	21.000 TL			21.000 TL														
Veri tabanı sunucusu	92 TL	3.300 TL			3.300 TL														
Web Sunucusu	71 TL	2.550 TL			2.550 TL														
Yerel Ağ Altyapısı	21 TL	750 TL			750 TL														
Geniş Ağı Altyapısı	7 TL	250 TL			250 TL														
Faturalar	700 TL																		
Büro Giderleri (Yıvecek-İpecek)	1.000 TL																		
Kira/İşe Giderleri	200 TL																		
Yatırım Alı Yaptırı																			
Faiz	660 TL	50.000 TL			50.000 TL														
Ahmet	6.000 TL	50.000 TL			50.000 TL														
Mehmet	5.000 TL																		
Hasan	4.000 TL																		
Hüseyin	3.000 TL																		
Ömer	2.500 TL																		
Ali	2.250 TL																		
Veil	2.250 TL																		
Danışmanlık Hizmeti	1.500 TL																		
Toplam																			

Şubat ayı iş planı incelendiğinde (Tablo 23) isterlerin belirlenmesi ve analiz safhasına ait işlemler için çaba serf edildiği görülmektedir. Sadece 2 kişi bu projede çalışmış ve bu aşamada görev almıştır. Diğer kişilerin başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların şubat ayına ait payları kodlama ile alakalı olmadığından herhangi bir geliştirme sürecinde raporlanmamıştır ve sonuç olarak “Ar-Ge Değil” sınıfına pay edilmiştir. Mavi ile işaretlenmemişler incelenirse, (Örneğin; MS Office) şu şekilde hesaplanmıştır. Toplamda 7 kişi 4 haftadan ilgili kaynağı kullanabilecekken sadece 2 kişi 4 hafta boyunca kullanmıştır. Dolayısı ile dağıtılacak rakam $(22)/28*8=6$ olarak bulunur. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta = 28 haftadır. Örneğin; büro giderleri şöyle hesaplanmıştır: İsterlerin belirlenmesi ve analiz süreci için dağıtılacak rakam $(1.000)/28*8=286$ TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 8 hafta ve bunun tamamı tek bir süreçte geçtiğinden matrahın tamamı aktarılır. Dağıtılacak rakam $(636)/8*8=636$ TL isterlerin belirlenmesi ve analiz sürecine yazılır.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Ahmet ve Mehmet, maaşları karşılığında tamamen isterlerin belirlenmesi ve analiz aşamalarında çalışmışlardır. Bu sebeple her birinin maaşı söz konusu faaliyete bölünmeden aktarılmıştır. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımı da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 8 hafta ve bunun tamamı tek bir süreçte geçtiğinden matrahın tamamı aktarılır. Dağıtılacak rakam $(1500)/8*8=1500$ TL isterlerin belirlenmesi ve analiz sürecine yazılır.

Mart ayı iş planı incelendiğinde (Tablo 24) isterlerin belirlenmesi ve analiz safhasına ait işlemler ile tasarım aşamasına ait işlemlerin yapılması için çaba serf edildiği görülmektedir. 2 kişi isterlerin belirlenmesi ve analiz aşaması için, 4 kişi de tasarım aşaması için bu projede çalışmıştır. Boş karelerde başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların mart ayına ait payları kodlama ile alakalı olmadığından herhangi bir geliştirme sürecinde raporlanmamıştır ve sonuç olarak “Ar-Ge Değil” sınıfına pay edilmiştir. Mavi ile işaretlenmemişler incelenirse, (Örneğin; Ana Bilgisayar) şu şekilde hesaplanmıştır. Toplam zaman, 7 kişi x 4 hafta= 28 hafta. Toplam zamandaki isterlerin belirlenmesi ve analiz sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam $(278)/28*8=79$ TL. Toplam zamandaki tasarım sürecinin payı ise şu şekilde hesaplanır. Dağıtılacak rakam $(278)/28*12=119$ TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta=28 haftadır. Örneğin; kırtasiye giderleri şöyle hesaplanmıştır: İsterlerin belirlenmesi ve analiz süreci için dağıtılacak rakam $(200)/28*8=57$ TL. Tasarım süreci için dağıtılacak rakam $(200)/28*12=86$ TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 20 hafta ve bunun bir kısmı isterlerin belirlenmesi ve analiz sürecinde diğer bir kısmı da tasarım sürecinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(613)/20*8=245$ TL isterlerin belirlenmesi ve analiz sürecine yazılır ve $613/20*12= 368$ TL de tasarım sürecine yazılır.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Ahmet ve Mehmet, maaşları karşılığında tamamen isterlerin belirlenmesi ve analiz aşamalarında çalışmışlardır. Bu sebeple her birinin maaşı söz konusu faaliyete bölünmeden aktarılmıştır. Sonraki 4 kişi tasarım sürecinde çalışmışlardır ve sağladıkları çabaları oranında maaşları sürece dağıtılmıştır. Hasan ve Hüseyin, tam olarak; Ömer ve Ali'nin

maařlarının ise yarısı aktarılmıřtır. Dađıtılamayan pay, Ar-Ge Deđil kısmına yazılır, bu proje iin alıřmamıřtır.

Danıřman creti tamamen bu proje iin dendiđinden bu harcamanın dađıtımı da geliřtirme sreleri arasında pay edilmelidir. rneđin; geliřtirme srelerinde geen sre 20 hafta ve bunun bir kısmı isterlerin belirlenmesi ve analiz srecinde diđer bir kısmı da tasarım srecinde gemiřtir. Dađıtım iin kendi iinde oranlama yapmak gereklidir. Dađıtılacak rakam $(1500)/20*8=600$ TL isterlerin belirlenmesi ve analiz srecine yazılır ve $1500/20*12= 900$ TL tasarım srecine yazılır.

Mart ayında yapılan iřler incelendiđinde (Tablo 24) toplamda 12.685 TL arařtırma, 12.027 TL geliřtirme ve 7.176 TL Ar-Ge olmayan řeklinde sınıflandırılmıřtır. Ar-Ge olmayan rakamın ok byk kısmının bařka projelerin arařtırma veya geliřtirme safhalarına dađıtılabileceđini unutmamak gerekir.

Tablo 24: Mart Ayı Harcama Kalemleri ve Muhasebeleştirilmesi

Açıklama	Mart '16												
	9	10	11	12	Yazılım Geliştirme Süreçleri								Satış Sonrası
Aylar					inyöz belirleme / Analiz	Tasarım	Kodlama	Test	Devreye Alma	Bakım	Destek	Araştırma Harcamasının Muhasebe Kaydı	
Harcama					Analiz	Tasarım	Kodlama	Test	Devreye Alma	Bakım	Destek	Araştırma Harcaması	Geliştirme Harcamasının Muhasebe Kaydı
Yazılım Altvapısı	Visual Studio 2013			230 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	23 TL	23 TL
	MSSQL Server Standard Edition 2012			603 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	34 TL	34 TL
	Crystal Report			94 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	MSS Team foundation server			568 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Windows 7			58 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
Donanım Altvapısı	MSS Office			22 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Ana Bilgisayar (Server)			278 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Kullanıcı Bilgisayarları			583 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Veri tabanı sunucusu			92 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Web Sunucusu			71 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Yerel Ağ Altvapısı			21 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Geniş Ağ Altvapısı			7 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Faturalar			700 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
Yatırım Giderleri	Büro Giderleri (Yıycak-İpecek)			1.000 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	543 TL	543 TL
	Kırtasiye Giderleri			200 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	814 TL	814 TL
				0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
				0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
Fazl	24 ay vadeli 50.000 TL kredi alındı.			613 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	245 TL	245 TL
				0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
Yazılım Ekibi	Ahmet			6.000 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Mehmet			5.000 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Hakan			4.000 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Hüseyin			3.000 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Ömer			2.500 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
	Ali			2.500 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
Danışmanlık Hizmeti	Veil			2.250 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
				1.500 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
				0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL
Toplam			31.889 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	0 TL	12.685 TL	12.027 TL	31.889 TL

NOT: Sadece kodlama aşamasında kullanılanlar.

Nisan ayı iş planı incelendiğinde (Tablo 25) isterlerin belirlenmesi ve analiz safhasına ait işlemler ile tasarım aşamasına ait işlemlerin yapılması için çaba serf edildiği görülmektedir. 2 kişi isterlerin belirlenmesi ve analiz aşaması için, 4 kişi de tasarım aşaması için bu projede çalışmıştır. Boş karelerde başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların nisan ayına ait payları kodlama ile alakalı olmadığından herhangi bir geliştirme sürecinde raporlanmamıştır ve sonuç olarak “Ar-Ge Değil” sınıfına pay edilmiştir. Mavi ile işaretlenmemişler incelenirse, (Örneğin; Kullanıcı Bilgisayarları) şu şekilde hesaplanmıştır. Toplam zaman, 7 kişi x 4 hafta= 28 hafta. Toplam zamandaki isterlerin belirlenmesi ve analiz sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam $(583)/28*8=167$ TL. Toplam zamandaki tasarım sürecinin payı ise şu şekilde hesaplanır. Dağıtılacak rakam $(583)/28*12=333$ TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta = 28 haftadır. Örneğin, fatura giderleri şöyle hesaplanmıştır: İsterlerin belirlenmesi ve analiz süreci için dağıtılacak rakam $(700)/28*8=200$ TL. Tasarım süreci için dağıtılacak rakam $(700)/28*16=400$ TL. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 24 hafta ve bunun bir kısmı isterlerin belirlenmesi ve analiz sürecinde diğer bir kısmı da tasarım sürecinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(588)/24*8=196$ TL isterlerin belirlenmesi ve analiz sürecine yazılır ve $(588)/24*16=392$ TL de tasarım sürecine yazılır.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere son çalışan hariç tüm çalışanlar çabalarını tüm ay boyunca aynı geliştirme sürecine harcamışlar dolayısı ile her birinin maaşı ilgili sürece aynen aktarılmıştır. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımı da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen

süre 24 hafta ve bunun bir kısmı isterlerin belirlenmesi ve analiz sürecinde diğer bir kısmı da tasarım sürecinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(1500)/24*8=500$ TL isterlerin belirlenmesi ve analiz sürecine yazılır ve $1500/24*16=1000$ TL tasarım sürecine yazılır.

Nisan ayında yapılan işler incelendiğinde (Tablo 25) toplamda 12.536 TL araştırma, 15.072 TL geliştirme ve 4.256 TL Ar-Ge olmayan şeklinde sınıflandırılmıştır. Ar-Ge olmayan rakamın çok büyük kısmının başka projelerin araştırma veya geliştirme safhalarına dağıtılabileceğini unutmamak gerekir.



Mayıs ayı iş planı incelendiğinde (Tablo 26) tasarım ve kodlama safhalarına ait işlemler için çaba sarf edildiği görülmektedir. 4 kişi önce tasarım sonra kodlama safhaları için çalışmıştır. Boş karelerde başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların Mayıs ayına ait payları kodlama ile alakalı olduğundan projede kullanılmaya başlamıştır dolayısı ile dağıtım tabii olacaktır. Bu kaynaklar sadece “kodcu” tabir edilen yazılımcılar tarafından kullanılır, dolayısı ile toplam süre 4 kişi * 4 hafta = 16 haftadır. Tüm yazılımcılar 3'er hafta kodlamada çalıştığından hesaplaması şu şekilde olacaktır. Örneğin; Visual Studio 2013 şöyle hesaplanmıştır: Dağıtılacak rakam $(230)/16*4=58$ TL kodlama sürecine yazılır. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Mavi ile işaretlenmemişler incelenirse, (Örneğin; Ana Bilgisayar) şu şekilde hesaplanmıştır. Toplam zaman, 7 kişi x 4 hafta= 28 hafta. Toplam zamandaki tasarım sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam $(278)/28*12=119$ TL. Toplam zamandaki kodlama sürecinin payı ise şu şekilde hesaplanır. Dağıtılacak rakam $(278)/28*4=40$ TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta = 28 haftadır. Örneğin; büro giderleri şöyle hesaplanmıştır: Tasarım süreci için dağıtılacak rakam $(1.000)/28*12=429$ TL. Kodlama süreci için dağıtılacak rakam $(1.000)/28*4=143$ TL. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 16 hafta ve bunun bir kısmı tasarım sürecinde diğer bir kısmı da kodlama sürecinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(564)/16*12=423$ TL kodlama sürecine gider ve $(564)/16*4=141$ TL de test sürecine yazılır.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Hasan, Hüseyin, Ömer ve Ali 3 hafta tasarım ve 1 hafta kodlama süreçlerinde çaba sarf etmişlerdir. Dolayısı ile her birinin maaşı $\frac{3}{4}$ oranında tasarım sürecine, $\frac{1}{4}$ oranında da kodlama

sürecine aktarılacaktır. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımı da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 16 hafta ve bunun bir kısmı tasarım sürecinde diğer bir kısmı da kodlama sürecinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(1500)/16*12=1125$ TL tasarım sürecine yazılır ve $1500/16*4=375$ TL kodlama sürecine yazılır.

Mayıs ayında yapılan işler incelendiğinde (Tablo 26) toplamda 0 TL araştırma, 16.140 TL geliştirme ve 15.700 TL Ar-Ge olmayan şeklinde sınıflandırılmıştır. Ar-Ge olmayan rakamın çok büyük kısmının başka projelerin araştırma veya geliştirme safhalarına dağıtılabileceğini unutmamak gerekir.

Tablo 26: Mayıs Ayı Harcama Kalemleri ve Muhasebeleştirilmesi

Açıklama	Mayıs '14				Faaliyet Süreci	Yazılım Geliştirme Süreci				Satış Sonrası			Araştırma Harcamasının Muhasebe Kaydı		
	17	18	19	20		Analiz	Tasarım / Kodlama	Test	Devreye Alma	Bakım	Destek	H. Kodu	Hesap Adı	Borç	Alacak
Ahmet (Proje Yöneticisi)															
Mehmet (Analist)															
Hasan (Deneyimli Yazılımcı)															
Hüseyin (Deneyimli Yazılımcı)															
Ömer (Başlangıç Seviyesi)															
Ali (Başlangıç Seviyesi)															
Veli (Testçi)															
Visual Studio 2013															
MS SQL Server Standard Edition 2012															
Crystal Report															
MS Team foundation server															
Windows 7															
MS Office															
Ana Bilgisayar (Server)															
Kullanıcı Bilgisayardan															
Veri tabanı sunucusu															
Web Sunucusu															
Yerel Ağ Altyapısı															
Geniş Ağ Altyapısı															
Faturalar															
Büro Giderleri (Niyacak-İpecek)															
Kirataşy Giderleri															
24 ay vadeli 50.000 TL kredi alındı.															
Ahmet															
Mehmet															
Hasan															
Hüseyin															
Ömer															
Ali															
Veli															
Danışman Ücreti															
Toplam															

Not: Sadece kodlama aşamasında kullanılanlar.

Haziran ayı iş planı incelendiğinde (Tablo 27) kodlama ve test safhalarına ait işlemler için çaba sarf edildiği görülmektedir. 4 kişi tamamen tasarım sürecinde çalışmıştır. 1 kişi de 3 hafta test sürecinde çalışmıştır. Boş karelerde başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların haziran ayına ait payları kodlama ile alakalı olduğundan projede kullanılmaya başlamıştır dolayısı ile dağıtıma tabii olacaktır. Bu kaynaklar sadece “kodcu” tabir edilen yazılımcılar tarafından kullanılır, dolayısı ile toplam süre 4 kişi * 4 hafta=16 haftadır. Tüm yazılımcılar bütün ay boyunca kodlamada çalıştığından hesaplaması şu şekilde olacaktır. Örneğin; Crystal Report şöyle hesaplanmıştır: Dağıtılacak rakam (94)/16*16=94 TL kodlama sürecine yazılır.

Mavi ile işaretlenmemişler incelenirse, (Örneğin; web sunucusu) şu şekilde hesaplanmıştır. Toplam zaman, 7 kişi x 4 hafta = 28 hafta. Toplam zamandaki kodlama sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam (71)/28*16=40 TL. Toplam zamandaki test sürecinin payı ise şu şekilde hesaplanır. Dağıtılacak rakam (71)/28*3=8 TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta=28 haftadır. Örneğin; faturalar şöyle hesaplanmıştır: Kodlama süreci için dağıtılacak rakam (700)/28*16=400 TL. Test süreci için dağıtılacak rakam (700)/28*3=75 TL. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 19 hafta ve bunun bir kısmı kodlama sürecinde diğer bir kısmı da test sürecinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam (539)/19*16=454 TL kodlama sürecine yazılır ve (539)/19*3=85 TL de test sürecine yazılır.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Hasan, Hüseyin, Ömer ve Ali tüm ay boyunca tek bir sürece çaba sarf etmişlerdir. Dolayısı ile her birinin maaşı tam olarak sürece aktarılmıştır. Testçi ise 3 hafta bu projede çalıştığından ¾ oranı

ile aktarım yapılmıştır. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımını da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 19 hafta ve bunun bir kısmı kodlama sürecinde diğer bir kısmı da test sürecinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(1500)/19*16=1263$ TL kodlama sürecine yazılır ve $1500/19*3=237$ TL test sürecine yazılır.

Haziran ayında yapılan işler incelendiğinde (Tablo 27) toplamda 0 TL araştırma, 19.308 TL geliştirme ve 12.507 TL Ar-Ge olmayan şeklinde sınıflandırılmıştır. Ar-Ge olmayan rakamın çok büyük kısmının başka projelerin araştırma veya geliştirme safhalarına dağıtılabileceğini unutmamak gerekir.

Tablo 27: Haziran Ayı Harcama Kalemleri ve Muhasebeleştirilmesi

Açıklama	Haziran '14				Faaliyet Türü	Yazılım Geliştirme Süreçleri				Satış Sonrası		Araştırma Harcamasının Muhasebe Kaydı		Araştırma Harcamasının Muhasebe Kaydı						
	21	22	23	24		Değerlendirme / Analiz	Tasarım	Kodlama	Test	Devreye Alınma	Bakım	Destek	Araştırma Harcaması	Geliştirme Harcaması	Ar-Ge Değeri	H. Kodu	Hesap Adı	Borç	Alacak	
Alımlar (Proje Yöneticisi)																750	Ar-Ge Gid.	0 TL	0 TL	
Mehmet (Analist)																750	268 B. Amortisman	0 TL	0 TL	
Hüseyin (Deneyimli Yazılımcı)																750	268 B. Amortisman	0 TL	0 TL	
Ömer (Başlangıç Seviyesi)																750	268 B. Amortisman	0 TL	0 TL	
Ali (Başlangıç Seviyesi)																750	268 B. Amortisman	0 TL	0 TL	
Veli (Testçi)																750	268 B. Amortisman	0 TL	0 TL	
Visual Studio 2013																750	Ar-Ge Gid.	230 TL	0 TL	
MS SQL Server Standard Edition 2012																750	Ar-Ge Gid.	603 TL	0 TL	
Crystal Report																750	Ar-Ge Gid.	94 TL	0 TL	
MS Team foundation server																750	Ar-Ge Gid.	568 TL	0 TL	
Windows 7																750	Ar-Ge Gid.	58 TL	0 TL	
MS Office																750	Ar-Ge Gid.	22 TL	0 TL	
Ana Bilgisayar (Server)																750	Ar-Ge Gid.	278 TL	0 TL	
Kullanıcı Bilgisayarları																750	Ar-Ge Gid.	583 TL	0 TL	
Veri tabanı sunucusu																750	Ar-Ge Gid.	92 TL	0 TL	
Web Sunucusu																750	Ar-Ge Gid.	71 TL	0 TL	
Yerel Ağ Altyapısı																750	Ar-Ge Gid.	21 TL	0 TL	
Geniş Ağ Altyapısı																750	Ar-Ge Gid.	7 TL	0 TL	
Faturalar																750	Ar-Ge Gid.	700 TL	0 TL	
Büro Giderleri (Yıyacak-İpecek)																750	Ar-Ge Gid.	1.000 TL	0 TL	
Kurtulaşye Giderleri																750	Ar-Ge Gid.	200 TL	0 TL	
																750	Ar-Ge Gid.	0 TL	0 TL	
24 ay vadeli 50.000 TL kredi alındı.																750	Ar-Ge Gid.	539 TL	0 TL	
Ahmet																750	Ar-Ge Gid.	6.000 TL	0 TL	
Mehmet																750	Ar-Ge Gid.	5.000 TL	0 TL	
Hüseyin																750	Ar-Ge Gid.	4.000 TL	0 TL	
Ömer																750	Ar-Ge Gid.	3.000 TL	0 TL	
Ali																750	Ar-Ge Gid.	2.500 TL	0 TL	
Veli																750	Ar-Ge Gid.	2.250 TL	0 TL	
Danışmanlık Hizmeti																750	Ar-Ge Gid.	1.500 TL	0 TL	
Toplam																				

Temmuz ayı iş planı incelendiğinde (Tablo 28) kodlama ve test safhalarına ait işlemler için çaba sarf edildiği görülmektedir. 4 kişi tamamen tasarım sürecinde çalışmıştır. 1 kişi de tamamen test sürecinde çalışmıştır. Boş karelerde başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların temmuz ayına ait payları kodlama ile alakalı olduğundan projede kullanılmaya başlamıştır dolayısı ile dağıtıma tabii olacaktır. Bu kaynaklar sadece “kodcu” tabir edilen yazılımcılar tarafından kullanılır, dolayısı ile toplam süre 4 kişi * 4 hafta=16 haftadır. Tüm yazılımcılar bütün ay boyunca kodlamada çalıştığından hesaplaması şu şekilde olacaktır. Örneğin; MS Team Foundation Server şöyle hesaplanmıştır: Dağıtılacak rakam (568)/16*16=568 TL.

Mavi ile işaretlenmemişler incelenirse, (Örneğin; Yerel Ağ Altyapısı) şu şekilde hesaplanmıştır: Toplam zaman, 7 kişi x 4 hafta = 28 hafta. Toplam zamandaki kodlama sürecinin payı ise; dağıtılacak rakam (21)/28*16=12 TL. Toplam zamandaki test sürecinin payı ise şu şekilde hesaplanır. Dağıtılacak rakam (21)/28*4=3 TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta=28 haftadır. Örneğin; kırtasiye giderleri şöyle hesaplanmıştır: Kodlama süreci için dağıtılacak rakam (200)/28*16=114 TL. Test süreci için dağıtılacak rakam (200)/28*4=29 TL. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 20 hafta ve bunun bir kısmı kodlama sürecinde diğer bir kısmı da test sürecinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam (514)/20*16=411 TL kodlama sürecine yazılır ve (514)/20*4=103 TL de test sürecine yazılır.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Hasan, Hüseyin, Ömer ve Ali tüm ay boyunca tek bir sürece (kodlama) çaba sarf etmişlerdir. Veli de tüm ay boyunca tek bir sürece (test) çaba sarf etmiştir. Dolayısı ile her birinin maaşı tam

olarak süreçlere aktarılmıştır. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımını da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 20 hafta ve bunun bir kısmı kodlama sürecinde diğer bir kısmı da test sürecinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(1500)/20*16=1200$ TL kodlama sürecine yazılır ve $1500/20*4=300$ TL de test sürecine yazılır.

Temmuz ayında yapılan işler incelendiğinde (Tablo 28) toplamda 0 TL araştırma, 19.950 TL geliştirme ve 11.840 TL Ar-Ge olmayan şeklinde sınıflandırılmıştır. Ar-Ge olmayan rakamın çok büyük kısmının başka projelerin araştırma veya geliştirme safhalarına dağıtılabileceğini unutmamak gerekir.

Tablo 28: Temmuz Ayı Harcama Kalemleri ve Muhasebeleştirilmesi

Açıklama	Temmuz '14				Faaliyet Sürücüsü	Yazılım Geliştirme Süreçleri				Satış Sonrası				Araştırma Harcaması	Geliştirme Harcaması	Ar-Ge Değili	Araştırma Harcamasının Muhasebe Kaydı	
	25	26	27	28		Yönetim / Analiz	Tasarım	Kodlama	Test	Devreye Alma	Bakım	Destek	H. Kodu				Hesap Adı	Borç
Yazılım Alt Yapısı																		
Ahmet (Proje Yöneticisi)																		
Mehmet (Analiz)																		
Hasan (Deneyimli Yazılımcı)																		
Hüseyin (Deneyimli Yazılımcı)																		
Ömer (Başlangıç Seviyesi)																		
Ali (Başlangıç Seviyesi)																		
Veli (Testçi)																		
Visual Studio 2013																		
MSSQL Server Standard Edition 2012																		
Crystal Report																		
MS Team foundation server																		
Windows 7																		
MS Office																		
Ana Bilgisayar (Server)																		
Kullanıcı Bilgisayarları																		
Veri tabanı sunucusu																		
Web Sunucusu																		
Yerel Ağ Altyapısı																		
Geniş Ağ Altyapısı																		
Faturalar																		
Büro Giderleri (Yiyecek-İpecek)																		
Kurtuluş Günüleri																		
Yönetime Giderleri																		
Faiz																		
24 ay vadeli 50.000 TL kredi alındı.																		
Ahmet																		
Mehmet																		
Hasan																		
Hüseyin																		
Ömer																		
Ali																		
Veli																		
Danışman Ücreti																		
Toplam																		

Not: Sadece kodlama aşamasında kullanılanlar.

Ağustos ayı iş planı incelendiğinde (Tablo 29) kodlama, test ve devreye alma safhalarına ait işlemler için çaba sarf edildiği görülmektedir. 1 kişi 3 hafta kadar devreye alma sürecinde, 4 kişi tamamen tasarım sürecinde ve 1 kişi de tamamen test sürecinde çalışmıştır. Boş karelerde başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların ağustos ayına ait payları kodlama ile alakalı olduğundan projede kullanılmaya başlamıştır dolayısı ile dağıtıma tabii olacaktır. Bu kaynaklar sadece “kodcu” tabir edilen yazılımcılar tarafından kullanılır, dolayısı ile toplam süre 4 kişi * 4 hafta = 16 haftadır. Tüm yazılımcılar bütün ay boyunca kodlamada çalıştığından hesaplaması şu şekilde olacaktır. Örneğin; veritabanı sunucusu şöyle hesaplanmıştır: Dağıtılacak rakam $(92)/16*16=92$ TL.

Mavi ile işaretlenmemişler incelenirse, (Örneğin; Genel Ağ Altyapısı) şu şekilde hesaplanmıştır: Toplam zaman, 7 kişi x 4 hafta = 28 hafta. Toplam zamandaki kodlama sürecinin payı ise; dağıtılacak rakam $(7)/28*16=4$ TL. Toplam zamandaki test sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam $(7)/28*4=1$ TL. Toplam zamandaki devreye alma sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam $(7)/28*3=1$ TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta = 28 haftadır. Örneğin; fatura giderleri şöyle hesaplanmıştır: Kodlama süreci için dağıtılacak rakam $(700)/28*16=400$ TL. Test süreci için dağıtılacak rakam $(700)/28*4=100$ TL. Devreye alma süreci için dağıtılacak rakam $(700)/28*3=75$ TL. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 23 haftadır ve bunun bir kısmı kodlama, test ve devreye alma süreçlerinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(488)/23*16=340$ TL kodlama sürecine gider, $(488)/23*4=85$ TL test sürecine yazılır ve $(488)/23*3=64$ TL de devreye alma sürecine yazılır.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Hasan, Hüseyin, Ömer ve Ali tüm ay boyunca tek bir sürece (kodlama) çaba sarf etmişlerdir. Veli, tüm ay boyunca tek bir sürece (test) çaba sarf etmiştir. Mehmet de 3 hafta kadar devreye alma sürecinde çaba sarf etmiştir. Dolayısı ile Mehmet hariç her birinin maaşı tam olarak süreçlere aktarılmıştır. Mehmet'in maaşının $\frac{3}{4}$ 'ü devreye alma sürecine aktarılmıştır. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımı da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 23 haftadır ve bunun bir kısmı kodlama, test ve devreye alma süreçlerinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(1500)/23*16=1.043$ TL kodlama sürecine yazılır, $1500/23*4=261$ TL test sürecine yazılır ve $1500/23*3=196$ TL devreye alma sürecine yazılır.

Ağustos ayında yapılan işler incelendiğinde (Tablo 29) toplamda 0 TL araştırma, 23.989 TL geliştirme ve 7.775 TL Ar-Ge olmayan şeklinde sınıflandırılmıştır. Ar-Ge olmayan rakamın çok büyük kısmının başka projelerin araştırma veya geliştirme safhalarına dağıtılabileceğini unutmamak gerekir.

Tablo 29: Ağustos Ayı Harcama Kalemleri ve Muhasebeleştirilmesi

Açıklama	Ağustos '14				Faaliyet Süreci	Yazılım Geliştirme Süreçleri				Satış Sonrası				Araştırma Harcaması	Geliştirme Harcaması	Ar-Ge Değili	Araştırma Harcamasının Muhasebe Kaydı		
	29	30	31	32		Yazılım Geliştirme / Analiz	Tasarım	Kodlama	Test	Devreye Alma	Bakım	Destek	H. Kodu				Hesap Adı	Borç	H. Kodu
Yazılım Alışverişi																			
Ahmet (Proje Yöneticisi)																			
Mehmet (Analist)																			
Hasan (Deneyimli Yazılımcı)																			
Hüseyin (Deneyimsiz Yazılımcı)																			
Ömer (Başlangıç Seviyesi)																			
Alli (Başlangıç Seviyesi)																			
Veil (Testçi)																			
Visual Studio 2013																			
MS SQL Server Standard Edition 2012																			
Crystal Report																			
MS Team foundation server																			
Windows 7																			
MS Office																			
Ana Bilgisayar (Server)																			
Kullanıcı Bilgisayarları																			
Veri tabanı sunucusu																			
Web Sunucusu																			
Yerel Ağ Altyapısı																			
Geniş Ağ Altyapısı																			
Faturalar																			
Büro Giderleri (Niycek-İpecek)																			
Kurtasiye Giderleri																			
24 ay vadeli 50.000 TL kredi alındı.																			
Ahmet																			
Mehmet																			
Hasan																			
Hüseyin																			
Ömer																			
Alli																			
Veil																			
Danışman Ücreti																			
Toplam																			

Not: Sadece kodlama aşamasında kullanılırlar.

Eylül ayı iş planı incelendiğinde (Tablo 30) kodlama, test ve devreye alma safhalarına ait işlemler için çaba sarf edildiği görülmektedir. Her çalışan çaba sarf ettiği süreçte 4 hafta çalışmıştır. 1 kişi devreye alma sürecinde, 4 kişi tasarım sürecinde ve 1 kişi de test sürecinde çalışmıştır. Boş karelerde başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların eylül ayına ait payları kodlama ile alakalı olduğundan projede kullanılmaya başlamıştır dolayısı ile dağıtımına tabii olacaktır. Bu kaynaklar sadece “kodcu” tabir edilen yazılımcılar tarafından kullanılır, dolayısı ile toplam süre 4 kişi * 4 hafta = 16 haftadır. Tüm yazılımcılar bütün ay boyunca kodlamada çalıştığından hesaplaması şu şekilde olacaktır. Örneğin; Visual Studio 2013 şöyle hesaplanmıştır: Dağıtılacak rakam (230)/16*16=230 TL.

Mavi ile işaretlenmemişler incelenirse, (Örneğin; Windows 7) şu şekilde hesaplanmıştır: Toplam zaman, 7 kişi x 4 hafta = 28 hafta. Toplam zamandaki kodlama sürecinin payı; dağıtılacak rakam (58)/28*16=33 TL. Toplam zamandaki test sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam (58)/28*4=8 TL. Toplam zamandaki devreye alma sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam (28)/28*4=8 TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta = 28 haftadır. Örneğin; büro giderleri şöyle hesaplanmıştır: Kodlama süreci için dağıtılacak rakam (1.000)/28*16=571 TL. Test süreci için dağıtılacak rakam (1.000)/28*4=143 TL. Devreye alma süreci için dağıtılacak rakam (1.000)/28*4=143 TL. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 24 haftadır ve bunun bir kısmı kodlama, test ve devreye alma süreçlerinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam (463)/24*16=308 TL kodlama sürecine yazılır, (463)/24*4=77 TL test sürecine yazılır ve (463)/24*4=77 TL de devreye alma sürecine yazılır.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Hasan, Hüseyin, Ömer ve Ali tüm ay boyunca tek bir sürece (kodlama) çaba sarf etmişlerdir. Veli, tüm ay boyunca tek bir sürece (test) çaba sarf etmiştir Mehmet de tüm ay boyunca tek bir sürece (devreye alma) çaba sarf etmiştir. Dolayısı ile her birinin maaşı tam olarak süreçlere aktarılmıştır. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımı da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 24 haftadır ve bunun bir kısmı kodlama, test ve devreye alma süreçlerinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(1500)/24*16=1.000$ TL kodlama sürecine yazılır, $1500/24*4=250$ TL test sürecine yazılır ve $1500/24*4=250$ TL devreye alma sürecine yazılır.

Eylül ayında yapılan işler incelendiğinde (Tablo 30) toplamda 0 TL araştırma, 25.319 TL geliştirme ve 6.420 TL Ar-Ge olmayan şeklinde sınıflandırılmıştır. Ar-Ge olmayan rakamın çok büyük kısmının başka projelerin araştırma veya geliştirme safhalarına dağıtılabileceğini unutmamak gerekir.

Ekim ayı iş planı incelendiğinde (Tablo 31) kodlama, test ve devreye alma safhalarına ait işlemler için çaba sarf edildiği görülmektedir. 1 kişi devreye alma sürecinde 3 hafta, 4 kişi tasarım sürecinde 1'er hafta ve 1 kişi de test sürecinde 3 hafta çalışmıştır. Boş karelerde başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların ekim ayına ait payları kodlama ile alakalı olduğundan projede kullanılmaya başlamıştır dolayısı ile dağıtım tabii olacaktır. Bu kaynaklar sadece "kodcu" tabir edilen yazılımcılar tarafından kullanılır, dolayısı ile toplam süre 4 kişi * 4 hafta = 16 haftadır. Dört yazılımcı 1'er hafta kodlamada çalıştığından hesaplaması şu şekilde olacaktır. Örneğin; MS SQL Server Ed. 2012 şöyle hesaplanmıştır: Dağıtılacak rakam (603)/16*4=151 TL.

Mavi ile işaretlenmemişler incelenirse, (Örneğin; MS Office) şu şekilde hesaplanmıştır: Toplam zaman, 7 kişi x 4 hafta = 28 hafta. Toplam zamandaki kodlama sürecinin payı; dağıtılacak rakam (22)/28*4=3 TL. Toplam zamandaki test sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam (22)/28*4=3 TL. Toplam zamandaki devreye alma sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam (22)/28*3=2 TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta = 28 haftadır. Örneğin; kırtasiye giderleri şöyle hesaplanmıştır: Kodlama süreci için dağıtılacak rakam (200)/28*4=29 TL. Test süreci için dağıtılacak rakam (200)/28*4=29 TL. Devreye alma süreci için dağıtılacak rakam (200)/28*3=21 TL. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 11 haftadır ve bunun bir kısmı kodlama, test ve devreye alma süreçlerinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam (436)/11*4=159 TL kodlama sürecine yazılır, (436)/11*4=159 TL test sürecine yazılır ve (436)/11*3=119 TL de devreye alma sürecine yazılır.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Hasan, Hüseyin, Ömer ve Ali tüm ay boyunca 1'er hafta kodlama sürecinde çaba sarf etmişlerdir. Veli, tüm ay boyunca tek bir sürece (test) çaba sarf etmiştir. Mehmet de tüm ay boyunca 3 hafta devreye alma sürecinde çaba sarf etmiştir. Dolayısı ile her birinin maaşı, bu oranda süreçlere aktarılmıştır. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımı da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 11 haftadır ve bunun bir kısmı kodlama, test ve devreye alma süreçlerinde geçmiştir. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(1500)/11*4=545$ TL kodlama sürecine yazılır, $1500/11*4=545$ TL test sürecine yazılır ve $1500/11*3=409$ TL devreye alma sürecine yazılır.

Ekim ayında yapılan işler incelendiğinde (Tablo 31) toplamda 0 TL araştırma, 12.488 TL geliştirme ve 19.225 TL Ar-Ge olmayan şeklinde sınıflandırılmıştır. Ar-Ge olmayan rakamın çok büyük kısmının başka projelerin araştırma veya geliştirme safhalarına dağıtılabileceğini unutmamak gerekir.

Kasım ayı iş planı incelendiğinde (Tablo 32) test ve devreye alma safhalarına ait işlemler için çaba sarf edildiği görülmektedir. 2 kişi devreye alma sürecinde 2'şer hafta ve 1 kişi test sürecinde 2 hafta çalışmıştır. Boş karelerde başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların kasım ayına ait payları kodlama ile alakalı iş yapılmadığından dağıtılmayacaktır.

Mavi ile işaretlenmemişler incelenirse, (Örneğin; ana bilgisayar) şu şekilde hesaplanmıştır: Toplam zaman, 7 kişi x 4 hafta = 28 hafta. Toplam zamandaki test sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam $(278)/28*2=20$ TL. Toplam zamandaki devreye alma sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam $(278)/28*4=40$ TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta = 28 haftadır. Örneğin; fatura giderleri şöyle hesaplanmıştır: Test süreci için dağıtılacak rakam $(700)/28*2=50$ TL. Devreye alma süreci için dağıtılacak rakam $(700)/28*4=100$ TL. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 6 haftadır. Bunun içinde test ve devreye alma süreçleri vardır. Dağıtım için kendi içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(410)/6*2=137$ TL test sürecine yazılır ve $(410)/6*4=273$ TL de devreye alma sürecine yazılır.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Hasan ve Ömer 2'şer hafta devreye alma sürecinde çaba sarf etmişlerdir. Veli de 2 hafta test sürecine çaba sarf etmiştir. Dolayısı ile her birinin maaşı, bu oranda süreçlere aktarılmıştır. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımı da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 6 haftadır. Bunun içinde test ve devreye alma süreçleri vardır. Dağıtım için kendi

içinde oranlama yapmak gereklidir. Dağıtılacak rakam $(1.500)/6*2=500$ TL test sürecine yazılır ve $1.500/6*4=1.000$ TL devreye alma sürecine yazılır.

Kasım ayında yapılan işler incelendiğinde (Tablo 32) toplamda 0 TL araştırma, 8.227 TL geliştirme ve 23.459 TL Ar-Ge olmayan şeklinde sınıflandırılmıştır. Ar-Ge olmayan rakamın çok büyük kısmının başka projelerin araştırma veya geliştirme safhalarına dağıtılabileceğini unutmamak gerekir.



Tablo 32: Kasım Ayı Harcama Kalemleri ve Muhasebeleştirilmesi

Açıklama	Kasım '14				Faaliyet Süreci	Yazılım Geliştirme Süreçleri				Satış Servisi		Araştırma Harcaması	Geliştirme Harcaması	Ar-Ge Değil	Araştırma Harcamasının Muhasebe Kaydı		
	41	42	43	44		Yazılım Geliştirme Süreçleri	Test	Deveye Alma	Bakım	Destek	H. Kodu				Hesap Adı	Borç	Alacak
Ahmet (Proje Yöneticisi)																	
Mehmet (Analist)																	
Hasan (Deneyimli Yazılımcı)																	
Hüseyin (Deneyimli Yazılımcı)																	
Ömer (Başlangıç Servisi)																	
Ali (Başlangıç Servisi)																	
Veli (Testçi)																	
Visual Studio 2013				230 TL													
MS SQL Server Standard Edition 2012				603 TL													
Crystal Report				94 TL													
MS Team foundation server				568 TL													
Windows 7				58 TL													
MS Office				22 TL													
Ana Bilgisayar (Server)				278 TL													
Kullanıcı Bilgisayarları				583 TL													
Veri tabanı sunucusu				92 TL													
Web Sunucusu				71 TL													
Yerel Ağ Altyapısı				21 TL													
Geniş Ağ Altyapısı				7 TL													
Faturalar				700 TL													
Büro Giderleri (Yıyacak-İpecek)				1.000 TL													
Kiraşive Giderleri				200 TL													
				0 TL													
24 ay vadeli 50.000 TL kredi alındı.				410 TL													
Ahmet				6.000 TL													
Mehmet				5.000 TL													
Hasan				4.000 TL													
Hüseyin				3.000 TL													
Ömer				2.500 TL													
Ali				2.500 TL													
Veli				2.250 TL													
Danışman Ücreti				1.500 TL													
Toplam				31.686 TL													

Not: Sadece kodlama aşamasında kullanılanlar.

Aralık ayı iş planı incelendiğinde (Tablo 33) sadece devreye alma safhalarına ait işlemler için çaba sarf edildiği görülmektedir. 2 kişi bu süreçte tüm ay boyunca çalışmıştır. Boş karelerde başka projelerde çalışabileceğini unutmamak gereklidir.

Yazılım ve donanım alt yapısına dair kaynakların meblağları aynı renklerde olanlar aynı prensiple dağıtılır. Mavi ile işaretlenmiş kaynakların aralık ayına ait payları kodlama ile alakalı iş yapılmadığından dağıtılmayacaktır.

Mavi ile işaretlenmemişler incelenirse, (Örneğin; kullanıcı bilgisayarları) şu şekilde hesaplanmıştır: Toplam zaman, 7 kişi x 4 hafta = 28 hafta. Toplam zamandaki devreye alma sürecinin payı şu şekilde hesaplanır. Dağıtılacak rakam $(583)/28*8=167$ TL. Dağıtılamayan pay Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Yönetim giderlerinin dağıtımı, süreçlerin toplam zaman içindeki payına göre hesaplanır. Toplam zaman, 7 kişi x 4 hafta = 28 haftadır. Örneğin; büro giderleri şöyle hesaplanmıştır: Devreye alma süreci için dağıtılacak rakam $(1.000)/28*8=286$ TL. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için kullanılmamıştır.

Kredinin tamamı bu proje için çekildiğinden faiz giderlerinin de dağıtımı geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 8 haftadır. Bunun içinde sadece devreye alma süreci vardır. Dağıtılacak rakamın tamamı bu sürece gidecektir. Dağıtılacak rakam $(383)/8*8=383$ TL.

Yazılım ekibinin ücretlerinin dağıtılması, çalışanın hangi faaliyette ne kadar süre geçirdiği ile alakalı olarak dağıtılmıştır. Örnekte görüldüğü üzere Hasan ve Ömer 4'er hafta devreye alma sürecinde çaba sarf etmişlerdir. Dolayısı ile her birinin maaşı, tam olarak bu sürece aktarılmıştır. Dağıtılamayan pay, Ar-Ge Değil kısmına yazılır, bu proje için çalışmamıştır.

Danışman ücreti tamamen bu proje için ödendiğinden bu harcamanın dağıtımı da geliştirme süreçleri arasında pay edilmelidir. Örneğin; geliştirme süreçlerinde geçen süre 8 haftadır. Bunun içinde sadece devreye alma süreci vardır. Dağıtılacak rakamın tamamı bu sürece yazılacaktır. $(1.500)/8*8=1.500$ TL.

Aralık ayında yapılan işler incelendiğinde (Tablo 33) toplamda 0 TL araştırma, 10.723 TL geliştirme ve 20.936 TL Ar-Ge olmayan şekilde sınıflandırılmıştır. Ar-Ge olmayan rakamın çok büyük kısmının başka projelerin araştırma veya geliştirme safhalarına dağıtılabileceğini unutmamak gerekir.

Tablo 33: Aralık Ayı Harcama Kalemleri ve Muhasebeleştirilmesi

Aylık Harcama Kalemleri	Aralık '14				Faaliyet Süreci / Açıklama	Yazılım Geliştirme Süreçleri				Satış Süremleri				Araştırma Harcaması		Geliştirme Harcaması		Araştırma Harcamasının Muhasebe Kaydı	
	45	46	47	48		İnşaat / Tasarım	Kodlama	Test	Devreye Alma	Bakım	Destek	Araştırma Harcaması	Geliştirme Harcaması	Araştırma Harcaması	Geliştirme Harcaması	H. Kodu	Hesap Adı	Borç	Alacak
Ahmet (Proje Yöneticisi)																			
Mehmet (Analist)																			
Hasan (Deneyimli Yazılımcı)																			
Hüseyin (Deneyimsiz Yazılımcı)																			
Ömer (Başlangıç Seviyesi)																			
Ali (Başlangıç Seviyesi)																			
Veli (Testçi)																			
Visual Studio 2013				230 TL															
MSSQL Server Standard Edition 2012				603 TL															
Crystal Report				94 TL															
MSSQL Team foundation server				568 TL															
Windows 7				58 TL															
MSS Office				22 TL															
Ana Bilgisayar (Server)				278 TL															
Kullanıcı Bilgisayarları				583 TL															
Veri tabanı sunucusu				92 TL															
Web sunucusu				71 TL															
Yerel Ağ Altyapısı				21 TL															
Geniş Ağ Altyapısı				7 TL															
Faturalar				700 TL															
Büro Giderleri (Yıvecek-İpecek)				1.000 TL															
Kırtasiye Giderleri				200 TL															
				0 TL															
Yatırım Giderleri																			
24 ay vadeli 50.000 TL kredi alındı.				383 TL															
Zarar																			
Ahmet				6.000 TL															
Mehmet				5.000 TL															
Hasan				4.000 TL															
Hüseyin				3.000 TL															
Ömer				2.500 TL															
Ali				2.500 TL															
Veli				2.250 TL															
Yazılım Ekibi																			
Danışman Ücreti				1.500 TL															
Denetimlik Hizmeti																			
Toplam				31.659 TL															

3.2.7. Geliştirme Sürecindeki İşlerin Dağılımı ve Aylık Harcamalar Özeti

Her yapılan harcamanın aylık bazda geliştirme süreçlerine dağıtılması ve muhasebeleştirilmesi yapılmıştır. Bu tip veri üretebilen bir proje yönetim aracı ile tablo 34'deki gibi geliştirme süreçlerine dayalı harcama özeti sunan bir tablo elde edilebilir. (Yılmaz ve Calayoglu, 2015b: 451)

Tablo 34’de her ay proje/projelere dağıtılacak gider kalemleri ve aylık toplamaları görülebilmektedir. Bunun altında, geliştirme süreçlerinde biriken tutarlar görülebilmektedir. Bunun da altında, geliştirme süreçlerinin araştırma, geliştirme veya ar-ge olmayan cinsinden biriken tutarlar görülebilmektedir. En sağda, her satırın toplamı hesaplanmıştır. 1 yıllık bu tabloda 381.607 TL’lik harcama dağıtılmış, bunun 53.197 TL+163.244 TL=216.441 TL’si örnekteki projeye ilgilidir ve geliştirme süreçlerine dağıtılmıştır. Kalan 165.165 TL’nin bir kısmı eş zamanlı yürütülen başka projeler var ise onların geliştirme süreçlerine dağıtılabileceği hatırlanmalıdır. Herhangi bir projeye dağıtılamayan giderler atıl kapasite olarak tanımlanmalı ve buna göre kayıtlara geçirilmelidir.

Muhasebe açısından yapılan hesaplamalar ve TMS 38’in hükümleri göz önüne alındığında ortaya çıkan sonuç şudur: Her ay yapılan araştırma harcamaları dönemin gider hesaplarına aktarılır, geliştirme harcamaları ise aktifleştirilerek varlık statüsünde kayıt edilir.

İşletme içi kullanılacak olan her projenin geliştirme harcamaları, 263 numaralı hesabın altında açılması gereken “Kullanım Amaçlı Geliştirilen Varlık” başlığının altında projenin adıyla açılacak bir alt hesapta takip edilmelidir. Her ayın sonundaki geliştirme harcamasının söz konusu hesapta biriktirilmesiyle beraber tablo 35’deki gibi bir görüntü oluşur.

Tablo 35: Aktifleştirilen Geliştirme Maliyetleri

Hesap Adı	Borç	Alacak
263 Araştırma ve Geliştirme Giderleri Hesabı	163.244	
263. 01 Kullanım Amaçlı Geliştirilen Varlıklar		
263. 01. 001 A projesi		
100/102/320 gibi hesaplar		163.244
(Geliştirme Harcamalarının Aktifleştirilmesi)		

Kullanım amaçlı geliştirilen yazılımların geliştirme süresi bir yıldan daha fazla olabilir. Bu durumda yukarıda detaylıca anlatılan işlemler, takip eden yıllar için de uygulanmalı ve 263 numaralı hesapta geliştirme harcamaları biriktirilmelidir. 263 numaralı hesabın ismine dair 3.1.2. Yararlı Ömür ve Amortisman bölümünde açıklamalar yapılmıştır.

Varlık geliştirilmesi bitip kullanıma başlandığı dönemden itibaren Aktifleştirme yapılmaz. İstisna olarak 1.2.7 başlığında belirtilen iyileştirici bakım sürecindeki

harcamalar geliřtirmeden sayılır ve varlıđın deđerinde artıř oluřturur. (Sardo ve Arcady, 1998: 50) (Luecke, Meeting ve Nardecchia, 1999: 30) Bu sebeple aktifleřtirmeye devam edilebilir.

Varlıđın iřletme iinde kullanımına bařlanmasıyla birlikte 263 numaralı hesapta biriken tutarlar mevcut hesap planına gre 260 Haklar Hesabına aktarılarak kaydedilmelidir. Ancak bu hesaba alternatif olarak sunulan neriler vardır, bunlar řu řekildedir: 265 Yazılımlar Hesabı (Elitař, Aydemir ve Demirel, 2007: 344) ve 263 Yazılım Programları Hesabı (Akdođan ve Sevilengl, 2007: 59) Anlařılan odur ki haklar hesabından satın alınan ve geliřtirilen yazılımların ıkartılması dřnlmektedir. Bilgisayar yazılımları, 260 Haklar Hesabının altında izleneceđine mstakil bir hesapta izlenmesi raporlama aısından daha uygun olur ve net bilgi sunar.

Yazılımların takip edildiđi hesaptan amortisman yntemleri aracılıđı ile giderleřtirme sađlanabilir. Ancak bu konuda da varlıđın yararlı mrnn sınırlı mı sınırsız mı olduđunun tespiti gerekmektedir.

3.2.8. Yararlı mr ve Amortisman

Yazılımlarda yararlı mr tespiti sonrasında amortisman ayırmak ancak mmkn olabilir. Yararlı mr konusu sınırlı veya sınırsız yararlı mr řeklinde tanımlanmaktadır. Kullanım amalı geliřtirilen yazılımların yararlı mrleri genelde sınırsızdır. nk yazılımdan greceđi fayda (kullanılabilir sre) tahmin edilemeyebilir. “2.1.4 Yazılımın Yararlı mr İle İlgili Hkmler ve Deđerlendirilmesi” kısmında detaylıca aıklamalar yapılmıřtır.

Kısaca, yararlı mrn belirlenemediđi durumlarda sınırsız mrl olarak addedilir, ve amortisman ayrılmaz. Ancak varlıđın faydasının bir sonu ngrlebildiđi andan itibaren amortisman ayrılabilir.

Yararlı srenin belirlenebilmesi ođu kez iřletmenin yazılımı kullanmayı beklediđi sre ile sınırlanabilir. Bu tahminin rasyonel temelleri olabilmesi iin ařađıdaki ifadelere cevap verilmesi gerekebilir.

- a) Yazılımın kullanımının diđer varlık veya iř sreleri ile iliřkili olup olmadıđı
- b) Teknik ve teknolojik aıdan deđiřimlere cevap verebilme durumu
- c) Sektr ihtiyalarının deđiřmesi

Yukarıdaki ve benzeri yollar ile yararlı sürenin belirlenebilmesi rasyonelleştirilemiyorsa sınırsız yararlı olarak tanımlanır ve belirliliğin olacağı döneme kadar amortisman ayrılmaz. Belirlilik olduğu andan itibaren amortisman ayırma işlemlerine başlanır. Yazılımın yararlı ömrünün belirli veya belirsiz olarak kabul edilmesi ve değişimleri, yönetim kurulu kararı ile kayıtlara geçirilmelidir. Bundan sonra gereken muhasebe işlemleri uygulanabilir.

Yararlı ömrü sınırlı olan kullanım amaçlı geliştirilen yazılımlara ömrünün bir bölümü tüketildikten sonra iyileştirici bakım uygulanarak varlığın ömrü uzatılabilir. Bu durumda varlığın net değeri üzerinden yeni yararlı ömür dikkate alınarak itfa payı hesaplanması uygundur. (Sonneltter ve Pacter, 1994: 49) Söz konusu durum, tablo 36'da örnek yardımıyla açıklanmıştır.

Örnek: 10.000 TL değerinde olan ve yararlı ömrü 5 yıl olarak tanımlanan bir yazılımın doğrusal yöntem ile amortismanı 3 yıl ayırdıktan sonra 3.000 TL iyileştirici bakım uygulanarak yararlı ömrüne 2 yıl daha eklenmiştir. Tüm yararlı ömür boyunca hesaplanan itfa payları aşağıdaki gibi olmalıdır.

Tablo 36: Değeri Artan Varlığın Yeni İtfa Payı Hesaplaması

Yararlı Ömürdeki Yıl	İtfa Payı Hesaplaması	İtfa Payı	Biriken İtfa Payı
1	$10.000 * 0,2 =$	2.000	2.000
2	$10.000 * 0,2 =$	2.000	4.000
3	$10.000 * 0,2 =$	2.000	6.000
<u>Net defter değeri:</u> Elde etme bedeli + değer arttırıcı bedeller – birikmiş itfa payları ($10.000 + 3.000 - 6.000 = 7.000$ TL)			
<u>Yeni Yararlı Ömür:</u> İlk yararlı ömür + eklenen ömür – tüketilen ömür ($5 + 2 - 3 = 4$ yıl)			
4	$7.000 * 0,25 =$	1.750	7.750
5	$7.000 * 0,25 =$	1.750	9.500
6	$7.000 * 0,25 =$	1.750	11.250
7	$7.000 * 0,25 =$	1.750	13.000

3.2.9. Yeniden Değerleme

Yeniden değerlendirme yapabilmek için ürünün aktif bir piyasasının olması gerekmektedir. Hâlâ alınıp satılması gerekmektedir. Kullanım amaçlı geliştirilen yazılımların aktif bir piyasasından bahsedilemez, ayrıca işletmeye özgü geliştirilen bir yazılımın emsalinden de bahsedilemez. Bu sebepler neticesinde yeniden değerlendirme yapma imkânı bulunmamaktadır. Yeniden değerlendirme yolu kapanmıştır.

Ürüne yeni özellikler katan, iyileştirici bakımdaki geliştirici işlem harcamaları, ürünün maliyetine eklenir. Bu konu “1.2.7. Bakım ve Destek” başlığı altında anlatılmıştır. Böylece yazılımın varlık değeri artabilir. Ancak bu artış ile yeniden değerlendirme karıştırılmamalıdır.

Kullanılmaya başlandıktan bir süre sonra yazılımın satışı yapılabilir veya yazılımın kullanılmasından vazgeçilebilir, böylece gelecek ekonomik fayda sağlama beklentisi ortadan kalkmış olur. Bu gibi durumlarda maddi duran varlıklarda olduğu gibi varlığı ve amortisman hesaplarını kapatarak bilanço dışı etmek gerekir. (Munter, 1997: 33)

3.3. Sipariş Üzerine Geliştirilen Yazılımların Muhasebeleştirilmesi

Bir müşterinin kendi özel ihtiyacı için bir yazılım firmasına, yazılım siparişi vermesiyle ortaya çıkan süreçtir. Geliştirici firma, yazılım geliştirme süreçlerinden geçerek siparişi teslim etmeye çalışır. Bu gelir türü, yazılım firmalarının oldukça fazla hasılat topladıkları bir sözleşme türüdür.

Literatürde sözleşme çerçevesinde satış amaçlı geliştirilen yazılımlar, IAS 11 ile muhasebeleşmesi gerektiği belirtilmiştir. (Suermann, 2006: 134) “Yazılımların Değerlemesi” başlıklı eserde TMS 11 ile değerlendirilmesi gerektiği açıkça ifade edilmektedir. (Sumer ve Erer, 2010: 29) Buna ilave olarak, Amerikan standartlarına göre raporlama yapanlar için birden fazla mali döneme sirayet eden projeleri ARB 45 ve SOP 81-1 standartları ile muhasebeleştirmek tavsiye edilmektedir. (Price of Water House Coopers, 2009: 159) Bu ifadelerde farklı raprolama veya muhasebeleştirme standartlarına atıf yapıldığı açıktır. Ancak, tüm bu standartların ortak özelliği inşaat sözleşmelerine yönelik oluşudur. Flanagan’ın yazılımların geliştirilme maliyetlerinin muhasebeleştirilmesine dair olan eserinde, başkası için geliştirilen yazılımlar için sözleşme standartlarının uygulanmasına yönlendirme yapılmıştır. (Flanagan, 2004: 4)

Bu noktada açıklanması gereken şudur: konumuz inşaat değildir, yazılımdır ancak konumuz kesinlikle sözleşmelerdir. Zaten, Uluslararası Muhasebe Standartları Kurulu (IASB) IFRS 15 - Müşterilerle Yapılan Sözleşmelerden Doğan Hasılat (Revenue from Contracts with Customers) standardını ilk bakış için 2008 yılının Aralık ayında çıkarmıştır. (Henry ve Holzmann, 2009: 77) En son versiyonu ise 2014 yılının Mayıs ayında çıkarmıştır. Bu standart, 2018 yılından itibaren zorunlu olacaktır. IAS 11 ve IAS 18 standartlarının yerine olacağı belirtilmiştir. (IFRS-15, 2015: md.6) Uzmanların görüşüne göre bu standarttan en çok etkilecek olan endüstrilerin başında uzun vadeli sözleşmeler yapan telekomünikasyon, yazılım geliştirme, gayrimenkul ve diğer sanayi firmaları gelmektedir. (Silvia, 2014: www.ifrsbox.com) Standardın amacı, sözleşme gelirlerinin tanımlanması, ölçülmesi ve açıklanmasıdır. Karşılaştırılabilirliği ve dönemselliği hedeflemektedir.

Sipariş yöntemiyle üretilen ve birden fazla hesap dönemine sarkan yazılım işlerinde mevcut durumda, TMS-11 İnşaat sözleşmeleri, TMS-18 Hasılat ve TMS-38 Maddi Olmayan Duran Varlıklar standartları esas yol gösterici olmaktadır. IFRS-15 yürürlüğe girmediğinden standardın hükümleri değerlendirilmeye alınmamıştır.

Hasılatın ölçümü çok önemli bir konudur. Çünkü hasılat kayıtları ile işletmenin dönemsel performansı ortaya çıkmakta ve kar-zarar durumu netleşmektedir. Bu sebeple gerçeğe uygun değer üreten (işin tamamlanan kısmı kadar hasılatın hesaplanıp paranın zaman değeri göz önünde bulundurularak kayıt edilmesi) bir muhasebe bilgi sistemine ihtiyaç vardır.

Bu bölümün ele alacağı konunun özeti şudur; siparişi yüklenen firmanın takip etmesi gereken harcama, maliyet ve gider kayıtları ile bunların Türkiye Muhasebe Standartları ilkelerine göre yapılması gereken kayıtlarına detaylı bir örnek ile açıklık getirebilmektir.

3.3.1. Hesaplama Metodu ve Kısıtlar

Sözleşme gelirin hesaplanması için TMS-11 İnşaat İşlemleri standardının tavsiye ettiği “Tamamlanma Oranı” yöntemidir. Tamamlanma oranı yöntemi ile tamamlanma yöntemi arasındaki farkı açıklayabilmek için uygulama örneği her iki yönteme göre hesaplanıp muhasebeleştirilmiştir.

Tamamlanma oranı yöntemi açıklanırken kullanılan örnekte tahsilatlarda vade farkına yer verilmemiştir. Tahsilatlar peşin olarak varsayılmıştır. Tamamlanma yönteminin anlatımı bittikten sonra çeşitli örnekler ile vade farkı durumunda yapılması gereken hesaplamalar ve muhasebe kayıtları açıklanmıştır. Bunun sebebi, her iki konunun (hasılat ve tamamlanma oranı) birbirinden farklı ancak iç içe olup, birbirini etkileyen olmasından dolayıdır. Böylece açıklamalar daha anlaşılabilir olmaktadır.

Örneklerde yapılan muhasebe kayıtlarında Katma Değer Vergisi dikkate alınmadığından hesaplanmamıştır.

3.3.2. Örnek Bir Uygulama ve Detayları

İster kendi kullanımı için olsun, isterse alınan bir siparişi teslim etmek için olsun yazılım geliştirme süreçleri aynıdır ve araştırma-geliştirme harcamalarının takip edilmesi gerekmektedir. Çünkü inşaat sözleşmeleri standardındaki hesaplama modellerine göre hep *sözleşmenin maliyeti* üzerinden hesaplamalar yapılmaktadır.

Örnek: Bir işletme, kendi iş süreçlerini ve kaynaklarını takip edebilmek için hazır bir paket program almayı, kendisi için yeni bir yazılım siparişi vermiştir. İnsan Kaynakları, Stok, Satın Alma, Satış Takibi, Finansal İşlemler, Muhasebe, Üretim, Raporlar gibi ana işlemlerin takip edildiği bir sistem istenmektedir. Bu kapsamda yazılım firması ile yapılan görüşmeler sonucunda projenin 2 yıl süreceği ve maliyetinin 230.000 TL olması tahmin edilmiştir. Satış sözleşmesine göre Sabit Fiyat tercihi ile 350.000 TL üzerinden anlaşma sağlanmıştır. (Yılmaz ve Calayoğlu, 2015b: 449-457) Tablo 37 sözleşme bilgilerini özetlemektedir.

Tablo 37: Tahmini Sözleşme Bilgileri

Toplam Satış Geliri	350.000 TL
Toplam Maliyet (araştırma + geliştirme)	230.000 TL
Toplam Kar	120.000 TL
Proje Tamamlanma Süresi	2 Yıl

1.Yıl: Yazılım firması 216.441,28 TL araştırma ve geliştirme harcaması yapmıştır. Dönem içinde 300.000 TL'lik hak ediş faturası kesilmiş ve 300.000 TL tahsilat yapılmıştır.

2.Yıl: Yazılım firması maliyetlerini revize ederek mevcuda ilave olarak 28.807,03TL'lik maliyetin daha olacağını tahmin etmiş ve gerekeni yaparak işi tamamlamıştır. Böylece başlangıçtaki tahmin edilen toplam maliyet 10.000 TL artmıştır. Dönem içinde 50.000 TL'lik hak ediş faturası kesilmiş ve 50.000 TL tahsilat yapılmıştır.

Birinci ve ikinci yılın işlem özeti tablo 38'de gösterilmiştir. Buna göre sözleşme yapılırken tahmin edilen maliyet güncellenerek ilave 10.000 TL olduğu anlaşılmaktadır.

Tablo 38: Dönemler İçindeki Nakit Akışı Özeti

	1.Yıl	2.Yıl	Toplam	Tahmin Edilen
Harcamalar Toplamı	216.441,28 TL	23.558,72 TL	240.000,00 TL	230.000 TL
Hakediş Tutarı	300.000,00 TL	50.000,00 TL	350.000,00 TL	
Tahsilat Tutarı	300.000,00 TL	50.000,00 TL	350.000,00 TL	

Çoğu işletme proje harcamalarını takip için aylık veya 3 aylık tablolar kullanır. Buradaki ana amaç tahmin edilen harcama ve tamamlanma düzeyinin gerçekleşen değerler ile devamlı karşılaştırarak projenin gidişatını kontrol altına almaktır. (Gentle, 2007: 87) Yazılım geliştirmeyle ilişkili olan kalemlere yapılan harcamaların hem aylık dönemler bazında muhasebeleştirilmesine imkân veren hem de araştırma-geliştirme ayrımını yapmaya yarayan tablo 39 ile ilk yılın harcamaları ortaya konmaya çalışılmıştır. Bu tablo, tablo 34 ile aynıdır. Oradaki geliştirme harcamaları kullanılarak bu konunun hesaplamaları ve açıklamaları yapılacaktır.

Tablo 39: Sözleşmeye Bağlı Yazılım Geliştirme Harcamalarının Özeti

Ayarlar	Yazılım Geliştirme Sürecindeki İşlerinin Dağılımı ve Aylık Harcamalar Özeti																																														
	Ocak '14		Şubat '14		Mart '14		Nisan '14		Mayıs '14		Haziran '14		Temmuz '14		Ağustos '14		Eylül '14		Ekim '14		Kasım '14		Aralık '14																								
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
Alınan (Proje Yöneticisi)																																															
Mehmet (Analist)																																															
Hüseyin (Deneyimli Yazılımcı)																																															
Ömer (Başlangıç Seviyesi)																																															
All (Başlangıç Seviyesi)																																															
Diğer																																															
Veil (Teşvik)																																															
Visual Studio 2013																																															
MS SQL Server Standard Edition 2012																																															
Crystal Report																																															
MS Team Foundation server																																															
Windows 7																																															
MS Office																																															
Ana Bilgisayar (Server)																																															
Kullanıcı Bilgisayarı																																															
Veri tabanı sunucusu																																															
Web Sunucusu																																															
Yerel Ağ Altyapısı																																															
Geniş Ağ Altyapısı																																															
Faturalar																																															
Büro Giderleri (Meycek-pçek)																																															
Kirazlık Giderleri																																															
Faiz 24 ay vadeli 50.000 TL kredi																																															
Alınan																																															
Mehmet																																															
Hüseyin																																															
Ömer																																															
All																																															
Veil																																															
Danışman Ücreti																																															
Aylık Giderlerin Toplamı	31.936 TL	31.912 TL				31.889 TL		31.864 TL		31.840 TL		31.815 TL		31.790 TL		31.764 TL		31.739 TL		31.712 TL		31.686 TL		31.659 TL		31.631 TL		31.604 TL		31.577 TL		31.550 TL		31.523 TL		31.496 TL		31.469 TL		31.442 TL		31.415 TL		31.388 TL			
İstisna Belirlemesi - Analiz																																															
Tasarım																																															
Kodlama																																															
Test																																															
Devreye Alma																																															
Toplam	13.999,92	13.976,37				13.952,82		13.929,27		13.905,72		13.882,17		13.858,62		13.835,07		13.811,52		13.787,97		13.764,42		13.740,87		13.717,32		13.693,77		13.670,22		13.646,67		13.623,12		13.599,57		13.576,02		13.552,47		13.528,92					
Arayışma Harcamaları																																															
Geliştirme Harcamaları																																															
Arayışma veya Geliştirme olmayan																																															
Toplam	31.935,97	31.911,42				31.886,87		31.862,32		31.837,77		31.813,22		31.788,67		31.764,12		31.739,57		31.715,02		31.690,47		31.665,92		31.641,37		31.616,82		31.592,27		31.567,72		31.543,17		31.518,62		31.494,07		31.469,52							

Tablo 39 incelendiğinde her ay yapılacak işlerin planı, bu işlerin kimler tarafından yapıldığı ve ne kadar sürdüğü raporlanabilmiştir. Renkler, yapılan işin hangi yazılım geliştirme sürecine dâhil olduğunu sembolize etmektedir. Üretim kaynaklarının hangi geliştirme safhasında (tasarım, kodlama, test ve devreye alma) kullanılmış ise kullanım zamanı esasına göre oranlanarak aylık döneme indirgenmiştir. Tablonun sol tarafında dikey biçimde yazılmış olanlar sözleşme maliyetlerinin gruplarıdır. Yazılım alt yapısı gibi. Yazılım ve donanım alt yapısına dair olan değerler, bu varlıkların ilgili aya düşen amortisman paylarıdır. Böylelikle her ay için her süreçte, ne kadarlık bir harcama yapıldığının anlaşılması mümkündür.

En alttaki kısımda ise yazılım geliştirme süreçleri incelenerek İsterlerin Belirlenmesi – Analiz için “Araştırma”, diğer süreçlerin toplamları da “Geliştirme” harcamaları olarak ay bazında özetlenmektedir.

Tablo 39’da “Araştırma veya Geliştirme olmayan” adı ile ifade edilen kalem aslında aynı süreçte devam eden diğer projelerin araştırma veya geliştirme harcamalarını ifade eder. Örnekteki projenin harcamalarından değildir. Gerçek hayatta aynı anda birden çok proje ile uğraşıldığından kaynaklar sürekli olarak A, B, C, vb. projelere katkı yapmaktadır. Böylece söz konusu harcamalar diğer projelerin geliştirme harcamalarını oluşturur. İyi bir proje yönetimi atıl kapasiteyi en aza indirendir.

İlk yılın harcamalar toplamı 381.607 TL’dir. Bu tutar, hem aylık giderler toplamı satırında, hem de araştırma-geliştirme harcamalarının toplamında gözükmektedir. Ancak yazılım geliştirme süreçlerine göre raporlanan tutarların toplamı 216.441 TL’dir. Bunun sebebi, eş zamanlı yürütülen diğer projelerin harcamalarının örnekteki projenin harcamalarından (Araştırma veya geliştirme olmayan) ayrı hesaplanması gerektiğindedir.

İkinci yılın ilk iki ayında devreye alma safhasında 23.559 TL’lik geliştirme harcaması daha yapılmış ve sipariş tamamlanarak teslim edilmiştir. Bu kısım için bir tablo daha gösterilmemiştir.

3.3.3. Tamamlanma Yöntemine Göre Hesaplama ve Raporlamalar

Tablo 40 ve 41’deki muhasebe kayıtları, tamamlanma yöntemine göredir. Ülkemizde VUK’a göre rapor üreten firmaların uygulamayı tercih edebildikleri yöntemdir.

Birinci yılda proje tamamlanmadığı için gelir ve gider raporlanmamış olup, vergi de ödenmemiştir. Harcamalar ve tahsilatlar 170 ve 350 numaralı hesaplarda toplanarak bekletilmiştir. İkinci yılda ise projenin tamamlanmasıyla birlikte bu hesaplardaki bakiyeler, gelir-gider kayıtlarına dönüşmüş ve vergi hesaplanabilmiştir.

Tablo 40: Tek Düzen Hesap Planına Göre Birinci Yılın Muhasebe Kayıtları

Hesap Adı	Borç	Alacak
170 Yıllara Yaygın İnşaat ve Onarım Maliyeti Hesabı	216.441	
170.01 Devam Eden Yazılım Sözleşmesi Maliyetleri		
100/102/320 gibi hesaplar		216.441
(Geliştirme maliyetleri için yapılan harcamalar toplamı)		
100/102/120 gibi hesaplar	300.000	
350 Yıllara Yaygın İnşaat ve Onarım Hakediş Bedelleri Hesabı		300.000
350.01 Devam eden Yazılım Sözleşmesi Hakediş Bedelleri		
(Müşteriye kesilen hakediş faturaları toplamı)		

Tablo 41: Tek Düzen Hesap Planına Göre İkinci Yılın Muhasebe Kayıtları

Hesap Adı	Borç	Alacak
170 Yıllara Yaygın İnşaat ve Onarım Maliyeti Hesabı	23.559	
170.01 Devam Eden Yazılım Sözleşmesi Maliyetleri		
100/102/320 gibi hesaplar		23.559
(Geliştirme maliyetleri için yapılan harcamalar toplamı)		
100/102/120 gibi hesaplar	50.000	
350 Yıllara Yaygın İnşaat ve Onarım Hakediş Bedelleri Hesabı		50.000
350.01 Devam eden Yazılım Sözleşmesi Hakediş Bedelleri		
(Müşteriye kesilen hakediş faturaları toplamı)		
740 Hizmet Üretim Maliyetleri Hesabı	240.000	
740.01 Yazılım Üretim Maliyeti		
170 Yıllara Yaygın İnşaat ve Onarım Maliyeti Hesabı		240.000
170.01 Devam Eden Yazılım Sözleşmesi Maliyetleri		
(Dönem sonu maliyet kaydı)		
741 Hizmet Üretim Maliyeti Yansıtma Hesabı	240.000	
741.01 Yazılım Üretim Maliyeti Y. H.		
740 Hizmet Üretim Maliyetleri Hesabı		240.000
740.01 Yazılım Üretim Maliyeti		
(Dönem sonu yansıtma yapılışı)		
622 Hizmet Üretim Maliyeti Hesabı	240.000	
622.01 Yazılım Üretim Maliyeti		
741 Hizmet Üretim Maliyeti Yansıtma Hesabı		240.000
741.01 Yazılım Üretim Maliyeti Y. H.		
(Yazılım maliyetinin gider hesaplarına transferi)		

350 Yıllara Yaygın İnşaat ve Onarım Hakediş Bedelleri Hesabı	350.000	
350.01 Devam eden Yazılım Sözleşmesi Hakediş Bedelleri		
600 Yurt İçi Satışlar Hesabı		350.000
600.01 Yazılım Tamamlama Gelirleri		
(Hasılat kaydı)		
691 Dönem Karı Vergi ve Diğer Yükümlülükler Karşılıkları Hesabı	22.000	
370 Dönem Karı Vergi ve Diğer Yükümlülükler Karşılıkları Hesabı		22.000
(Vergi Karşılığının Kaydı)		

Yukarıdaki işlemlerin sonuçları tablo 42’de özetlenmektedir. Dönemsellik ilkesine ters düşecek şekilde raporlama yapılmış olmaktadır. Birinci yıl hiç gelir ve gider hesapları işlem görmemiştir.

Tablo 42: Tek Düzen Hesap Planına Göre Projenin Kar-Zarar Özeti

	T1	T2
Toplam Gelir	0	350.000 TL
Toplam Gider	0	240.000 TL
Kar/Zarar	0	110.000 TL
Vergi %20	0	22.000 TL
Net Kar	0	88.000 TL

3.3.4. Tamamlanma Oranı Yöntemine Göre Hesaplama ve Raporlamalar

Ülkemizde TFRS’ye göre rapor üreten firmaların uygulamaları gereken kayıt biçimidir. Bunun için önce tamamlanma yüzdesine göre hesaplamaların yapılması gerekmektedir. Bu şu şekilde hesaplanır: Fiili gerçekleşen maliyet değeri ile tahmin edilen değere olan oranı sayesinde tamamlanma oranı bulunur. $(216.441/230.000=0,941)$. Bulunan bu oran ile sözleşme geliri ile oranlama yapılarak dönemin tamamlanan iş miktarına isabet eden gelir hesaplanır. $(350.000*0,941=329.350)$ Brüt taahhüt karı, satış gelirinden satışların maliyeti çıkarılarak bulunur. $(329.350-216.441=112.909)$

İkinci yılın sonuçları hesaplanırken ilave 10.000 TL’lik maliyet artışını kümülatif olarak düşünmek gerekmektedir. O güne kadar katlanılan fiili maliyetler (önceki dönemlere ait) ile işin tamamlanması için katlanılan fiili maliyetler (içinde bulunulan döneme ait) toplamını $(216.441+23.559=240.000)$, o güne kadar tahmin edilen maliyetler (önceki dönemlere ait) ve işin tamamlanması için gereken tahmini maliyetler toplamına

(230.000+10.000=240.000) bölünmesi ile işin tamamlanma oranı bulunur. (240.000/240.000=%100) Demek ki 2. Yıl yapılan geliştirme harcamaları, toplamın (%100-%94,1=)%5,9'dur.

Birinci ve ikinci yılın harcamalarına bağlı olarak bulunan tamamlanma oranı çerçevesinde döneme isabet eden gelir, kar ve vergi rakamları tablo 43'te özetlenmiştir.

Tablo 43: Projenin Tamamlanma Aşamalarına Göre Kar-Zarar Özeti

	T1	T2	Toplam
Satış Geliri	329.350 TL	20.650 TL	350.000 TL
Satışların Maliyeti	-216.441 TL	-23.559 TL	-240.000 TL
Brüt Taahhüt Karı	112.909 TL	-2.909 TL	110.000 TL
Tamamlanma Oranı	94,10%	5,90%	100,00%

Sabit fiyat üzerinden sözleşme yapıldığından geliştirme maliyetlerindeki artıştan satış fiyatı etkilenmemektedir. Maliyet artı kar yöntemine göre sözleşme yapılmış olsaydı geliştirme maliyetlerindeki artış sebebiyle satış fiyatı da etkilenecekti.

Birinci yılın muhasebe kayıtları tablo 44'te gösterilmiştir. Birinci yılda harcanan 216.441 TL'lik miktarın araştırma ve geliştirme cinsinden sınıflandırılmasıyla birlikte ilk yılın harcamalarının büyük bir kısmı (163.244 TL) aktifleştirilmiştir. Diğer kısmı (53.197 TL) ise giderleştirilmiştir. Aktifleştirilen kısım satış amaçlı olduğu için amortismanına tabi tutulmamaktadır.

Hasılat, mal veya hizmetin alıcıya teslim edilmesiyle doğar, tahsilatın gerçekleşme zamanına ve hasılatın doğduğu zamana göre muhasebe kayıtlarında paranın zaman değerinin hesaplanarak ertelenen faiz geliri veya gideri kalemlerine ayrıştırılarak kaydedilmesini gerektirir. Örnekte kesilen fatura tutarı kadar (300.000 TL) tahsilat peşin olarak (300.000 TL) alındığından vade farkı doğmamıştır. Hasılat tutarı, hesaplanan değerden (329.350 TL) daha küçük olduğundan gerçekleşen hasılat değeri gelir olarak kaydedilmiştir. Gelir tutarı hesaplanan değerden daha fazla olsaydı, bu durumda sadece hesaplanan kadarı gelir yazılırdı. Fazlası bilanço hesaplarında tutulması gerekir. Hesaplanan gelir ile tahsil edilen gelir arasında fark olması da yine kayıtları değiştirmektedir.

Tablo 44: TFRS'ye Göre Birinci Yılın Muhasebe Kayıtları

Hesap Adı	Borç	Alacak
263 Araştırma ve Geliştirme Giderleri Hesabı	163.244	
263.01 Satış Amaçlı Geliştirilen A Projesi		
750 Araştırma ve Geliştirme Giderleri Hesabı	53.197	
750.01 Satış Amaçlı Geliştirilen A Projesi		
100/102/320 gibi hesaplar		216.441
(Araştırma ve Geliştirme harcamalarının ayrıştırılması)		
100/102/120 gibi hesaplar	300.000	
600 Yurt İçi Satışlar Hesabı		300.000
600.01 Yazılım Tamamlama Gelirleri		
(Müşteriden tahsilatlar toplamı)		
751 Araştırma ve Geliştirme Giderleri Yansıtma Hesabı	53.197	
751.01 Satış Amaçlı Geliştirilen A Projesi		
750 Araştırma ve Geliştirme Giderleri Hesabı		53.197
750.01 Satış Amaçlı Geliştirilen A Projesi		
(Dönem sonu yansıtımaların yapılışı)		
630 Araştırma ve Geliştirme Giderleri Hesabı	53.197	
751 Araştırma ve Geliştirme Giderleri Yansıtma Hesabı		53.197
(Araştırma harcamalarının giderleştirilmesi)		
691 Dönem Karı Vergi ve Diğer Yükümlülükler Karşılıkları Hesabı	49.361	
484 Ertelenmiş Vergi Yükümlülüğü Hesabı		49.361
(Ertelenmiş vergi etkisinin kaydı)		

İkinci yılın muhasebe kayıtları tablo 45'de gösterilmiştir. Harcamaların tamamı (23.559 TL), geliştirme safhasına ait olduğundan tamamı aktifleştirilmiştir. Aktifleştirilen miktarın tamamı proje sonunda 750 hesabına aktarılarak giderleştirilmiştir.

Gelir tutarı, projenin tamamlanması sebebiyle kalan fark olan 50.000 TL tahsil edildiğinden gelir olarak yazılmıştır. Ancak proje devam etseydi ve hesaplanan değerden (20.650 TL) daha fazla tahsilat söz konusu olduğunda, aradaki fark bilanço hesaplarında bekletilirdi. Daha sonraki dönemlerde gelir hesabına aktarılarak mahsuplaşma sağlanmış olurdu. Amaç dönemsellik ile fiiliyatın çelişmemesidir.

Tablo 45: TFRS'ye Göre İkinci Yılın Muhasebe Kayıtları

Hesap Adı	Borç	Alacak
263 Araştırma ve Geliştirme Giderleri Hesabı	23.559	
263. 01 Satış Amaçlı Geliştirilen A Projesi		
100/102/320 gibi hesaplar		23.559
(Araştırma ve Geliştirme harcamalarının ayrıştırılması)		
100/102/120 gibi hesaplar	50.000	
600 Yurt İçi Satışlar Hesabı		50.000
600.01 Yazılım Tamamlama Gelirleri		
(Müşteriden tahsilatlar toplamı)		
750 Araştırma ve Geliştirme Giderleri Hesabı	186.803	
750.01 Satış Amaçlı Geliştirilen A Projesi		
263 Araştırma ve Geliştirme Giderleri Hesabı		186.803
263. 01 Satış Amaçlı Geliştirilen A Projesi		
(Varlığın Giderleştirilmesi)		
751 Araştırma ve Geliştirme Giderleri Yansıtma Hesabı	186.803	
751.01 Satış Amaçlı Geliştirilen A Projesi		
750 Araştırma ve Geliştirme Giderleri Hesabı		186.803
750.01 Satış Amaçlı Geliştirilen A Projesi		
(Dönem sonu yansıtımaların yapılışı)		
630 Araştırma ve Geliştirme Giderleri Hesabı	186.803	
751 Araştırma ve Geliştirme Giderleri Yansıtma Hesabı		186.803
(Araştırma harcamalarının giderleştirilmesi)		
484 Ertelenmiş Vergi Yükümlülüğü Hesabı	49.361	
284 Ertelenmiş Vergi Varlığı Hesabı		27.361
691 Dönem Karı Vergi ve Diğer Yükümlülükler Karşılıkları Hesabı		22.000
(Ertelenmiş vergi hesaplarının birbirileriyle mahsubu)		
691 Dönem Karı Vergi ve Diğer Yükümlülükler Karşılıkları Hesabı	22.000	
484 Ertelenmiş Vergi Yükümlülüğü Hesabı		22.000
(Ertelenmiş vergi etkisinin kaydı)		

TFRS'ye göre raporlama yapmadan evvel önce tamamlanma yüzdesi yöntemine göre hesaplama yapılmalı, böylece hesaplanan tamamlanma gideri ve geliri bulunur. Gelirin tahsilatına dair miktar ve vade gibi konular satış kayıtlarını etkileyeceğinden bu açıdan da bakıldıktan sonra raporlama yapılmalıdır. Böylece dönemsellik ve gerçeğe uygunluk ilkeleri en iyi şekilde uygulanmış olur. Farklılaşan sonuçlar, daha iyi anlatılabilmek için tablo 46'da gösterilmiştir.

Tablo 46’da görüldüğü üzere hesaplanan ile rapor edilen rakamların vardığı sonuç aynıdır. Değişen ise dönemsel olarak elde edilmeyen gelir ve harcamaların bir kısmının aktifleştirilmesinden doğan karlılık ve vergi farkıdır.

Tablo 46: Hesaplanan Değerler ile Kayıtlarda Kullanılan Değerler Farkı

	T1		T2		Toplam
	Hesaplanan	Rapor Edilen	Hesaplanan	Rapor Edilen	
Satış Geliri	329.350 TL	300.000 TL	20.650 TL	50.000 TL	350.000 TL
Satışların Maliyeti	-216.441 TL	-53.197 TL	-23.559 TL	-186.803 TL	-240.000 TL
Brüt Taahhüt Karı	112.909 TL	246.803 TL	-2.909 TL	-136.803 TL	110.000 TL
Tamamlanma Oranı	94,10%		5,9%		100,00%
Ert. Vergi Yük/Var.	22.582 TL	49.361 TL	-582 TL	-27.361 TL	22.000 TL
Vergi Sonrası Net Kar	90.327 TL	197.442 TL	-2.327 TL	-109.442 TL	88.000 TL

VUK’a göre yapılan kayıtlar da aynı sonuçlara vardır, ancak ilk yıl hiç gelir ve gider raporlanmamıştır.

3.3.5. Vade Farkı Olması Durumunda Yapılması Gereken Kayıtlar

Örnekte kesilebilecek fatura tutarı 329.350 TL iken kesilen fatura tutarı, peşin olarak tahsil edilebilecek tutar olan 300.000 TL’dir. Bu sebeple herhangi bir vade farkı oluşmamış ve bunun için bir hesaplama yapılmamıştır. Ancak vadeli bir ödeme söz konusu olduğunda hesaplama metodu ve muhasebe kayıtları farklılaşacaktır.

Vade farkı olduğunda hasılatın hesaplanmasında belirlenen faiz oranı üzerinden bugünkü değer veya gelecek değer formülleriyle hesaplama yapıp, çıkan sonuçlar fatura üzerinde vade farkı hanesinde belirtilmelidir. (Güngörmüş ve Boyar, 2009: 179) Tanımlar ve formüller aşağıdaki gibidir:

Gelecek değer, “bugünkü bir paranın belirli bir faiz oranı üzerinden, belirli bir süre sonra ulaşacağı değeri ifade eder.” (Usta, 2008: 6)

$$\text{Gelecek Değer} = \text{Bugünkü Değer} \times (1+f)^n$$

Bugünkü değer, gelecekte gerçekleşmesi beklenen bir nakit akımının belirli bir faiz oranı ile bugüne indirgenmesidir. (Usta, 2008: 9)

$$\text{Bugünkü Değer} = \text{Gelecek Değer} / (1+f)^n$$

Vade farkını doğuran çeşitli örnekler ile TMS-18 Hasılat standardının tavsiye ettiği şekilde muhasebeleştirmeler yapılacaktır.

Örnek 7: Fatura tutarı 300.000 TL olan, 3 ay vadeli tahsilat için aylık % 2 vade farkı uygulanacağı konusunda işletmeler anlaşmıştır.

Bugünkü değer üzerinden kesilen tutarın gelecekte ödenmesi icap ettiğinden gelecekte elde edilecek 300.000 TL'nin içinde hem hasılat hem de paranın zaman değeri olacaktır. Bu sebeple fatura tutarının tamamı satış hasılatı olarak muhasebeleştirilmemelidir. Zamana bağlı bir gelir ortaya çıktığından vadeler geldikçe hesaplanan kazanılmamış faizler, satış geliri olarak değil faiz geliri olarak kaydedilmelidir. Faiz geliri ve satış gelirini birbirinden ayırabilmek için tablo 47'de gerekli hesaplamalar yapılmıştır.

Tablo 47: Örnek 7'nin Hesaplanması

Vade	Gelecek Değer	Aylık Faiz	Kazanılmamış Faiz
1. Ay	$300.000/(1+0,02)=294.117,65$	5.882,35 TL	5.882,35 TL
2. Ay	$300.000/(1+0,02)^2=288.350,63$	5.761,01 TL	11.649,37 TL
3. Ay	$300.000/(1+0,02)^3=282.696,70$	5.653,93 TL	17.303,30 TL
	Toplam	17.303,30 TL	

Bu hesaplamalara göre fatura tarihindeki ve sonraki vadelerde yapılması gereken muhasebe kayıtları tablo 48'de gösterilmiştir.

Tablo 48: Örnek 7'nin Muhasebe Kaydı

Hesap Adı	Borç	Alacak
120 Alıcılar Hesabı	300.000,00	
600 Yurt İçi Satışlar Hesabı		282.696,70
600.01 Yazılım Tamamlama Gelirleri		
382 Ertelenmiş Gelirler Hesabı		17.303,30
(Fatura tarihindeki kayıt)		
382 Ertelenmiş Gelirler Hesabı	5.882,35	
642 Faiz Gelirleri Hesabı		5.882,35
(1. Aydaki kayıt)		
382 Ertelenmiş Gelirler Hesabı	5.761,01	
642 Faiz Gelirleri Hesabı		5.761,01
(2. Aydaki kayıt)		
382 Ertelenmiş Gelirler Hesabı	5.653,93	
642 Faiz Gelirleri Hesabı		5.653,93
(3. Aydaki kayıt)		
100 Kasa Hesabı	300.000	
120 Alıcılar Hesabı		300.000
(Tahsilat kaydı)		

Örnek 8: Tamamlanan işin oranına göre en fazla kabul edilebilir dönem geliri 329.350 TL'dir. Bu tutarda fatura kesilmiştir. Faturanın yarısı peşin tahsil edilmiştir. Kalanı, 3 ay sonra aylık % 2 vade farkı uygulanarak tahsil edilecektir.

Vade farkının isabet edeceği matrah fatura tutarının ödenmeyen kısmı olduğundan paranın zaman değeri buna göre hesaplanmalıdır. Faiz geliri ve satış gelirini birbirinden ayırabilmek için tablo 49'da gerekli hesaplamalar yapılmıştır.

Tablo 49: Örnek 8'in Hesaplanması

Vade	Gelecek Değer	Aylık Faiz	Kazanılmamış Faiz
1. Ay	$164.675/(1+0,02)=161.446,08$	3.228,92 TL	3.228,92 TL
2. Ay	$164.675/(1+0,02)^2=158.280,47$	3.165,61 TL	6.394,53 TL
3. Ay	$164.675/(1+0,02)^3=155.176,93$	3.103,54 TL	9.498,07 TL
	Toplam	9.498,07 TL	

Bu hesaplamalara göre fatura tarihindeki ve sonraki vadelerde yapılması gereken muhasebe kayıtları tablo 50'de gösterilmiştir.

Tablo 50: Örnek 8'in Muhasebe Kaydı

Hesap Adı	Borç	Alacak
100 Kasa Hesabı	164.675,00	
120 Alıcılar Hesabı	164.675,00	
600 Yurt İçi Satışlar Hesabı		319.851,93
600.01 Yazılım Tamamlama Gelirleri		
382 Ertelemiş Gelirler Hesabı		9.498,07
(Fatura tarihindeki kayıt)		
382 Ertelemiş Gelirler Hesabı	3.228,92	
642 Faiz Gelirleri Hesabı		3.228,92
(1. Aydaki kayıt)		
382 Ertelemiş Gelirler Hesabı	3.165,61	
642 Faiz Gelirleri Hesabı		3.165,61
(2. Aydaki kayıt)		
382 Ertelemiş Gelirler Hesabı	3.103,54	
642 Faiz Gelirleri Hesabı		3.103,54
(3. Aydaki kayıt)		
100 Kasa Hesabı	164.675,00	
120 Alıcılar Hesabı		164.675,00
(Tahsilat kaydı)		

Görüldüğü üzere vadeli tahsilatlarda fatura tutarının tamamı satış geliri olarak kayıt edilememektedir. Bu sebeple satış değerleri düşmektedir. Bunun alternatifi olarak vadeli tahsilatlarda satış hasılatının düşürülmemesi için bugün kesilecek fatura tutarına paranın zaman değeri de eklenebilir.

Örnek 9: İşin tamamlanma oranına göre satış hasılatı 329.350 TL'dir. Bedel, 3 ay sonra aylık % 2 vade farkı uygulanarak tahsil edilecektir. Bugün ne kadarlık fatura kesilmelidir?

Gelecekteki değer bilindiğine göre paranın zaman değeri dikkate alınarak yapılan genişletmeyle faiz geliri + bugünkü satış geliri tutarına erişilebilir. Faiz geliri ve satış gelirini birbirinden ayırabilmek için tablo 51'de gerekli hesaplamalar yapılmıştır.

Tablo 51: Örnek 9'un Hesaplaması

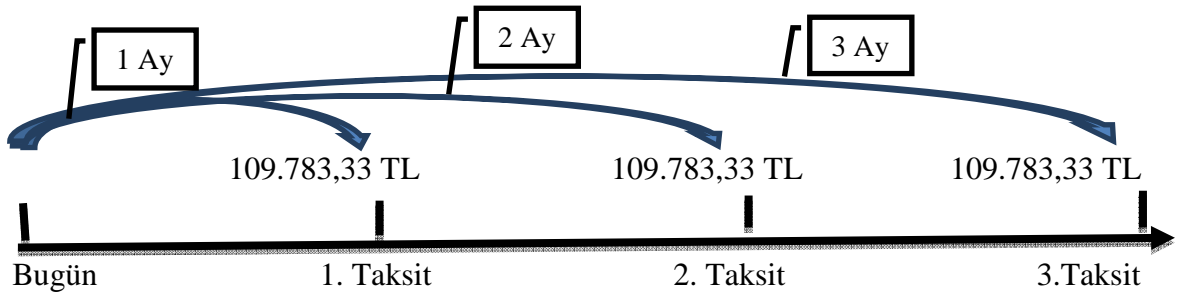
Vade	Bugünkü Değer	Aylık Faiz	Kazanılmamış Faiz
1. Ay	$329.350 \cdot (1+0,02) = 335.937,00$	6.587,00 TL	6.587,00 TL
2. Ay	$329.350 \cdot (1+0,02)^2 = 342.655,74$	6.718,74 TL	13.305,74 TL
3. Ay	$329.350 \cdot (1+0,02)^3 = 349.508,85$	6.853,11 TL	20.158,85 TL
	Toplam	20.158,85 TL	

Bu hesaplamalara göre fatura tarihindeki ve sonraki vadelerde yapılması gereken muhasebe kayıtları tablo 52'de gösterilmiştir.

Tablo 52: Örnek 9'un Muhasebe Kaydı

Hesap Adı	Borç	Alacak
120 Alıcılar Hesabı	349.508,85	
600 Yurt İçi Satışlar Hesabı		329.350,00
600.01 Yazılım Tamamlama Gelirleri		
382 Ertelenmiş Gelirler Hesabı		20.158,85
(Fatura tarihindeki kayıt)		
382 Ertelenmiş Gelirler Hesabı	6.587,00	
642 Faiz Gelirleri Hesabı		6.587,00
(1. Aydaki kayıt)		
382 Ertelenmiş Gelirler Hesabı	6.718,74	
642 Faiz Gelirleri Hesabı		6.718,74
(2. Aydaki kayıt)		
382 Ertelenmiş Gelirler Hesabı	6.853,11	
642 Faiz Gelirleri Hesabı		6.853,11
(3. Aydaki kayıt)		
100 Kasa Hesabı	349.508,85	
120 Alıcılar Hesabı		349.508,85
(Tahsilat kaydı)		

Örnek 10: Maksimum hasılat tutarı olan 329.350 TL, peşin fiyatına kredi kartıyla 3 taksitte tahsil edilecektir. Bankanın uyguladığı aylık faiz % 2'dir. Eğer vadeler beklenecek ise ara tahsilatlar olacağından her bir tahsilata olan uzaklığa göre ayrı ayrı faiz hesaplamak gereklidir.



Şekil 15: Taksitlerin Beklenme Süreleri

Faiz geliri ve satış gelirini birbirinden ayırabilmek için tablo 53’de gerekli hesaplamalar yapılmıştır.

Tablo 53: Örnek 10’un Hesaplaması

Taksitle	Vade	Gelecek Değer	Aylık Faiz	Kazanılmamış
1. Taksit	1. Ay	$109.783,33/(1+0,02)=107.630,72$	2.151,61 TL	2.151,61 TL
		Toplam	2.151,61 TL	
2. Taksit	1. Ay	$109.783,33/(1+0,02)= 107.630,72$	2.151,61 TL	2.151,61 TL
	2. Ay	$109.783,33/(1+0,02)^2=105.520,3$	2.110,41 TL	4.263,02 TL
		Toplam	4.263,02 TL	
3. Taksit	1. Ay	$109.783,33/(1+0,02)= 107.630,72$	2.151,61 TL	2.151,61 TL
	2. Ay	$109.783,33/(1+0,02)^2=105.520,3$	2.110,41 TL	4.263,02 TL
	3. Ay	$109.783,33/(1+0,02)^3=103.451,2$	2.069,03 TL	6.332,05 TL
		Toplam	6.332,05 TL	

Bu hesaplamalara göre fatura tarihindeki ve sonraki vadelerde yapılması gereken muhasebe kayıtları tablo 54’te gösterilmiştir.

Tablo 54: Örnek 10'un Muhasebe Kaydı

Hesap Adı	Borç	Alacak
108 Diğer Hazır Değerler Hesabı	329.350,00	
600 Yurt İçi Satışlar Hesabı		316.875,32
600.01 Yazılım Tamamlama Gelirleri		
382 Ertelenmiş Gelirler Hesabı		12.747,68
(Fatura tarihindeki kayıt)		
102 Bankalar Hesabı	109.783,33	
108 Diğer Hazır Değerler Hesabı		109.783,33
(1. Taksitin tahsili)		
382 Ertelenmiş Gelirler Hesabı	6.457,84	
642 Faiz Gelirleri Hesabı		6.457,84
(1. Aydaki kayıt) (Tüm birinci ay faizleri)		
102 Bankalar Hesabı	109.783,33	
108 Diğer Hazır Değerler Hesabı		109.783,33
(2. Taksitin tahsili)		
382 Ertelenmiş Gelirler Hesabı	4.220,81	
642 Faiz Gelirleri Hesabı		4.220,81
(2. Aydaki kayıt) (Tüm ikinci ay faizleri)		
102 Bankalar Hesabı	109.783,34	
108 Diğer Hazır Değerler Hesabı		109.783,34
(3. Taksitin tahsili)		
382 Ertelenmiş Gelirler Hesabı	2.069,03	
642 Faiz Gelirleri Hesabı		2.069,03
(3. Aydaki kayıt)		

Kredi kartı ile yapılan taksitli satışlarda eğer vade günü beklenmeden hemen paraya kavuşulması istenirse bankanın net olarak vereceği tutar hasılat olarak kabul edilmelidir. Kesinti miktarı faiz gideri olarak yorumlanmalıdır. Böylece zamanı gelecek olan bir kazanç ortadan kalkmış olur.

BÖLÜM 4: YAZILIM FİRMALARININ FİNANSAL DURUM TABLOLARI ve DİPNOTLARININ ANALİZİ

Bu bölümde yazılım muhasebesi için uygulanan modelin tutarlılığı, piyasadaki önemli firmaların finansal durum raporlarındaki açıklamaların analiz edilmesiyle test edilecek ve uygulayıcıların durum tespiti yapılacaktır.

Bu analizi gerçekleştirirken önce Forbes dergisinin 2014 yılı Mayıs ayı raporunda en değerli 2000 firma içinde olan 17 yazılım firmasının yazılım muhasebeleştirme konusundaki raporlamaları analiz edilecektir.

Daha sonra Borsa İstanbul'da Bilişim ve Teknoloji Endeksinde yer alan yazılım geliştiren firmaların finansal durum ve dipnotları analiz edilecektir.

4.1.En Değerli Yazılım Firmalarının Analizi

Tablo 55'de yazılım sektöründe piyasa değeri en yüksek olan firmaların maddi olmayan duran varlıklarının tüm varlıklarına oranı verilmiştir. Listedeki firmalardan SAP, Amadeus IT Holdings ve Dassault Systemes IFRS'e göre raporlama yapmaktadır. Diğerleri US-GAAP'e göre raporlama yapmaktadır. Sonuçlara göre, HCL Technologies, Check Point Software ve VeriSign firmalarının aktifleştirdiği maddi olmayan duran varlık miktarı toplam varlıklarına kıyasla çok düşüktür. Üçü de US-GAAP'e göre raporlama yapmaktadır. Bunun yanında en fazla orana sahip olan üç firma sırasıyla Amadeus IT Holdings, Dassault Systemes ve Fiserv'dir. Bunlardan ilk ikisi IFRS'e göre raporlama yapmaktadır. Üçüncüsü ise US-GAAP'e göre raporlama yapmaktadır.

US-GAAP'a göre aktifleştirmenin eşiği "teknolojik yeterlilik" Bu kavrama göre aktifleştirmeye başlama seviyesi işletmelerin rahatlıkla esnetebildiği bir yapı sunmaktadır. Çünkü teknolojik yeterlilik iki şekilde mümkündür. Birinci yöntem göre detaylı program tasarımı yapılır ve bundan sonra aktifleştirmek mümkün olur. İkincisi ise detaylı program tasarımı yapılmaz ve sadece ürün tasarımı yapıp çalışan modelin tasarlananla uyumluluğu açısından incelendiği ve müşteri testine çıkılmak üzere olduğu andır. İkinci yöntem, harcamaların aktifleştirilmesini ürün geliştirme süresinin sonuna doğru gerçekleştirir. (Mulford & Roberts, 2006: 4)

Tablo 55: En Değerli Yazılım Firmalarının Analizi

Sıra No	Firma Adı	Ülkesi	Toplam Varlık (A)	Maddi Olmayan Duran Varlık (B)	Oran (B/A)
1	Microsoft	ABD	172.384.000 \$	6.981.000 \$	0.04
2	Oracle	ABD	90.344.000 \$	6.137.000 \$	0.07
3	SAP	Almanya	38.507.000 €	4.608.000 €	0.12
4	Vmware	ABD	15.216.000 \$	748.000 \$	0.05
5	Salesforce.com	ABD	9.152.930 \$	481.917 \$	0.05
6	Adobe Systems	ABD	10.785.829 \$	469.662 \$	0.04
7	Intuit	ABD	5.201.000 \$	199.000 \$	0.04
8	Amadeus IT Holdings	İspanya	6.165.442 €	2.352.937 €	0.38
9	HCL Technologies	Hindistan	29.980 .	84 .	0.00
10	Dassault Systemes	Fransa	4.968.057 €	1.170.626 €	0.24
11	Fiserv	ABD	9.337.000 \$	2.003.000 \$	0.21
12	CA	ABD	12.016.000 \$	1.063.000 \$	0.09
13	Symantec	ABD	13.539.000 \$	768.000 \$	0.06
14	Check Point Software	İsrail	4.948.818 \$	14.085 \$	0.00
15	Citrix Systems	ABD	5.512.007 \$	390.717 \$	0.07
16	Amdocs	ABD	5.185.277 \$	181.227 \$	0.03
17	VeriSign	ABD	2.154.859 \$	21.344 \$	0.01

Kaynak: Firmaların 2014 Dönemi Finansal Tabloları

US- GAAP'a göre maddi olmayan duran varlık aktifleştirmesi yapan firmaların bilançoları incelendiğinde açıkça görülmektedir ki firmalar, çalışan model yöntemine göre hesaplama yapmaktadır. Bu da bu tür firmaların aktifleştirme miktarlarını düşürmektedir.

Örneğin; Microsoft firması kodlama ve test süreçlerinin yüksek riskli kısımlarının biterek ürünü müşteriye bırakmadan evvel ki noktadan itibaren araştırma ve geliştirme giderlerini aktifleştirmeye başlamaktadır. (Microsoft, 2014: 37) Adobe firması, kritik bir hatanın olmadığı ve çalışan bir prototipin olduğu andan itibaren araştırma ve geliştirme giderlerini aktifleştirmeye başlamaktadır. İç kullanım için geliştirilen yazılımlarda ise önemli bir kısmının bitip kullanım amacına hazır olduğu andan itibaren harcamalarını aktifleştirmektedir. (Adobe Systems, 2014: 70)

IFRS'e göre raporlama yapan firmalar hesaplama modelimize daha yakın yani daha fazla aktifleştirme oranına sahip olan işletmelerdir. SAP firması, araştırma ve geliştirme giderlerini geliştirmeden olup olmadığını tanımlayamadığını ve bu tür giderlerin tamamını gerçekleştiklerinde gider yazdıklarını bildirmiştir. (SAP, 2014: 173) Firmaların rapor ettiği maddi olmayan duran varlıklar diğer kalemlerden oluşmaktadır. Oysa SAP, ayrıştırma yapabilse oldukça fazla miktarda aktifleştirme yapacaktır. Tıpkı Amadeus IT Holdings ve Dassault Systemes gibi. Amadeus IT Holdings, kullanım amaçlı geliştirdiği yazılımların aktifleştirmesinde borçlanma maliyetlerini özel borçlanma olduğunu ifade ederek hesaplamış ve eklemiştir. (Amadeus IT Holdings, 2014: 40) Bu kalemi aktifleştirmeye dahil eden tek firmadır. Devlet teşviklerini katmadığını bildirmiş ve diğer geliştirme harcamalarından bahsetmemiştir. Dassault Systemes firması ise hesaplamasına personel maliyetlerini, yazılım, donanım ve demirbaşların amortismanlarını, bilgisayar bakımı ve donanım giderleri, bilgisayar ağı ve iletişim giderleri ile çeşitli devlet teşviklerini katmaktadır. (Dassault Systemes, 2014: 78) Böylece aktifleştirilen değer büyümektedir.

Firmaların aktifleştirdiği maddi olmayan duran varlık kalemleri incelendiğinde daha özel yorumlar yapılabilecektir. Tablo 55'deki firmaların sırasına göre aktifleştirdikleri maddi olmayan duran varlık kalemleri aşağıdaki gibidir.

1. Teknoloji bazında, Sözleşme bazında, Pazarlamayla ilişkili, Müşteriyle ilişkili
2. Yazılım destek anlaşmaları ve ilişkileri, Donanım destek anlaşmaları ve ilişkileri, Geliştirilmiş teknoloji, Temel teknoloji, Müşteri ilişkileri ve sözleşmeleri, Saas ve Paas anlaşmaları ve ilişkileri, Marka, Araştırma ve geliştirme
3. Yazılım ve veri tabanı lisansları, Edinilen teknoloji, Müşteri ilişkileri ve Diğer maddi olmayan duran varlıklar
4. (Sunucu cevap vermediği için dipnotlara erişilemedi.)
5. Edinilen gelişmiş teknoloji, Müşteri ilişkileri, Ticari isim ve Marka, Alan hakları
6. Satın alınan teknoloji, Müşteri sözleşmeleri ve ilişkileri, Marka, Teknoloji kullanımı için edinilen haklar, Yazılım yerelleştirme, Diğer
7. Satın alınan teknoloji, Ticari isim ve logo, Müşteri listesi, Rekabeti sınırlandırıcı anlaşmalar
8. Patent marka ve lisans, Teknoloji ve içerik, Sözleşmesel ilişkiler, Diğer

9. Yazılım
10. Yazılım, Müşteri ilişkileri, Diğer
11. Müşteriyle ilişkili, Edinilen yazılım ve teknoloji, Ticari isim, Aktifleştirilen yazılım geliştirme maliyeti, Satın alınan yazılım
12. Satın alınan yazılım, İç kullanım için geliştirilen yazılım, Diğer
13. Müşteri İlişkileri, Geliştirilmiş teknoloji, Sınırlı ömürlü marka, Patent, Sınırsız ömürlü marka
14. Temel teknoloji, Marka ve ticari isim, Müşteri ilişkileri
15. Ürünle ilişkili
16. Temel teknoloji, Müşteri ilişkileri, Satın alınan yazılım ve entelektüel haklar, Diğer
17. İç kullanım için geliştirilen yazılım, Dışarıdan alınan sözleşme bazlı geliştirmeler ve danışmanlık hizmetleri

Anlaşılabacağı üzere firmaların faaliyetlerinin çeşitliliğine göre benzer ve farklı konularda raporlamalar yapılmıştır. Hesaplamaların derinliği (harcama kalemi çeşitliliği) değişim gösterse de az ya da çok listedeki tüm firmalar konumuza dair başlıklarda (altı çizili olanlar) aktifleştirme yapmıştır. Şuanda aktifleştirme yapan firmalardan Microsoft, Adobe Systems ve Citrix Systems 1999 yılındaki finansal tablolarında yazılım geliştirme harcamalarını aktifleştirmeden raporlamıştır. Öne sürülen gerekçe, hesaplanamaz bulmalarıdır. (Sandino ve Kaplan, 2001: 9) SOP 98-1'in yayınlanmasıyla birlikte Amerikan standartlarına göre raporlama yapan firmalar için aktifleştirmenin açıklaması genel olarak ifade edilmiştir. Ancak hesaplama dâhil edilen harcama kalemlerinde farklılıklar görülmektedir. Bunun sebebi araştırma ve geliştirme harcamalarını ayırtıramama veya harcama kalemlerini süreçlere dağıtamama problemlerinden ötürüdür. Zaten, Jan-Christoph Suermann'ın doktora tezinde ifade ettiğine göre kendi geliştirdiği yazılımı aktifleştirmeyen firmaların sundukları gerekçeler şunlardır: (Suermann, 2006: 184) Maliyetlerin maddi olmaması, araştırma-geliştirme faaliyetlerini birbirinden ayıramama ve teknik olarak uygulanabilmesinden emin olamama.

4.2. BIST'deki Bilişim ve Teknoloji Endeksindeki Firmaların Analizi

Borsa İstanbul'da işlem gören bilişim ve teknoloji endeksinde toplamda 15 firma bulunmaktadır. Bu firmaların çoğu yazılım geliştirmeyip bilgisayar donanımı alıp satmakta veya internet servisi sağlamaktadır. Dolayısı ile yazılım geliştirmediklerinden aktifleştirmeleri de beklenmez. Yazılım üreterek çalışmamız açısından mercek altına alınabilecek olan firmalara dair bilgiler tablo 56'da verilmiştir.

Toplam varlık miktarındaki maddi olmayan duran varlık oranına bakıldığında en büyük değere Logo firması ulaşmaktadır. Aktifleştirilen tutarın sadece personel giderlerinden olduğu bildirilmektedir. (Logo, 2014: 41)

Karel firmasının ar-ge giderleri incelendiğinde hesaplamaya dahil ettiği kalemler şunlardır: personel giderleri; hammadde ve malzeme-prototip giderleri; bina makine demirbaş ve kira giderleri; seyahat ve konaklama giderleri; araç giderleri; amortisman ve itfa payları; danışmanlık giderleri; elektrik ve haberleşme giderleri; laboratuvar giderleri ve diğerdir. (Karel, 2014: 48) Tüm bu giderlerin bir kısmını gidere, bir kısmını maddi olmayan duran varlığa bir kısmını da stoğa pay etmektedir. Bu kadar çeşitli harcama kalemleri ile geliştirme çalışmaları arasında ilişki kurulabilmesi firmanın yapısının incelenmesi için merak uyandırmaktadır.

Link firmasının ar-ge giderleri incelendiğinde sadece amortisman giderleri görülmektedir. (Link, 2014: 29) Ancak, hangi harcama kalemini aktifleştirdiğini bildirmemiştir.

Aselsan firması, en fazla miktarda maddi olmayan duran varlık raporlayandır, fakat tüm varlıklarının içindeki oranı itibari ile listede dördüncüdür. Çünkü Aselsan'ın faaliyet konusu gereği maddi varlık cinsinden olan işlemleri daha fazladır. Geliştirme maliyetlerini oluşturan kalemler şunlardır: Malzeme, personel ve tasarım giderleri ile amortisman paylarıdır. (Aselsan, 2014: 71) Böylece, aktifleştirmeye en tutarlı ve tam harcama kalemi ekleyen firma konumundadır.

Netaş firmasının aktifleştirdiği geliştirme harcamaları; personel giderleri ve genel üretim giderlerinin bir kısmıdır. (Netaş, 2014: 14)

Alcatel firması ar-ge giderleri incelendiğinde sadece personel giderleri görülmektedir. (Alcatel, 2014: 39) Ancak, hangi harcama kalemini aktifleştirdiğini bildirmemiştir.

Tablo 56: Yerel Aktifleştirme Yapan Firmaların Analizi

Firma Adı	Toplam Varlık (A)	Maddi Olmayan Duran Varlık (B)	Oran (B/A)	(B) nin Detayı	Aktifleştirilen Geliştirme Harcamaları
Logo	166.141.594 TL	37.728.856 TL	0,23	Geliştirme Maliyetleri- Gelişmiş Teknoloji- Müşteri İlişkileri- Rekabet sınırlayıcı Sözleşme-D.M.O.D.V.	Personel
Karel	322.457.215 TL	60.563.659 TL	0,19	Haklar-Geliştirme harcaması-Şirket içi yazılım geliştirme	Tüm araştırma ve geliştirme harcamalarından gidere, stoğa ve M.O.D.V. a pay veriyor.
Link	14.012.515 TL	2.371.171 TL	0,17	Haklar-Geliştirme Giderleri-D.M.O.V	Belirsiz
Aselsan	4.985.433.774 TL	438.506.438 TL	0,09	Haklar-Geliştirme Giderleri-Bilgisayar Yazılımları	Malzeme, Personel, Amortisman, Tasarım
Netaş	901.106.101 TL	32.886.087 TL	0,04	Satın Alınan M.O.D.V.- Bilgisayar Yazılımı-Sözleşmeye Bağlı Müşteri İlişkileri	Personel, Genel Üretim Giderleri
Alcatel	322.509.671 TL	2.155.249 TL	0,01	Haklar-Geliştirme Maliyetleri-D.M.O.V.	Belirsiz

Kaynak: Firmaların 2014 Dönemi Finansal Tabloları

Her yazılım üreten firmanın TMS 38'e göre personel giderlerini, yazılım ve donanımların itfa paylarını kesinlikle aktifleştirmesi gerekmektedir. Çünkü bunlar olmazsa olmaz olan giderlerdendir. Bunun yanına ilişki kurabildiği diğer harcamaları da ekleyebilmesi gerçeğe uygun bir raporlama ortaya çıkarır.

Personel harcamaları, bir yazılım projesi için tespitinin karmaşası ve gerçekleşme sıklığı bakımından en kritik harcama kalemidir. Daha sonra sırasıyla genel giderler, danışmanlık hizmetleri, yazılımların ve donanımların itfa payları gelmektedir. (Gentle, 2011: 49) Personel giderlerinin toplam miktarının ne kadar olduğu aylık maaş üzerinden rahatlıkla hesaplanabildiğinden basittir. Ancak süreçlere bunu dağıtmak karmaşıktır. Keza, itfa paylarının dönem payını hesaplamak kolaydır ancak hangi sürece dağıtılması

gerektiđi daha zordur. Danışmanlık giderlerinin ise süreçlere dağıtılması basittir çünkü hangi süreç için harcama yapıldığı bellidir.

İşletmelerin aktifleştirdikleri varlıkların değerini hesaplamaya dâhil ettikleri harcama kalemlerinde bu denli farklılık olmasının sebebi raporlamayı oluşturanların ulaşabildikleri verilerin imkânından dolayıdır. Yani işletmeler hangi veriye ulaşabiliyorsa onunla raporlama yapabilmektedir. Diğer etkenler ise işletme stratejisi, yorum farkı ve niyet faktörleridir.

Ana tartışma konusu, Türkiye'deki firmalar, yaptıkları harcamaları hemen dönem giderine yazarak daha az vergi ödeme istemelerinden ileri gelmektedir. Hâlbuki geliştirme harcamalarını aktifleştirerek ileriki dönemlerde gider yazmanın yatırımcı-hissedara, firmaya, denetçiye, analizci-araştırmacıya ve düzenleyiciye bir takım faydaları vardır. (Wyatt ve Abernethy, 2008: 101-103)

- Yatırımcı ve hissedara olan faydalar: Aktifleştirilen maddi olmayan varlığı fark eden yatırımcının firmadan beklentisi artacak ve somutlaşan beklenti ile yatırım kararı alabilecektir. Somut veriler karşılaştırılabilirliği sağlayacaktır. Böylece sektörde işletmenin durumu belli olabilecektir.
- Firmaya olan faydalar: Maddi olmayan varlık aktifleştirilmesi yöneticilere yatırımcılarla diyalog kurmak için standartlaşmış ve yorumlanabilir bir yol sunar. (Healy ve Palepu, 1993: 5) Bununla beraber daha az açıklama yapma riski oluşturmaz yani idari cezaların oluşma ihtimalini azaltır. En önemlisi ise piyasa değeri arttıkça borçlanma maliyetleri azalır, fonlanma ve likiditede ferahlama sağlanır.
- Denetçiye olan faydalar: Maddi olmayan duran varlık ile ilgili gider hesaplarının yorumu için kavramsal çerçeve hazırdır. Aktifleştirme yapılması denetçinin yükünü azaltır. Çünkü varlıklar açıklanır ve değerlerin kaynakları tespit edilebilir. Böylece denetlenen ve denetçi arasında maddi olmayan duran varlığın tanımlanma kriterleri açısından anlaşmazlıklar azalır.
- Analizci ve araştırmacıya olan faydalar: Veri toplama ve işleme sayesinde ek bilgiler açığa çıkabilir, böylece yeni yorumlar yapılabilir ve yeni mali stratejiler geliştirilebilir.

- Düzenleyiciye olan faydalar: Etkili bilgiye ulaşmada düşük maliyet imkânı elde edilir, düşük maliyetle kamu yararını koruma sağlanır.

Önemli olan kar değil, varlıkları arttırmak, piyasa değeri ve kamu yararınıdır. Bu sebeple daha fazla halka açık yazılım firmasının olması hem ülkemiz hem de yukarıda sayılan tarafların tamamı için kazançlı olacaktır.



SONUÇ ve DEĞERLENDİRME

Bu bölümde, önceki bölümlerde izah edilen konuların sonuçları ve sonraki araştırmacılara tavsiyeler arz edilecektir.

Satın Alınan Yazılımlar

Yazılımların satın alınması söz konusu olduğunda bunun nasıl muhasebeleştirilmesi gerektiğine dair ikinci bölümde gerekçeler ortaya konmuştur, üçüncü bölümde ise örnekler ile açıklanmıştır.

Satın alan taraf için, elde ettiği kullanım hakkı sayesinde yazılımdan gelecekte ekonomik fayda elde edebilmesinden dolayı varlık kabul edilir. Eğer satın alınan yazılımın maliyeti her sene devlet tarafından belirlenen demirbaş sınırının altında kalırsa, aktifleştirmeden hemen dönemin giderine yazılabilir. Bu sınırın üzerinde bir elde etme maliyeti var ise TDHP'na göre 260 Haklar Hesabına kayıt edilmelidir. Buna ilave olarak kullanıma hazır hâle getirmeye yönelik harcamalar ve değerini arttırıcı harcamalar da maliyete ilave edilir. İşlerliğinin sürdürülmesine yönelik harcamalar ise oluştuğları anda gider yazılır.

Haklar, tüketildikçe itfa payları hesaplanarak giderleştirilir. Satın alınan yazılımların itfa payının hesaplanması konusunda TMS 38 standardı gereği varlığın yararlı ömrünün sınırlı/sınırsız cinsinden tanımlanması gerekmektedir. Ancak buna göre itfa payının hesaplanabilmesi söz konusudur. Şekil 12'de özetlemektedir.

Satın alınan yazılımların yararlı ömürleri lisans süreleri ile sınırlı olduğu kabul edilmelidir ki bu durumda hem varlığın ömrü sınırlandırılmış olur, hem de itfa payı hesaplanabilir. Hesaplama modeli olarak TMS 16 standardındaki yöntemlerden biri seçilerek yapılabilir.

Hesaplanan itfa payının kaydı da varlığın kullanım yerine göre değişen hesaplarda borç olarak kayıt edilebilir. Alacak kaydı hep 268 Birikmiş Amortismanlar Hesabı (-) olmalıdır.

Yazılımın kullanıldığı yer üretim faaliyeti ile ilgiliyse üretim safhasının araştırma veya geliştirme dönemlerine denk gelen itfa payları şekil 13'de görüldüğü biçimde kayıt edilmelidir.

Tek düzen hesap planında bulunan mevcut hesap isimlerinin TMS standartlarına uyumu açısından bazı hesap isimleri önerilmiştir. 263 ve 750 numaralı hesapların isimleri “Araştırma ve Geliştirme Giderleri”dir. Aynı isimli bu iki hesaptan 263 numaralı olanda geliştirme harcamaları, 750 numaralı olanda araştırma giderleri kayıt edilmektedir. O hâlde isimlerinin buna göre düzeltilmesi öze daha uygundur.

Satın alınan yazılımların değerlemesi konusu incelendiğinde varlığın aktif bir piyasasından bahsetmek mümkündür. Bu sebeple fiyat değişimlerinden etkilenmesi durumunda şekil 14’deki yol haritasına göre muhasebe kayıtlarını yapmak gereklidir.

Varlığın yararlı ömrü içerisinde değerinde artış veya azalış olması durumunda şekil 14’deki gibi hareket etmek gereklidir. Varlığın yararlı ömrü bittiğinde yeniden değerlemeden bahsedilemez. Sadece hurda değerinden bahsetmek gündeme gelebilir. Ancak söz konusu varlığı 3. kişilere satma yetkisi yazılımı alan tarafta hiçbir zaman olmayacağından “hurda” değerinden de bahsetmek mümkün olamaz.

Yerli ve yabancı işletmelerin 2014 yılı finansal durum tabloları ve dipnotları incelendiğinde haklar, edinilen yazılımlar, satın alınan yazılımlar, yazılım lisansları, bilgisayar yazılımları, yazılımlar ve diğer maddi olmayan duran varlıklar gibi başlıklar altında ifade edilerek raporlama yapılmıştır. Açıklamalara göre maliyet değeri üzerinde aktifleştirilmiştir.

Yabancı firmaların amortisman süreleri 1-10 yıl arasında değişmektedir. Yazılımların yararlı ömürleri ayrı ayrı hesaplanmakta ve satın alınan yazılımlar kaleminin amortisman ortalama süresi oluşmaktadır.

Yerli firmaların amortisman süreleri 1-15 yıl arasında değişmektedir. Yazılımların yararlı ömürlerine Aselsan, 1-2 yıl; Link ve Escort 3-5 yıl; Alcatel ve Netaş 5 yıl; Karel ise 3-15 yıl süre biçmektedir. Neye göre bu sürelerin tayin edildiği belirtilmemektedir. Biraz genelleyici yaklaşım bulunmaktadır.

333 sıra numaralı VUK Genel Tebliği ekli listedeki açıklamaya göre “4.3 Bilgisayar yazılımları” sınıflandırması uyarınca 3 yılda ve %33,3 amortisman oranı ile itfa edilmesi istenmektedir. 3 yılın açıklaması bu olabilir. Diğer süreler işletmelerin kendi özel durum veya yorumları olabilir. Aselsan firması modelimize en yakın değerleri kullanmaktadır.

Satın alınan yazılımların TMS 38'e göre muhasebeleştirilmesinin hükümleri açık, net ve hesaplaması basit olduğundan uygulayıcıların aktifleştirmelerinde aksaklığa rastlanılmamıştır. Amortisman süreleri çeşitli olup VUK'un tavsiyesine uyma ağırlığı vardır.

Kullanım Amaçlı Geliştirilen Yazılımlar

Kullanım amaçlı yazılım geliştirmesi söz konusu olduğunda bunun nasıl muhasebeleştirilmesi gerektiğine dair ikinci bölümde gerekçeler ortaya konmuştur, üçüncü bölümde, örnekler ile açıklanmış ve dördüncü bölümde firmaların davranışları analiz edilmiştir.

TMS 38 standardında tam olarak yazılım geliştirme süreçlerinden bahsedilmese de satır aralarındaki ifadelerden yorumlar çıkartılabilmektedir. Buna ilave olarak Amerikan standartlarındaki hükümler incelendiğinde yazılım geliştirme süreçlerinin hangisinin araştırma veya geliştirme safhasına dâhil olduğu net olarak anlaşılabilir. Buna göre isterlerin belirlenmesi ve analiz aşamaları araştırma safhası iken tasarım, kodlama, test ve devreye alma aşamaları da geliştirme safhası olarak kabul edilmektedir.

Araştırma safhası ile ilişkili olan harcamalar dönem giderine yazılırken geliştirme safhası ile ilişkili olan harcamalar ise aktifleştirilir. Safhalar ile harcamaların ilişkisi kaynakların zaman birimi cinsinden tüketilmesine göre kurulması tavsiye edilmektedir.

Geliştirmesi tamamlanan yazılımın yararlı ömrü ölçülmelidir. Bunda bir sınırlılık var ise itfa payı hesaplanabilirken; herhangi bir makul sınırlılık yok ise sınırsız yararlı ömürlü varlık kabul edilir ve itfa payı hesaplanmaz.

Yeniden değerlendirme modeli uygulanabilmesi için aktif bir piyasa şartı gereklidir. Bu şart sağlanmadığından uygulama yolu kapanmıştır.

Yazılıma yapılabilecek iyileştirici bakım (Bakınız: 1.2.7. numaralı başlık) ile aktifleştirmeye ilaveler devam edebilir.

Yerli ve yabancı işletmelerin 2014 yılı finansal durum tabloları ve dipnotları incelendiğinde araştırma ve geliştirme, aktifleştirilen yazılım geliştirme maliyetleri, iç kullanım için geliştirilen yazılım, geliştirme maliyetleri, geliştirme harcamaları, geliştirme giderleri ve şirket içi yazılım geliştirme gibi başlıklar altında ifade edilerek raporlama yapılmıştır.

Analiz edilen firmaların aktifleştirdikleri yazılım geliştirme maliyetleri çeşitli gider yeri ve türünden oluşturulduğu fark edilmiştir. (Bakınız 4.2.)

Yazılım projelerinin etkin bir yönetime ihtiyaç duyduğu muhakkaktır; hem etkin ve verimli bir proje yönetimi hem de standartlara uygun finansal raporlama yapabilmek için gerekli verilerin toplanabilmesi gerekir. Bunu sağlayabilecek en hazır yazılımlar proje yönetim yazılımlarıdır.

Standartlara uygun şekilde mali raporlama yapabilmek için piyasadaki proje yönetim yazılımlarını kullanmak da yetmemektedir, proje yönetim yazılımlarının yazılım geliştirme süreçlerini, araştırma-geliştirme cinsinden kategorize etmesi gereklidir. Bunlara ek olarak harcamaların ve amortisman gibi diğer giderlerin faaliyetlere dağıtılması ve muhasebe sistemiyle entegre olması gerekir ki bu sayede standartlara uygun finansal raporlama yapabilmek mümkün olsun. Aksi hâlde standartlara uygun finansal raporlama yapma niyetinde olursa bile ilgili hesaplamaları doğru bir şekilde yapabilmek için çok efor harcanması gerekir.

İşletmelerin standartlara uygun şekilde raporlama yapması istendiğine göre bunu sağlayabilecekleri araçların da piyasada mevcut olması gerekmektedir. Piyasadakiler, projenin iyi yönetilmesi için hazırlanmıştır. Yazılım mühendislerinin söz konusu yazılımlara ekleyecekleri yukarıdaki işlevler ile muhasebenin ihtiyaç duyduğu bilgiler elde edilebilir. Böylece Jan-Christoph Suermann'ın doktora tezinde ifade ettiği gerekçeler ile aktifleştirme yapmayanların mazeretleri büyük ölçüde giderilmiş olur.

Ancak unutulmamalıdır ki TMS 38 yazılım süreçlerinin hangisinin araştırma veya geliştirme olduğunu net şekilde ifade etmemektedir. “TMS Yorum 32: Maddi Olmayan Duran Varlıklar – İnternet Sitesi Maliyetleri”nde olduğu gibi yazılım geliştirme sürecinin tanımlanması ve buna bağlı olarak giderleştirilecek ve aktifleştirilecek harcamaların sınıflandırılmasında kaynağa ihtiyaç vardır. Bu konuda yeni bir TMS yorumu gereklidir.

Sipariş Üzerine Geliştirilen Yazılımlar

Sipariş üzerine yazılım üretimi, yazılım işletmelerinin gelir türlerinden sadece biridir. Dolayısı ile firmaların bu konudaki işlemlerini raporlaması hem kendileri için, hem de bilgi kullanıcıları için kritiktir. Çünkü hem aktifleştirilecek olan tutar, hem de hizmet maliyeti belirlenecek ve karlılık etkilenecektir.

Birden fazla mali döneme sirayet eden yazılım geliştirme taahhüt sözleşmeleri, TMS 11 İnşaat Sözleşmeleri prensiplerine göre muhasebeleşir. Standart, işi yapmayı taahhüt eden işletmeye işin tamamlanma oranı ve dönem gelirinin tespiti konularında sorumluluk yüklemektedir. İşin tamamlanma oranının tespit edilebilmesi için de proje harcamalarını takip edebilecek bir finans sisteminin olmasını beklemektedir. TMS-38'e göre güvenilir raporlama sadece bu şartlar altında sağlanabilir.

Sözleşmeye göre yazılım geliştirmek, önceki başlıkta anlatılan geliştirme süreçlerini yaşattırır. Dolayısı ile yazılım geliştirme projesini takip etmek için gerekli olan ve önerilen tüm metod ve tavsiyeler şimdi de lazımdır. Raporlamadaki fark tabii olduğu standarttan ve hasılat boyutundan ileri gelmektedir.

Modelimizde dönemsellik, maliyet esası, tam açıklama ve sosyal sorumluluk gibi ilkelere uygun bir kayıt düzeni oluşturmak amaçlanmıştır.

Yerli ve yabancı işletmelerin 2014 yılı finansal durum tabloları ve dipnotları incelendiğinde sözleşme bazında, müşteri sözleşmeleri, sözleşmesel ilişkiler ve sözleşmeye bağlı müşteri ilişkileri gibi başlıklar altında ifade edilerek raporlama yapılmıştır. Bu başlıklar altında müşterilerine taahhüt ettikleri aktifleştirilen hizmet maliyetleri bulunmaktadır. Birden fazla mali döneme sirayet ettiğinden maddi olmayan duran varlıklar altında rapor edilmiştir.

Bu tür varlıkların itfası genelde hizmet ediminin taahhüt edildiği süre ile sınırlı olacak şekilde itfasına karar verilmiştir.

Bu çalışmadan hareketle gelecekte yazılım mühendislerinin çabalarıyla mevcut proje yönetim yazılımlarına yapılacak bir yama ile standartların istediği ayrıştırılmayı (araştırma veya geliştirme) sağlayabilen ve ayrışanları olması gereken biçimde muhasebeleşirmek üzere entegre olabilen yeni bir proje yönetim yazılımı elde edilebilir.

Türkiye Muhasebe Standartları'nda yazılımların aktifleştirilmesine yönelik düzenlenmiş bir standart olmadığından farklı yorumların önüne geçilmesi zor olacaktır. Bu sebeple "TMS Yorum 32 Maddi Olmayan Duran Varlıklar – İnternet Sitesi Maliyetleri"nin açıklandığı gibi kullanım amaçlı yazılım geliştirme ve satış amaçlı geliştirmenin maliyetlerini açıklayan standardize edilmiş bir yoruma ihtiyaç vardır.

KAYNAKÇA

Kitaplar

- Akbulut, A. (2012). *Karşılaştırmalı TMS/TFRS-Vergi Uygulamaları ve Sonuçları*. Ankara: Maliye Hesap Uzmanları Derneği.
- Alican, F. (2006). *Ekonomik ve Sosyal Boyutlarıyla Dünya'da ve Türkiye'de Yazılım Sektörü*. İstanbul: İletişim Yayınları.
- Arora, A. ve A. Gambardella. (2005). *From Underdogs to Tigers: The Rise and Growth of the Software Industry in Brazil, China, India, Ireland, and Israel*. New York: Oxford University Press.
- Atbaş, H. (2012). *Kaliteli Yazılım Nasıl Geliştirilir?*. İstanbul: Pusula Yayıncılık.
- Boehm, B. W. (1988). *A Spiral Model of Software Development and Enhancement*. USA: IEEE Computer.
- Çatıkkaş, Ö. ve C. Şuekinçi. (2013). *TMS-11 İnşaat Sözleşmeleri Standardına Göre İnşaat Muhasebesi ve Vergi Uygulamaları*. İstanbul: Maliye Hesap Uzmanları Derneği.
- Evans, J. R. (1997). *Production & Operations Management: Quality, Performance, and Value*. Minnesota: West Publishing Company.
- Gençoğlu, Ü. G., Y. Özerhan ve S. Karabınar. (2013). *Türkiye Finansal Raporlama Standartları*. Sakarya: Sakarya Üniversitesi Sürekli Eğitim Merkezi.
- Gentle, M. (2007). *IT Success! Towards a New Model for Information Technology*. West Sussex: John Wiley & Sons.
- Gentle, M. (2011). *An Introduction to IT Project Financials-Budgeting, Cost Management and Chargebacks*. Micheal Gentle: www.lulu.com.
- Gürbüz, A. (2010). *Yazılım Test Mühendisliği*. İstanbul: Papatya Yayıncılık.
- Jones, C. (2010). *Software Engineering Best Practices - Lessons from Successful Projects in the Top Companies*. New York: McGraw-Hill.
- International Institute of Business Analysis (IIBA). (2006). *A Guide to the Business Analysis Body of Knowledge.2*. Baskı. Toronto: www.iiba.org.
- Karabınar, S. (2011). *Türkiye'de Hasılat Ölçümü ve Raporlamasında Sorunlar ve Çözüm Önerileri*. Bursa: Ekin Yayınevi.
- Kliem, R. L. (2008). *Effective Communications for Project Management*. New York: Taylor & Francis Group.

- McConnell, S. (2006). *Software Estimation: Demystifying the Black Art*. Washington: Microsoft Press.
- Mirza, A. A. ve G. J. Holt. (2011). *Practical Implementation Guide and Workbook. for IFRS*. (3. Baskı). New Jersey: John Wiley and Sons.
- Nizam, A. (2014). *Yazılım Proje Yönetimi*. İstanbul: Papatya Yayıncılık.
- Örten, R., H. Kaval ve A. Karapınar. (2012). *TMS-TFRS Uygulama ve Yorumları*. 6 baskı. Ankara: Gazi Kitabevi.
- Pressman, R. S. (2009). *Software Engineering: A Practitioner's Approach*. 7. Baskı. New York: McGraw-Hill Science.
- Project Management Institute. (2000). *A Guide to the Project Management Body of Knowledge*. Pennsylvania: PMI.
www.cs.bilkent.edu.tr/~cagatay/cs413/PMBOK.pdf. (30.11.2015)
- Sarıdoğan, E. (2008). *Yazılım Mühendisliği*. İstanbul: Papatya Yayıncılık.
- Sarıdoğan, E. (2011). *Yazılım Mühendisliği Temelleri*. İstanbul: Papatya Yayıncılık.
- Sommerville, I. (2011). *Software Engineering*. Boston: Addison-Wesley.
- Şahin, M. (1996). *Üretim Yönetimi*. Eskişehir: Anadolu Üniversitesi.
- Usta, Ö. (2008). *İşletme Finansı ve Finansal Yönetim*. 3. Baskı. Ankara: Detay Yayıncılık.

Sürekli Yayınlar

- Akbulut, Y. ve B. Marşap. (2006). Maddi Duran Varlıklarda Değer Düşüklüğünün "TMS 36: Varlıklarda Değer Düşüklüğü" Standardı Kapsamında İncelenmesi ve İMKB'de İşlem Gören Şirketlerdeki Uygulamaların Değerlendirilmesi. *Muhasebe Bilim Dünyası Dergisi*, 8.4, 85-116.
- Akdoğan, N. (2000). Araştırma Geliştirme Maliyetlerinin Kapsamı ve Muhasebeleştirilme Esasları (TMS-15). *Muhasebe ve Denetime Bakış*, 1.1, 13-17.
<http://www.turmob.org.tr/TurmobWeb/EKutuphane/EKIcerikDetay.aspx?enc=1EkNSf+wNjLVDesyolc0i3Pc17vwVttARKeaLMfJq1u9+O9TQTgcRF7iGexiXgoULjRiqMTFUBI=> . (15.10.2014)
- Akdoğan, N. ve O. Sevilengül. (2007). Türkiye Muhasebe Standartlarına Uyum İçin Tek Düzen Hesap Planında Yapılması Gereken Değişiklikler. *Mali Çözüm Dergisi*. 84, 29-70.
<http://archive.ismmmo.org.tr/docs/malicozum/84malicozum/02%20nalanakdogan.pdf> (29.02.2016)
- Ali, A. S., F. T. Anbari ve W. H. Money. (2008). Impact of Organizational and Project Factors on Acceptance and Usage of Project Management Software and Perceived Project Success. *Project Management Journal*, 39.2, 5-33.
<http://dx.doi.org/10.1002/pmj.20041>. (09.11.2015)
- Calayoğlu, İ. ve R. Yılmaz. (2015). Kullanım Amaçlı İşletme İçinde Geliştirilen Yazılımların Türkiye Muhasebe Standartları'na Göre Muhasebeleştirilmesi. *İşletme ve İktisat Çalışmaları Dergisi*, 3.4, 130-153.
<http://www.isletmeiktisat.com/index.php/iicd/issue/view/12> (15.11.2015)
- Dulaney, W. (2000). Capitalizing Software and Creating Business Value. *Journal of Accountancy*, 33-35.
<http://www.journalofaccountancy.com/issues/2000/jul/capitalizingsoftwareandcreatingbusinessvalue.html> (12.01.2016)
- Elitaş, C., O. Aydemir ve B. L. Demirel. (2007). Teknoloji Geliştirme Bölgeleri ve Tekdüzen Hesap Planı Açısından Bir Değerlendirme. *Afyonkarahisar Kocatepe Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*. 9.1, 337-347.
<http://www.iibfdergi.aku.edu.tr/9-1.html> (25.03.2016)
- Ergün, Ü. (2001). Araştırma Geliştirme Maliyetlerinin Muhasebeleştirilmesi Standardının Uygulama İlke ve Yöntemleri (TMS-15). *Muhasebe ve Denetime Bakış*, 2.2. 65-80.
<http://www.turmob.org.tr/TurmobWeb/EKutuphane/EKIcerikDetay.aspx?enc=1EkNSf+wNjIacMTbRBbhd6x9jFh6hDT19kX2v0Kqnx4RLmRxxwRmhF4Wja/E4+gQUlzEXYkxQWLI=>. (15.10.2014)

- Gökgöz, A. (2013). Diğer Kapsamlı Gelirler ve Muhasebeleştirilmesi. *Muhasebe ve Finansman Dergisi*, 9.57, 23-38.
http://journal.mufad.org.tr/index.php?option=com_content&view=article&id=692%3Adiger-kapsamli-gelirler-ve-muhasebelestirilmesi&catid=938%3Asayi-57-ocak-2013&Itemid=55&lang=tr. (23.05.2014)
- Günaydın, H. M. ve S. Bolposta. (2002). Proje Yönetimi Uygulamalarının Yapımında Kullanımı Etkileyen Faktörler. *Ege Mimarlık*, 12.42, 27-30.
www.egemimarlik.org/42/42-4.pdf. (25.06.2014)
- Güngörmüş, A. H. ve E. Boyar. (2009). TMS-2 Stoklar Standardında Alışlarla İlgili Olarak Ortaya Çıkan Vade Farklarına Yönelik Bir Uygulama Önerisi. *Vergi Sorunları Dergisi*. 244, 172-179. http://enderboyar.com/wp-content/uploads/2015/05/8-Vergi-Sorunlar%C4%B1-2009-vadeli_alis.pdf (03.03.2016)
- Healy, P. M. ve K. G. Palepu. (1993). The Effect of Firms' Financial Disclosure Strategies on Stock Prices. *Accounting Horizons*, 7.1, 1-11.
<http://search.proquest.com/docview/208903714?accountid=16693>. (21.10.2015)
- Henry, E. ve O. J. Holzmann. (2009). Contract-Based Revenue Recognition. *Journal of Corporate Accounting & Finance*, 20.5, 77-81.
<http://dx.doi.org/10.1002/jcaf.20518> (23.10.2015)
- Luecke, R. W., D. T. Meeting ve M. R. Nardecchia. (1999). SOP 98-1: Accounting for the Costs of Computer Software Developed or Obtained for Internal Use. *Ohio CPA Journal*, 58.1, 29-33.
<http://connection.ebscohost.com/c/articles/1664816/sop-98-1-accounting-costs-computer-software-developed-obtained-internal-use> (12.01.2016)
- Munter, P. (1997). AcSEC Tackles Accounting for Internal-Use Software. *Journal of Corporate Accounting & Finance*, 9.1, 29-33.
<http://dx.doi.org/10.1002/jcaf.3970090105>. (11.12.2015)
- Munter, P. (2002). A Quick Guide to Accounting for Software and Web Sites. *Journal of Corporate Accounting & Finance*, 13.5, 25-32.
<http://dx.doi.org/10.1002/jcaf.10082>. (11.12.2015)
- Munter, P., T. Moores ve G. Sanders. (2006). Software Costs - A Growing Accounting Issue. *Journal of Corporate Accounting & Finance*, 2.2, 159-164.
<http://dx.doi.org/10.1002/jcaf.3970020206>. (11.12.2015)
- Okur, M. (2007). Yazılım Endüstrisinin Geleceği Gelişmekte Olan Ülkeler ve Türkiye. *Journal of Yasar University*, 2.7, 651-660. http://journal.yasar.edu.tr/wp-content/uploads/2012/05/No_7_vol2_03_okur.pdf. (16.04.2015)

- Olsen, L. ve T. R. Weirich. (2010). New Revenue-Recognition Model. *Journal of Corporate Accounting & Finance*, 22.1, 55-61. <http://dx.doi.org/10.1002/jcaf.20644>. (12.12.2015)
- Saban, M. ve M. Genç. (2005). Araştırma ve Geliştirme Faaliyetlerinin IAS-38 Maddi Olmayan Varlıklar Kapsamında Muhasebeleştirilmesi. *Mali Çözüm Dergisi*, (70), 123-133. <http://www.istanbulsmmmmodasi.org.tr/yayinlar.asp?Gid=1&Yid=70>. (14.08.2014)
- Sandino, T. ve R. S. Kaplan. (2001). Accounting for Computer Software Development Costs. *Harvard Business School*, 1-9. <https://cb.hbsp.harvard.edu/cbmp/product/102034-PDF-ENG>. (06.07.2015)
- Sardo, A. ve A. Arcady. (1998). Accounting for Internal-Use Software. *Journal of Corporate Accounting & Finance*, 9.4, 47-53. <http://dx.doi.org/10.1002/jcaf.3970090405>. (22.12.2015)
- Sarıay, İ. (2012). Türkiye Muhasebe Standartlarına Göre Maddi Duran Varlıkların Değerlemesine İlişkin Uygulamalar. *Electronic Journal of Vocational Colleges*, 2.1, 110-124. http://www.ejovoc.org/May_2012.aspx. (23.03.2015)
- Sonnelitter, R. J. ve P. Pacter. (1994). Accounting for Software. *Journal of Corporate Accounting & Finance*, 6.2, 43-58. <http://dx.doi.org/10.1002/jcaf.3970060205>. (15.12.2015)
- Sumer, H. ve M. Erer. (2010). Yazılımların (TMS 38'e göre) Değerlemesi. *Mali Çözüm Dergisi*, (98), 25-40. <http://www.istanbulsmmmmodasi.org.tr/yayinlar.asp?Gid=1&Yid=98>. (05.10.2013)
- TBD Genel Kurul Kararı. (2014). Yazılım Üretim Sektörü Bildirgesi. *Bilişim Kültürü Dergisi*, 42.162, s. 58. www.bilisimdergisi.org/s162/pages/s162_web.pdf. (23.11.2015)
- Uşul, H. ve O. Özdemir. (2007). Türkiye Muhasebe Standartlarına Göre İnternet Sitesi Maliyetlerinin Muhasebeleştirilmesi. *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi*, 12.3, 121-132. <http://dergipark.ulakbim.gov.tr/sduiibfd/article/download/5000122603/5000112908>. (27.11.2015)
- Walker, R. G. ve G. R. Oliver. (2005). Accounting for Expenditure on Software Development for Internal Use. *Abacus*, 41.1, 66-91. <http://dx.doi.org/10.1111/j.1467-6281.2005.00170.x>. (22.12.2015)
- Wyatt, A. ve M. Abernethy. (2008). Accounting for Intangible Investments. *Australian Accounting Review*, 18.2, 95-107. <http://dx.doi.org/10.1111/j.1835-2561.2008.0012.x>. (20.12.2015)

Yılmaz, R. ve İ. Calayođlu. (2015, a). Accounting For Purchased Software Based On Turkish Accounting Standards. *Journal of Economics, Finance and Accounting*, 2.2, 181-195.
www.pressacademia.org/images/documents/jefa/archives/vol_2_issue_2/04.pdf. (04.07.2015)

Yılmaz, R. ve İ. Calayođlu. (2015, b). Accounting for Developed Software on Contract. *Journal of Economics, Finance and Accounting*, 2.3, 438-460.
http://www.pressacademia.org/images/documents/jefa/archives/vol_2_issue_3/10.pdf. (02.10.2015)



Tezler

Arpacı, M. C. (2010). Türkiye'de Yazılım Sektöründe Sağlanan Teşvikler ve Maliyetleme: Türkiye ve Hindistan Karşılaştırılması. *Yüksek Lisans Tezi*. Ankara, Gazi Üniversitesi, Sosyal Bilimler Enstitüsü, İşletme Anabilim Dalı.

Kır, E. (2007). Yazılım Sektöründe Proje Yönetimi. *Yüksek Lisans Tezi*. İstanbul, Kadir Has Üniversitesi, Sosyal Bilimler Enstitüsü, İşletme Anabilim Dalı.

Suermann, J. C. (2006). Bilanzierung von Software nach HGB, US-GAAP und IFRS - Integrative Analyse der Regelungen zu Ansatz, Bewertung und Umsatzrealisation von Software aus Hersteller- und Anwendersicht. *Doktora Tezi*. Würzburg, Julius-Maximilians Universität. https://opus.bibliothek.uni-wuerzburg.de/frontdoor/deliver/index/docId/1671/file/Druckversion_Dissertation.pdf. (06.12.2014)

Bildiriler

- Albrecht, A. (1979). Measuring Application Development Productivity. *IBM Application Development Symposium*. California: IBM. 83-92.
- Arora, A., & Gambardella, A. (2004). The Globalization of the Software Industry. *National Bureau of Economic Research*. Massachusetts: The MIT Press. 1-32. <http://www.nber.org/chapters/c10805>. (15.08.2014)
- Carmel, E. (2003). The Software Expertin Nations: Success Factors. *The Globalization of the Software Industry*. Hong Kong: City University of Hong Kong. 1-12. <http://www.ejisd.org/ojs2/index.php/ejisd/article/view/78/78>. (15.08.2014)
- Fırat, H. (2008). Maddi Duran Varlıklarda Deęerleme Esaslarının TMS ve VUK Ađısından İncelenmesi. *VI. Muhasebe Uygulamaları ve Vergi Mevzuatı Sempozyumu*. Antalya: Ankara Serbest Muhasebeci Mali Múşavirler Odası. 103-130.

Raporlar

- Adobe Systems.* (2014). Annual Report. California. <http://www.adobe.com/investor-relations/financial-documents.html>. (20.11.2015)
- Alcatel.* (2014). Finansal Tablolar ve Bağımsız Denetçi Raporu. İstanbul. <https://www.kap.gov.tr/bildirim-sorgulari/finansal-tablolar.aspx>. (22.11.2015)
- Amadeus IT Holdings.* (2014). Annual Report. Madrid. http://www.investors.amadeus.com/english/quarterly_financial_info/annual_reports/. (20.11.2015)
- Aselsan.* (2014). Finansal Tablolar ve Bağımsız Denetçi Raporu. Ankara. <https://www.kap.gov.tr/bildirim-sorgulari/finansal-tablolar.aspx>. (22.11.2015)
- Batı Akdeniz Kalkınma Ajansı (Baka).* (2012). Yazılım Sektörü Raporu. Antalya. <http://www.baka.org.tr/uploads/1357649691YAZILIM-SEKTORU-RAPORU-3ARALIK.pdf>. (23.07.2014)
- Dassault Systemes.* (2014). Annual Report. Vélizy-Villacoublay Cedex. <http://www.3ds.com/investors/annual-reports/reports/2014-annual-report-1/>. (20.11.2015)
- Flanagan, P. (2004). *Accounting for Software Development Costs: Applying SOP 98-1*. Christopher Technology Consulting: Virginia. www.christophertechnology.com/docs/SOP%2098-1%20+%20EITF%2000-2%20Application.pdf. (19.03.2015)
- Fois, S. ve R. Lysonick. (2012). *Analyzing the Global Software Industry*. Paris: SIA Partners.
- Forbes. (2014). *Global 2000 List - The World Biggest Public Companies*. http://www.forbes.com/global2000/list/#page:1_sort:0_direction:asc_search:filter:Software%20%26%20Programming_filter:All%20countries_filter:All%20states. (18.08.2014)
- Karel.* (2014). Finansal Tablolar ve Bağımsız Denetçi Raporu. İstanbul. <https://www.kap.gov.tr/bildirim-sorgulari/finansal-tablolar.aspx>. (22.11.2015)
- Link.* (2014). Finansal Tablolar ve Bağımsız Denetçi Raporu. İstanbul. <https://www.kap.gov.tr/bildirim-sorgulari/finansal-tablolar.aspx>. (22.11.2015)
- Logo.* (2014). Finansal Tablolar ve Bağımsız Denetçi Raporu. Kocaeli. <https://www.kap.gov.tr/bildirim-sorgulari/finansal-tablolar.aspx>. (22.11.2015)
- Microsoft.* (2014). Annual Report. Washington. <https://www.microsoft.com/investor/AnnualReports/default.aspx>. (20.11.2015)
- Mulford, W. C. ve J. Roberts. (2006). Capitalization of Software Development Costs: A Survey of Accounting Practices in the Software Industry. *Georgia Tech*,

- Financial Analysis Lab.* <https://smartech.gatech.edu/handle/1853/15598>. (10.05.2015)
- Netaş.* (2014). *Finansal Tablolar ve Bağımsız Denetçi Raporu.* İstanbul. <https://www.kap.gov.tr/bildirim-sorgulari/finansal-tablolar.aspx>. (22.11.2015)
- Price of Water House Coopers. (2009). *Software Revenue Recognition: A User-Friendly Guide for Navigating through the Many Complexities.* www.pwc.com/en_US/us/cfodirect/assets/pdf/accounting-guides/pwc-revenue-recognition-march-2009.pdf. (02.07.2015)
- Price of Water House Coopers. (2012). *Global Software 100 Leaders.* <http://www.pwc.com/gx/en/technology/publications/global-software-100-leaders/compare-results.jhtml>. (15.08.2014)
- SAP.* (2014). *Annual Report.* Walldorf. <http://www.sap.com/corporate-en/about/investors/newsandreports/financial-reports.html>. (20.11.2015)
- Tiryakioğlu, M. (2012). *Türkiye'de Yazılım Sektörü.* Ankara: Stratejik Düşünce Enstitüsü. http://www.sde.org.tr/userfiles/file/TURKIYEDE_YAZILIM_%20SEKTORU.pdf. (22.08.2014)
- TOBB Türkiye Telekomünikasyon Meclisi. (2012). *Telekomünikasyon Bilişim ve Yazılım Alanında Teşvikler ve Mali Destekler.* Ankara. <http://www.tobb.org.tr/Documents/yayinlar/2013/Telekom%C3%BCnikasyon%20Bili%C5%9Fim%20ve%20Yaz%C4%B1m%20Alan%C4%B1nda%20Te%C5%9Fvikler%20ve%20Mali%20Destekler.pdf>. (17.03.2015)
- Türkiye Bilişim Derneği.* (2013). *Değerlendirme Raporu.* Ankara. http://www.tbd.org.tr/usr_img/temp/2013_TBD_Degerlendirme_Raporu.pdf. (02.09.2014)
- Türkoğlu, Y. (2006). *Yazılım Sektörü Dış Pazar Araştırması.* Ankara: T.C Başbakanlık Dış Ticaret Müsteşarlığı İhracatı Geliştirme Etüd Merkezi.
- West, D. (2002). *Planning a Project with the Rational Unified Process.* California: Rational Software Development Company

Standartlar

- IFRS-15. (2015). *Revenue From Contracts With Customers*. International Accounting Standards Board. Londra.
- KOBİ-TFRS. (2013). *Küçük ve Orta Büyüklükteki İşletmeler İçin Türkiye Finansal Raporlama Standardı*. Kamu Gözetimi Kurumu. Ankara.
- SFAS-86. (1985). *Accounting For The Costs Of Computer Software To Be Sold, Leased, Or Otherwise Marketed*. Financial Accounting Standards Board. Connecticut.
- SOP-98-1. (1998). *Accounting for the Costs of Computer Software Developed or Obtained for Internal Use*. American Institute of Certified Public Accountants. New York.
- TMS-11. (2015). *İnşaat Sözleşmeleri*. Kamu Gözetimi Kurumu. Ankara.
- TMS-16. (2015). *Maddi Duran Varlıklar*. Kamu Gözetimi Kurumu. Ankara.
- TMS-18. (2015). *Hasılat*. Kamu Gözetimi Kurumu. Ankara.
- TMS-38. (2015). *Maddi Olmayan Duran Varlıklar*. Kamu Gözetimi Kurumu. Ankara.

Sözlükler

Bilişim Terimleri Sözlüğü. (2004). "Yazılım". Ankara: Asil Yayın.

<http://www.bilimsözlugu.net/lisans%20anla%C5%9Fmas%C4%B1>. (05.07.2015)

Türk Dil Kurumu Sözlüğü. (1981).

http://www.tdk.gov.tr/index.php?option=com_bts&arama=kelime&guid=TDK.GTS.53aad67984ae14.80290016. (25.06.2014)



İnternet Kaynakları

- Anık, S. (2014). Yazılım Sektörü AR-GE Teşviği ile Büyüyecek. *Bloomberht*.
<http://www.bloomberght.com/haberler/haber/1524307-yazilim-sektoru-ar-ge-tesvigi-ile-buyuyecek>. (22.09.2014)
- Beck, K. (2001). *Çevik Geliştirme İlkeleri*.
<http://www.agilemanifesto.org/iso/tr/principles.html>. (25.01.2015)
- Borandağ, E. (2012a). Yazılım Proje Yönetimi Ders Notları - Bölüm 1 - Yazılım Proje Yönetimine Giriş. *Maltepe Üniversitesi. Yazılım Mühendisliği*.
<http://slideplayer.biz.tr/slide/2463179/>. (07.10.2013)
- Borandağ, E. (2012b). Yazılım Proje Yönetimi Ders Notları - Bölüm 4 - Yazılım Büyüklük ve Emek Kestirimi. *Maltepe Üniversitesi. Yazılım Mühendisliği*.
<http://slideplayer.biz.tr/slide/2698438/>. (07.10.2013)
- Çankır, B. (2010). *Sabit Kıymetlerin Kayıtlardan Çıkarılmasında Özellikli Durumlar*.
<http://www.burhaneray.com/Sabit-Kiyemterin-Kayitlardan-Cikarilmasinda-Ozellikli-Durumlar/731> (10.04.2015)
- Gelirler İdaresi Başkanlığı. *Hesap Planı Açıklamaları*.
<http://www.gib.gov.tr/fileadmin/mevzuatek/eski/muhsisteb1ekmuh5c.htm>. (08.03.2015)
- Güneş, D. (2014). Yazılım Sektörü AR-GE Teşviği ile Büyüyecek. *Bloomberht*.
<http://www.bloomberght.com/haberler/haber/1524307-yazilim-sektoru-ar-ge-tesvigi-ile-buyuyecek>. (22.09.2014)
- Kansu, C. A. (2013). *Proje Yönetimi Bilgi Alanları*.
<http://canatufkansu.com/2013/02/05/proje-yonetimi-bilgi-alanlari/>. (11.02.2015)
- Özbilgin, İ. G. (2010). *Yazılım Geliştirme Süreçleri ve ISO 27001 Bilgi Güvenliği Yönetim Sistemi*. <https://www.bilgiguvenligi.gov.tr/yazilim-guvenligi/yazilim-gelistirme-surecleri-ve-iso-27001-bilgi-guvenligi-yonetim-sistemi.html>. (25.11.2014)
- Özerçen, E. (2012). *TMS 18 Nolu Standarda Göre Hasılatın Ölçümü ve Muhasebeleştirilmesi*.
<http://www.muhasibetr.com/yazarlarimiz/emreozercen/003/>. (23.07.2015)
- Putra, L. D. (2009). *Capitalization and Amortization of Software Cost*. <http://accounting-financial-tax.com/2009/05/capitalization-and-amortization-of-software-cost/>. (12.01.2015)
- Silvia, M. (2014). *IFRS 15 vs. IAS 18: Huge Change Is Here!*
<http://www.ifrsbox.com/ifrs-15-vs-ias-18/>. (06.08.2015)
- Stanford University. (2015). *Capitalization of Software Development Costs*. Gateway to Financial Activities.

http://web.stanford.edu/group/fms/fingate/staff/capitalequip/capital_software.html. (15.01.2015)

Temur, B. (2013). *Yazılım Geliştirme Süreç Modelleri Nelerdir ?*

<http://salyangoz.com.tr/blog/2013/11/02/digerleri/yazilim-gelistirme-surec-modelleri-nelerdir/>. (24.01.2015)

Yılmaz, G. (2007). Yazılım Mühendisliği Ders Notları Bölüm 6:Gerçekleştirim. *İstanbul Kültür Üniversitesi. Yazılım Mühendisliği*. <http://web.iku.edu.tr/~gyilmaz/>. (19.05.2015)

Yılmaz, G. (2007). Yazılım Mühendisliği Ders Notları Bölüm 7:Yazılım Doğrulama ve Geçerleme. *İstanbul Kültür Üniversitesi. Yazılım Mühendisliği*. <http://web.iku.edu.tr/~gyilmaz/>. (19.05.2015)

Yücalar, F. (2014). Yazılım Ölçütleri - Bölüm 5 - Yazılım Emek Ölçütleri ve Kestirim Modelleri. *Maltepe Üniversitesi. Yazılım Mühendisliği*. (07.10.2014)

<http://global.britannica.com/biography/John-Wilder-Tukey>. (27.06.2014)

<http://www.innova.com.tr/uygulama-gelistirme-yasam-dongusu.asp#Tab3-tab>
(30.01.2015)

<http://www.projeegitimmerkezi.com/>. (08.09.2014)

<http://www.statista.com/statistics/208652/global-packaged-software-revenue-since-2010/>.
(20.08.2014)

<http://webrazzi.com/2012/09/03/turkiye-it-sektoru-igucu-raporu/>. (25.08.2014)

ÖZGEÇMİŞ

1985 yılında İstanbul'da doğan İlker Calayoğlu; ilk ve orta eğitimini Hattat Rakım İlköğretim Okulu'nda okuyup, sonra İstanbul Anadolu Turizm ve Otelcilik Meslek Lisesi'nde eğitimine devam etmiştir. Mesleğinin devamı olarak Okan Üniversitesi, Uygulamalı Bilimler Yüksekokulu'nda Turizm ve Otelcilik Bölümü'nü burslu bitirerek lisans mezunu olmuştur.

Mesleki stajlarla birlikte 16 yaşında çalışma hayatına başlamıştır. Bir müddet çalıştıktan sonra lisans eğitiminin sonlarından itibaren seyahat acenteciliği ve uluslararası havayolu biletleme konularıyla ilgilenmeye başlamıştır. Bu eğitimlerden sonra kazandığı sertifikalar ile Okan Turizm ve Seyahat AŞ'de biletleme ve muhasebe pozisyonlarında başarı ile çalışmıştır.

Muhasebe ile fiili olarak tanışması neticesinde kariyerini bu alana doğru yönlendirmeye karar vermiştir. Bu sebeple lisans eğitimi biter bitmez kariyerinde yükselmek için yüksek lisansa başlamış ve 2011 yılında Okan Üniversitesi'nde Muhasebe ve Denetim Yüksek Lisansını Tezli olarak bitirmiştir.

Lisans mezuniyetinden sonra Okan Üniversitesi Uygulamalı Bilimler Yüksekokulu ile iletişimini kesmeyen yazar; yüksek lisansın dersleri sırasında Muhasebe ve Finansal Yönetim bölümünde Araştırma Görevlisi olarak 2009 yılında çalışmaya başlamıştır. 2013 yılında doktora yeterlilik aşamasını tamamlamasıyla birlikte, o günden bu güne Öğretim Görevlisi olarak çalışmayı sürdürmektedir.

Bilgisayarlı Muhasebe, Muhasebe Bilgi Sistemleri, İleri Excel Çalışmaları, E-Belge, E-Bildirge ve E-Beyanname, Bilgisayara Giriş, Muhasebeye Giriş, Dış Ticaret İşlemleri ve Muhasebe gibi dersler vermektedir.