



**T.C.
DÜZCE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**WEB VE MOBİL UYGULAMA TABANLI BAKIM-ONARIM VE
VARLIK YÖNETİM SİSTEMİNİN GELİŞTİRİLMESİ**

KENAN KOÇER

**YÜKSEK LİSANS TEZİ
ELEKTRİK ELEKTRONİK VE BİLGİSAYAR MÜHENDİSLİĞİ
ANABİLİM DALI**

**DANIŞMAN
Dr. Öğr. Üyesi SERDAR BİROĞUL**

DÜZCE, 2018

T.C.
DÜZCE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**WEB VE MOBİL UYGULAMA TABANLI BAKIM-ONARIM VE
VARLIK YÖNETİM SİSTEMİNİN GELİŞTİRİLMESİ**

Kenan Koçer tarafından hazırlanan tez çalışması aşağıdaki jüri tarafından Düzce Üniversitesi Fen Bilimleri Enstitüsü Elektrik Elektronik ve Bilgisayar Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Dr. Öğr. Ü. Serdar BİROĞUL

Düzce Üniversitesi

Jüri Üyeleri

Dr. Öğr. Ü. Serdar BİROĞUL

Düzce Üniversitesi

Dr. Öğr. Ü. Sinan TOKLU

Düzce Üniversitesi

Doç. Dr. Aydın ÇETİN

Gazi Üniversitesi

Tez Savunma Tarihi: 17/04/2018

BEYAN

Bu tez çalışmasının kendi çalışmam olduğunu, tezin planlanmasından yazımına kadar bütün aşamalarda etik dışı davranışımın olmadığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara kaynak gösterdiğimi ve bu kaynakları da kaynaklar listesine aldığımı, yine bu tezin çalışılması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını beyan ederim.

17 Nisan 2018

Kenan Koçer

TEŞEKKÜR

Yüksek lisans öğrenimimde ve bu tezin hazırlanmasında gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocam Dr. Öğr. Ü. Serdar Biroğul'a en içten dileklerimle teşekkür ederim.

Bu çalışma boyunca yardımlarını ve desteklerini esirgemeyen sevgili eşim ve kızıma, maddi manevi desteğini esirgemeyen şirketim Bimser Çözüm A.Ş. ailesine sonsuz teşekkürlerimi sunarım.

17 Nisan 2018

Kenan Koçer

İÇİNDEKİLER

	<u>Sayfa No</u>
ŞEKİL LİSTESİ	IX
ÇİZELGE LİSTESİ	XI
KISALTMALAR.....	XII
SİMGELER	XIII
ÖZET	XIV
ABSTRACT	XV
1. GİRİŞ.....	1
2. BAKIM YÖNETİMİ.....	3
2.1. BAKIM YÖNETİMİ KAVRAMLAR	3
2.1.1. Bakım Yöntemlerinin Karşılaştırılması.....	4
2.2. BAKIM SİSTEMİNDE BİLGİSAYARIN ROLÜ	5
3. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİ	6
3.1. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİNİN ÖZELLİKLERİ....	8
3.2. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİNİN İŞLETMELERE SAĞLADIĞI FAYDALAR	10
3.3. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİNİN SİSTEM MİMARİSİ	11
3.3.1. Rest Mimarisi ve Restful Servisler	11
3.3.2. Sistem Kapsamı.....	11
3.3.2.1. Boysweb'in Altyapı Katmanı	12
3.3.2.2. Boysweb'in Orta Katmanı	13
3.3.2.3. Boysweb'in İstemci Katmanı	14
3.3.3. Boysweb' in Kullanıcı Arayüzü	14
3.3.4. Boysweb' in Konfigürasyonu	14
3.3.5. Boysweb'in Mimari Avantajları ve Kısıtları.....	15

3.4. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİ KATMANLARI.....	16
3.5. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİ DİAGRAMLARI	17
3.5.1. İş Talebi	17
3.5.2. İş Emri Veritabanı Diagramı.....	17
3.6. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİ DAĞILIM DİAGRAMI	18
3.7. USE-CASE DİAGRAMI	19
3.7.1. İş Talebi Diagramı	19
3.7.2. İş Emri Diagramı	21
3.8. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİNİN ANA MODÜLLERİ	27
3.8.1. Varlık Yönetim Sistemi	277
3.8.2. Bakım Yönetim Sistemi.....	28
3.8.3. İş Talepleri.....	29
3.8.4. İş Emirleri.....	30
3.8.5. İş Talimatları.....	30
3.8.6. Planlı Bakım-İş Adımları	30
3.8.7. Bakım Planları	30
3.8.8. Koruyucu Bakım Tanımları	30
4. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİ YAZILIM GELİŞTİRMELERİ	31
4.1. BOYSWEB ÖNBELLEKLEME YAKLAŞIMLARI.....	31
4.1.1. AppFabric Nedir?	31
4.1.2. Boysweb Neden AppFabric'e İhtiyaç Duyar.....	311
4.1.3. Redis Nedir, Redis ve AppFabric Farkları.....	355
4.1.4. Redis Kurulumu ve Konfigurasyonu	35
4.1.5. Önbellekleme İşlemleri Sonucu Kazanımlar.....	39
4.2. PERİYODİK(GÜNLÜK) KONTROL MODÜLÜNÜN GELİŞTİRİLMESİ	40
4.2.1. Günlük Kontrol Şablonu.....	42
4.2.2. Günlük Kontrol Formu	43
4.2.3. Periyodik Kontrol Modülü Değerler, Grafikler ve Kazanımlar	48
4.2.3.1. DevExpress Nedir	48

4.2.3.2. <i>Günlük Kontrol Grafikler</i>	49
4.3. MALZEME SAYIMI MODÜLÜNÜN GELİŞTİRİLMESİ	51
4.3.1. <i>Malzeme Sayımı Modülü Geliştirmeler</i>	52
4.3.2. <i>Mobil Uygulama Üzerinden Malzeme Sayımı Verilerinin Gönderimi</i> ..	57
4.3.3. <i>Modülün eklenmesiyle gerçekleşen kazanımlar</i>	58
4.4. BAKIM TALİMATI TARİHÇE VE ONAY SİSTEMİ GELİŞTİRİLMESİ	58
4.4.1. <i>Bakım Talimatı Tarihçesi</i>	59
4.4.2. <i>Bakım Talimatı Onay Sistemi</i>	60
4.4.3. <i>Bakım Talimatı Akış Yapısı</i>	62
4.4.4. <i>Bakım Talimatı Geliştirmeleri Sonucunda Elde Edilen Kazanımlar</i>	64
4.5. GRAFİKSEL VARLIK AĞACI MODÜLÜNÜN GELİŞTİRİLMESİ	64
4.5.1. <i>Grafiksel Varlık Ağacı Modülü Geliştirilmesi Sonucu Kazanımlar</i>	66
4.6. İŞ EMRİ BAKIM DURUMLARI SEÇİMİ GELİŞTİRMESİ	67
4.7. VARLIK KORDİNAT BİLGİLERİ GELİŞTİRMESİ	69
4.8. VARLIK TAŞIMA MODÜLÜNÜN GELİŞTİRİLMESİ	70
4.8.1. <i>Varlık Taşıma Modülünün Geliştirilmesi Sonucu Elde Edilen Kazanımlar</i>	74
4.9. İŞ EMRİ FORMLARININ MOBİL ÜZERİNDEN ALINMASI	74
4.10. ELEKTRONİK İMZA UYGULAMASININ ENTEGRE EDİLMESİ	75
4.10.1. <i>Elektronik İmza ve Sayısal İmza</i>	75
4.10.2. <i>Elektronik İmza Prosedürü</i>	75
4.10.3. <i>Elektronik İmza Altyapısı</i>	76
4.10.3.1. <i>Kriptografi</i>	77
4.10.3.2. <i>Açık Anahtar Şifreleme</i>	78
4.10.3.3. <i>Güvenli Elektronik İmza Oluşturma ve Doğrulama Araçları</i>	78
4.10.4. <i>Elektronik İmza Uygulamasının Boysweb Üzerine Eklenmesi</i>	79
4.10.5. <i>Elektronik İmza Uygulaması Kazanımları</i>	83
4.11. GELİŞTİRİLEN UYGULAMANIN KULLANIMI	84
4.12. GELİŞTİRİLEN UYGULAMANIN TEST DURUMLARI	91
4.12.1. <i>Periyodik(Günlük) Kontrol Modülü Test Senaryosu</i>	91
4.12.2. <i>Bakım Talimatı Geliştirmesi Test Senaryosu</i>	97
4.12.3. <i>Malzeme Hareketi Test Senaryosu</i>	99
4.12.4. <i>Varlık Taşıma Test Senaryosu</i>	102

4.12.5. Birim Test Metodları	104
5. SONUÇLAR VE ÖNERİLER	112
6. KAYNAKLAR	117
ÖZGEÇMİŞ	119



ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1. Bakım Modeli Şeması.	3
Şekil 2.2. Bakım Yönetimi Yaklaşımları.	4
Şekil 3.1. Boysweb’ in Sistem Mimarisi.	12
Şekil 3.2. Boysweb’ in Katmanları	16
Şekil 3.3. İş Talebi Veritabanı Diagramı.	17
Şekil 3.4. İş Emri Veritabanı Diagramı.	18
Şekil 3.5. Boysweb Dağılım Diagramı.	19
Şekil 3.6. Boysweb İş Talebi Use – Case Diagramı.	20
Şekil 3.7. Boysweb İş Emri Use – Case Diagramı.	25
Şekil 3.8. Boysweb İş Emri Kayıt Aşamasını Gösterir Use – Case Diagramı.	26
Şekil 3.9. Kapatılan ve ya İptal Edilen İş Emri Use – Case Diagramı.	27
Şekil 3.10. Boysweb Varlık Ağacı Gösterimi.	28
Şekil 3.11. Bakım Yönetim Sistemi Modülleri.	29
Şekil 4.1. Boysweb AppFabric Uygulama Şeması.	32
Şekil 4.2. AppFabric Kurulumu Sonrası Kurulması Gereken Windows Servisi.	32
Şekil 4.3. AppFabric Kurulumu Sonrası Konfigurasyon Ayarlarının Yapılacağı Klasör.	33
Şekil 4.4. Boysweb Uygulaması AppFabric Konfigurasyonu Terminali.	34
Şekil 4.5. Redis Kurulum Konfigurasyon Dosyası.	36
Şekil 4.6. Redis Masaüstü Uygulaması Önbelleğe Alınan Verinin Gösterilmesi.	37
Şekil 4.7. AppFabric – Redis Tenolojisi Alt Yapısı İle Geliştirilen Boysweb Uygulaması İlk Yükleme Süresi Karşılaştırması.	38
Şekil 4.8. AppFabric – Redis Uygulamaları Önbelleklenmiş Veri Karşılaştırması.	38
Şekil 4.9. Günlük Kontrol Modülü.	41
Şekil 4.10. Günlük Kontrol Şablonu.	42
Şekil 4.11. Günlük Kontrol Şablonunun Doldurulması.	42
Şekil 4.12. Günlük Kontrol Şablonu Veritabanı İlişkileri Şeması.	43
Şekil 4.13. Günlük Bakım Kontrol Formu Ana Ekranı.	44
Şekil 4.14. Kontrol Formu İş Adımları Listesi.	44
Şekil 4.15. Kontrol Formu Ölçüm Paketleri Listesi.	45
Şekil 4.16. Günlük Kontrol Servisi Ayar Ekranı.	46
Şekil 4.17. Servisten Otomatik Tetiklenen Günlük Kontrol Formları.	46
Şekil 4.18. Otomatik Oluşan Günlük Kontrol Formlarının Logları.	47
Şekil 4.19. Günlük Kontrol Formu Veritabanı İlişkileri Şeması.	48
Şekil 4.20. Periyodik Bakım Gerçekleşme Raporu Tasarım Ekranı.	49
Şekil 4.21. Sarfyeri Bazında Kontrol Formlarının En Çoklar Analizi.	50
Şekil 4.22. Sarfyeri Bazında Kontrol Formlarının Grafiği.	51
Şekil 4.23. Mobil Servis Malzeme Sayımı Sınıf Yapısı.	53
Şekil 4.24. Mobil Uygulama Malzeme Sayımı Sınıf Yapısı.	53
Şekil 4.25. Mobil Uygulama Liste Ekranı Sayfa Dizaynı.	54

Şekil 4.26. Malzeme Sayımı Modülü Fonksiyonları.	55
Şekil 4.27. Boysweb Malzeme Sayımı Barkod Okuyucudan Malzeme Seçimi.	56
Şekil 4.28. Boysweb Uygulaması Malzeme Sayımı Ana Ekranı.	57
Şekil 4.29. Mobil Uygulama JsonString Bilgisi.	57
Şekil 4.30. Restfull Servis JsonString Bilgisi.	58
Şekil 4.31. Talimat - Tarihçe Veritabanı İlişkisi.	60
Şekil 4.32. Bakım Talimatı Tarihçe Sayfası.	60
Şekil 4.33. Bakım Talimatı Onay Bilgileri.	61
Şekil 4.34. Bakım Talimatı Akış Tasarımı.	63
Şekil 4.35. Bakım Talimatı Onay Aşamaları.	64
Şekil 4.36. Grafiksel Varlık Ağacı Dizayn.	65
Şekil 4.37. Grafiksel Varlık Ağacı İzleme Modu.	66
Şekil 4.38. İş Emri Bakım Durumu Seçim İşlemi.	67
Şekil 4.39. İş Emri Bakım Durumu Özelliğinin Mobil Versiyona Uyarlanması.	68
Şekil 4.40. Varlık Navigasyon Bilgisinin Mobil Ekranda Gösterilmesi.	69
Şekil 4.41. Varlık Kordinat Bilgileri İlişkisel Veritabanı	70
Şekil 4.42. Varlık Taşıma Ekranları.	71
Şekil 4.43. Varlık Taşıma Fonksiyonunun Mobil Uygulama Üzerinde Geliştirilmesi. .	72
Şekil 4.44. Varlık Taşıma Raporunun Dizayn Edilmesi.	73
Şekil 4.45. Mobil İş Emri Form Yazdırma Özelliği.	74
Şekil 4.46. Elektronik İmza Algoritması.	77
Şekil 4.47. Elektronik İmza Doğrulama Aracı.	78
Şekil 4.48. Elektronik İmza Doğrulama Tipi.	79
Şekil 4.49. Boysweb Uygulaması İçerisine Elektronik İmzanın Eklenmesi.	80
Şekil 4.50. Boysweb Uygulaması İmzalama Arayüzü.	81
Şekil 4.51. Boysweb Satınalma Talep Formunun İmzalanması.	82
Şekil 4.52. Elektronik İmzası Atılan Dosyanın İmzalama Durumunun Gösterilmesi. ...	83
Şekil 4.53. Periyodik(Günlük) Kontrol Şablon Ekranı.	84
Şekil 4.54. Periyodik(Günlük) Kontrol Form Ekranı.	85
Şekil 4.55. Malzeme Sayımı Modülü Ana Ekranı.	85
Şekil 4.56. Malzeme Sayımı Modülü Sayım Ekranı.	86
Şekil 4.57. Malzeme Sayımı Modülü Liste Ekranı.	86
Şekil 4.58. Bakım Talimatı Ekranı.	87
Şekil 4.59. Bakım Talimatı Onay Bilgileri.	87
Şekil 4.60. Grafiksel Varlık Ağacı Çizimi.	88
Şekil 4.61. Grafiksel Varlık Ağacı Fonksiyonları.	89
Şekil 4.62. Mobil Uygulama İş Emri Listesi.	89
Şekil 4.63. Satınalma Talep Formu Elektronik İmza Ekranı.	90
Şekil 4.64. Elektronik İmzası Atılan Dosyanın İmzalama Durumunun Gösterilmesi. ...	91
Şekil 4.65. Birim Test Proje Yapısı	104

ÇİZELGE LİSTESİ

	<u>Sayfa No</u>
Çizelge 3.1. Minimum BOYSWEB Test ve Canlı Sistem Gereksinimi.....	15
Çizelge 4.1. Günlük Kontrol Şablonu Oluşturma Test Senaryosu	91
Çizelge 4.2. Günlük Kontrol Formu Oluşturma Test Senaryosu.....	94
Çizelge 4.3. Günlük Kontrol Formu Kapatma Test Senaryosu	94
Çizelge 4.4. Bakım Talimatı Oluşturma Test Senaryosu.....	97
Çizelge 4.5. Malzeme Giriş Hareketi Oluşturma Test Senaryosu	99
Çizelge 4.6. Malzeme Çıkış Hareketi Oluşturma Test Senaryosu.....	101
Çizelge 4.7. Varlık Taşıma Test Senaryosu.....	103
Çizelge 4.8. Birim Test Metodları	105

KISALTMALAR

BOYS	Bakım Onarım Yönetim Sistemi
BOYSWEB	Bakım Onarım Varlık Yönetim Sistemi
CADES	CMS Gelişmiş Elektronik İmza
FTP	Dosya Transfer Protokolü
HTTP	Hiper Metin Aktarım Protokolü
MTTR	Onarıma Kadar Geçen Ortalama Süre
MTBF	Ardışık İki Arıza Arasında Beklenen Ortalama Süre
PADES	PDF Gelişmiş Elektronik İmza
XADES	XML Gelişmiş Elektronik İmza

SİMGELER

Dk

Dakika



ÖZET

WEB VE MOBİL UYGULAMA TABANLI BAKIM-ONARIM VE VARLIK YÖNETİM SİSTEMİNİN GELİŞTİRİLMESİ

Kenan KOÇER

Düzce Üniversitesi

Fen Bilimleri Enstitüsü, Elektrik Elektronik ve Bilgisayar Mühendisliği Anabilim Dalı
Yüksek Lisans Tezi

Danışman: Dr. Öğr. Üyesi Serdar Biroğul

Nisan 2018, 118 sayfa

İşletmeler, bakım-onarım çalışmalarını takip edebilmek için bilgi işlem tabanlı sistemlere ihtiyaç duymaktadır. Bakım-onarım sisteminin sorunsuz çalışması için kaydedilen verilerin ve bu verileri elde edebilmek için yapılan işlemlerin saklanması gerekir. İşletme yöneticileri ellerindeki kayıtlı verilerle, sistemlerden çekecekleri grafikler ve raporlar neticesinde işletmelerin bakım-onarım durumlarını takip edebilmektedirler. İşletmeler sahip oldukları sistemlerine yeni özellikler, modüller, grafikler, sayfalar, fonksiyonlar kazandırarak sistemlerinin daha kullanışlı, anlaşılır ve hızlı olmasını istemektedirler. Bakım-onarım sistemleri hakkında yapılan önceki araştırmalar bakım-onarım sistemlerinin işletmelere nasıl entegre edildiği, diğer sistemler ile bakım-onarım sistemlerinin nasıl birlikte çalışabileceği konularına odaklanmaktadır. Bu araştırmalarda bakım-onarım sistemlerinin nasıl geliştirildiği, alt yapısının ne olduğu, hangi teknolojiler kullanıldığı bilgilerine yer verilmemektedir. Bizim çalışmamızda üniversite sanayi işbirliği kapsamında varolan bir bakım-onarım sistemine yeni özellikleri eklenirken yazılım geliştirme süreçlerinin nasıl işletildiği, hangi teknolojilerin kullanıldığı, test adımlarının neler olduğu bilgilerine yer verilmektedir. Ayrıca çalışmamızda eklenen yeni özellikler sonucunda işletmelerin hangi kazanımlara sahip oldukları bilgisine değinilmektedir. Bu tez çalışmasında temel amacımız kullanılmakta olan bakım-onarım sisteminin işletme ihtiyaçlarını karşılayacak şekilde revize edilmesi, elektronik imza altyapısına geçilmesi, mobil uygulamasının geliştirilmesi ve bu geliştirmeler sırasında yazılım hayat döngüsünün işletilmesi, testlerinin gerçekleştirilmesi, geri bildirimlerin değerlendirilmesidir.

Anahtar sözcükler: Bakım-onarım sistemi, Elektronik imza, Mobil uygulama, Test, Yazılım geliştirme süreçleri.

ABSTRACT

DEVELOPMENT OF MAINTENANCE-REPAIR AND ASSET MANAGEMENT SYSTEM BASED ON WEB AND MOBILE APPLICATION

Kenan KOÇER

Düzce University

Graduate School of Natural and Applied Sciences, Department of Electrical Electronic
and Computer Engineering

Master's Thesis

Supervisor: Asist. Prof. Dr. Serdar BİROĞUL

April 2018, 118 pages

Enterprises (Operations) need IT-based systems in order to carry out repair and maintenance works. To ensure the proper functioning of the repair and maintenance system, it is necessary to obtain and store data on the operations that are carried out. Managers of the enterprise can track the maintenance and repair status of the enterprises on the basis of the recorded data on their hand, graphics and reports to be drawn from the systems. Enterprises want their systems to be more useful, understandable and faster by giving new features, modules, graphics, pages, functions to their systems. Previous research on maintenance-repair systems focuses on how maintenance-repair systems are integrated into operations, how other systems and maintenance-repair systems can work together. In these surveys, there is no information on how maintenance-repair systems are developed, what the infrastructure is, what technologies are used. In our study, when new features are added to a maintenance-repair system existing within the scope of university industry cooperation, information about how software development processes are used, which technologies are used, and test steps are given. In addition, the new features added to our work call for information about what benefits they have. The main aim of this thesis is to revise the maintenance-repair system in order to meet the needs of the business, to apply to the electronic signature infrastructure, to develop the mobile application and to operate the software life cycle, to carry out the tests and to evaluate the feedbacks.

Keywords: Electronic signature, Maintenance-repair system, Mobile application, Software development processes, Testing.

1. GİRİŞ

İşletmelerin verimliliklerini yükseltmelerinin arkasında bakım-onarım sistemleri yer almaktadır. Bakım-onarım sistemleri aracılığı ile işletmeler varlıklarını yönetebilmekte, varlıklarına ait planlı ve plansız bakımlarını düzenli olarak gerçekleştirebilmektedirler. Personel yönetimlerini ve personel verimliliklerini bakım-onarım sistemleri üzerinden takip edebilmektedirler. Bakımlar için kullanılan malzemelerinin yönetimlerini ve ambar durumlarını bakım-onarım sistemi üzerinden yönetebilmektedirler. Kullanılan malzemelerinin satınalma işlemlerini yönetebilmeleri için yine bakım-onarım sistemine ihtiyaç duyulmaktadır. İşletmeler bakım-onarım sistemi üzerinden yönetilen durumlar sonucunda elde edilen işlenmiş verilerin dönüşlerini grafikler, raporlar, dosyalar halinde sistemden çekebilmektedirler.

Bakım-onarım sistemleriyle alakalı yapılan benzer çalışmalarda bakım-onarım sistemlerinin işletmelere nasıl uygulandığı, uygulama sürecinde hangi aşamalardan geçildiği ve bakım-onarım sisteminin işletmeye nasıl katkı sağlandığı üzerinde durulmuştur. Ayrıca bazı araştırmalarda bakım-onarım sistemi ile diğer sistemlerin nasıl entegre edilebileceği ve birbirleriyle veri alışverişini nasıl sağlayabileceği araştırılmıştır.

Araştırmamızda hali hazırda ticari olarak kullanılan bakım-onarım sisteminin daha kullanışlı hale getirilmesi için gerekli olan ve diğer bakım-onarım sistemlerinde olmayan işletmenin bakım süreçlerinin tamamını kapsayabilmesini sağlayacak yeni fonksiyonların ve modüllerin bakım-onarım sistemine kazandırılması süreci üzerinde durulmuştur. Bu süreç kapsamında üniversite, sanayi işbirliği içerisinde ürünü kullanan kullanıcılardan gereksinimler toplanarak analizler gerçekleştirilmiştir. Yazılım geliştirme sürecine geçildiğinde yazılım geliştirme süreci yaşam döngüsü irdelenmiş, geliştirme sırasında ve sonrasında test işlemleri gerçekleştirilmiş, test senaryoları tanımlanmış ve birim testler yapılmıştır.

Yapılan çalışmalar sonucunda bakım-onarım sistemi önbellekleme işlemini gerçekleştirebilmek için güncel ve popüler olan

ak kullanılan iki uygulama, bakım-onarım sistemi altyapısına eklenip test edilmiştir. Hangi önbellekleme yaklaşımı daha hızlı, kullanışlı ve işlevsel ise o uygulama bakım-onarım sistemi altına entegre edilmiştir.

İşletmeler üretim için kullandıkları ellerinde varolan malzemelerin yönetimini gerçekleştirebilmek için belirli periyotlarla malzeme sayımı yapmaktadırlar. Geliştirilen malzeme sayımı modülü sayesinde işletmeler mobil cihazları ile malzemelerini anlık sayarak güncel ambar miktarlarını sistemde tutabilmektedirler.

Bakımdan sorumlu olan son kullanıcılar işletmelerinin krokisini tek bir ekrandan görmek istemektedirler. Hangi noktada problem varsa oraya direkt müdahale etmek istemektedirler. Grafiksel varlık ağacı modülünün geliştirilmesi sayesinde işletmeler tek ekrandan varlıklarının bakımlarını kontrol edebilmektedirler.

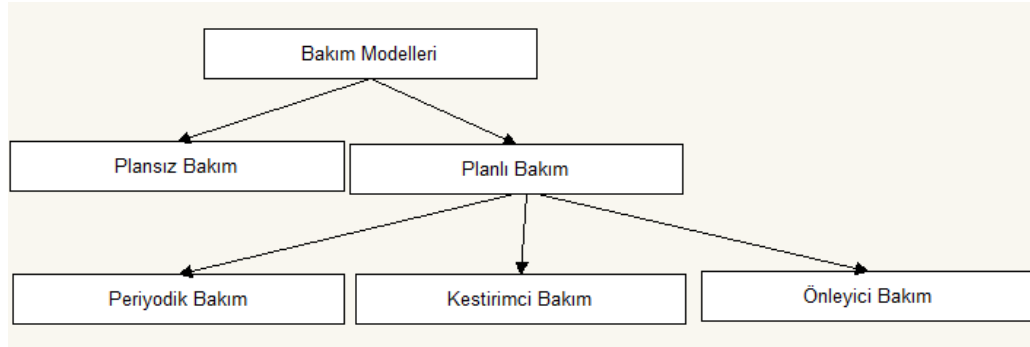
İşletmeler kullandıkları malzemelerinin satınalma işlemlerini sistem üzerinden gerçekleştirmektedirler. Gerçekleştirilecek satınalma talepleri için formlar oluşturulmakta ve satınalma işlemi gerçekleştirilebilmesi için işletme yöneticilerin onayları ile satınalma işlemi tamamlanmaktadır. Tamamlanan satınalma işlemi sonrası satınalma formlarının çıktıları alınmakta ve bu formlar ıslak imza ile imzalanarak dökümanlar fiziksel olarak saklanmaktadır. Sistemin tamamen sayısal ortama geçmesi ve fiziksel olarak form saklama işleminin önüne geçilmesi için bakım-onarım sistemine elektronik imza eklenmiştir. İmzalanması gereken dosyaların sanal ortamda imzalanarak saklanması sağlanmıştır. Excel formlarda bakımlarını takip eden şirketlerin daha kolay ve kullanılabilir bir ortama taşınması için bakım-onarım sistemine geçerek bakımlarında hız, güvenlik ve kalite sağlamalarının yanında aynı zamanda elektronik imza uygulaması eklenerek kağıtsız evrak yönetimi sağlamışlardır.

Araştırmamızda bakım-onarım sisteminin ne olduğu ve nasıl kullanıldığı bilgilerinin aktarılması yerine diğer araştırmalardan farklı olarak kullanılan bir sistemin nasıl iyileştirilebileceği, iyileştirme kapsamında hangi teknolojilerin kullanıldığı, yazılım yaşam döngüsünün nasıl işletildiği, test işlemlerinin nasıl gerçekleştirildiği, geliştirme tamamlandıktan sonra kullanıcılara bu geliştirmelerin nasıl yarar sağlandığı bilgilerine yer verilmiştir.

2. BAKIM YÖNETİMİ

2.1. BAKIM YÖNETİMİ KAVRAMLAR

Bakım, bir parçanın istenilen ömür çevrimi içerisinde, parçanın gerekli fonksiyonlarını yerine getirebilecek şekilde koruması veya muhafaza edilmesi için bütün teknik eylemlerin bir bütünü olarak tanımlanabilir [1]. Bakım yönetimi, bakım önceliklerini, stratejilerini, planlamasını, kontrolünü, denetlenmesini ve organizasyonda ekonomik açıdan iyileştirmeleri içeren metotlar gibi sorumluluk ve uygulamalar ile ilgili bütün yönetim aktivitelerinin belirlenmesi olarak tanımlanabilir [2]. Yetersiz bakım yönetimi sonunda makina arızaları meydana gelir. Malzeme yönetilemez. İşletmede üretim durabilir. Üretimde meydana gelen gecikme veya iptaller ancak bakımın programlı ve kurallara uygun yapılmasıyla önlenabilmektedir [3].



Şekil 2.1. Bakım modeli şeması [4].

Şekil 2.1’de bakım modeli şeması gösterilmiştir. Bakım modeli planlı ve plansız bakım olarak ikiye ayrılır. Plansız bakım, işletmelerde arıza çıktıkça yapılan bakım ve onarım şeklindedir. Bu nedenle, onarım sırasında üretim kaybı fazla olmaktadır. Planlı bakım, kontrollü ve düzenli bir biçimde yapılan ve içinde her türlü bakım çeşidini içeren bir bakım çeşididir.

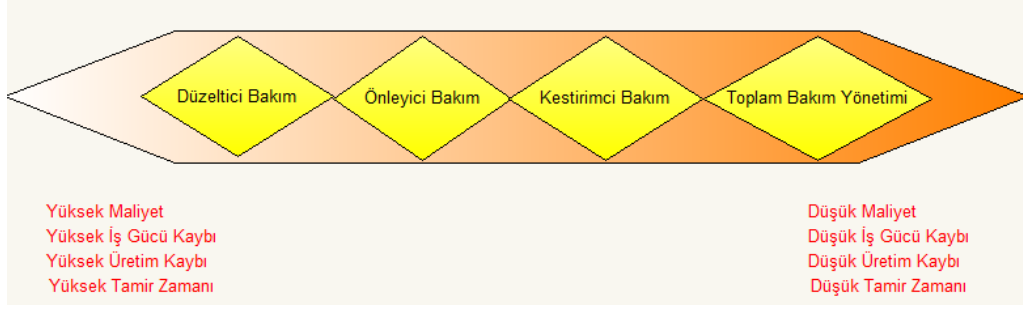
Arızı (Plansız) bakım, ekipmanlara işlevlerini yitirinceye, bozuluncaya kadar bakım yapılmamasını baz almaktadır. En ilkel bakım sistemidir. Bakım, ekipman arızalandıktan sonra yapılır. Periyodik bakım, ekipmanların işlevlerini yerine getirdikleri süre içerisinde

önceden belirlenmiş zaman periyodlarında bakım işlemlerinin yapılmasıdır. Önleyici bakım, arızaya neden olabilecek temel faktörlerin ortadan kaldırılarak makinenin çalışmaya dayanıklı hale getirilmesidir. Problemlerin önceden tanınması ve giderilmesi işlemine önleyici bakım denir [5]. Kestirici bakım yönteminde, ekipmanların çalışma koşulları ve karakterleri göz önünde bulundurulur. Yapılan program çerçevesinde, üretimi durdurmadan bazı parametrelerin kontrolleri ve ölçümleri yapılmaktadır. Sonuçlar dahilinde ekipmanda oluşabilecek arızaların gelişimi incelenir. Hata belirlenirse gerekli yedek parça temin edilir, üretim durdurulur, kısa süre içerisinde bakım yapılır ve tekrar üretime devam edilir [6].

Toplam verimli bakım (TVB) tüm çalışanların katılımının ön görüldüğü, küçük grup faaliyetleri aracılığı ile gerçekleşen verimli bakım olarak tanımlanabilir. İşletmede üretim faaliyetleri içerisinde çalışan kaynak ve ekipmanların tamamının katılımını gerektirir. Operatörlere üzerlerinde çalıştıkları ekipmanların otonom bakım sorumluluğunu getirir. Arızaları önleyen, ekipman verimliliğini arttıran bir yaklaşımdır. Japon Fabrika Bakım Enstitüsü (JIMP) tarafından geliştirilen, ekipmanların sıfır arıza ve minimum üretim kayıplarına sahip olmalarını hedefleyen bir kavramdır [7].

2.1.1. Bakım Yöntemlerinin Karşılaştırılması

Bakım yöntemleri arasında en yüksek maliyet, işgücü, üretim kaybı ve tamir zamanına sahip olanı düzeltici bakımdır. Kestirimci bakım planlaması, makinalar üzerinden, periyodik aralıklar ile alınan, fiziksel parametre ölçümlerinin zaman içindeki eğilimlerini izleyerek, makina sağlığı hakkında geleceğe yönelik bir kestirimde bulunma yöntemidir [8]. Toplam verimli bakım ise en etkin ve gelişmiş bakım yaklaşımıdır. Toplam verimli bakım, çalışanların bilgi ve becerilerinin arttırılması, kullanılan ekipmanların en iyi şekilde korunması, tüm bakım faaliyetlerinin bilgisayar ortamında takip edilmesi ve gerekli önlemlerin alınmasıyla sıfır kaza, sıfır hata ve sıfır plansız duruşu amaçlayan bir yönetim sistemidir [9]. Şekil 2.2’de bakım yönetimi yaklaşımları gösterilmiştir.



Şekil 2.2. Bakım yönetimi yaklaşımları.

2.2. BAKIM SİSTEMİNDE BİLGİSAYARIN ROLÜ

Bakım yönetim sistemi iş gücünü, talepleri, ekipman tarihçelerini, ekipman listesini ve ağacını, ekipman parçalarının listesini ve lokasyon bilgisini, mevcut malzeme miktarı ve ambar durumlarını, koruyucu bakım zaman ve sayaç planlarını, gider ve maliyet miktarlarını tutmalıdır.

Bakım sistemi yazılımında ;

- Şirket içerisinde benimsenmiş kodlama yapısı ve hiyerarşisine uyum sağlayabilmesi gerekir.
 - İstenen formatlarda rapor üretilmesi ve rapor formatının zaman içerisinde geliştirilmesi sağlanabilmelidir.
 - Şirket ile birlikte yazılımda genişleyebilmeli ve esnek olabilmelidir.
 - Program taşınabilir cihazlarda da kullanılabilmelidir.
 - Farklı veritabanları ve sunucularla optimize edilebilmelidir.
 - Veri girişlerinde ve programın kullanımının sürdürülmesinde iş gücü az olmalı, zaman kaybı olmamalıdır.
 - Farklı işletim sistemlerinde çalışabilmelidir.
 - Ekipman veya arıza geçmişi kayıtları tutulabilmelidir.
 - Bakım programında bakım işleri zaman planlarının oluşturulması sağlanmalıdır.
- Personel görevlendirmesi, malzeme stok kontrolü yapılmalıdır. Maliyet takibi, arızaların kök nedenlerinin takibi, devre dışı kalma süresi, geleceğe yönelik iyileştirme faaliyetleri hesaplanabilmelidir.

3. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİ

Bakım–onarım varlık yönetim sistemi, firmalarda var olan ekipman ve makinaların bakım süreçlerinin planlı ve sağlıklı bir şekilde tutulmasını ve yapılmasını sağlar. Bu işler yapılırken harcanan malzeme ve yedek parçaların stok kontrollerinin yapılmasını sağlar. Bakımı yapan personel ve kaynakların çalışma bilgilerini ve işçilik süresi ve işçilik maliyetlerini tutar. Miktarı azalan yedek parça ve malzemelerin satın alma yönetimini ve satın alma süreçlerini tutar.

Önceden ve hali hazırda ticari olarak kullanılan BOYSWEB sisteminin gereksinimleri karşılama yetersiz kalmasından dolayı bu çalışma paralelinde yenilenmiştir. Müşteri istekleri doğrultusunda yeni geliştirmeler yapılmıştır. Bu geliştirmeler kapsamında yeni modüller ve fonksiyonlar eklenmiştir. Kullanıcı dostu ve daha fazla kullanıcı ihtiyacına cevap verebilecek hale getirilmiştir.

Yapılan çalışmalar sonucunda Boysweb uygulamasında önbellekleme işlemini gerçekleştirebilmek için güncel ve popüler olarak kullanılan iki uygulama, Boysweb altyapısına eklenip konfigure edilerek test edilmiştir. AppFabric ve Redis uygulamalarından hangisinin daha az maliyetli olacağı, hangisinin daha hızlı ve programı yormayacak şekilde önbellekleme işlemini gerçekleştirebileceğini test edebilmek için önce AppFabric uygulaması Boys uygulamasına dahil edilmiş hız ve kabiliyetleri gözden geçirilmiş daha sonra Redis uygulaması Boys uygulamasına dahil edilmiş, hız ve kabiliyetleri gözlemlenmiştir. Redis uygulaması Appfabric uygulamasına oranla verileri daha hızlı önbelleğe almakta ve daha hızlı uygulamaya tekrar geri dönmektedir.

İşletmelerdeki bazı ekipmanların arızı bakım yapılmasına gerek kalmadan, duruş yaşanmadan önce bakımlarının günlük olarak yapılması gerekir. Bu ekipmanların kontrolü hayati önem taşıyabilmektedir. Bu ekipmanların bozulması ekipman hattını tamamen duruşa geçirebilmektedir. Tamiri uzun sürmekte yedek parça ve işçilik maliyeti ekipman maliyetine yaklaşmaktadır. Duruş süresinde meydana gelen duruş kaybı maliyetinin telafisi zor olmaktadır. Bu ekipmanların bakımlarının periyodik olarak yapılmaması durumunda oluşacak arıza, çalışan personellerin hayatını ve çalışma

güvenliğini riske atmaktadır. İşletmeler bu olumsuzlukların önüne geçebilmek için bakımlarının günlük olarak oluşmasını ve bakım personellerinin önüne düşmesini istemektedirler. Bu gereksinimi gerçekleştirmek için periyodik(günlük) kontrol modülü geliştirilmiştir.

İşletmeler üretim için kullandıkları ellerinde varolan malzemelerin yönetimini gerçekleştirebilmek için belirli periyotlarla malzeme sayımı yapmaktadırlar. Ambar sorumlularının ambara girerek malzemeleri tek tek sayıp daha sonra sisteme kaydederek vakit harcamalarını istememektedir. Ayrıca manuel olarak gerçekleştirilen bu uygulamada hata yapma riski çok yüksektir. Bunun yerine Boysweb'in mobil versiyonu üzerinde geliştirilen yeni bir modül ile her ambar sorumlusunun elinde bulunan el terminali ya da cep telefonu ile uygulamaya girerek cihazın kamerasından ambar içerisindeki malzeme üzerindeki barkod numarasını okutarak ve ambardaki miktarını girerek o barkodlu malzemeye ait sayım bilgisinin sisteme girişini sağlamak istemektedirler. Bu gereksinimin gerçekleştirilebilmesi için malzeme sayımı modülü geliştirilmiştir.

Bakımdan sorumlu olan son kullanıcılar işletmelerinin krokisini tek bir ekrandan görmek ve hangi noktada problem varsa oraya direk müdahale etmek istemektedir. Alışveriş merkezlerindeki güvenlik odası gibi düşünülebilir. Alışveriş merkezlerindeki güvenlik şefi güvenlik odasında ekranların başında oturarak güvenlik kamerası sayesinde tüm alışveriş merkezini gözlemleyebilmektedir. Alışveriş merkezinin herhangi bir mağazasında tartışma, hırsızlık ya da huzursuz edici bir hareketlilik gördüğünde güvenlik görevlilerine telsiz ile ulaşarak o bölgeye yönlendirmektedir. Güvenlik görevlileri ilgili mağazaya giderek huzursuzluğu ortadan kaldırmakta ve telsizle şeflerini durumdan haberdar etmektedirler. Bu işleyişi Boysweb uygulamasına taşıyabilmek adına grafiksel varlık ağacı modülü geliştirilmiştir. Grafiksel varlık ağacı modülü sayesinde işletmelerde yöneticiler tek ekrandan varlıklarını takip ederek arızalarını görebilmekte ve müdahale edebilmekte ve varlık haritalarını istedikleri şekilde dizayn edebilmektedir.

İşletmelerin sahip oldukları ekipmanlarının bazı durumlarda yerlerinin değişmesi gerekmektedir. Örneğin bir bilgisayar sisteme tanımlandıktan sonra bir ekranda bilgisayara bağlı olarak sisteme tanımlanır. Bu ekran bilgisayarın alt ekipmanı olarak sistemde seçilir. Daha sonra bu ekranın bozulması durumunda bu ekranın yerinden alınarak yeni ekranın bu ekran yerine getirilmesi gerekir. Çıkarılan ekranında hurdaların içerisine atılması gerekir. Bu senaryonun Boysweb üzerinde işletilebilmesi için varlık

taşıma modülü geliştirilmiştir.

İşletmeler kullandıkları malzemelerinin satınalma işlemlerini sistem üzerinden gerçekleştirmektedir. Gerçekleştirilecek satınalma talepleri için formlar oluşturulmakta ve satınalma işlemi gerçekleştirilebilmesi için işletme yöneticilerin onayları ile satınalma işlemi tamamlanmaktadır. Tamamlanan satınalma işlemi sonrası satınalma formlarının çıktıları alınmakta ve bu formlar ıslak imza ile imzalanarak dökümanlar fiziksel olarak saklanmaktadır. Sistemin tamamen sayısal ortama geçmesi ve fiziksel olarak form saklama işleminin önüne geçilmesi için Boysweb uygulamasına bu tez çalışmasında elektronik imza da eklenmiştir. İmzalanması gereken dosyaların sanal ortamda imzalanarak saklanması sağlanmıştır. Excel formlarda bakımlarını takip eden şirketlerin daha kolay ve kullanılabilir bir ortama taşınması için Boysweb uygulamasına geçerek bakımlarında hız, güvenlik ve kalite sağlamalarının yanında aynı zamanda elektronik imza uygulaması eklenerek kağıtsız evrak yönetimi sağlanmıştır.

3.1. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİNİN ÖZELLİKLERİ

- Bakım yönetim sisteminde uluslararası standartlarda gerekli olan tüm bilgi ve raporlamalar elde edilebilmektedir.
- Koruyucu bakım tablosu hazırlanıp, kullanılacak yedek parça malzeme ve iş gücü kaynakları tanımlanarak koruyucu bakımın zamanı geldiğinde otomatik olarak iş emri oluşturulur.
- Kullanılan ekipmanların kalibrasyon işleri BOYSWEB ile takip edilebilir.
- Kullanılan kestirimci bakım sistemleri ile kestirimci bakım uygulamaları yönetilebilir.
- BOYSWEB Mobil ile iş emirleri direkt olarak mobil cihazlara gönderilip bakım iş emirleri ile ilgili bütün işler mobil olarak yönetilebilir.
- İşletmedeki tüm demirbaşlar BOYSWEB ile yönetilebilir.
- Makina grubu, makina, sarf yeri ve iş emri bazında bakım maliyetleri iş gücü maliyeti, malzeme maliyeti ve toplam maliyet olarak istenilen döviz kurunda alınabilmektedir.
- BOYSWEB, iç veya dış kaynak kullanımı ile ilgili karar destek sistemi oluşturabilir.

- Belirlenen periyotlarda, koruyucu bakımlar için gerekli malzeme ve iş gücü kaynağı tespit edilmekte ve bakım planlaması yapılabilmektedir.
- Belli standart talimatlara göre bakım işlemlerinin yapılması sağlandığı için hata yapma olasılığı azalmaktadır. İş gücü verimliliği artmakta ve işletmenin bakım yönetimi ile ilgili oldukça kapsamlı bilgi havuzu oluşmaktadır.
- Bakım personelinin performansı gerek departman gerekse kişi bazında analiz edilebilmektedir.
- İşletmelerde bulunan özel ekipmanların (elektrik motorları, pompalar vs.) takibi yapılabilmekte ve bunlarla ilgili bakım bilgileri raporlanabilmektedir.
- Koruyucu bakım planlarının canlı ve sağlıklı bir şekilde takip edilmesiyle üretim planlama personeline daha etkin üretim planlama yapılmasında yardımcı olmaktadır.
- Makina grubu ve makina bazında alınan bakım tarihçeleri ile makinenin bakım geçmişine ilişkin detaylı bilgi alınabilir.
- Alınan raporlar ile makine ömürleri hakkında bilgi sahibi olunabilir.
- Arıza türlerine göre alınan raporlarla belli arızaların oluş sıklığı ve kronik arızalar tespit edilebilir.
- Bakımların iş tiplerine (elektrik, mekanik vs.) göre dağılımı takip edilebilir.
- Koruyucu bakımların arıza bakımlara oranı BOYSWEB ile takip edilebilmektedir.
- Yedek parça/malzemelerin yıllık kullanım miktarları takip edilebilir. Bu şekilde yedek parça/malzemelerin minimum stok seviyeleri tespit edilebilir.
- Kullanılan yedek parça/malzemeler sarf yerleri ve makineler bazında takip edilebilir.
- En çoklar Analizi kullanılıp BOYSWEB' den 4 000'in üzerinde grafik elde edilerek işletme her açıdan analiz edilebilir.
- Grafik İle Maliyet Analizi kullanılarak seçilen parametre bazında istenilen düzeyde maliyet analiz grafikleri elde edilebilir.
- İşletmeler elektronik imza sayesinde satınalma formlarını dijital ortamda bilgi güvenliği içinde saklayabilmektedirler.
- İş Emri formları mobil uygulama üzerinden yazıcıya gönderilerek bakım

personellerinin çıktılarının istenilen ortamdan alınması sağlanmaktadır.

- Ekipman kordinatlarının mobile aktarılması ile ekipmanların mobil uygulamadan konumları belirlenerek navigasyon ile ekipmanların yollarının bulunması sağlanabilir.
- İş Emri bakım durumlarının mobil uygulama üzerinden seçilmesi sayesinde bakım personelleri işlere başlama-bitirme tarih ve saatlerini anlık olarak istedikleri yerden girebilmektedirler.
- Grafikselsel varlık ağacı modülü sayesinde işletmeler varlıklarının kontrolünü ve takiplerinin tek ekran üzerinden izleyip gerçekleştirebilmektedirler.
- Malzeme sayımı modülü sayesinde bakım personelleri malzemelerinin yönetimini ve sayımlarını mobil cihazları üzerinden hızlı ve pratik şekilde gerçekleştirebilmektedirler.
- Peroyidik(Günlük) kontrol modülü sayesinde işletmeler günlük bakımlarını otomatik olarak gerçekleştirebilmektedirler.

3.2. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİNİN İŞLETMELERE SAĞLADIĞI FAYDALAR

Boysweb'i kullanan firmalarda yönetim seviyesinde yapılan incelemelerde yetersiz kalan bakım-onarım yönetim sistemine oranla şu sonuçlar gözlenmiştir.

- Bakım maliyetlerinde %15-%30 tasarruf
- Yedek parça stok maliyetlerinde %15-%20 azalma
- Duruş sürelerinde en az %25 azalma
- Arıza sayısında %40-%80 azalma
- Arızaya müdahale süresinde azalma
- Bakım bilgilerini detay bazda, istenildiği anda, doğru olarak analiz edebilme imkanı
- Denetimlerde bakım açısından şirketin olumlu vizyon sunması
- Bakım bütçelerinin daha sağlıklı bir şekilde oluşturulup yönetilmesi
- Bakım planlaması yapabilme imkanı
- Ekipmanların bakım kontrolünün iyi yapılması sonucu daha kaliteli üretim

yapılması

- Ürün maliyetlerinin bakım maliyetlerinin azalmasından dolayı düşmesi
- Bakım personelinin eğitim ihtiyaçlarının daha sağlıklı bir şekilde tespit edilmesi

3.3. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİNİN SİSTEM MİMARİSİ

3.3.1. Rest Mimarisi ve Restful Servisler

Tcp Protokolü, Internet protokollerinden birisidir. Ağlar arasında veri aktarımını sağlamak amacıyla tasarlanmıştır. HTTP, HTTPS, FTP gibi protokoller TCP yapısında yazılmışlardır. Çalışma mantığı sunucu ile bağlantı sağlanır, veri transferi yapılır, bağlantı koparılır.

Http, tarayıcılar ile web sunucuları arasında ortak bir dildir. Http Tcp protokolü üzerine kurulmuş bir protokoldür. İstek varsa bağlantı kurulur, cevap geldiğinde bağlantı sonlanır. 80 portu üzerinde çalışır. İstek geldiğinde bir bilgi ve başlık şeklinde veri gönderilir. Cevap geldiğinde bağlantı sonlandırılır.

Soap Web Servis, Xml şeklinde haberleşme mantığına sahip Tcp protokolü üzerine kurulu servislerdir.

Soap mesaj yapısı içerisinde envelope, header, body, fault terimleri yer almaktadır. Envelope, soapa ait isteklerin ve bu isteklere verilen cevapların verilerini içerir. Header, metadata bilgilerini içerir. Body, sadece soapda bulunur. XML formatında veriler iletilir ve XML formatında cevaplar alınır. Fault, dönecek cevapları içerir.

Rest, veri transferini hızlı bir şekilde istemci ile sunucu arasında yapan servis modelidir. Verilerin içerikleri küçük tutulduğu için hızlıdır ve HTTP üzerinde çalışır.

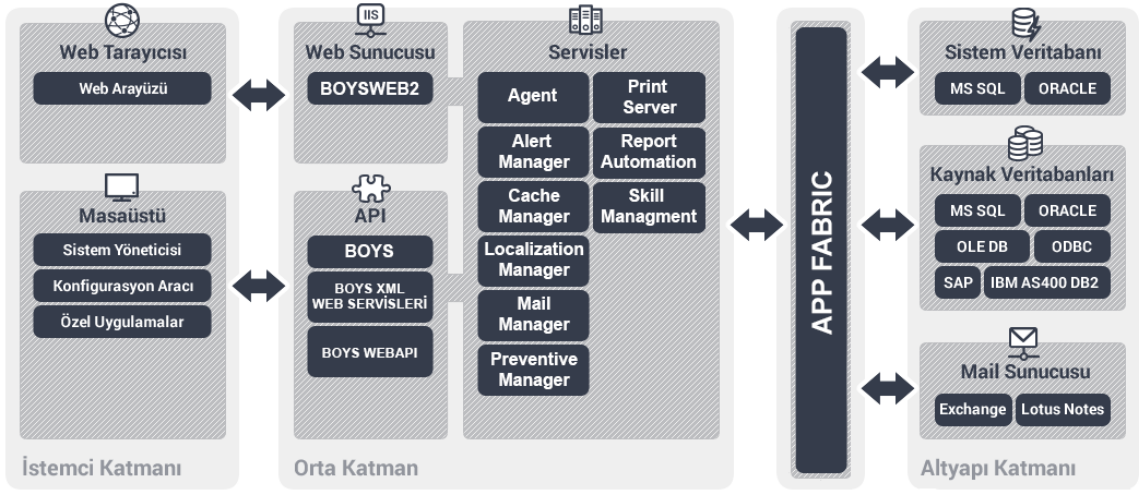
Restful servisler üzerinde XML, JSON, Text datalar gönderilip alınabilir. Soap servislerde sadece xml yapısı kullanılmaktadır ve soap servislerde mesajla birlikte header bilgiside gönderilmelidir. Restful servislerin kullanımı ve güncellemesi kolay olduğundan seçim nedenidir.

3.3.2. Sistem Kapsamı

BOYSWEB Varlık ve Bakım Yönetim Sistemi .Net framework'ü üzerinde servis tabanlı mimariye (Service Oriented Architecture - SOA) sahip ve Internet Information Services (IIS) üzerinde çalışan web tabanlı bir sistemdir. Servise dayalı mimari, birbirinden

bağımsız çalışan servis ve uygulamaların birbirleri ile sistemsel olarak bütünleşmesine imkan sağlayan bir sistem tasarımı anlayışıdır [10]. SOA teknolojisindeki servisler yapboz parçalarına benzetilebilir. Servislerde yapılan yer değiştirmeler ve fonksiyonel değişiklikler yapboz parçalarının yerlerinden sökülüp istenilen yere takılması ya da yeni parçaların eklenmesi kadar esnek bir işlemdir. Süreçler bir bütün halinde bulunsalardı, parçalı bir yapı olmasaydı, yeni eklentiler ekleyip çıkarmak, parçaları birleştirmek hiç de kolay olmazdı [11].

Sistem veri kaynağı olarak Microsoft SQL Server ve Oracle veritabanlarını kullanabilmektedir. Kısıtlı kullanımda süreç tasarımı (Bakım iş talebi ve satın alma süreçleri için), sistem yönetimi gibi istemci araçları servisler üzerinden sunucuya 80 portundan iletişime geçmektedir. Bu sayede istemcilere ekstra sistemsel ayar (port açma gibi) yapılmasına gerek kalmamaktadır. Sistem mimarisi; altyapı, orta ve istemci olarak 3 katmandan oluşmaktadır [12]. Şekil 3.1’de bu 3 katmanın birbirleri ile iletişimi ve içerikleri gösterilmiştir.



Şekil 3.1. Boysweb' in sistem mimarisi.

3.3.2.1. Boysweb'in Altyapı Katmanı

Sistem veri tabanı ve mevcut operasyonel sistemleri içerir. Sistem veri tabanı, kaynak veri tabanları ve mail server olarak 3 bölümden oluşmaktadır.

- Sistem veri tabanı: BOYSWEB sisteminin veri kaynağı olarak kullanacağı veri tabanıdır. Microsoft SQL ya da Oracle veri tabanı kullanılabilir.
- Kaynak veri tabanları: Sistem mevcut operasyonel sistemlere ve veri tabanlarına erişim yaparak ilgili verileri bakım emirleri içerisinde ve/veya malzeme yönetim sistemi

içerisinde kullanabilmektedir. Bunlardan bazıları MS Sql, Oracle, Ole Db, Odbc, AS 400 DB2, SAP, LOGO Tiger, LOGO Unity, NETSIS ve Xml Web Servisleridir. Ayrıca harici veri bağlayıcısı da tanımlama imkanı mevcuttur.

- Mail Sunucusu: Sistem üzerinden gönderilecek maillerin gönderimini sağlayacak servistir.

3.3.2.2. Boysweb'in Orta Katmanı

Sunucu bileşenlerinin yer aldığı katmandır. BOYSWEB'nin tüm servislerinin yanı sıra, IIS Web server ve BOYSWEB API'ları da bu katman üzerinden çalışmaktadır. Servisler kendi aralarında ve alt yapı katmanı ile haberleşmektedir. Web server ve API'lara servis hizmeti vermektedir. Web server ve API'lar da istemci ile servisler arasında ara birim olarak çalışıp yalıtılmış bir yapı sağlamaktadır.

Aşağıda BOYSWEB Servislerinin genel işlevleri açıklanmıştır;

- Agent: Zamanlanmış bakım işlerini ve bazı küçük veri aktarım işlerini yapar.
- Alert Manager: Uygulama içerisinde, kullanıcı tarafından ayarlanan uyarı işlerinin yürütülmesinden sorumludur.
- Cache Manager: Çok kullanılan verilerin sunucu hafızasına yüklenerek kullanıma hazır hale getirilmesinden sorumludur.
- Localization Manager: Uygulama ara yüzünün çok dilli yapıda çalışmasını sağlar.
- Mail Manager: Uygulamanın e-posta gönderim işlerinin yerine getirilmesinden sorumludur.
- Preventive Manager: Periyodik olarak sistemi tarayarak, bakım için zamanı gelen periyodik bakım tanımlarından iş emri oluşturmaktan sorumludur.
- Print Server: Yazdırılma listesinde bekleyen iş talep ve iş emri formlarının ilgili yazıcılara gönderilmesinden sorumludur.
- Report Automation: Uygulama içerisinden, kullanıcı tarafından belirlenen raporların otomatik dağıtılmasından sorumludur.
- Skill Management: Bakım iş emirlerine yeteneğe göre otomatik bakım sorumlusu atanmasından sorumludur.

3.3.2.3. Boysweb'in İstemci Katmanı

İstemci katmanı son kullanıcılara ve yönetici kullanıcılara hizmet veren uygulamalardan oluşmaktadır. Web ve masaüstü uygulamaları olarak 2'ye ayrılmaktadır.

- Web: Son kullanıcılar için web tarayıcıları üzerinden erişebildikleri web tabanlı bir uygulamadır.
- Masaüstü: BOYS uygulamasının yönetici araçlarından oluşan Windows tabanlı masaüstü uygulamalarıdır.
- ConfigurationEditor: Uygulama kullanım parametrelerinin ayarlandığı ve izlendiği yönetici aracıdır.
- Workflow Studio: Bakım iş talebi ve satın alma süreçlerinin kurgulandığı kısıtlı kullanım lisanslı iş akış yönetici aracıdır.
- ServerConfigurationEditor: Uygulama veritabanı bağlantısının ayarlandığı yönetici aracıdır.
- AppFabricConfigurator: Bir önbellek (Cache), gerekli yapılandırmayı bilen tüm programlarca erişilebilir, kullanılabilir/ayarlanabilir ortak hafıza alanıdır. Her bir önbellek alanı kendisine ait tekil ismi üzerinde kullanıma sunulmaktadır. Bölgeler(Region), önbellekler içerisinde yer alan yapılar olup ek veri taşıyıcı alanlardır [13]. Bölgeler üzerinden önbellekler ile standart işlemleri yapabilir, verilen anahtar üzerinden istenilen nesneye ulaşılabilir. Bu özelliklerinin yanında; tag adı verilen ve string veri türünde veri tutabilen alanlar sayesinde bölge içerisindeki tüm nesnelerde arama yapılabilir [13]. AppFabricConfigurator, Boys üzerinde cache ve region oluşturma, silme ve yönetme işlemlerinin yapıldığı AppFabric yardımcı yönetici aracıdır.

3.3.3. Boysweb' in Kullanıcı Arayüzü

HTML5, CSS3 ve ASP.NET tabanlı web uygulaması ile kullanıcılara kullanışlı, rahat ve hızlı bir ara yüz sunmaktadır. Ayrıca IOS ve Android mobil işletim sistemlerine uygun native uygulamaları ile her platformda kullanıcılarına hizmet sunmaktadır.

3.3.4. Boysweb' in Konfigürasyonu

Boys kullanan firmalar sistemi kullanmaya başlayacaklarında öncelikle müşteri sunucularında bir test ortamı hazırlanması gerekir. Bu test ortamında sistemin hız testi, stres testi yapılır. Kullanıcıların programa girerek aşırı yüklenmeleri sonucunda ne gibi

aksiliklerle karşılaşılacağı saptanır. Oluşan problemler çözülerek sistem sorunsuz hale getirilir. Test ve canlı sistemlere Boys'un kurulumu ve düzgün çalışması için firmadan sunucu özellikleri önceden belirlenen sunucuları kurmaları istenir. Test ve Canlı Sistem için gerekli, önerilen konfigürasyon özellikleri aşağıda Çizelge 3.1'de gösterilmiştir.

Çizelge 3.1. Minimum BOYSWEB test ve canlı sistem gereksinimi.

Test Sistemi Gereksinimleri	
Sunucu İşlemcisi	Min Intel Xeon 4 çekirdekli işlemci
Sunucu Ram	16 GB
Sunucu İşletim Sistemi	Windows 2008 veya üstü 64 Bit
Web Server - Framework	IIS7 veya üstü, Microsoft .NET Framework 4,
Veritabanı	MS SQL Server 2008 veya üstü
DB Disk Alanı	100 GB
Uygulama Disk Alanı	50 GB
Canlı Sistem Gereksinimleri	
Sunucu İşlemcisi	Min Intel Xeon 4 çekirdekli işlemci
Sunucu Ram	32 GB
Sunucu İşletim Sistemi	Windows 2008 veya üstü 64 Bit
Web Server - Framework	IIS7 veya üstü, Microsoft .NET Framework 4,
Veritabanı	MS SQL Server 2008 veya üstü
DB Disk Alanı	500 GB
Uygulama Disk Alanı	200 GB

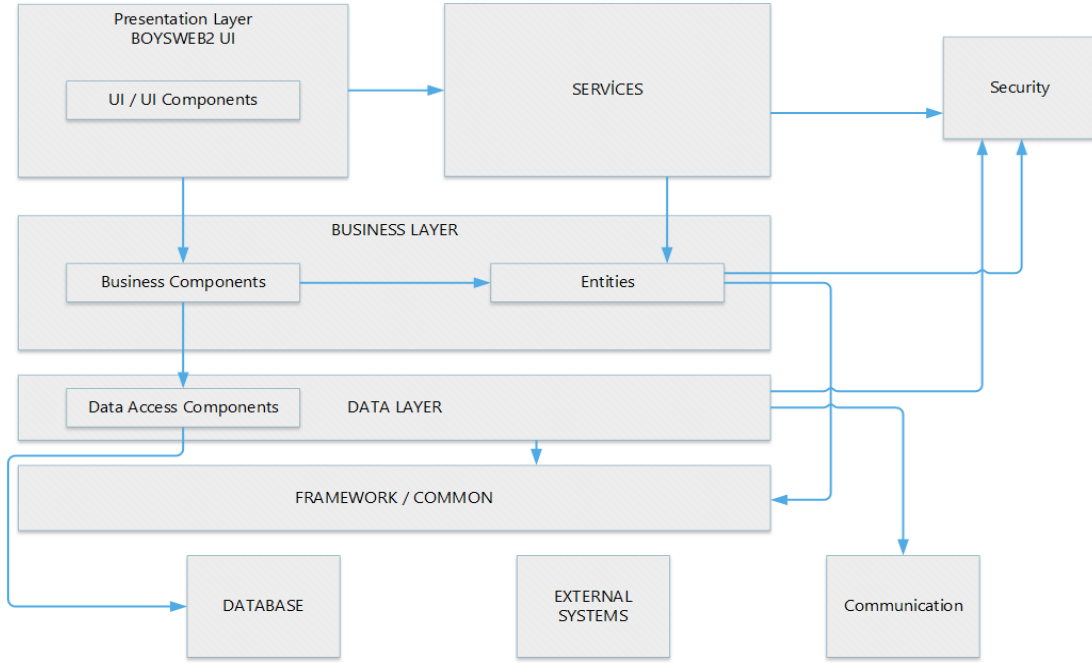
3.3.5. Boysweb'in Mimari Avantajları ve Kısıtları

Windows Server AppFabric; web uygulamaları ve orta katman hizmetler için geliştirilmiştir. Barındırma ve ön belleğe alma gibi özellikleri sunar. Windows Server işletim sisteminin kullanılabilirliğini geliştirmektedir. Windows Server AppFabric, barındırma özelliği sayesinde, Internet Information Services (IIS) ve Windows Etkinleştirme Hizmeti (WAS) ve .NET platformları için ek yönetim araçları içermektedir. Windows Server AppFabric'in sunmuş olduğu ön belleğe alma özelliği sayesinde, yüksek performans gerektiren .NET platformu uygulamaları ve ASP.NET uygulamaları için genişletilmiş bir platform sağlamaktadır. AppFabric kurulum sihirbazı üzerinden ve komut istemi parametreleri kullanılarak yüklenebilmektedir. Ayrıca birden fazla bilgisayar üzerindeki bellek kapasitesini birleştirerek, .NET uygulamaları kolayca ve düşük bir maliyetle ölçeklenebilir. BOYSWEB AppFabric mimarisi üzerinde çalıştığı için yüksek erişilebilirlik ve ölçeklenebilirlik senaryolarına uyum sağlayabilmektedir. Web tabanlı bir uygulama olması her yerden sistemi kullanılabilir kılmaktadır. Uygulama popüler yeni nesil tarayıcılarda sorunsuz çalışabilmektedir. Bununla beraber IOS ve Android işletim sistemlerinde de native uygulama desteği mevcuttur. BOYS .Net tabanlı

bir uygulama olduğundan dolayı sadece Windows işletim sistemi üzerinde çalışabilmektedir.

3.4. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİ KATMANLARI

BOYS uygulamasındaki katmanların bağılıkları aşağıda Şekil 3.2’de detaylı şekilde anlatılmıştır.



Şekil 3.2. Boysweb' in katmanları.

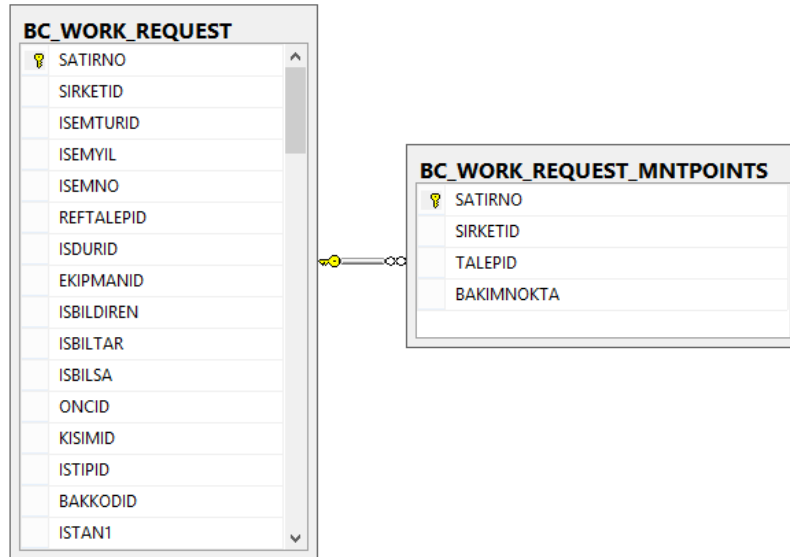
- **Presentation Layer:** Verinin anlaşılır halde son kullanıcıya sunulduğu katmandır. Bu katmanda BOYSWEB web arayüzü ve native mobil uygulamalar bulunmaktadır. Bu katmanın servis katmanı ve Business katmanına bağılılığı vardır.
- **Services:** Uygulamada arka planda çalışan ve veri dağıtım ve iletişim görevlerini üstlenen katmandır. Bu katmanın security ve entity katmanlarına bağılılığı vardır.
- **Business Layer:** Uygulamanın iş mantığının yer aldığı katmandır. Data, entity ve framework katmanlarına bağılılığı vardır.
- **Data Layer:** Veritabanına erişimin sağlandığı ve transaction'ların yönetildiği katmandır. Database, framework ve communication katmanına bağılılığı vardır.

3.5. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİ DİAGRAMLARI

Burada Boys programında kullanılan çok fazla veritabanı ve tabloları mevcuttur. Bu bölümde hepsi anlatılamayacağı için örnek bir veritabanı ve tablo üzerinden anlatım yapılmıştır.

3.5.1. İş Talebi

Bakımcı operatörler gördükleri arızaları iş talebinden sisteme bildirirler. Bu arızalara ilişkin bakım riskleri, bakım noktaları, koruyucu önlemleri, bağlı dokümanları da kayda eklenir. Girilen verilere ilişkin BC_wok_request tablosu ana iş talebi kaydının tutulduğu tablodur. Bu tablo iş talebine ait numara, arıza bildiriliş tarih ve saati, talebe ait açıklamalar gibi ana kayda ilişkin verilerin tutulduğu tablodur. Bu iş talebine ait birden fazla bakım noktası, bakım riski, koruyucu önlemler için ayrı tablolar mevcuttur. Ana tablo olan bc_work_request tablosu ile detay tablo olan bc_work_request_mntpoints tablosu 1-n ilişki içerisinde. Bir iş talebinin birden fazla bakım noktası olabilir. İlişkilendirme aşağıda Şekil 3.3'deki gibi ikincil anahtar(talepid) yardımı ile 1-n ilişki kurularak sağlanır.

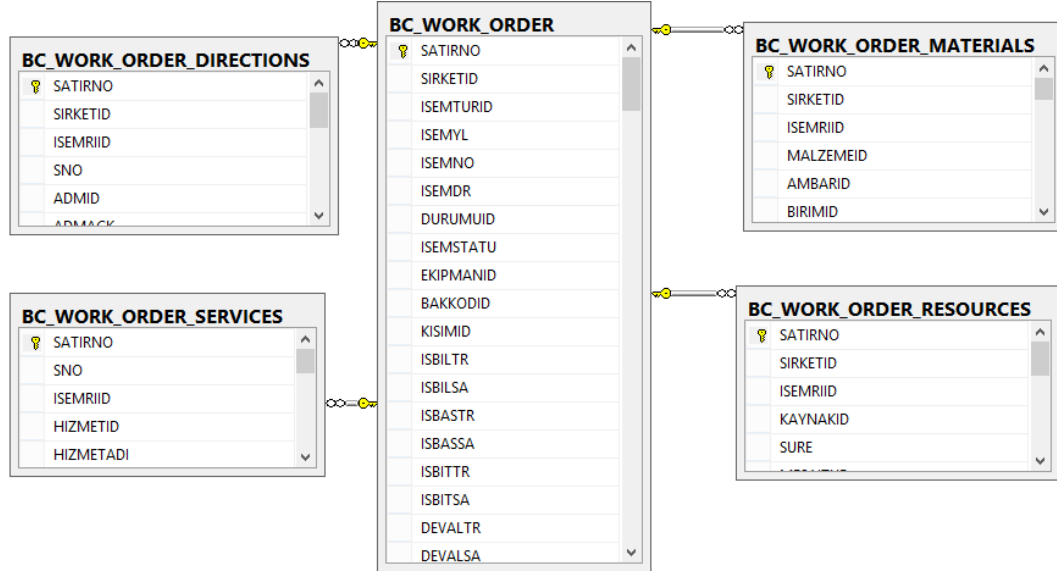


Şekil 3.3. İş talebi veritabanı diagramı.

3.5.2. İş Emri Veritabanı Diagramı

İş Emri ana tablosu bc_work_order tablosudur. Bu tabloda iş emrine ait ana bilgiler tutulur. İş emri numarası, iş emri yılı, iş emri durumu, iş emri statüsü, iş emrinin bildiriliş

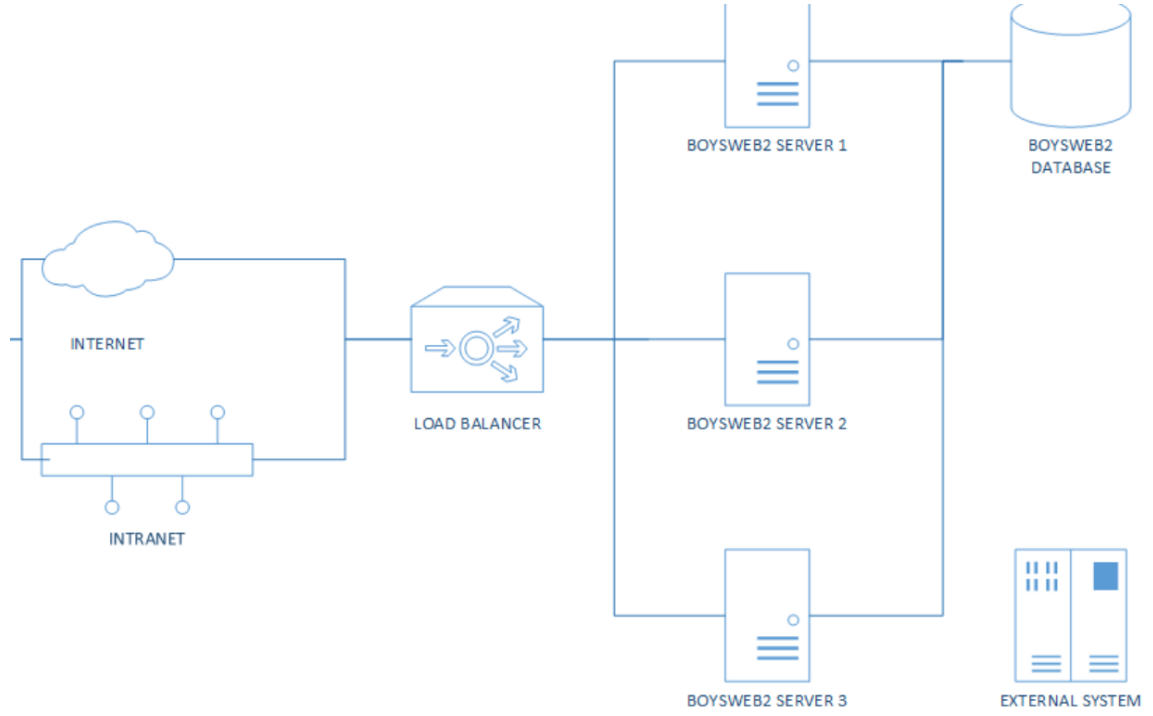
tarih ve saati, tamamlanma tarih ve saati gibi. Periyodik bakım iş emrine ait iş adımlarının tutulduğu tablo bc_work_order_directions tablosudur. Bir iş emrine ait birden fazla iş adımı olabileceği için 1-n ilişki kurulmuştur. İkincil anahtar olarak isemriid kullanılmıştır. İlişkilendirme aşağıda Şekil 3.4’deki gibi ikincil anahtar (isemriid) yardımı ile 1-n ilişki kurularak sağlanır.



Şekil 3.4. İş emri veritabanı diagramı.

3.6. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİ DAĞILIM DİAGRAMI

Dağılım diyagramı, yazılımın üzerinde çalışacağı donanımın fiziksel mimarisinin ortaya konduğu bakış açısıdır. Şekil 3.5’de Boys sisteminin nasıl çalıştığını göstermektedir.



Şekil 3.5. Boysweb dağılım diagramı.

3.7. USE-CASE DİAGRAMI

Aşağıda iş talebi ve iş emrine ait use-case tanımı ve diagramları tasarlanmıştır. İş talebi ve iş emrinin nasıl kullanıldığı çalışma mantığı ve senaryoları belirlenmiştir.

3.7.1. İş Talebi Diagramı

USE CASE-1

ADI: Hızlı Talep

AKTÖRLER: Kullanıcı

ÖN KOŞULLAR: Sisteme login olunur , Kullanıcılara ilgili yetkiler verilir.

SON DURUM: Kayıt tamamlanır.

BAŞARILI SENARYO:

1. Genel bilgiler ve bağlı dökümanlar tablaları mevcuttur.
2. Talep açılacak olan varlık seçilir.
3. İş tipi , bakım arıza kodu ve Talep açıklaması alanları girilir.
4. Kayıt tamamlanır.

ALTERNATİF YOLLAR

USE CASE-2

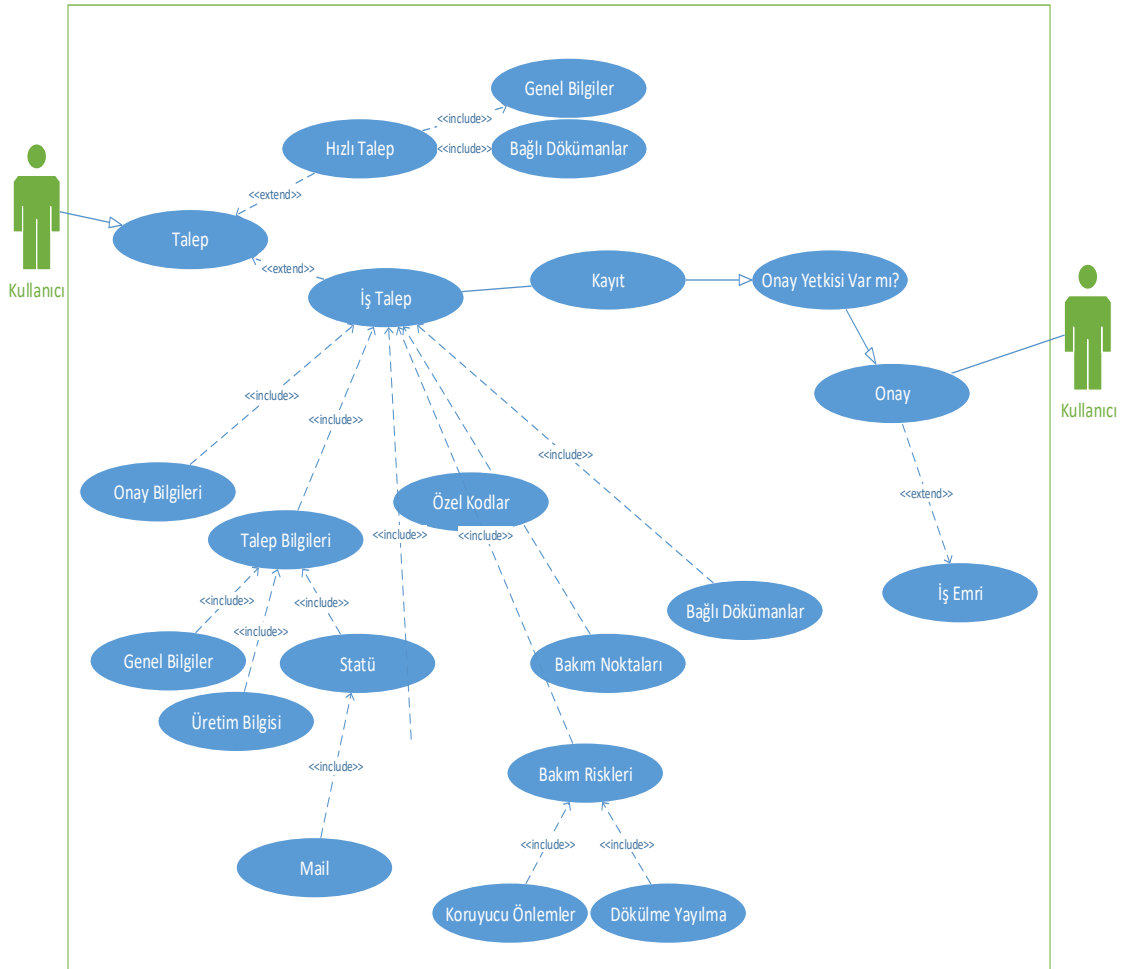
ADI: Yeni Talep

AKTÖRLER: Kullanıcı, Onay verecek kullanıcı

ÖN KOŞULLAR: Sisteme login olunur, Kullanıcıya talep onaylama yetkisi verilir.

SON DURUM: Yeni iş talebi sisteme eklenir.

Bir iş talebinin ilk kullanıcılar tarafından sisteme girilmesi, onaylardan geçmesi ve sisteme girildikten sonra sistemde yapabileceği aksiyonları gösteren diagram örneği Şekil 3.6’da gösterilmiştir.



Şekil 3.6. Boysweb iş talebi use – case diagramı.

BAŞARILI SENARYO:

1. Bakım menüsü altından iş talep sayfasına girilerek ekle butonu ile yeni sayfa açılır.
2. İş emri türü tanımlarda belirlediğimiz varsayılan iş emri türü olarak atanır.
3. İş emri numarası iş emri türü üzerinden tanımladığımız sayaç aralıklarına göre otomatik atanır.

4. İş talebinin açılacağı varlık seçilir.
5. Bu varlığa ait sarfyeri, kısım bilgileri otomatik olarak doldurulur.
6. Bu iş talebine ait bakım arıza kodu, iş tipi, bakım önceliği, bilgilerinin girilmesi gerekir.
7. Arıza oluşma tarihi ve bildiriliş tarihi tarihleri girilir. Bu tarihler iş emrine aktarılacaktır.
8. Talep açıklaması bilgisi girilir.
9. Onay bilgileri içerisinde, bakım onayı yapıp yapılmayacağı, iş güvenliği sağlanıp sağlanmadığı, onaylayan bilgileri, işin sorumlusu, onay tarihleri girilir.
10. Özel Kodlar sekmesindeki alanlar kişiselleştirilerek talep açarken girilmek istenen farklı bilgiler buralara girilebilir.
11. Bakım riski tabında riskler listesinden bu iş talebi için hangi riskler öngörülüyor ise seçilebilir. Bunun için risklerin önceden tanımlanmış olması gerekir.
12. Dökülme yayılma ve koruyucu önlemler ise burada tanımlanabilecek değerlerdir. Dökülebilecek kimyasalların ve risklere karşı nasıl koruyucu önlemler alınacağını bilgisi buraya girilir.
13. Bakım noktaları sekmesinde yapılacak olan bakımın hangi noktalara yapılacağı etkin olarak seçilir.

3.7.2. İş Emri Diagramı

USE CASE-1

ADI: Talepden Gelen İş emri

AKTÖRLER: Kullanıcı

ÖN KOŞULLAR: Sisteme login olunur, Kullanıcılara ilgili yetkiler verilir.

SON DURUM: Kayıt tamamlanır.

BAŞARILI SENARYO:

1. Sisteme iş talebi açılır.
2. Bu talebe ait iş emri sistem tarafından otomatik oluşturulur.
3. Kayıt tamamlanır.

ALTERNATİF YOLLAR

USE CASE-2

ADI: Yeni İş Emri

AKTÖRLER: Kullanıcı

ÖN KOŞULLAR: Sisteme login olunur, kullanıcıya yeni iş emri açma yetkisi verilir, yeni yıla ait iş emri sayaç bilgilerinin tanımlanmış olması gerekir.

SON DURUM: Yeni iş emri sisteme tanımlanır.

BAŞARILI SENARYO:

1. Bakım menüsü altından iş emri sayfasına girilerek ekle butonu ile yeni sayfa açılır.
2. İş emri türü tanımlarda belirlediğimiz varsayılan iş emri türü olarak atanır.
3. Default olarak iş emri açık seçili ve üretim devam ediyor gelir.
4. İş emri numarası iş emri türü üzerinden tanımladığımız sayaç aralıklarına göre otomatik atanır.
5. İş emrinin açılacağı varlık seçilir.
6. Bu varlığa ait sarfyeri, kısım bilgileri ve varlık üzerinden iş emri varsayılan bilgileri tanımlanmış ise bu bilgiler otomatik olarak doldurulur.
7. Eğer tanımlanmamış ise bu iş emrine ait bakım arıza kodu, iş tipi, bakım önceliği, bilgilerinin girilmesi gerekir.
8. İş emrinin başlangıç tarihi, bitiş tarihi, devreye alma tarihi gibi tarih bilgileri default olarak sistem tarihinden çekilerek gelir. İstenirse bu bilgiler iş emri açılırken düzenlenir istenirse daha sonra açık olan iş emrini kapatmak için sistem girişi yapıldığında düzenlenir.
9. Bu tarih ve saat aralıklarına göre işçilik süresi, bakım süresi, müdahale süresi gibi sürelerin hesaplanması sağlanır.
10. Arıza nedeni, çözümü ve bu işi kimin yapacağı seçilir.
11. Talep açıklaması ve yapılan işin açıklaması bilgileri girilir.
12. Bildirildiği mesai seçilir.
13. İşçilik maliyeti, malzeme maliyeti, hizmet maliyeti gibi bilgiler diğer sekmelerdeki personel, hizmet, malzeme alanları doldurulduğunda hesaplanır.
14. Statü kodu, tanımlar altında statüler içerisinde varsayılan tanımlamaları yapılabilir ve bizim değiştirmemize gerek kalmadan iş emri durumuna göre Kayıt aşamasında statü kodu değişir. Manuel olarak da değiştirilebilir.
15. Değişen statüye göre statü tanımları içerisinde eposta ya da sms gönder checki işaretli ise ve mail ve sms adresleri doldurulmuş ise sms ve mail olarak iş emri durumu iletilecektir.
16. Çalışan Personel sekmesinde bu iş emri için hangi personeller çalışacaklar ise bu

personellerin çalışma saatleri ve mesaieleri nasıl olacak ise ona göre seçim yapılır. Genel bilgiler sekmesinde seçilen işin sorumlusu kullanıcının onayına göre bu sekmeye çalışan personel olarak eklenebilir.

17. Personeller eklenirken çalışma süreleri iş emri çalışma süresi aralığı içerisinde olmalıdır.

18. Bu İş emri için malzeme kullanılmış ise kullanılan malzemeler sekmesine eklenecektir.

19. Burada malzeme seçimi, seçilen varlığın malzemeleri içinden, varlık grubunun malzemeleri içerisinde, ya da bakım kodunun malzemeleri içerisinde yapılabilir.

20. Seçilecek malzemenin girilen miktarının seçilmiş ambardaki miktarından küçük olması gerekir.

21. Bu iş emri için hizmet kullanılmış ise hizmetler sekmesinden seçilir.

22. Bu iş emrinin yapılması için bazı iş adımlarının yapılmış olması gerekiyor ise bu iş adımlarının seçimi sağlanır. Eğer iş emri periyodik bakım iş emri değil ise, İş emri periyodik bakım iş emri ise iş adımları periyodik bakım tanımları sayfasından tanımlanan bakımın bakım planı içerisindeki iş adımları olarak atanacaktır.

ALTERNATİF YOLLAR

1. Varlık üzerinde iş emri varsayılanı tabımdaki bilgiler doldurularak iş emri açılır iken varlık seçildiğinde bu verilerin otomatik doldurulması sağlanabilir.

2. Yapılacak entegrasyonlara göre iş emirlerinin sisteme otomatik oluşması sağlanabilir.

USE CASE-3

ADI: Periyodik Bakım İş Emirleri

AKTÖRLER: Sistem, Kullanıcı

ÖN KOŞULLAR: Sisteme login olunur, Kullanıcılara ilgili yetkiler verilir, periyodik bakımlar tanımlanır, periyodik bakım servisi çalıştırılır.

SON DURUM: İş Emri Oluşturulur

BAŞARILI SENARYO:

1. Sisteme iş adımları tanımlanır.
2. İş adımları bakım planlarına bağlanır.
3. Bakım planları periyodik bakım tanımları altında periyodik bakımlara bağlanır.
4. Periyodik bakım servisi çalıştırılır.

5. Verilen tarih aralıkları i periyotlar ya da sayaçlara göre iş emirleri oluşturulur.
6. Ya da periyodik bakım tanımları altında iş emri oluştur butonu kullanılarak bunlara gerek kalmadan iş emri oluşturulabilir.

ALTERNATİF YOLLAR

1. Periyodik bakım tanımları altında iş emri oluştur butonu kullanılarak bunlara gerek kalmadan iş emri oluşturulabilir.

USE CASE-4

ADI: İş Emri Değişiklikleri

AKTÖRLER: Sistem, İş Emri Değiştirme Yetkisi olan kullanıcılar

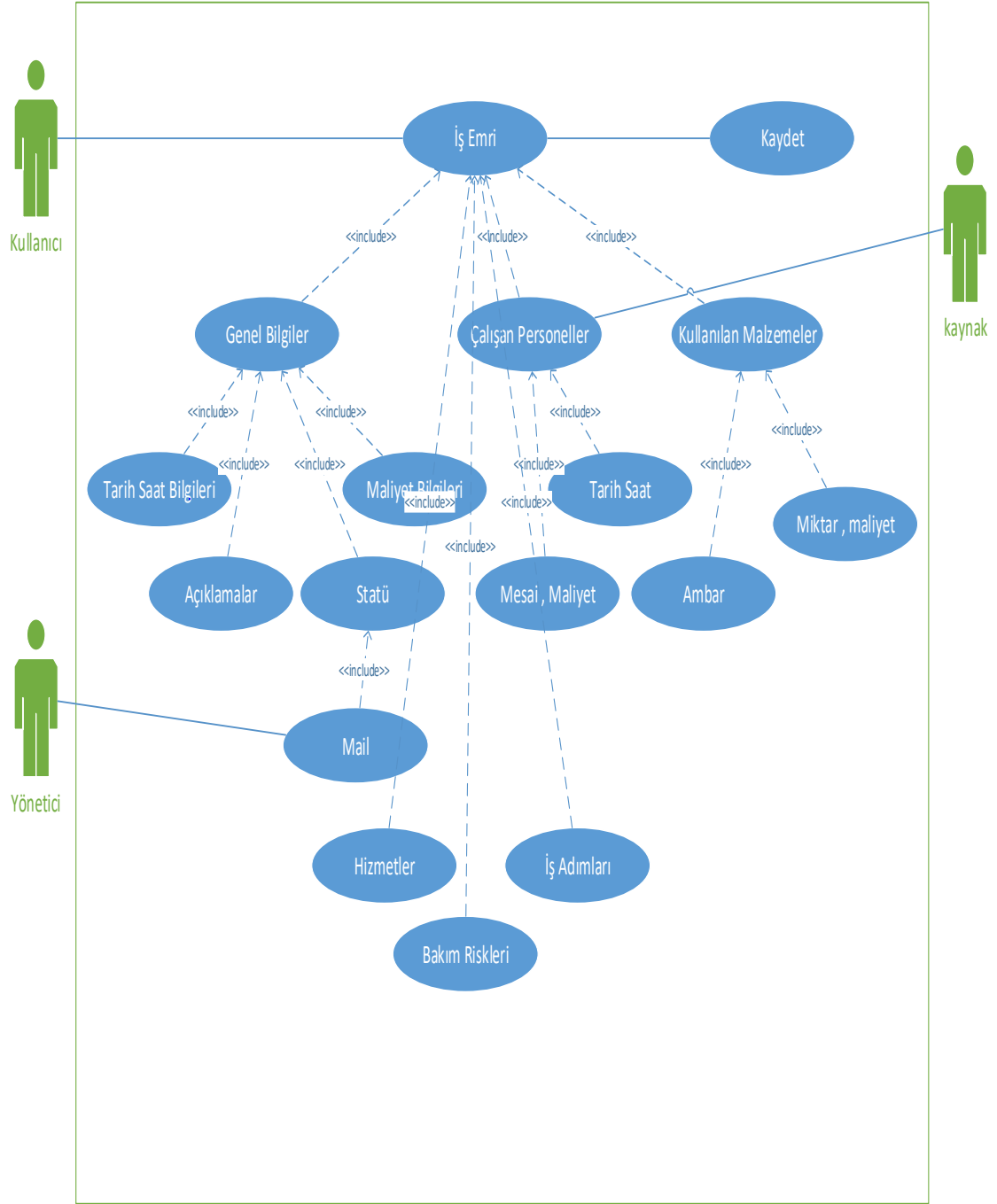
ÖN KOŞULLAR: Sisteme login olunur, Kullanıcılara ilgili yetkiler verilir

SON DURUM: İş Emri Kapatılır.

BAŞARILI SENARYO:

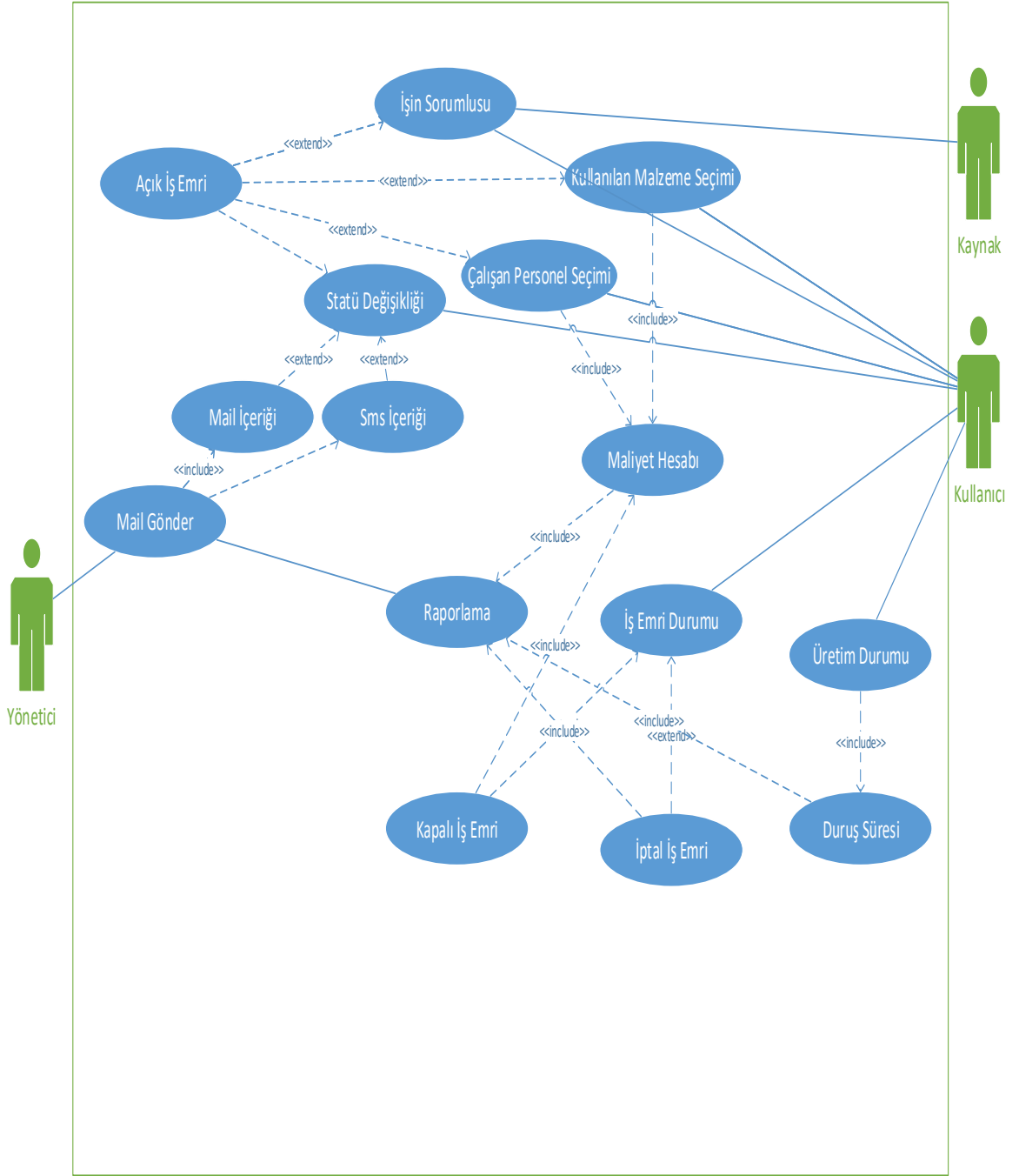
1. Kapalı ve İptal Edilen iş emirlerine yetki dahilinde girilebilir.
2. Açık iş emirleri için emir içerisine girerek, saat ve tarih aralıkları değiştirilerek çalışma süresi, duruş süresi gibi bilgiler güncellenebilir.
3. Statü değişiklikleri ile tanımlanan kullanıcılara ilgili bilgiler mail atılabilir.
4. Üretim durumu değiştirilerek üretimin durup durmadığı, durmuşsa kaç dakikalık bir duruş olduğu, duruştan kaynaklanan maliyet kaybı gibi bilgiler hesaplatılabilir.
5. İş Emri Durumu Açık, kapalı, iptal seçenekleri ile iş emrinin durumu belirlenir.
6. İş emri durumu kapalı seçildiğinde maliyetler hesaplanır, statü kapalı iş emrine çekilir ve bu statüde bilgilendirme maili atılabilir. Hesaplanan maliyet bilgileri, süre bilgileri raporlar ile yöneticilere sunulabilir.
7. İş Emri iptal edilerek yine iptal iş emri ile alakalı raporlar ve analizler çekilebilir.

Bir iş emrinin sisteme kaç farklı yolla tanımlanabileceği ve sisteme tanımlanan iş emri üzerinde işin sorumlusunun değişiklik yaptığında nasıl aksiyonlar alınabileceğini gösteren diagram Şekil 3.7’de gösterilmiştir. İş emri sisteme 3 yolla girilmektedir. Zamanı gelen periyodik bakım sisteme otomatik olarak iş emri oluşturabilir. Bir kullanıcı sisteme iş talebi girdiğinde gerekli onaylardan geçerek bu iş talebi iş emrine dönüşüp sistemde oluşabilir. Kullanıcı sıfırdan kendisi iş emri girebilir.



Şekil 3.8. Boysweb iş emri kayıt aşamasını gösterir use – case diagramı.

Sisteme girilen iş emrinin işin sorumlusu tarafından kapatılması ve ya iptal edilmesi sonrasında yöneticilere mail bilgilendirmesi, çekilen raporlarda bu iş emirleri ile ilgili maliyet bilgileri, arıza sıklığı, üretim durumunu gösteren diagram Şekil 3.9’da gösterilmiştir.



Şekil 3.9. Kapatılan ve ya iptal edilen iş emri use – case diagram.

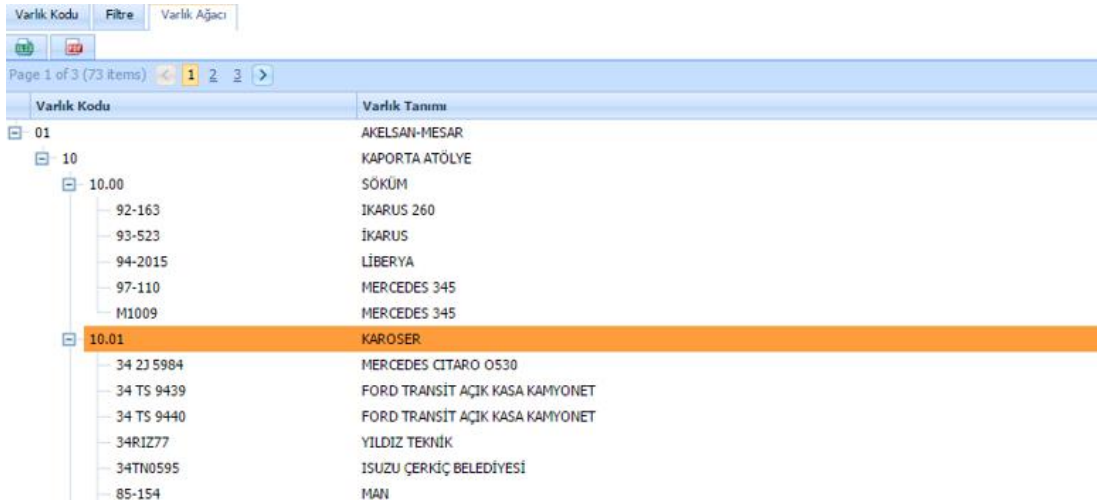
3.8. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİNİN ANA MODÜLLERİ

3.8.1. Varlık Yönetim Sistemi

Boys varlık yönetimi tanımlarında işletmeler, sarf yerleri ve kısımlar olarak 3 ayrı kırınım vardır. Bir firma birden fazla işletmeden oluşabilir. Varlık menüsü altına bu işletmeler listelenir. Bu işletmeler sarf yerlerine göre ayrılır. Sarf yeri kısımlarına ayrılır. Örneğin bir blok içerisinde binalar vardır. Her binada daireler vardır. Daireler içerisinde de eşyalar

vardır. Eşyaların sonsuz sayıda alt parçaya ayrılarak sisteme tanımlanabilir. Bir buzdolabını sisteme tanımlamak istendiğinde işletme olarak B blokları, sarfyeri olarak 3. Bina, kısım olarak 2. Kat, varlık olarak buzdolabı denir. Daha sonra alt varlık olarak buzdolabının motoru seçilebilir. Böylece buzdolabı motorunda arıza tespit edildiğine bu motora ilişkin iş açılabilir.

Varlıkların sarf yeri, kısım bilgileri değiştirilmek istendiğinde ya da başka bir kısma taşınmak istenebilir. Kullanıcıya kolaylık sağlaması açısından varlık ağacı denilen hiyerarşik bir form tasarlanmıştır. Varlık Ağacı, varlıkların daha kolay bulunabilmesini sağlar. Şekil 3.10'da varlık ağacında varlıklar ve bu varlıkların lokasyon içerisindeki kırımları gösterilmiştir.



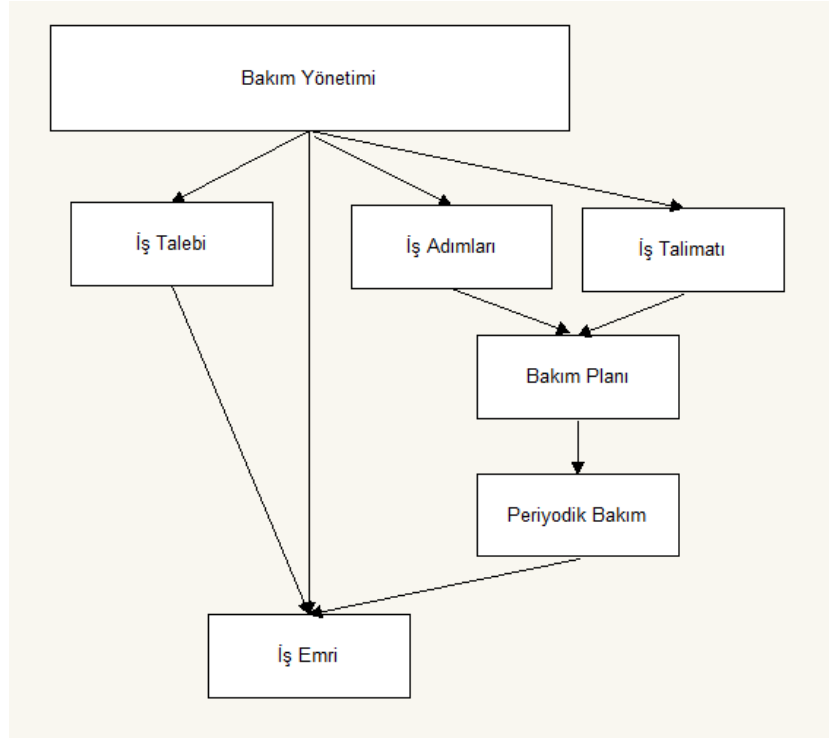
Varlık Kodu	Varlık Tanımı
01	AKELSAN-MESAR
10	KAPORTA ATÖLYE
10.00	SÖKÜM
92-163	İKARUS 260
93-523	İKARUS
94-2015	LİBERYA
97-110	MERCEDES 345
M1009	MERCEDES 345
10.01	KAROSER
34 23 5984	MERCEDES CİTARO O530
34 TS 9439	FORD TRANSİT AÇIK KASA KAMYONET
34 TS 9440	FORD TRANSİT AÇIK KASA KAMYONET
34R1Z77	YILDIZ TEKNİK
34TN0595	ISUZU ÇERKİÇ BELEDİYESİ
85-154	MAN

Şekil 3.10. Boysweb varlık ağacı gösterimi.

3.8.2. Bakım Yönetim Sistemi

BOYS bakım yönetim sistemi modülünde sisteme tanımlanmış olan ekipmanlar için arızı bakım, koruyucu bakım, kestirimci bakım, planlı bakım gibi bakımların yapılmasını sağlar. Bu bakımlar sisteme iş emirleri olarak açılır. Arızı bakım için iş emri oluşturulmadan önce bir ekipmanda arızı bakım olduğunu bildiren üretimin durup durmadığını haber veren iş bildirimi açılır. Bu bildirim BOYSWEB sisteminde iş talebi denir. Açılan iş talebi operatörün onayından geçerek iş emrine dönüşür. Planlı ve koruyucu bakımlar sisteme bildirilmeden önce, planlı bakımın hangi adımlar tamamlanarak yapılacağı bilgisinin tutulması gerekir. Öncelikle iş adımları belirlenir ve sisteme tanımlanır. İş adımları bir plan dahilinde uygulanacağı için bu iş adımları bakım planlarına bağlanır ve bir paket haline getirilir. Yapılacak olan planlı bakımların hangi

tarikh, saatlerde hangi periyotlarda ya da hangi sayaç değerlerinde yapılacağı sisteme tanımlanır. Bu tanıma periyodik bakım tanımı denir. İlgili bakım tanımı içerisine bakım planı eklenir. Günü ya da sayacı gelen bakım için sistem tarafından iş emri oluşturulur. İşi yapacak operatörlerin ekranına bu iş emirleri düşer.



Şekil 3.11. Bakım yönetim sistemi modülleri.

Şekil 3.11’de bakım yönetim sistemi içerisinde yer alan modüller yer almaktadır. Bakım yönetim sisteminin ana modülleri iş talebi ve iş emri modülleridir. İş talebi ve iş emri modülleri üzerinden bakım işlemleri gerçekleştirilmektedir. Planlı bakım işlemleri için ise periyodik bakım modülü kullanılmaktadır. Periyodik bakım modülünün işleyebilmesi için sistemde iş adımları veya iş talimatları tanımlanmalıdır. İş adımları veya iş talimatları bakım planlarına bağlanır. Bakım planları ise periyodik bakım tanımlarına bağlanarak günlük, aylık, yıllık ya da sayaç değerlerine göre periyodik bakımlar oluşturulur. Zamanı gelen periyodik bakımlar için iş emirleri oluşturularak bakımların gerçekleştirilmesi sağlanır.

3.8.3. İş Talepleri

İş talebi bir işin yapılmasına yönelik gelen ilk ihbar bilgi olarak kabul edilir. İnceleme sonucunda şayet iş bildiriminde belirtilen işin yapılması gerekiyorsa, iş emri açılır. İş bildirimleri bakım birimi tarafından yapılabileceği gibi işletme veya başka birimler

tarafından da yapılır.

3.8.4. İş Emirleri

BOYS bakım yönetim sisteminde, bakım ile ilgili faaliyetler iş emirleri oluşturularak tanımlanır. İş emirleri menüsü sistemin en çok kullanılan kısmıdır. Yapılan veya yapılacak bakımın tanımlanması, ilgili birim, bakımın türü, bakımın tipi, tarihi, önceliği, işin süresi, harcanan iş gücü, maliyeti, görevli personeller, duruş gerektirip-gerektirmediği, harcanan malzemeler ve işin detay açıklaması iş emirlerinde belirtilir.[14]

3.8.5. İş Talimatları

BOYS iş talimatları menüsü, işlerin standartlaştırılması, kolaylaştırılması ve yapılacak özel işler hakkında personele bilgi aktarılması amacı ile oluşturulmuştur. Kalite Yönetim Sistemi iş talimatları bu menü altında oluşturulabilmektedir. Sistem bu şekilde kalite yönetim sistemi ile uyumlu halde çalışmaktadır.

3.8.6. Planlı Bakım-İş Adımları

Planlı bakım yapısı gereği öncelikle iş adımları oluşturulur. İş adımları oluşturulurken işi yapanların tecrübelerinden oluşan daha önce hazırlanılmış olan bir takım dokümantasyonlardan yararlanılır. Varlıkların tedarik edildiği satıcılardan alınan bakım dokümantasyonları kullanılır.

3.8.7. Bakım Planları

Bakım planı hangi iş adımlarından oluşması gerekiyorsa iş adımları havuzundan ilgili iş adımları seçilerek oluşturulur. Böyle bir yapı kullanıcıya bazı kolaylıklar sunmaktadır. Aynı bakım planının uygulandığı varlıklar için tek bir bakım planı oluşturularak ilgili bütün varlıklarla ilişkilendirilebilir.

3.8.8. Koruyucu Bakım Tanımları

Planlı bakımda varlık ilişkisinin kurulduğu adım planlı bakım tanımlamaları adımdır. Varlıklarla ilgili planlı bakım tanımlamaları yapılırken bir önceki adımda oluşturulan bakım planları kullanılmaktadır.

4. BAKIM ONARIM VARLIK YÖNETİM SİSTEMİ YAZILIM GELİŞTİRMELERİ

4.1. BOYSWEB ÖNBELLEKLEME YAKLAŞIMLARI

Önbellekleme web sayfasının sıklıkla erişileceği sayfaların bir kısmını ya da tamamını veya belirli bir verinin bellekle tutulması işlemidir. Bu işlemi performansın iyileştirilmesi için yapılmaktadır. Bir verinin bellekten okunması veritabanı ya da diskten okunmasından çok daha hızlı sonuç vermektedir. Önbelleklemede dikkat edilmesi gereken nokta belleğe alınacak verinin iyi optimize edilmesi gerekir. Böylece bellek şişmesi ve programın hata vermesinin önüne geçilmiş olur.

4.1.1. AppFabric Nedir?

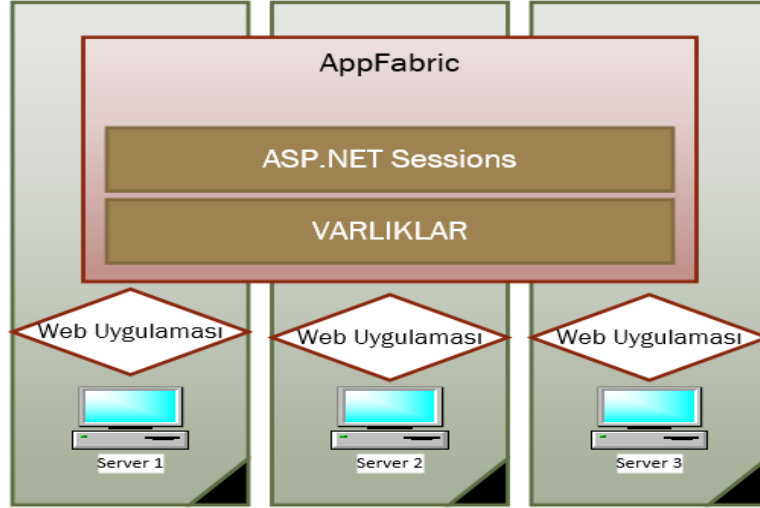
IIS (Internet Information Server), web sayfalarının yayınlanmasını ve web uygulamalarının çalışmasını sağlayan, istemcilerden HTTP ve FTP üzerinden gelen talepleri Microsoft Windows sunucu tabanlı işletim sistemlerinde karşılayan birimdir [15].

AppFabric, IIS üzerinde çalışır ve önbellek verilerini bilgisayar belleğinde saklar. Böylece sıklıkla kullanılacak olan veriler için sunucuya gidip gelmek yerine bilgisayar belleğinde saklanan bu veriler kullanarak programda hızdan ve maliyetten kazanç sağlanabilmektedir [16]. Bu kazancı sağlayabilmek ve belleğin çok fazla şişmesinin önüne geçebilmek için hangi verilerin appfabric tarafından cachelenmesi gerektiği kararının güzel verilmesi gerekir. Boys uygulaması appfabric üzerinde kullanıcı bilgilerini (kullanıcı adı, şifre vb.), uygulamada kullanılan gridlerin özelliklerini, uygulama ekranlarında kullanılan yazıların çoklu dil dosyalarını, listeleri, uygulamadan geri gönderilecek mesajları, uygulama ekranlarının sayfa renklerini saklamaktadır.

4.1.2. Boysweb Neden AppFabric'e İhtiyaç Duyar

Boys'u bir işletme tarafından kullanılırken, işletmenin bünyesinde yer alan varlıkların tamamı boys sistemine tanımlanır. Veritabanında kayıtlı olan her bir varlık için binlerce

kullanıcı boys sistemine girerek bu varlıklara ait arızaları bildirmek için veritabanına iş talebi ve iş emri kayıtları atacaktlardır. Böylece veritabanına milyonlarca başvuru olacaktır. Veritabanı bu yükü ne ölçüde kaldırabilecek ve bu isteklere ne kadar hızla yanıt verebilecek soruları ortaya çıkmaktadır. Kullanıcıların çok fazla beklemlerin önüne geçmek, hatta aşırı yüklenmelerden kaynaklı sistem çökmelerinin önüne geçebilmek adına AppFabric uygulamasından yardım alınmaktadır.



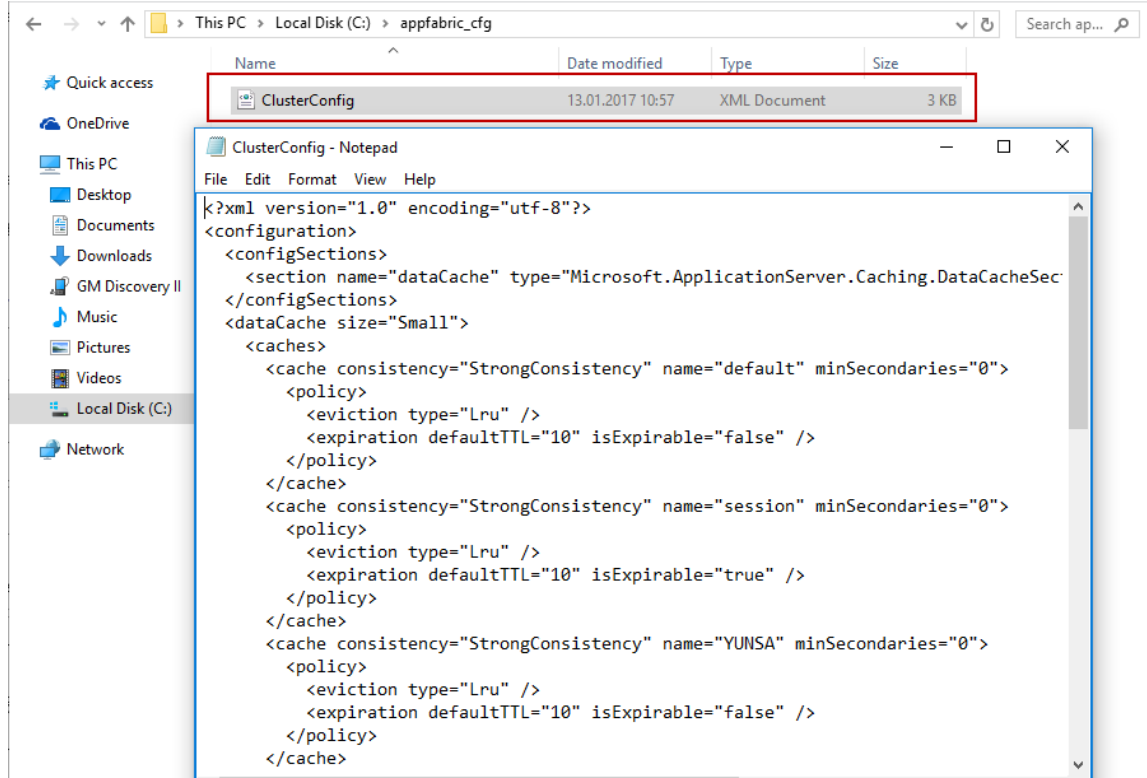
Şekil 4.1. Boysweb AppFabric uygulama şeması.

AppFabric uygulaması ile kullanıcılar, varlıklar, varlıklar sayfasındaki sayfa özellikleri ve düzenleri, sayfada kullanılan listeler, mesajlar önbellekte tutulmaktadır. Böylece yeni bir varlık eklenmediği veya var olan bir varlık güncellenmediği takdirde tüm varlıklar önbellek sunucularına yazılacaktır. Boys'a giren tüm kullanıcılar varlıklara ulaşmak için veritabanına başvurmayacaktır. Verileri önbellek sunucularından hazır olarak alacaklardır. Bu sayede veritabanına yapılan başvuru sayısı ve yükü azalacaktır. Eskiye nazaran çekilecek olan veriler için harcanan süre kısılacaktır. Şekil 4.1'de farklı sunucular üzerinden boys uygulamasına erişilmeye çalışıldığında AppFabric uygulaması ile önbelleğe alınan verilerin şeması yer almaktadır.

Services (Local)		Name	Description	Status	Startup Type	Log On As
AppFabric Caching Service		AppFabric Caching Service	Serves as an in-memory ...	Stopped	Manual	Network S...
AppFabric Event Collection Service		AppFabric Event Collection Service	Collects events from an ...	Running	Automatic	Local Service
AppFabric Workflow Management Service		AppFabric Workflow Management Service	Manages the execution o...	Running	Automatic	Local Service

Şekil 4.2. AppFabric kurulumu sonrası kurulması gereken windows servisi.

AppFabric uygulamasını kurduktan sonra. Şekil 4.2’de gösterilen şekilde bir windows servis kurulumunun gerçekleştirilmesi gerekir. Bu servis çalıştığı sürece appfabric uygulaması önbelleğe verileri kaydetmeye devam edecektir. Önbelleğin temizlenmesi istendiğinde bu servisin tekrar çalıştırılması yeterli olacaktır.



Şekil 4.3. Appfabric kurulumu sonrası konfigürasyon ayarlarının yapılacağı klasör.

Appfabric uygulamasının configure edilebilmesi için Şekil 4.3’deki gibi bir klasör oluşturulur. Bu klasöre hangi bilgisayarlardan boys uygulamasına ve appfabric uygulamasına erişilmesi isteniyor ise ona göre paylaşım yetkisi verilir. Böylece appfabric uygulaması önbelleğe verileri alırken yetki problemiyle karşılaşılmasının önüne geçilir. Bu klasör içerisine xml türünde bir dosya oluşturulur. Boys uygulaması birden fazla veritabanı ile çalışabilmektedir. Her bir veritabanı boys uygulaması içerisinde sunucu olarak tanımlanmaktadır. Bu durumda her bir sunucu için ayrı ayrı önbellekleme işlemi yapılması gerekmektedir. Oluşturduğumuz ClusterConfig.xml dosyası içerisinde her bir sunucu için ayrı ayrı konfigürasyon bilgisi ayarlaması yapılır. Böylece her bir sunucu için ayrı ayrı önbellekleme işlemi yapılabilir. Bu konfigürasyon ayarında appfabric uygulamasının hangi milisaniyeler arasında önbelleğe alacağı verileri kontrol edeceği, bu sunucu için kaç birim belleğe kadar önbelleğe alma işlemine devam edileceği gibi bilgiler girilir.

```
Administrator: Windows PowerShell

PS C:\WINDOWS\system32> use-cachecluster
PS C:\WINDOWS\system32> stop-cachecluster

HostName : CachePort      Service Name      Service Status Version Info
-----
kkocer.bimser.local:22233 AppFabricCachingService DOWN          3 [3,3][1,3]

PS C:\WINDOWS\system32> start-cachecluster

HostName : CachePort      Service Name      Service Status Version Info
-----
kkocer.bimser.local:22233 AppFabricCachingService UP              3 [3,3][1,3]

PS C:\WINDOWS\system32> get-cache

CacheName      [Host]
-----
Regions
-----
default
session
YUNSA          [kkocer.bimser.local:22233]
PageSettings(PPrimary)

CIMTAS
TEST
DOCO

PS C:\WINDOWS\system32> use-cachecluster
PS C:\WINDOWS\system32> stop-cachecluster

HostName : CachePort      Service Name      Service Status Version Info
-----
kkocer.bimser.local:22233 AppFabricCachingService DOWN          3 [3,3][1,3]

PS C:\WINDOWS\system32> start-cachecluster

HostName : CachePort      Service Name      Service Status Version Info
-----
kkocer.bimser.local:22233 AppFabricCachingService UP              3 [3,3][1,3]

PS C:\WINDOWS\system32> get-cache

CacheName      [Host]
-----
Regions
-----
default        [kkocer.bimser.local:22233]
Default_Region_0640(Primary)
Default_Region_0918(Primary) Default_Region_0678(P
rimary)
Menus(Primary) Default_Region_0775(Primary)

session
YUNSA          [kkocer.bimser.local:22233]
Default_Region_0640(Primary)
PageSettings(PPrimary) Default_Region_0567(Primary)

Default_Region_0678(Primary) Menus(Primary)
Default_Region_0918(Primary)
```

Şekil 4.4. Boysweb uygulaması AppFabric konfigürasyonu terminali.

Appfabric servisinin tekrar çalıştırılması, boys uygulamasına eklenen yeni sunucuların appfabric uygulamasına tanımlanıp konfigürasyonunun yapılması, appfabric uygulamasında hangi sunucuların önbelleklendiğini ve bu sunucunun hangi özelliklerinin önbelleklendiğini anlık görebilmek için Windows PowerShell uygulamasından yararlanılmaktadır.

Use-CacheCluster : AppFabric uygulaması komutlarını kullanılacaktır.

New-Cache SUNUCU1: Boys uygulamasının kullanacağı yeni sunucuları

tanımlamaktadır.

Remove-Cache SUNUCU1: Boys uygulamasının kullanmaktan çıkardığı sunucuyu Appfabric uygulamasında önbelleklemekten çıkarması için kullanılır.

Start-CacheCluster: AppFabric servisini çalıştırarak önbellekleme işlemine başlanılmasını sağlar.

Stop-CacheCluster: AppFabric servisini durdurarak önbellekleme işlemine son verilmesini sağlar.

Get-Cache: Hangi sunucuda hangi verilerin önbellekleme işlemine alındığını anlık gösterir.

Şekil 4.4’de SUNUCU1 sunucusunun pagesettings ve menus verilerini önbelleğe aldığı görülmektedir. Bu işlemi yapabilmek için SUNUCU1 sunucusuna yani SUNUCU1 veritabanına girilerek iş talebi sayfasına giriş yapmamız yeterli olmuştur. Böylece iş talebi sayfasını ve ana ekrandaki menü bilgilerini AppFabric önbelleğe almıştır.

4.1.3. Redis Nedir, Redis ve AppFabric Farkları

Açık kaynak kodlu bir key-value (anahtar - değer) mantığıyla çalışan bir depolama aracıdır. AppFabric veriyi bellekte tutar ama bu veriyi dosyaya yazmaz. Sunucu açık kaldığı sürece veriyi saklar. Sunucu kapatıldığında veriyi bellekten siler. Ancak Redis belirli aralıklara veriyi dosyalara kaydedebilir [17].

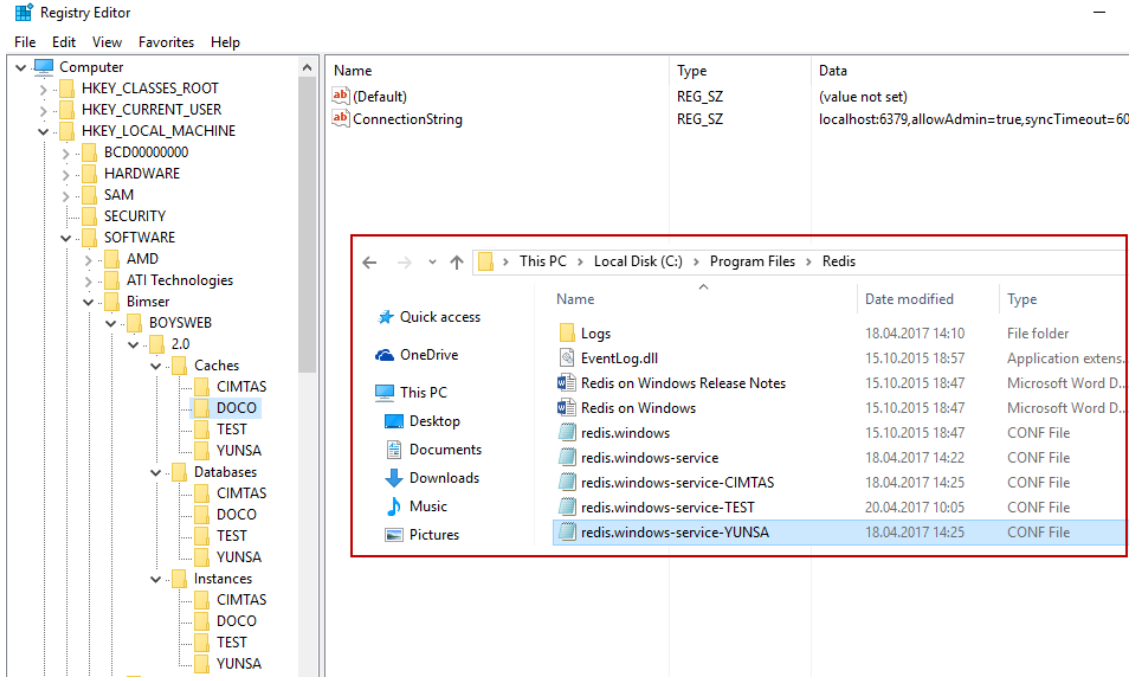
Appfabric sadece veri giriş ve çıkışı yapmaktadır. Veriyi alır sonra kullanılmak istendiğinde veriyi geri verir. Ancak Redis veri yapıları kullandığı için birçok fonksiyonu desteklemektedir.

Redis ile string (karakter kümesi), hash (Map<String, String>), list (liste), set (küme), sorted set (sıralı küme) veri tiplerinde veriler saklanabilir [18].

4.1.4. Redis Kurulumu ve Konfigürasyonu

Redis uygulamasının, redis desktop manager uygulamasının exe’leri çalıştırılır ve uygulama kurulumları yapılır.

Şekil 17’de Redis.windows-service Redis’in konfigürasyon dosyasıdır. Her sunucu için ayrı konfigürasyon dosyası tanımlaması yapılır. Regedit altında databases ve Instance altına Caches olarak bir klasör tanımlanır. Her sunucu için cache isminde klasör caches üst klasörü altına tanımlanır.



Şekil 4.5. Redis kurulum konfigürasyon dosyası.

Şekil 4.5’de Caches klasörü altına tanımlanan cache klasörleri için connectionstring key i tanımlanır. Default port 6379 dur. Yeni eklenecek her sunucu için bu port 1 arttırılır. Her sunucu farklı portta önbellekleme yapmaktadır. Bu sebepten klasör yapısı regedit altında Şekil 4.5’deki gibi dizayn edilmiş olur. Her bir sunucu için yeni bir windows servis oluşturulur.

ConnectionString:

localhost:6379,allowAdmin=true,syncTimeout=60000,connectTimeout=15000

Redis Desktop Manager 0.9.3.817

Connect to Redis Server

EBA

- db0 (7)
 - Config.Keys
 - DMIIdentities
 - Identities
 - Organization
 - OsPropertyCache
 - PasswordCache
 - Security
- db1 (0)
- db2 (0)
- db3 (0)
- db4 (2)
- db5 (0)
- db6 (0)
- db7 (4)
- db8 (0)
- db9 (0)

EBA::db0::DMIIdentities

HASH: DMIIdentities

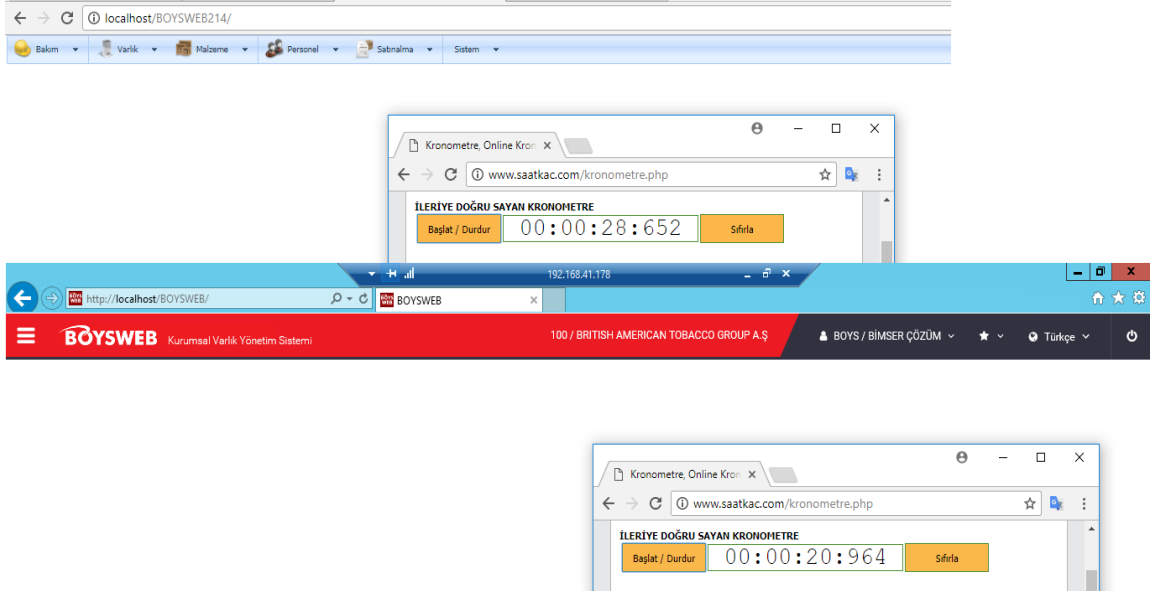
row	key	value
1	1.FOLDERSECURITY	6
2	1.FOLDERS	30
3	1.FILES	73
4	1.FILEPARENTS	73
5	1.FILEDATASET	72
6	1.FILEDATA	72
7	2.FOLDERSECURITY	10
8	2.FOLDERS	1
9	3.FOLDERSECURITY	3
10	3.FOLDERS	2
11	4.FOLDERSECURITY	1
12	4.FOLDERS	33
13	4.FILES	185
14	4.FILEPARENTS	185
15	4.FILEDATASET	185
16	4.FILEDATA	185

Key: size: 0.00 bytes

Şekil 4.6. Redis masaüstü uygulaması önbelleğe alınan verinin gösterilmesi.

Şekil 4.6’da Redis masaüstü uygulaması önbelleklenmiş verinin yapısı gösterilmektedir. Connect to redis server butonu yardımı ile her bir sunucu için yeni bir instance name klasörü tanımlanılmıştır. Her bir cache klasörü içerisinde 15 adet db yer almaktadır. Bunlardan db0 içerisinde label, list gibi XML yapıdaki listelerin yer aldığı dosyalar tutulmaktadır. Db1 de company, user bilgileri tutulmaktadır. Aynı şekilde ilk 5 db dolu olarak gelmektedir. Diğer db ler boştur doldurma işlemleri ar-ge sırasında kod tarafından geliştirilmektedir. Cache kontrolü sırasında JSON değeri seçilerek cachelenmiş verilere Şekil 4.6’daki gibi erişilmektedir.

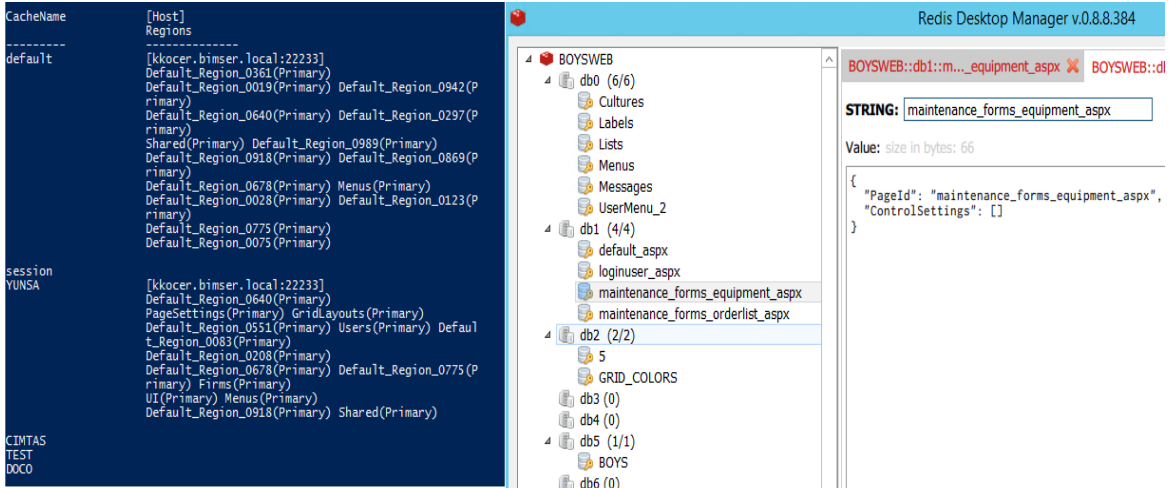
Redis uygulamasında yer alan 15 adet db değeri ve bu değerler içerisinde eklenebilecek alt kırılım verileri ile AppFabric uygulamasına göre Redis uygulaması önbelleğe alınacak verilerin yönetilmesi konusunda daha kullanışlıdır. Daha çok sayıda ve daha fazla verinin önbelleğe alınması, okunması, yönetilmesi ağaç mantığı ile sağlanmaktadır. Böylece önbellek verileri daha düzenli tutulmaktadır.



Şekil 4.7. Appfabric – Redis teknolojisi alt yapısı ile geliştirilen boysweb uygulamasının ilk yüklenme süresi karşılaştırması.

Şekil 4.7’de Önbellekleme ilkelerinde Appfabric teknolojisi kullanılarak geliştirilen Boysweb 2.14 versiyonunda önbelleğe alınmış veriler silindikten sonra uygulama ilk açılış süresi 28.6 saniyedir. 28.6 saniyede appfabric uygulamasının önbelleğine arka planda istediğimiz veriler alınmakta ve sayfa çağrılarak yüklenmektedir.

Önbellekleme ilkelerinde Redis teknoloji kullanılarak geliştirilen Boysweb 2.15 versiyonunda önbelleğe alınmış veriler silindikten sonra uygulama ilk açılış süresi 20.9 saniyedir. 20.9 saniyede redis uygulamasının önbelleğine istenen veriler alınmış ve sayfa yüklenmiştir.



Şekil 4.8. Appfabric – Redis uygulamaları önbelleklenmiş veri karşılaştırması.

Şekil 4.8’de appfabric ve redis uygulamalarının önbelleğe aldıkları verilerin yapısı gösterilmiştir. Şekil 4.7’de 28.6 saniyede yüklenen Boysweb 2.14 uygulamasının önbelleğini tutan Appfabric uygulamasının tuttuğu veriler Şekil 4.8’de sağ kısımda gösterilmiştir. Aynı şekilde Şekil 4.7’de 20.9 saniyede yüklenen Boysweb 2.15 uygulamasının önbelleğini tutan redis uygulamasının tuttuğu veriler Şekil 4.8’de sol kısımda gösterilmiştir.

Appfabric uygulaması içerisinde firma, menu, sayfa isimleri, sayfa ayarları, sayfa tagları tutulmaktadır. Appfabric teknolojisinin bir masaüstü programı olmadığı için Powershell uygulaması aracılığı ile tutulan verilerin sadece taglarına ulaşılabilir.

Redis uygulaması içerisinde firma, menu, sayfa isimleri, sayfa ayarları, sayfa tagları, açılan her sayfanın kontrol ayarları tutulmaktadır. Ayrıca Redis uygulaması içerisinde 15 adet db ismi verilen boş önbellek depoları bulunmaktadır. Tutulan bu verilere ek olarak eklenebilecek yeni kümelenmiş veriler boş db alanlarında gruplar halinde yeni isimler verilerek tutulabilir. Bu kontrolleri ve ayarları yapabilmek için Redis teknolojisinin masaüstü uygulaması Şekil 4.8’de görüleceği şekilde mevcuttur. Masaüstü uygulaması sayesinde önbelleğe alınmış veriler Boysweb uygulamasının arka kısmında kodlama aşamasında görülmesi yerine direk masaüstü uygulaması içerisinde kümelenmiş şekilde JSON vb. veri tiplerinde Şekil 4.8’deki gibi görüntülenebilir. Ayrıca önbelleklenmesi istenen yeni veriler yine masaüstü uygulaması sayesinde kolaylıkla önbellek teknolojisine aktarılabilir.

Redis uygulaması daha fazla önbelleğe alınmış veri ve daha düzenli grup veriler şeklinde 20.9 saniyede Boysweb uygulamasını yükleyebilirken, appfabric uygulaması daha az veri önbelleğe alarak, verileri kümelemeden ve geliştiriciye direk göstermeden 28.6 saniyede Boysweb uygulamasını yükleyebilmiştir.

4.1.5. Önbellekleme İşlemleri Sonucu Kazanımlar

Boysweb uygulamasında önbellekleme işlemini gerçekleştirebilmek için güncel ve popüler olarak kullanılan iki uygulama, Boys altyapısına eklenip konfigure edilerek test edilmiştir. AppFabric ve Redis uygulamalarından hangisinin daha az maliyetli olacağı, hangisinin daha hızlı ve programı yormayacak şekilde önbellekleme işlemini gerçekleştirebileceğini test edebilmek için önce AppFabric uygulaması Boys uygulamasına dahil edilmiş hız ve kabiliyetleri gözden geçirilmiş daha sonra Redis uygulaması Boys uygulamasına dahil edilmiş, hız ve kabiliyetleri gözlemlenmiştir.

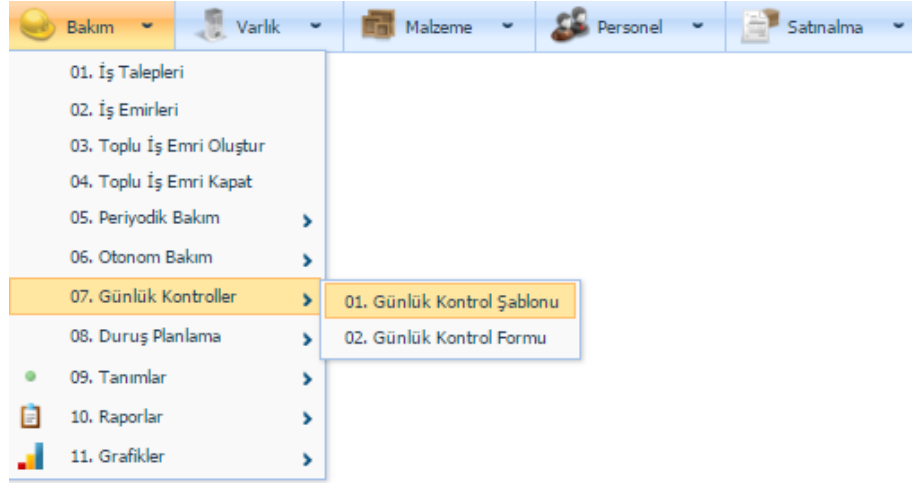
Appfabric ve Redis uygulamaları için iki uygulamanın da önbelleğe alınmış verilerinin listesi çekildiğinde Redis uygulamasında önbelleğe alınan veri daha düzenli ve daha okunabilir durumdadır. Redis uygulaması daha çok ve daha küçük ölçekli verileri önbellekleyebilmekte ve Şekil 4.8’de de görülebileceği üzere bu verileri kümeleyerek düzenli halde önbellekte saklayabilmektedir. Ayrıca istendiği takdirde Redis uygulaması önbelleğe aldığı verileri dosyaya yazabilmektedir. Böylece sunucu kapatıldığında dahi geliştiricinin kaybolmasını istemediği veriler dosyalarda saklanarak sunucu açıldığında tekrar kullanılabilir hale getirilmektedir. Bununla birlikte Redis uygulaması Appfabric uygulamasına oranla verileri daha hızlı önbelleğe almakta ve daha hızlı uygulamaya tekrar geri dönmektedir.

4.2. PERİYODİK(GÜNLÜK) KONTROL MODÜLÜNÜN GELİŞTİRİLMESİ

Firmalardaki bazı ekipmanların arızı bakım yapılmasına gerek kalmadan, duruş yaşanmadan önce bakımlarının günlük olarak yapılması gerekir. Bu ekipmanların kontrolü hayati önem taşıyabilmektedir. Bu ekipmanların bozulması ekipman hattını tamamen duruşa geçirebilmektedir. Tamiri uzun sürmekte yedek parça ve işçilik maliyeti ekipman maliyetine yaklaşmaktadır. Duruş süresinde meydana gelen duruş kaybı maliyetinin telafisi zor olmaktadır. Bu ekipmanların bakımlarının periyodik olarak yapılmaması durumunda oluşacak arıza, çalışan personellerin hayatını ve çalışma güvenliğini riske atmaktadır.

Örneğin buhar kazanında sensör yerleştirilen noktalardan günlük olarak ölçümler yapılmaktadır. Ölçüm sonuçlarında sıcaklık, basınç, nem gibi değerler tolerans değerlerinin üzerinde ya da altında çıkar ise sistem tarafından acil kodunda iş emri oluşturulmaktadır. Bu durumu mevcut BOYSWEB sistemi üzerinde tutulabilmektedir. Ancak bir işletmede buhar kazanı ya da buhar kazanı tarzında risk sınıfına girecek ekipman olmadığı durumda her bir ekipman için ayrı ayrı iş adımları ve ölçüm tanımları belirlenerek iş emri oluşturulmaktadır. Bu durum sistemde karışıklığa sebep olmaktadır. Operatörler acil olarak yapılması gereken ve günlük olarak oluşan bu iş emirlerini diğer iş emirleri içerisinde sızdırmaktadır. Personellerin çalışma planlarını bu iş emirlerine göre ayarlamaya çalışılır. Bu durum iş takibini karmaşık hale getirmektedir. Bazı gözden kaçan iş emirleri günlük yapılması gereken bakımın yapılmamasına sebep olmaktadır. Firmanın bu iş emirleri yüzünden iş güvenliği riskini çığnediği yönünde olumsuz dönüşler almasına sebep olmaktadır. Şekil 4.9’da günlük kontrol modülünün bakım yönetimi altında hangi

menüde yer aldığı gösterilmektedir. Hangi alt katmanlardan oluştuğu bilgisi yer almaktadır. Günlük kontrol modülü kontrol şablonu ve bu şablonun tetiklenmesi sonucu kontrol formlarından meydana gelmektedir.



Şekil 4.9. Günlük kontrol modülü.

Böyle bir modülün geliştirmesinin amacı; birden fazla ekipman için birden fazla bakım planı kullanılarak oluşturulacak olan bakımların günlük olarak toplu oluşmasının sağlanmasıdır. Günlük kontrol formlarında her bir ekipman için her bir bakım planı içerisindeki iş adımının yapılması sağlanır. İş adımı yapıldıysa ne kadar sürede yapıldığı bilgileri girilir. Sorun olan iş adımları için otomatik iş emri oluşturulması sağlanır.

Ekipmanın bakımında ölçümler yapılması gerekiyor ise; önce sisteme ölçüm türleri (sıcaklık, basınç, nem vb.) tanımlanır. Ekipman üzerindeki ölçüm noktalarının seçilmesi sağlanır. Bu ölçümün gerçekleşmesi sırasında seçilen ölçüm türlerinin ortam şartları ne olması gerektiği belirlenir. Standart değerlerinin ne olması gerektiği bilgilerinin girildiği ölçüm paketleri tanımlanır. Ekipmanlar için hangi ölçüm paketleri kullanılacak ise bu ölçüm paketleri de seçilir. Günlük kontrol gerçekleştirilirken ölçüm değerleri dışında olan ölçüm paketi için sorun olduğu sisteme bildirilir. O ekipmana ait ölçüm noktası baz alınarak yeni iş emri oluşturulur. İlgili operatörlerin lokasyona gitmeleri istenir. Sonuç olarak kritik ekipmanların günlük kontrolü yapılarak tespit edilen hasarlar ya da kusurlar belirlenir. Bu hasarın üretim durmadan ya da ekipman bozulmadan önce onarımı ve bakımı gerçekleşir. Sistemden çekilen raporlar neticesinde de o ekipman için iş gücü ve bakım maliyeti miktarı hesaplanır. Ne sıklıkla arıza oluşmuş bilgileri toplanarak yöneticilere sunulur.

4.2.1. Günlük Kontrol Şablonu

Günlük kontrol şablonunda tüm şablonların listelendiği bir sayfa mevcuttur. Bu sayfaların içerisinde değişiklik yapılması ve yeni şablon oluşturulabilmesi için tasarlanmış ikinci bir sayfa mevcuttur. Şekil 4.10 tüm şablonların listelenmesi için hazırlanan sayfadır.

Günlük Kontroller

Filtre

1

2

10

12

Drag a column header here to group by that column

Kayıt Kullanımda	Şablon Numarası	Bakım / Arıza Ko	Bakım / Arıza Tai
	16	1000	ARIZA
	17	1000	ARIZA
	12	1003	KALİBRASYON
	10	1004	AYLIK SÖZLEŞME
	11	1005	YILLIK SÖZLEŞME
	6	1005	YILLIK SÖZLEŞME
	9	1005	YILLIK SÖZLEŞME
	18	1006	Görevli
	8	1006	Görevli
	13	1006	Görevli

Şekil 4.10. Günlük kontrol şablonu.

Günlük kontrol şablonu oluşturulurken hangi varlıklar için kontrol yapılacağı bilgisi varlıklar sekmesinde girilmektedir. Bu varlıklara hangi iş adımları uygulanacağı bilgisi bakım planları sekmesinde girilmektedir. Şekil 4.11 günlük kontrol şablonuna ölçüm paketleri, iş adımları ve ekipman bilgileri girileceği ekranları göstermektedir.

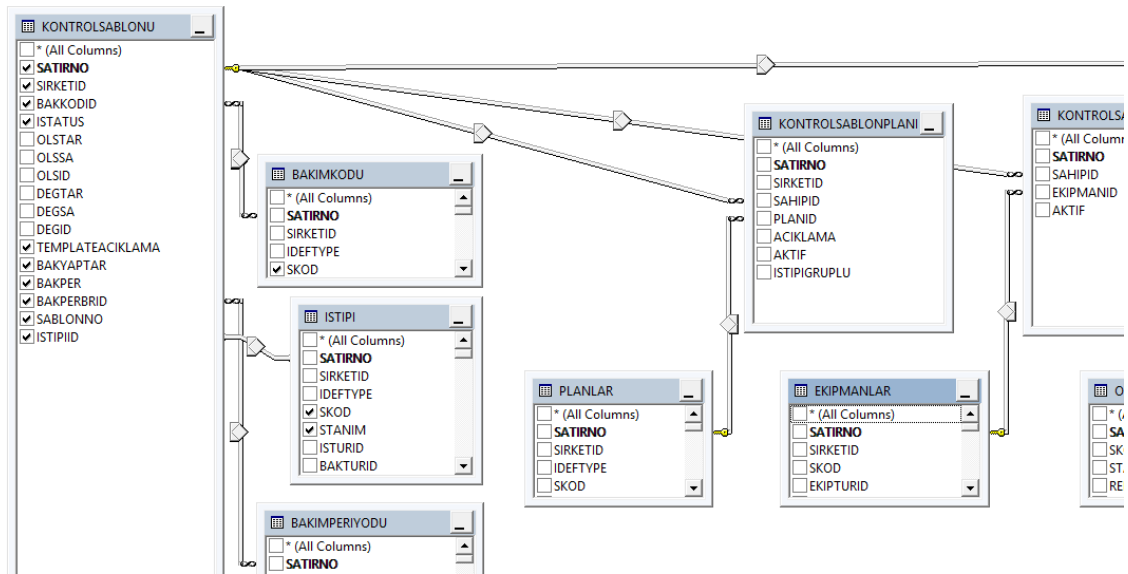
Günlük Kontrol Bilgileri		
Şablon Numarası:	17	
Bakım / Arıza Kodu:	1000 ARIZA	
Varlıklar	Bakım Planı	
Aktif	Varlık Kodu	Varlık Tanımı
☒	107.CR-03	CR - 03 SOĞUK ODA TERMOMETRE / TERMOSTATLARI
☒	111.CR-07	CR - 07 SOĞUK ODA TERMOMETRE / TERMOSTATLARI
☒	124.CR-16	CR - 16 SOĞUK ODA TERMOMETRE / TERMOSTATLARI

Varlıklar	Bakım Planı	Ölçüm Paketleri	Açıklama
Plan Kodu	Plan Tanımı	Açıklama	
10000	AKÜLÜ FORKLİFT 3 AYLIK BAKIM PLANI		
10007	ARAÇ KALDIRMA LİFTİ (BİMSAN) HAFTALIK BAKIM PLANI		
10026	BOŞALTIMA PLATFORMU OTOMATİK KAPİ AYLIK BAKIM PLANI		

Varlıklar	Bakım Planı	Ölçüm Paketleri	Açıklama
Ölçüm Paketi Kodu	Ölçüm Paketi Tanımı		
AOE-002	10 GR ÖLÇÜM NOKTASI		
AOE-004	50 GR ÖLÇÜM NOKTASI		

Şekil 4.11. Günlük kontrol şablonunun doldurulması.

Günlük kontrol şablonu ana ekranında bakım periyodu, iş tipi seçimi, bakım arıza kodu bilgileri girilir. Bu bilgiler Şekil 4.12’de gösterilen KONTROLSABLONU tablosunda tutulur. Bakım periyodu satır numarası, iş tipi satır numarası, bakım kodu satır numarası KONTROLSABLONU tablosuna kaydedilir. Bu satır numaraları ile 1- 1 ilişki içerisinde olan BAKIMKODU tablosu, ISTIPI tablosu, BAKIMPERİYODU tablosundaki satır numaraları ilişki içerisinde olduklarıdır. Günlük kontrol şablonu sayfasında hangi ekipmanlara kontrol uygulanacağı bilgisinin girilmesi sağlanır. Bu bilgiyi veritabanı üzerinde sağlamak için KONTROLSABLONU ana tablosundaki satır numarası ile 1 – n ilişkisi içerisinde KONTROLSABLONUUEKIPMANI tablosundaki SAHIPID alanı eşleşmektedir. Böylece birden fazla ekipman bir kontrol şablonunda tanımlanabilir. Aynı şekilde KONTROLSABLONUPLANI ve KONTROLSABLONUOLCUMPAKETI tabloları üzerindeki SAHIPID alanları ile KONTROLSABLONU tablosu satır numarası 1 – n ilişkisi içerisinde olduklarıdır. Böylece birden fazla ölçüm paketi ve birden fazla bakım planı bir günlük kontrol şablonu içerisine tanımlanabilir.



Şekil 4.12. Günlük kontrol şablonu veritabanı ilişkileri şeması.

4.2.2. Günlük Kontrol Formu

Asıl işlemler günlük kontrol formu üzerinden işlemektedir. Oluşturulan şablonlar günlük olarak günlük kontrol formuna dönüştürülerek listeye düşmektedir. Bu liste üzerinden operatörler bakımlara müdahale ederler. Günlük bakım kontrol formu içerisinde tüm kontrol formlarının listelendiği bir ekran vardır. Ayrıca oluşan kontrol formlarının tamamlanıp kapatılmasını sağlayan ikinci bir ekran vardır. Şekil 4.13 günlük bakım

kontrol formu ekranı tüm formların listelendiği ekrandır. Bu ekran üzerinde hangi tarihler arasında hangi bakım arıza kodunda hangi kontrol formunun olduğu gösterilir. Kontrol Formunun statüsünün açık, kapalı, iptal statülerinden hangisine ait olduğu renklendirme ile gösterilir.

Günlük Bakım Kontrol Formu Filtre

1 1 100 23

Drag a column header here to group by that column

Kor	Kayıt Kullanımında	Kontrol No	Bakım / Arıza Ko	Bakım / Arıza Tai	Başlangıç Tarihi	Bitiş Tarihi
		41	1000	ARIZA	12.10.2015	12.10.2015
		42	1000	ARIZA	12.10.2015	12.10.2015
		44	1000	ARIZA	27.10.2015	27.10.2015
		46	1000	ARIZA	18.11.2015	18.11.2015
		47	1000	ARIZA	18.11.2015	18.11.2015
		33	1003	KALİBRASYON	18.8.2015	18.8.2015
		34	1003	KALİBRASYON	19.8.2015	19.8.2015
		35	1003	KALİBRASYON	19.8.2015	19.8.2015
		36	1003	KALİBRASYON	21.8.2015	21.8.2015

Kapalı Günlük Kontrol

Şekil 4.13. Günlük bakım kontrol formu ana ekranı.

Şekil 4.14’de kontrol formu üzerinde ekipmanlara iş adımlarının hangi sırada uygulandıkları listelenmektedir. Formu kapatacak olan operatör hangi iş adımlarını yaptığını ve hangi iş adımları yaparken sorun ile karşılaştığını bu form üzerinden sisteme aktarır. Aynı mantık ölçüm paketleri içinde geçerlidir. Hangi varlıklara hangi ölçüm paketleri uygulandığı listesi kontrol formu ölçüm paketleri sekmesinde listelenmektedir. Kullanıcı hangi ölçüm noktasında ölçüm dışı değerlerle karşılaştığının bilgisi bu ekranda sisteme aktarır.

Kontrol Formu Bilgileri

Genel Bilgiler Kontrol Yapılan Varlıklar Çalışan Personel İş Adımları Ölçüm Paketleri

İş Adımları			107.CR-03			
SNO	İş Adımı Kodu	İş Adımı Tanımı	Yapıldı mı?	Sorun Var mı?	Süre	Yapıldı mı?
0	1007	ATEŞLEME FONKSİYONU	☐	☐	0	☐
0	1008	AYAKLARIN VE PABUÇLARIN SAĞLAMLIĞI	☒	☒	12	☐
1	1008	AYAKLARIN VE PABUÇLARIN SAĞLAMLIĞI	☐	☐	0	☐
1	1070	EMNİYET SİVİÇLERİ	☐	☐	0	☐
2	1120	HALATLAR	☐	☐	0	☒
3	1159	KAPI YAYI	☐	☐	0	☐
4	1170	KIZAK VE MAKARALAR	☐	☐	0	☐

Şekil 4.14. Kontrol formu iş adımları listesi.

Kontrol formunda bakımı yapan operatör genel bilgiler sekmesinde statüyü kapalıya çeker. Başlangıç tarihi ve bitiş tarihi değerleri doldurulur. Bakım şablonundaki varlıklar kontrol formu üzerine aktarılmış şekilde gelir. Bakım sırasında alt ekipman ya da farklı bir ekipmana da bakım uygulanmış ise bu varlıklarda operatör tarafından kontrol formuna eklenebilir.

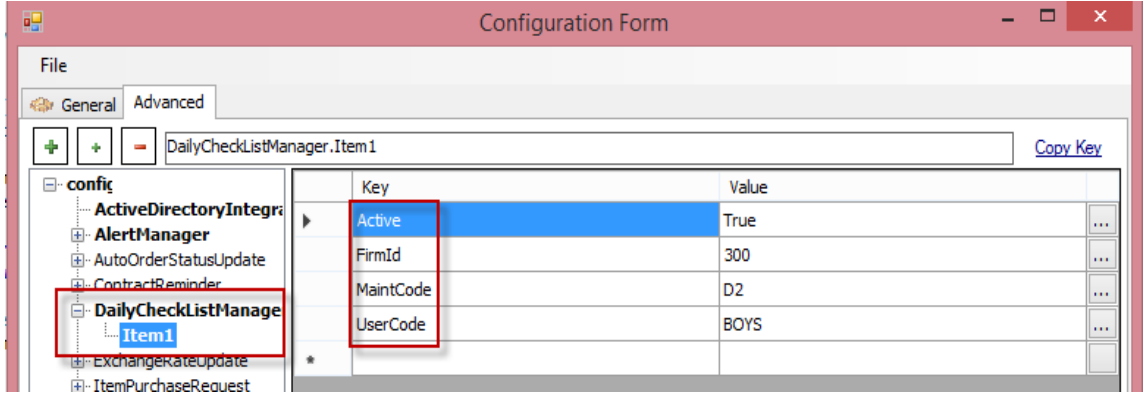
Çalışan personeller sekmesinde bakımı hangi personel ya da personellerin yaptığı, hangi zaman diliminde hangi mesaide yaptığı, kaç dakikalık bir süre harcayarak bu işi tamamladığı verileri girilir. Böylece bu bakımı yapmak için ne kadar bir maliyet ortaya çıktığı bilgisi hesaplanmış olur.

İş adımları sekmesinde, kontrol şablonunda seçilen bakım planlarının içerisinde tanımlı olan iş adımları kontrol formu iş adımları sekmesine aktarılır. Ekranın üst kısmında ekipmanlar, yan kısmında iş adımları listesi gelir. Böylece bir ekipman için bir iş adımının yapılıp yapılmadığı verisi girilebilir.

Kontrol Formu Bilgileri						
Genel Bilgiler	Kontrol Yapılan Varlıklar	Çalışan Personel	İş Adımları	Ölçüm Paketleri		
Ölçüm Paketleri		107.CR-03				
Ölçüm Paketi Kodu	Ölçüm Paketi Tanımı	Standart Değer	Ölçüm Değeri	Birim	Alt Kabul Değeri	Üst Kabul Değeri
AOE-002	10 GR ÖLÇÜM NOKTASI	10,00	0,00	GR	9,5	10,5
AOE-004	50 GR ÖLÇÜM NOKTASI	50,00	0,00	GR	49,5	50,5

Şekil 4.15. Kontrol formu ölçüm paketleri listesi.

Şekil 4.15’de kontrol formu ölçüm paketi sekmesi gösterilmiştir. Bu sekmede üst kısımda ekipmanlar yan kısımda ölçüm paketleri listelenen bir ekran mevcuttur. Bu tabloda hangi ekipman için hangi ölçüm paketi kullanılacaksa ve bu ölçüm paketi içerisindeki ölçüm noktasında ekipmanın standart değeri, ölçülen değeri ve bu değerlerin sisteme girilmesi sağlanır. Daha sonra standart değerler dışarısında çıkan ölçüm değeri için ilgili ekipmana iş emri oluşturulur.



Şekil 4.16. Günlük kontrol servisi ayar ekranı.

Kontrol şablonlarının günlük olarak kontrol formuna dönüşmesinin iki yolu vardır. İlki kontrol şablonu altındaki yeni kontrol formu butonu ile, ikincisi yazılmış olan günlük kontrol servisinin ayarlarının yapılarak çalıştırılması ile olur. Şekil 4.16’da günlük kontrol servisinin ayarlarının yapıldığı ekran görüntüsü verilmiştir. Bu ekran sayesinde hangi günde hangi saatte şablonun forma dönüşmesi isteniyorsa ayarlama yapılır.

Per	Kayıt Kullanında	Kontrol No	Bakım / Arıza Ko	Bakım / Arıza Tai	Başlangıç Tarihi	Bitiş Tarihi	Kontrol Açıklaması
		3	A1	SÖKÜM	21.12.2015	21.12.2015	TEST
		5	A1	SÖKÜM	23.12.2015	23.12.2015	TEST
		8	A1	SÖKÜM	23.12.2015	23.12.2015	TEST
		15	D2	ELEKTRİK SİSTEMLERİ	24.12.2015	24.12.2015	
		16	D2	ELEKTRİK SİSTEMLERİ	24.12.2015	24.12.2015	
		17	D2	ELEKTRİK SİSTEMLERİ	24.12.2015	24.12.2015	
		18	D2	ELEKTRİK SİSTEMLERİ	24.12.2015	24.12.2015	

Şekil 4.17. Servisten otomatik tetiklenen günlük kontrol formları.

Günlük kontrol şablonları tanımlanmış olan kayıtlar günlük olarak otomatik kontrol formuna dönüşmektedir. Kontrol formuna dönüşen kayıtlar personellerin Şekil 4.17’deki Günlük Kontrol Şablonu ana ekranına listelenmektedir. Personeller günlük olarak gelen formları ekranda görerek günlük kontrollerini gerçekleştirmekte ve bu formları kapatmaktadır.

Event Viewer				
App Name	BOYSWEB2 DailyCheckList I	Date	User	Text
Date Time	24.12.2015			Message
Hour	10:00 - 11:00			
	Refresh			
Type	All			
User				
Text				
Message				

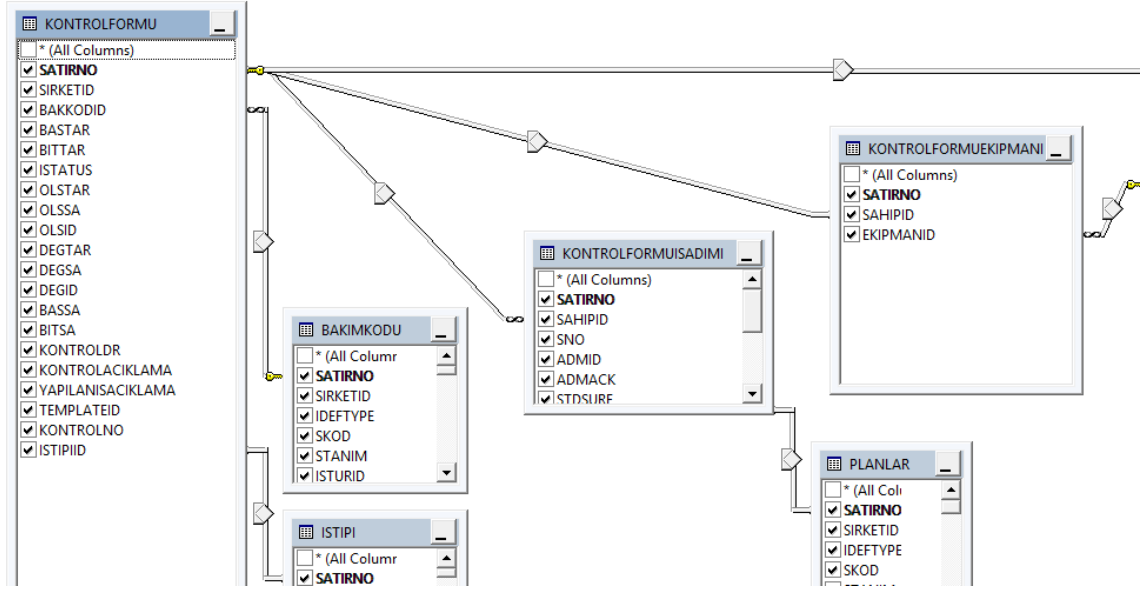
10:51:40	Service	KillProcess	Process Id: 2556
10:51:40	Service	KillProcess	Process exited already. Process Id: 15604
10:51:40	Service	Service stopped	
10:51:17	BISENERJI	Process exited	Process exited, process will be start again
10:51:17	Service	Starting mainte...	Starting instance service maintaining
10:50:52	YUNSA	Waiting	
10:50:52	YUNSA	Process compla...	FirmId: 300
10:50:52	YUNSA	There is no any...	FirmId: 300
10:50:52	YUNSA	Creating forms	SELECT * FROM V_DC_TEMPLATES WHERE SATIRN

Şekil 4.18. Otomatik oluşan günlük kontrol formlarının logları.

Otomatik oluşan günlük kontrol formlarının takip edilmesini sağlayan ekran Şekil 4.18’de gösterilmiştir

Bu ekrandan hangi kontrol formları oluşmuş, hangi kontrol formları hataya düşmüş görülebilir ve sisteme müdahale edilebilir.

Günlük kontrol formu ana ekranında seçilen bakım arıza kodu ve iş tipi seçimlerinin veritabanı kaydı BAKIMKODU, ISTIPI tablolarının satır numaraları ile KONTROLFORMU ana tablosunun ISTIPIID ve BAKKODID alanlarının 1 – 1 ilişki kurularak sağlanır. Günlük kontrol formu üzerinde birden fazla seçilebilen iş adımı, ölçüm paketi, ekipmanların veritabanı kaydı ise KONTROLFORMU ana tablosundaki satır numarası ile KONTROLFORMUISADIMI, KONTROLFORMUEKIPMANI, KONTROLFORMUOLCUMPAKETI 1 – n ilişki kurularak sağlanır. Şekil 4.19’daki veritabanı ilişkileri şeması bu durumu özetlemektedir.



Şekil 4.19. Günlük kontrol formu veritabanı ilişkileri şeması.

4.2.3. Periyodik Kontrol Modülü Değerler, Grafikler ve Kazanımlar

Son kullanıcının gireceği veriler doğrultusunda elde etmek istediği sonuçları kullanıcılara sunabilmek için Boysweb uygulaması üzerinde bazı raporlar ve grafikler geliştirilmektedir. Bu grafiklere çekilecek verilerin anlaşılır ve kullanıcı dostu olabilmesi için Visual Studio üzerinde DevExpress eklentisi kullanılmıştır.

4.2.3.1. DevExpress Nedir

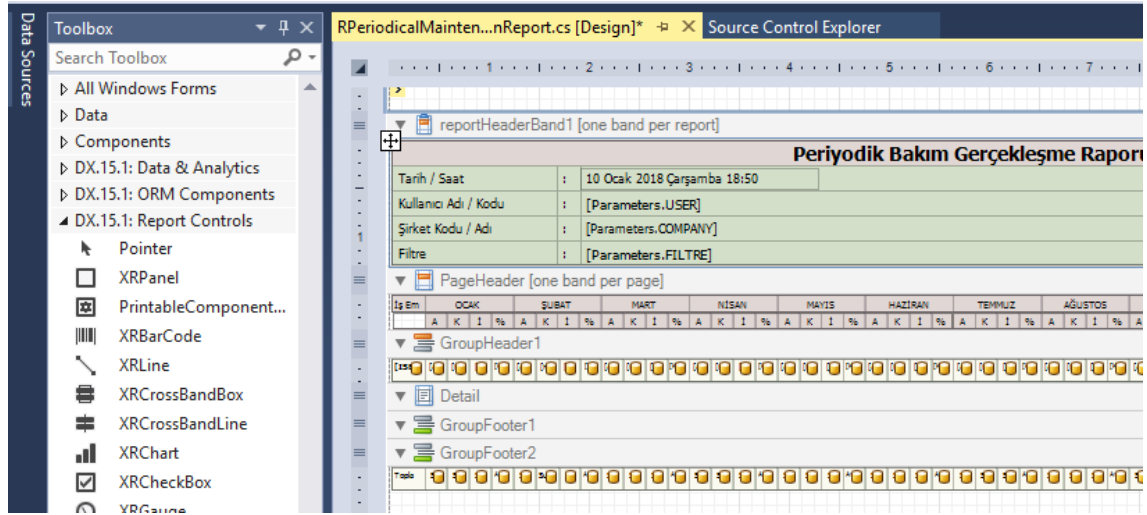
DevExpress, merkezi California'da bulunan küçük bir özel mülkiyetli işletme olarak 1998'de kuruldu ve mevcut teknolojilerle yeni nesil uygulamalar sağlıyor. Visual Studio ile ortaklık kurdu ve Windows Forms, ASP.NET'deki denetimlere ek özellikler sağladı [19].

Raporlama hizmetleri kısmına DevExpress paketinde XtraReports denir ve Visual Studio'ya tamamen entegre edilerek Windows Forms, ASP.NET ve LightSwitch gibi çeşitli platformlarda işlevsellik sağlar. Bu paket, bir geliştiricinin son kullanıcıya kendi raporlarını oluşturmaya izin veren bir son kullanıcı tasarımcı görevi görmektedir [20].

DevExpress tarafından sağlanan son kullanıcı formu, son kullanıcıya raporlar tasarlayabilir veya değiştirebilir. Bu formun en büyük avantajı, kullanıcıya ihtiyaç duyacakları tüm özelliklerin formda bulunması ve formun kullanıcı ihtiyaçlarına göre uyarlanması seçeneğidir [19].

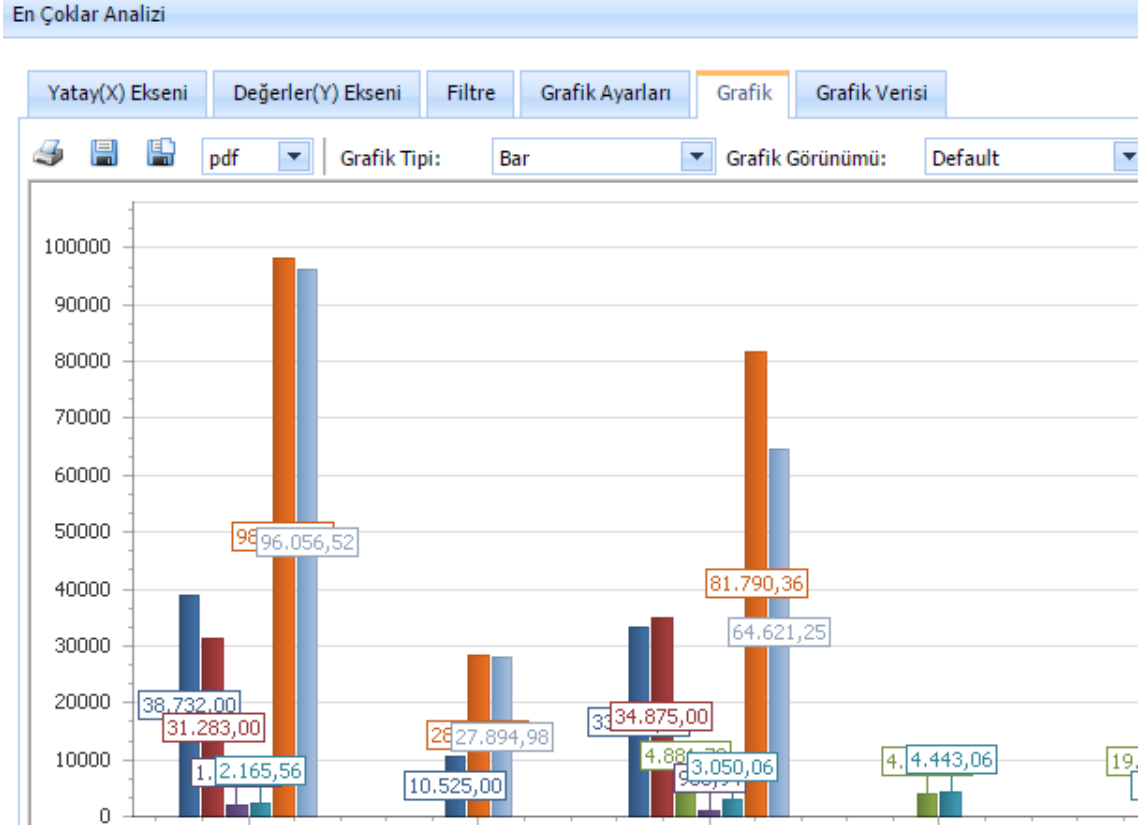
4.2.3.2. Günlük Kontrol Grafikler

Visual Studio üzerine eklenen Devexpress uygulaması kullanılarak Periyodik bakım gerçekleştirme raporu tasarlanmıştır. Rapora çekilecek veriyi Sql Server sunucusu üzerinde oluşturulan veritabanı içerisindeki tablolar birleştirilerek bir tablo yardımıyla alınmıştır. Alınan veri dperiodicalmaintenancerealizationreport1 isimli datasource tanımlanarak içerisine aktarılmıştır. Rapor için filtre paneli oluşturuldu. Devexpress uygulaması yardımıyla Visual Studio uygulaması içerisinde rapor dizayn edilmiştir. Dizayn aşaması Şekil 4.20’de gösterilmiştir.



Şekil 4.20. Periyodik bakım gerçekleştirme raporu tasarım ekranı.

Bakım operatörleri günlük kontrol şablonu oluşması sonucu önlerine gelen kontrol ve bakım işini tamamlayıp kontrol şablonunu kapatırlar. Kapatılan kontrol şablonunu gerçekleştirirken sorunla karşılaştıkları ekipmanlar için sistem tarafında otomatik iş emirleri oluşturulur. Oluşan iş emirleri neticesinde bakım operatörlerinin önlerine gelen yeni bakımları yapıp iş emrini kapalı statüsüne alırlar. Bu işlerin tamamlanmasında kullanılan kaynak sayısı, bu kaynakların işçilik süreleri hesaplanır. Ekipmanlar, oluşan arızada duruş yaşamışlarsa duruş süresi hesaplanır. Bakımların yapıldığı toplam bakım süresi hesaplanır. İşin bildirilişinde bakımın yapıp tamamlanmasına kadar geçen süre (Bakım + müdahale süresi) hesaplanır. Varlıkların bulundukları lokasyonlara (sarfyerine) göre operatörlerin bilgisi dahilinde hesaplanan veriler yöneticilere mail olarak gönderilir. Şekil 4.21’de en çoklar analizi grafiği ekranı gösterilmiştir.

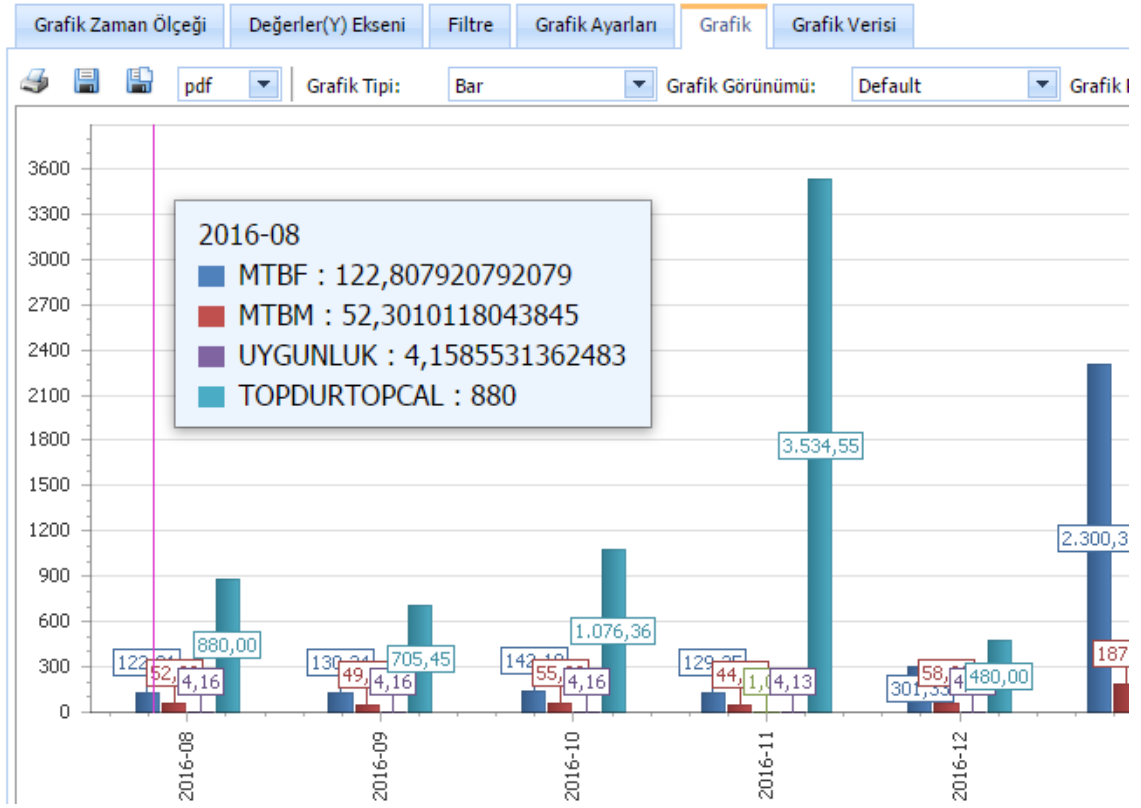


Şekil 4.21. Sarfyeri bazında kontrol formlarının en çoklar analizi.

Onarıma kadar geçen ortalama süre (MTTR) ve Onarılabilir sistemler için olası ardışık iki arıza arasında beklenen ortalama süre (MTBF) bilgileri hesaplanır. Ekipman, kısım, sarfyeri, işletme, kullanıcı, iş tipi, bakım arıza kodu gibi veriler bazında toplanan sürelerle göre grafikler ortaya çıkarılır. Oluşturulan bu grafik ekranına MtbF/Mttr Analizi denir. Şekil 4.22’de MTBF – MTTR dağılım grafiği ekranı mevcuttur.

$$MTBF = \text{Çalışma Süresi} / \text{Plansız Duruş Sayısı} \Rightarrow 1229 / 10 = 122.8 \text{ dk}$$

formülüne göre firmanın 2016 yılının 8. Ayında açılan işlerde, yapılan bakımlarda toplam 1228 dk çalıştığı varsayılmaktadır. 10 adet bakımda da makinanın durduğu varsayıldığında MTBF değeri 122.8 olarak çıkmış olacaktır. Burada firma MTBF değerini yükseltmek için plansız duruş sayısını azaltmak isteyecektir. Bu hedefi günlük kontrollerini düzenli olarak yeni yazılan periyodik (günlük) kontrol modülünden yaparak gerçekleştirebilir. Bakımlarını planlı bakım modülünden planlayıp düzenli olarak takip etmesi gerekir. Böylece bakımları planlı olarak yapılan ekipmanlar sağlıklı çalışacakları için ani duruşlara sebep olmazlar.



Şekil 4.22. Sarfyeri bazında kontrol formlarının grafiği.

4.3. MALZEME SAYIMI MODÜLÜNÜN GELİŞTİRİLMESİ

Firmalar mevcut işleyişlerini sürdürebilmeleri için hizmet oluşturmakta veya ürün ortaya koymaktadır. Bu süreçte hammadde almakta onları işlemekte daha sonra piyasaya sunmaktadırlar. Gelen hammaddelerin muhafaza edilip gerektiğinde kullanılabilmesi için bu hammaddeleri bünyelerinde bulunan ambarlarda tutmaktadırlar. Kullanılacak olan hammaddelere gerektiğinde pratik ulaşabilmeleri için ambarlarda yer, raf, göz gibi alanlar oluşturmaktadırlar. Hangi hammadde hangi ambarda, bu ambarın neresinde gibi bilgileri tutabilmeleri için bir sisteme ihtiyaç duymaktadırlar. Kullanılan hammaddelerin sonucunda bir ambarda ne kadarlık bir harcama olmuş geriye ne kadar hammadde kalmış bilgilerine ulaşabilmek için ambar sorumluları belirli periyotlarla ambarlara girerek hammaddeleri tek tek saymakta ve harcanan, kalan hammaddeleri tek tek not almaktadır. Bu aktiviteye malzeme sayımı denilir. Sayım tamamlandıktan sonra ambar sorumlusu aldığı notları sisteme girerek sistemdeki verileri belirli periyotlarla güncellemektedir.

Boysweb uygulamasında malzeme yönetimi modülü üzerinde firmalar hammadde bilgilerini tutabilmekte ve hangi malzemelerinin hangi ambarlarda olduğunu, bu ambarlar

içerisinde hangi yer, raf, göz üzerinde yer aldığını sisteme girebilmektedirler. Ayrıca malzeme hareketleri fonksiyonları aracılığı ile hangi malzeme hangi tarihte ambara girmiş, hangi tarihte ambardan çıkarak firma içerisinde kullanılmış ya da başka ambara transfer olmuş gibi bilgileri de boysweb uygulaması üzerinden yönetilebilmektedir.

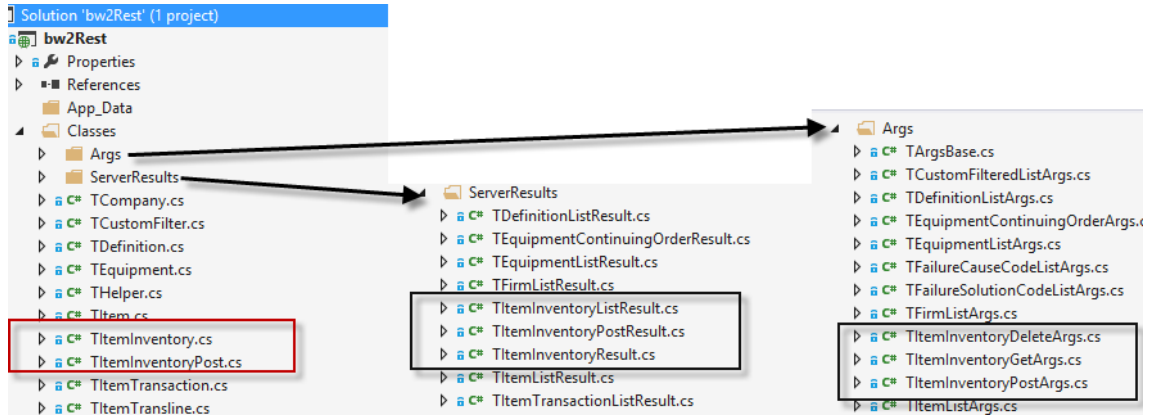
Firmaların Boysweb uygulaması içerisinde malzeme yönetiminde olmasını istediği yeni bir fonksiyon malzeme sayımı modulüdür. Firmalar belirli periyotlarda ambar sorumlularının ambara girerek malzemeleri tek tek sayıp daha sonra sisteme kaydederek vakit harcamalarını istememektedir. Ayrıca manuel olarak gerçekleştirilen bu uygulamada hata yapma riski çok yüksektir. Bunun yerine Boysweb'in mobil versiyonu üzerinde geliştirilen yeni bir modül ile her ambar sorumlusunun elinde bulunan el terminali yada cep telefonu ile uygulamaya girerek cihazın kamerasından ambar içerisindeki malzeme üzerindeki barkod numarasını okutarak ve ambardaki miktarını girerek o barkodlu malzemeye ait sayım bilgisinin sisteme girişini sağlamak istemektedirler.

Android ortamında geliştirme yapmak için eclipse juno uygulaması kullanılmıştır. SDK ve AVD manager uygulamalarında bilgisayara indirilerek yapılan geliştirmeler farklı emülatörlerde denenmiş ve test edilmiştir.

Mobil uygulama ile web uygulama arasındaki bağlantıyı sağlayan veri aktarılmasına yardımcı olan servisler vardır. Bu servisler sayesinde mobil uygulamadan gelen veriler veritabanına aktarılır. Veritabanından çekilen veriler web uygulama ile açılır. Böylece web ve mobil uygulama aynı veritabanı üzerinden aynı verilerin yönetilmesini sağlamış olurlar.

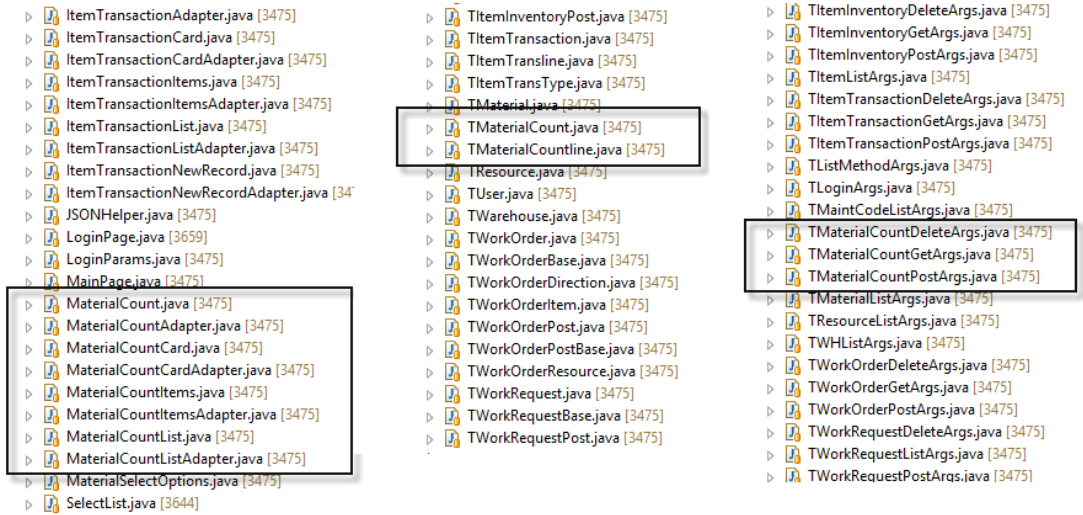
4.3.1. Malzeme Sayımı Modülü Geliştirmeler

Şekil 4.23'de mobil uygulamanın kullanmakta olduğu Bw2Rest isimli Restful servise malzeme sayımı işlemleri için eklenen yeni sınıflar gösterilmiştir. ServerResult klasörü altında ListResult, PostResult, InventoryResult olmak üzere sonuçların nesnelerini döndürecek sınıflar eklenmiştir. Daha sonra Args klasörü altına DeleteArgs, GetArgs, PostArgs sınıfları eklenerek post, get ve delete işlemleri sırasında kullanılacak olan nesnelere gönderilecek argümanlar tanımlanır. TItemInventory ve TItemInventoryPost sınıflarında malzeme sayımlarını çekmek ve veriyi göndermek ile alakalı işlemler yer almaktadır.



Şekil 4.23. Mobil servis malzeme sayımı sınıf yapısı.

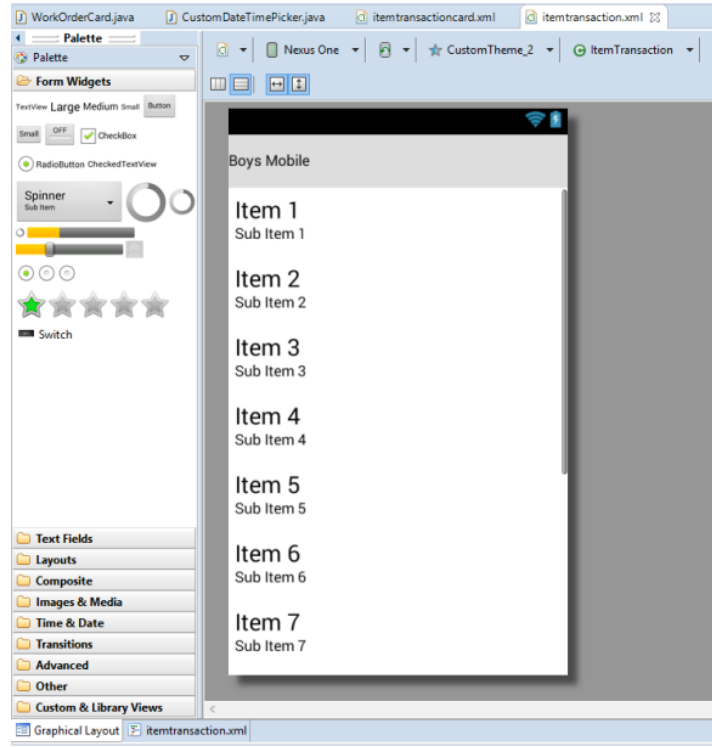
Bw2RestLib.svc sınıfı içerisinde metotlar tanımlanmıştır. DeleteItemInventory metodu malzeme sayımını siler. GetItemInventory metodu bugün oluşturan malzeme sayımlarını listeler. GetItemInventoryList tüm malzeme sayımı sonuçlarını listeler şeklinde methodlar mevcuttur.



Şekil 4.24. Mobil uygulama malzeme sayımı sınıf yapısı.

Şekil 4.24’de mobil uygulama geliştirme arayüzü üzerinde tanımlanmış olan sınıfların yapıları gösterilmiştir. TMaterialCount servisteki TitemInventory sınıfına karşılık gelmektedir. TMaterialCountGetArgs sınıfı serviste TitemInventoryGetArgs sınıfına karşılık gelmektedir. TMaterialCountListResult sınıfı serviste TitemInventoryListResult sınıfına karşılık gelmektedir. Asıl işlemler ise com.bimsercozum.boysandroid27 klasörü altındaki sınıflarda gerçekleştirilmektedir. Bu sınıflar içerisinde TO, TOArgs,

TOServerResults nesneleri ve tanımlamaları çağırılarak kullanılır. MaterialCount ve MaterialCountAdapter sınıfları malzeme sayımı ana sayfasının işlemlerini gerçekleştirir. MaterialCountCard, MaterialCountCardAdapter sınıfları malzeme sayımına yeni kayıt eklerken yada güncellerken girilen sayfanın işlemlerini gerçekleştirir. MaterialCountItem, MaterialCountItemAdapter sınıfları malzeme sayımı eklerken malzeme seçmek için girmiş olduğumuz sayfanın işlemlerini gerçekleştirir. MaterialCountList, MaterialCountListAdapter sınıfları malzeme sayımı sonucu oluşturulan sayımların listelendiği sayfaların işlemlerini gerçekleştirir.

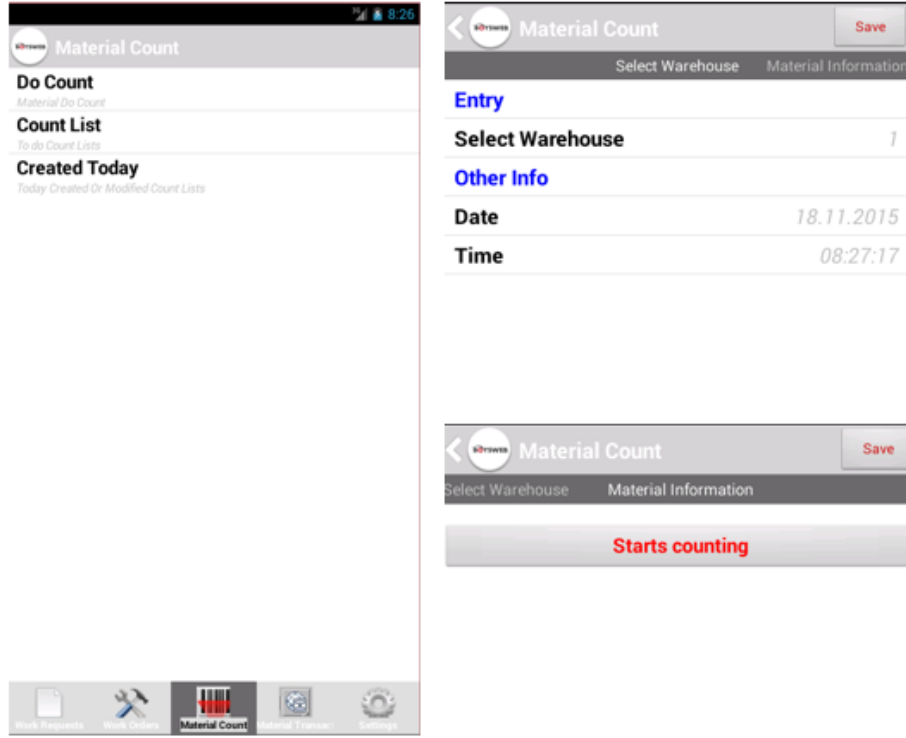


Şekil 4.25. Mobil uygulama liste ekranı sayfa dizaynı.

Şekil 4.25’de xml yapıları içerisinde sayfa tasarımları yer almaktadır. Malzeme sayım listesi bu sayfada tasarlanmıştır. Her bir Item içerisinde malzeme sayımının bilgileri (sayım numarası, malzeme bilgileri) yer almaktadır.

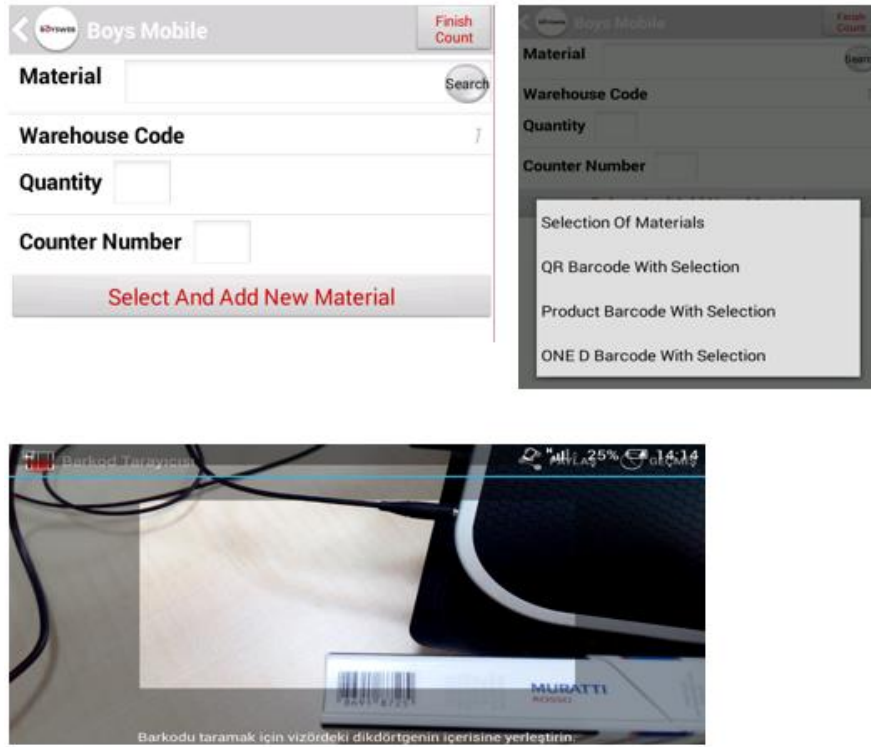
Res altındaki values klasörü içerisindeki xml yapısında uygulama içerisinde tutulan text değerler, mesajlar, uyarıların listesi yer almaktadır ve tag değerleri vardır. Bu tag değerleri programda kullanılarak textlerden bağımsız bir kodlama standardı geliştirilir.

Values-EN, Values-RU olarak eklenen her yeni klasör içerisindeki xml yapısında text alanı değiştirdiğimiz dile göre program çoklu dil desteği sağlamaktadır. Cihaz hangi dilde çalışıyor ise program o dile otomatik geçmektedir.



Şekil 4.26. Malzeme sayımı modülü fonksiyonları.

Şekil 4.26’da Boys Mobil uygulaması malzeme sayımı modülü sayfaları yer almaktadır. Malzeme sayımı listesi, malzeme sayımına başlama fonksiyonu, bugün oluşturulan malzeme sayımlarının listesi şekil 4.26’da sağ kısımda yer almaktadır. Uygulamaya yeni tab elenerek Malzeme sayımı bu tab üzerinden yapılmaktadır. Yeni sayımı tıklayarak girdiğimizde buradan tarih, saat ve ambar seçeriz, ekranı çekerek sayıma başla butonu ile malzemeleri seçeceğimiz sayfaya geçebiliriz.



Şekil 4.27. Boysweb malzeme sayımı barkod okuyucudan malzeme seçimi.

Şekil 4.27’de mobil uygulama üzerinde malzeme sayımına başlanıldığında malzemeleri barkod üzerinden nasıl okunabileceğine dair bilgiler yer almaktadır. Malzeme Seçim sayfasında ara butonu yardımıyla açılan listede malzemelerin barkod okuyucu ile mi yoksa listeden mi seçilmesi gerektiği bilgisi seçilir. Listedenden denilirse malzemelerin listelendiği liste açılır ve seçim gerçekleştirilir. Barkod ile seçim yapılacak ise program android cihaz üzerinde barkod okuyucu program kurulu olup olmadığını kontrol eder eğer kurulu değilse şu adresten kurulumu yapın diye uyarı döndürür. Eğer kurulu ise barkod okuyucu cihaz arka planda çalıştırılarak kamera açılır. Okunan barkod bilgisine göre malzeme bulunarak malzeme kodu alanı malzeme alanına aktarılır. Daha sonra yeni malzeme ekle diyerek aynı işlemleri tekrar ettirip seçilen tüm malzemeleri malzeme listesinde tutulması sağlanabilir. Sayımı bitir butonu yardımıyla listedeki malzeme bilgileri dönülen sayfaya aktarılır. Aktarılan malzemeleri gördükten sonra kaydet diyerek malzeme sayımı tamamlanır. Gelen ekranda sayımı listele yada bugün oluşturulanları listele denildiğinde malzeme sayımı listesi listeye aktarılır. Aktarım işlemleri Restful servis üzerinden Rest protokolü mantığına göre JSON olarak yapılmaktadır.

Malzeme Sayımı

Filtre

1

1

10

2

Drag a column header here to group by that column

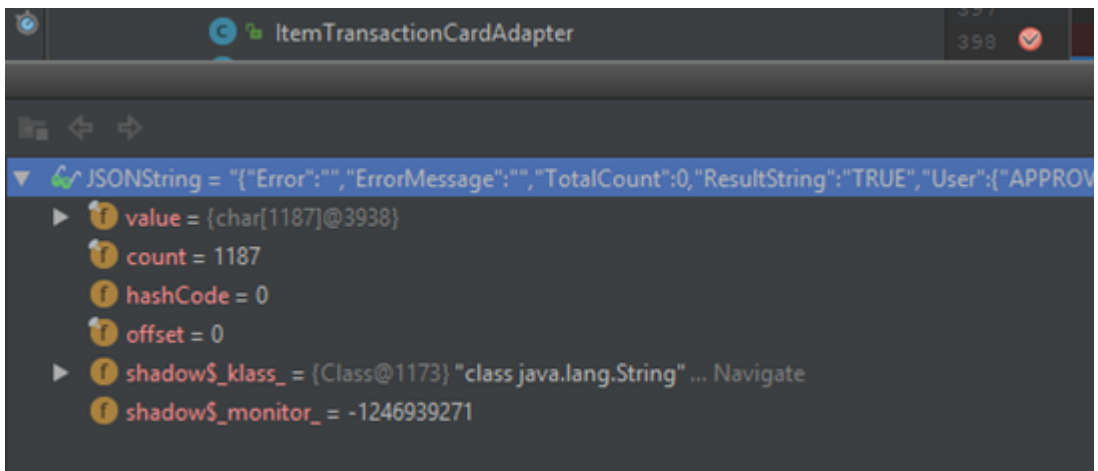
Sayım Statüsü		Malzeme Kodu	Malzeme Tanımı	Ambar Kodu
		1000	Fiat Doblo Opar Olio	1
		10000	Bulağık Yıkama Makinesi Su Seviye Prosestabi	1

Şekil 4.28. Boysweb uygulaması malzeme sayımı ana ekranı.

Şekil 4.28’de malzeme sayımı ana ekranı yer almaktadır. Bu ekran mobil uygulama ile gerçekleştirilen sayımların BOYSWEB uygulaması içerisinde görüntülenmesi ve takip edilmesini sağlar. Tamamlanan sayımların malzeme hareketine dönüştürülmesi işlemi de bu sayfa altında gerçekleştirilmektedir.

4.3.2. Mobil Uygulama Üzerinden Malzeme Sayımı Verilerinin Gönderimi

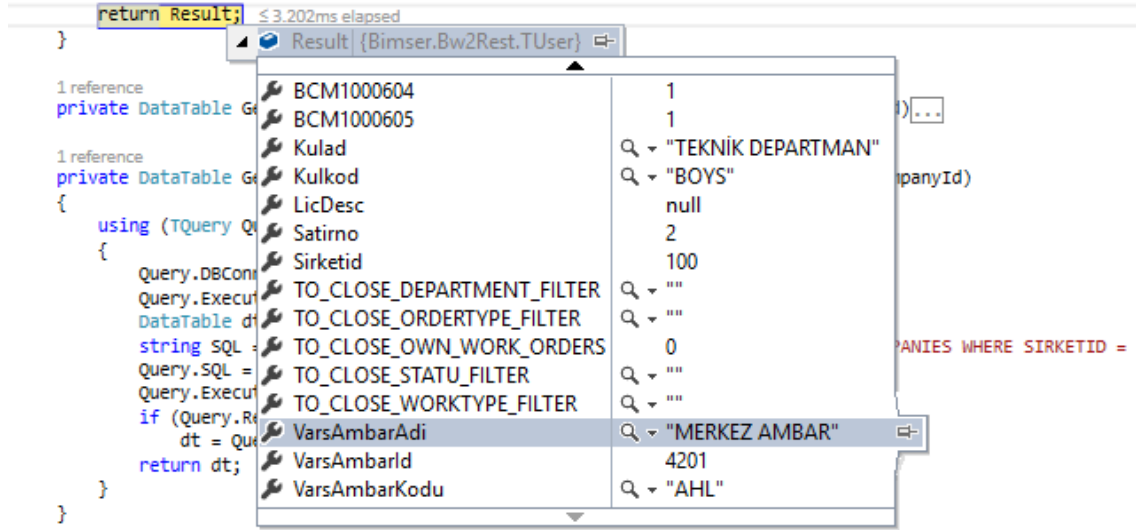
Mobil uygulama çalışması sırasında sunucu ile istemci arasında veri transferi işlemlerini Restful servis aracılığı ile gerçekleştirmektedir. Mobil uygulama kullanıcının bir aktivitesi olduğunda Retful Servisi çağırılmaktadır. Servis aracılığıyla sunucu üzerindeki bilgiler mobil uygulamaya iletilmektedir. Mobil uygulama Jsonstring olarak birleştirip aldığı bu verileri ayrıştırmakta ve kullanıcı aktivitesini gerçekleştirmektedir. Sonucunu tekrar Restful servisi çağırarak dönmektedir.



Şekil 4.29. Mobil uygulama JsonString bilgisi.

Örneğin mobil uygulama giriş ekranı sırasında kullanıcı bilgilerine ihtiyaç duymaktadır. Restful servisi çağırarak kullanıcı bilgilerini istemektedir. Kullanıcı bilgileri Restful

servis tarafından sunucudan çekilerek mobil uygulamaya gönderilmektedir. Şekil 4.29’da mobil uygulamaya gelen Jsonstring türündeki verinin ekran görüntüsü yer almaktadır. Mobil uygulama bu stringi ayrıştırarak giriş ekranında ilgili alanlara aktarmaktadır. Böylece giriş işlemi için gerekli veriler kullanıcı giriş ekranına gönderilmiş olur.



Şekil 4.30. Restfull servis JsonString bilgisi.

Şekil 4.30’da mobil uygulama tarafından istenen kullanıcı bilgileri sunucudan çekilmekte ve Restful servise gönderilmek üzere düzenlenmektedir. Mobil uygulamaya gönderilen kullanıcı bilgilerine göre uygulamaya giriş işlemi tamamlanmaktadır.

4.3.3. Modülün Eklenmesiyle Gerçekleşen Kazanımlar

Ambar sayımı yapan kullanıcıların iş yükü ve zaman kaybı azaltılmıştır. Android cihaz ile barkod okutarak hızlı bir şekilde malzemeleri saymaktadırlar. Tüm sayım sonuçlarını bir sayfa üzerinden takip edebilmektedirler. Miktarı göre malzeme hareketleri yapabilmektedirler. Bu gelişmeler neticesinde malzemelerin sayım yönetimlerini çok daha hızlı ve pratik şekilde gerçekleştirebilirler. İşletmelerin ambar sorumlularının tek tek sayarak not ettikleri verileri mobil cihazlar aracılığıyla barkodlar üzerinden okutarak 3 adımda malzeme hareketine dönüştürebilecekleri bir yapı geliştirilmiştir.

4.4. BAKIM TALİMATI TARİHÇE VE ONAY SİSTEMİ GELİŞTİRİLMESİ

Bakım türlerinden olan periyodik bakımlar içerisindeki işleyiş iki farklı yoldan gerçekleştirilmektedir. Boysweb sistemi üzerinde varlıklara periyodik olarak

gerçekleştirilmek üzere bakımlar tanımlanır. Tanımlanan bakımlara periyotlar verilir. Bu periyotlar 1 gün, 6 ay ya da sayaç bazında 1000 km, 100 birim gibi verilebilir. Verilen periyotlara göre zamanı ya da sayacı gelen periyodik bakımlar servis tarafından tetiklenerek iş emri oluşturulur.

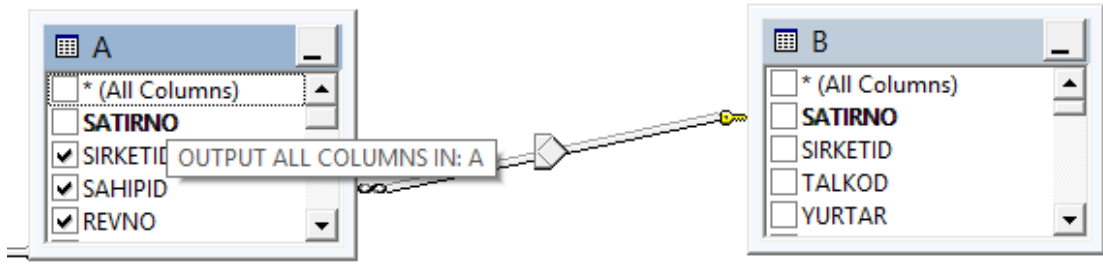
İş adımlarını kullanmak istemeyen kullanıcılar bakım talimatlarını sisteme girerek periyodik bakım üzerinden iş emrine aktarırlar. Bakımcılarda bu talimatlar doğrultusunda iş emrini gerçekleştirebilir.

Varlıkların bakımları sırasında hayati önem taşıyan bakım talimatlarının her varlık için güncel ve anlaşılır şekilde iş emrine eklenmiş olması gerekmektedir. Yanlış bakım talimatı iş emrini gerçekleştiren bakımcının varlığa yanlış müdahalesine sebep olabilir. Böylece işletme periyodik olarak bakımlarını gerçekleştirerek daha uzun süre hayatta kalmasını ve duruş yaşamamasını istediği varlığını tamamen kaybederek büyük zarara uğrayabilir. Ya da yanlış talimata maruz kalan varlık bakımı tamamlandıktan sonra duruş yaşayabilir. Buda işletmenin bakım yönetim sistemleri kullanarak kar etmelerini sağlamak yerine zarara geçmelerine neden olabilir.

Olumsuz senaryoları minimuma indirebilmek için bakım talimatlarının yetkilere bağlanarak belirli kişilerin bu talimatlarda değişiklik yapmaları istenmektedir. Değişiklik yapacak personelin gözünden kaçabilecek yanlışlıkların önüne geçebilmek adına bakım talimatı arkasına onay sistemi eklenmesi istenmektedir. Bir personel bakım talimatında değişiklik yaptığında bakım talimatının tekrar kullanılır hale gelebilmesi için üst yöneticinin onayına gider. Üst yöneticisi onay verdiğinde isteğe göre bir üst yöneticiye yönlendirilebilir. Ya da onay tamamlanabilir. Onaylardan geçen bakım talimatı tekrar kullanılabilir hale getirilir. Ayrıca değişikliğe uğrayan her bir bakım talimatı için kimin tarafından değişikliğe uğradığı hangi tarih ve saatlerde değişikliğe uğradığı bilgileri bakım talimatı tarihçesi altına kayıt atılır. Böylece bakım talimatı değişiklikleri ileride oluşacak olumsuz durumlarda kanıt niteliği taşıması için kayıt altına alınmış olur.

4.4.1. Bakım Talimatı Tarihçesi

Şekil 4.31’de bakım talimatı ile bakım talimatı tarihçesi arasındaki veritabanı ilişkisi yer almaktadır. Bunun için öncelikle bakım talimatı tablosu ile bir tarihçe tablosu oluşturularak ilişki kurulması gerekmektedir. BC_DIRECTION_HISTORY tablosu oluşturularak BC_DIRECTIONS tablosu ile 1- n ilişkisi kurulmuştur. Direction tablosunun satır numarası Direction_History tablosunun Sahipid’inde tutularak bağlantı sağlanır.



Şekil 4.31. Talimat - tarihçe veritabanı ilişkisi.

Bakım talimatında herhangi bir alanda meydana gelen değişiklik için talimat tarihçesi alanına yeni kayıt atılarak değiştirilen kısım değişmiş olarak ve kimin değiştirdiği kaydı da yazdırılarak gösterilmelidir. Bunun için tablolar birleştirilerek Direction tablosundaki her alan Direction_History tablosunda yer almasının sağlanması gerekmektedir. Böylece kod kısmında tarihçe çerçevesinde veriler listelenirken veriler view üzerinden çekilerek değişen tüm veriler gösterilmiş olur.

Bakım Talimat Kodu	Bakım Talimatı Tanımı
AOE-01	Euramet/cg-18/v.3 , EA-04/02
AZS-01	TS 10303, ITS 90, EA-04/02, TESTO, ATAGO
AZS-02	
CR-01	
DT-01	
HOE-01	
IT-01	
KL-01	
MKT-01	
MT-01	

Revizyon Tarihi	Talimat Kodu	Talimat Tanımı	Revizyon Numarası	Revize Eden Kodu
13.1.2016	AZS-01	TS 10303, ITS 90, EA-04/02, TESTO, ATAGO	3212	10000
13.1.2016	AZS-01	TS 10303, ITS 90, EA-04/02, TESTO, ATAGO	3212	10000
13.1.2016	AZS-01	TS 10303, ITS 90, EA-04/02, TESTO, ATAGO	3212	10000
13.1.2016	AZS-01	TS 10303, ITS 90, EA-04/02, TESTO, ATAGO	3212	10000

Şekil 4.32. Bakım talimatı tarihçe sayfası.

Şekil 4.32’de tasarlanan tarihçe ekranı gösterilmektedir. Bakım talimatında değişen her alan için yeni kayıt atılarak hangi tarih ve saatte kim hangi değişiklik yapmış kaydının kullanıcıya gösterilmesi sağlanmış olur.

4.4.2. Bakım Talimatı Onay Sistemi

Değiştirilecek olan bakım talimatları için kullanıcıların kendi belirledikleri kişilerin onayları verilerek talimatların değiştirilebilmesinin sağlanması gerekmektedir. Onay

işleminde mantık her bir onay işlemi için yeni bir kayıt oluşturmak ve bu onay kaydına bir id numarası vererek bu id üzerinden onay sürecini takip etmeyi ve kimin üzerinde onayda bekleyeceği, kimin onayladığı bilgilerinin erişilmesini sağlamaktır.

Bakım talimatı tablosuna onay bilgilerini tutabilmesi için alanlar eklenmiştir. Bu alanlar, Flowid onay kaydının satır numarasını tutmaktadır. Flowuser üzerinde onay ile alakalı kullanıcı bilgileri tutulmaktadır. FlwFinish ile onayın tamamlanıp tamamlanmadığı bilgisi tutulmaktadır. Böylece tamamlanan akışlar üzerinde işlem yapılamaz gibi bazı bir takım kontroller sağlanabilir. Flowstartdate, Flowstarttime onayın başlangıç tarih ve saatini, Flowenddate, Flowendtime bilgilerinde onayın bitiş tarih ve saati bilgileri tutulmaktadır. FlowDesc üzerinde onay açıklaması tutulmaktadır. Onaylayanid alanında kimin onayladığı bilgisi yer almaktadır. Onaytar, Onaysa alanlarında onaylanma tarih ve saati bilgileri tutulmaktadır. Eskiflow alanında bir önceki onaya ait satırnumarası bilgileri tutularak önceden gerçekleşen akışların tarihçeleri listelenebilirmesi işlemi için kullanılır. Statu alanında onayda mı bekliyor, onaylandığını, süreç bitimi gibi bilgilerin tutulması için hazırlanmış bir alandır.

Bakım Talimatı

Bakım Talimat Kodu:* talimat1

Bakım Talimatı Tanımı: talimat1

Talimat Tipi: Bakım Talimatı

Statü: Kapalı

Genel Bilgiler | Talimat | Bakım Kodları | Bağlı Dokümanlar | Onay Bilgileri

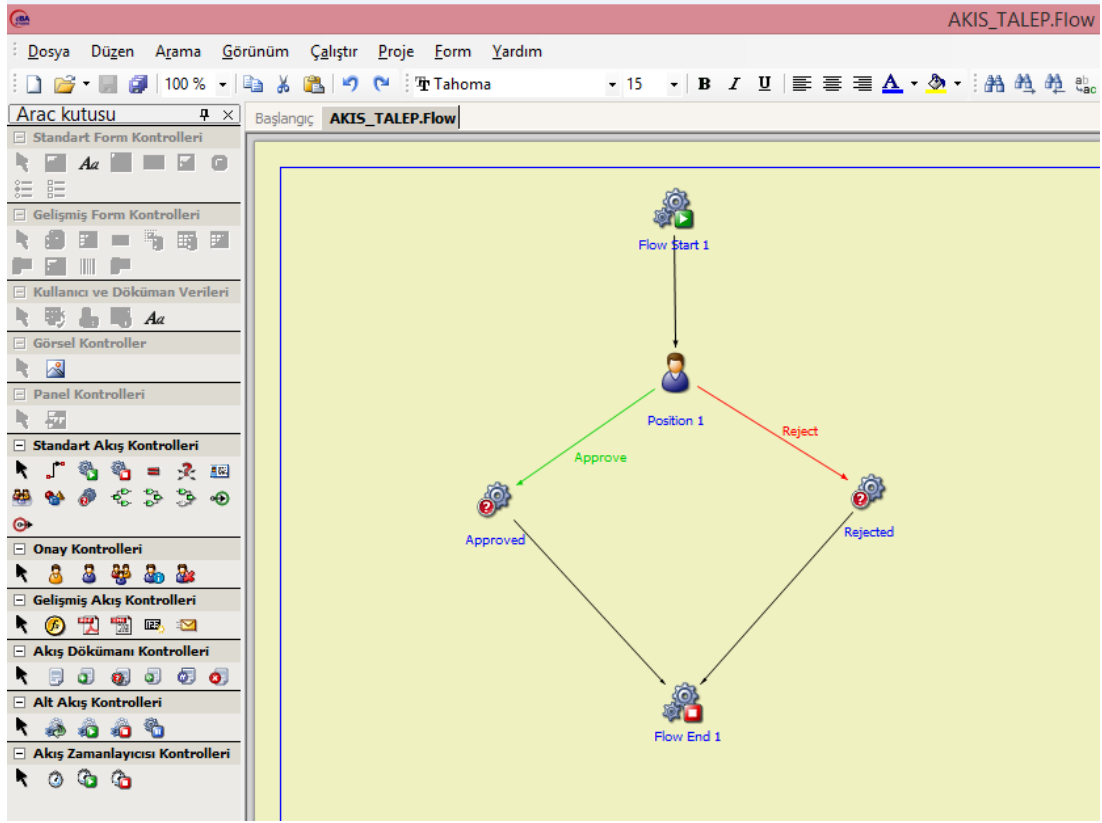
Onaylayanın Kodu	Onaylayan	Onay Seviyesi	Onaya Gönderilme Tarihi	Onay Durumu
100				test 100
200				TEST 200

Şekil 4.33. Bakım talimatı onay bilgileri.

Şekil 4.33’de bakım talimatlarının eklendiği sayfaya veritabanına eklenen statü alanının akıştaki onayın adımlarına göre değişmesini sağlamak ve kullanıcıya göstermek için (açık, kapalı, iptal) bilgisinin gösterileceği Statü alanı eklenmiştir. Onayın hangi aşamada olduğu ve kimlerin onayından hangi tarih ve saatte geçtiği bilgisine ulaşılabilmesi Onay bilgileri kısmı eklenmiştir. Bakım talimatlarının listelendiği ana sayfaya onay verilecek bakım talimatlarının daha kolay bulunması ve onaylanması işleminin sağlanması amacıyla Onay durumları ve Kimin onaylayacağı filtreleri eklenmiştir. Böylece onaylanacak kayıtların ana ekranda süzülerek çekilmesi sağlanmıştır.

4.4.3. Bakım Talimatı Akış Yapısı

Şekil 4.34’de Eba uygulama geliştirme platformu üzerinde bakım talimatlarının hangi kullanıcıların onayından geçeceği bilgisi göz önüne alınarak akış tasarlanmıştır. Akışın nasıl olacağı ve kimlerin onayından geçeceğini belirlemek için Eba WorkFlow Studio uygulaması kullanılarak iş akışı tasarlanmıştır. Position1 de seçilen kişiye göre onay işlemi yapılmaktadır. Pozisyon1 akışı onaylarsa işlem tamamlanmaktadır. Reddederse işlem bitmektedir. Birden fazla pozisyon seçilebilir ve akış dallandırılabilir. Tamamlanan iş akış projesine bir isim verilir. Verilen isim şirket altında EBA ayarları kısmında bakım talimatı değişiklik onayı bölümüne yazılır. Böylece akış ile BOYSWEB’in entegre çalışması sağlanmış olur. Position1 de akışı kimin onaylayacağı bilgisi pozisyon özellikleri sekmesinden verilir. Kullanıcı akışı onayladığında yada reddettiğinde akış bitecektir. Bu işlemler ile birlikte geliştirmeler tamamlanır.



Şekil 4.34. Bakım talimatı akış tasarımı.

Test için bakım talimatında bir değişiklik yapıldığında önce Position1’ de seçilen kullanıcının onayına gider. Onaylandığında değişiklik kabul görmüş olur ve talimat yeni hali ile kullanıma başlayabilir. Akış reddedilirse talimat kullanım dışı haline gelir ve kullanılamaz. Talimatta yapılan değişiklik bakım talimatı tarihçesine kayıt olarak atılır.

Şekil 4.35. Bakım talimatı onay aşamaları.

Şekil 4.35’de geliştirilmiş olan bakım talimatı onay sistemine göre bir talimat değiştirildiğinde Position1’de seçili olan kullanıcı için onay ekranlarının nasıl gösterildiği onay işleminin nasıl gerçekleştirildiği bilgisine yer verilmiştir.

4.4.4. Bakım Talimatı Geliştirmeleri Sonucunda Elde Edilen Kazanımlar

Bakım talimatının belirli aralıklarla güncellenmesi gerekmektedir. Bu güncellemeleri belirli kullanıcıların yapması ve bu kullanıcıların yaptığı güncellemelerinde üst yöneticilerin onayları alınarak tamamlanması sağlanmıştır. Böylece yapılacak olan güncelleme işlemi sonrası birden fazla kişinin onayından geçen yeni talimatta problem çıkma olasılığı azaltılmıştır. Ayrıca ileride çıkabilecek problemler sonucunda problemin kimden kaynaklandığının öğrenilebilmesi tarihçe sayesinde sağlanmıştır. Böylece bakım için yüksek önem taşıyan bakım talimatlarının güvenliği çift taraflı olarak kontrol altına alınmıştır.

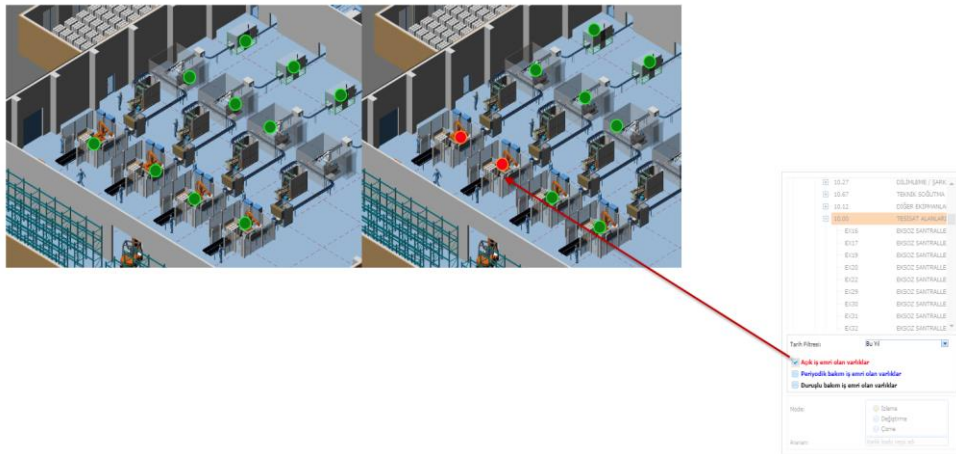
4.5. GRAFİKSEL VARLIK AĞACI MODÜLÜNÜN GELİŞTİRİLMESİ

Bakımdan sorumlu olan son kullanıcılar işletmelerinin krokisini tek bir ekrandan görmek istemektedirler. Hangi noktada problem varsa oraya direk müdahale etmek

istemektedirler. Alışveriş merkezlerindeki güvenlik odası gibi düşünülebilir. Alışveriş merkezlerindeki güvenlik şefi güvenlik odasında ekranların başında oturarak güvenlik kamerası sayesinde tüm alışveriş merkezini gözlemleyebilmektedir. Alışveriş merkezinin herhangi bir mağazasında tartışma, hırsızlık ya da huzursuz edici bir hareketlilik gördüğünde güvenlik görevlilerine telsiz ile ulaşarak o bölgeye yönlendirmektedir. Güvenlik görevlileri ilgili mağazaya giderek huzursuzluğu ortadan kaldırmakta ve telsizle şeflerini durumdan haberdar etmektedirler.

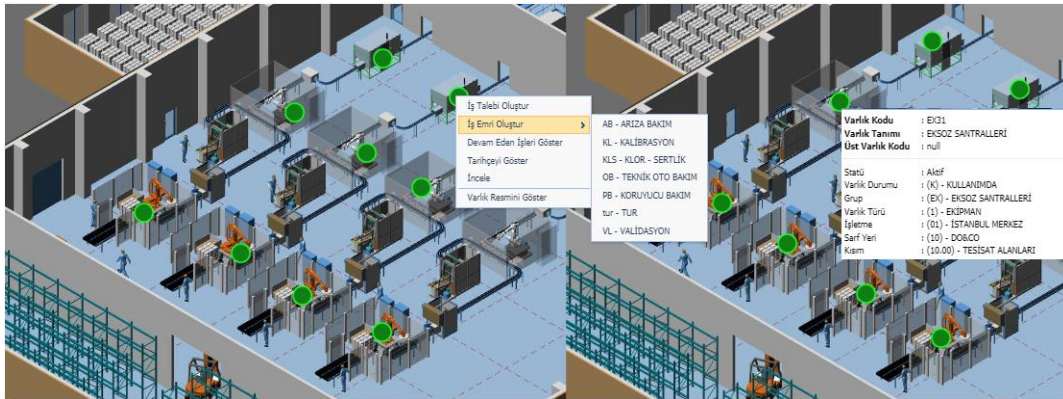
Bu işleyişi Boysweb uygulamasına taşıyabilmek adına grafiksel varlık ağacı modülü geliştirilmiştir. Devexpress uygulamasının AspxTreeList bileşeni kullanılarak yönetim panelindeki varlık ağacı yüklenmektedir. Varlık, kısım, sarfyeri, işletme üzerlerine harita resmi alanı açılmıştır. Bu tanımların herhangi birisinde bir harita resminin yolu seçilip kaydedildiğinde ve yönetim panelinde ilgili tanım seçildiğinde sayfanın kalan kısmına yerleştirilmiş olan resim birleşeni içerisine harita resmi yüklenmektedir.

Alışveriş merkezini işletme olarak düşünülebilir. Alışveriş merkezinin her bir katı sarfyeri, her kattaki 15 mağazalık bölümler kısım olarak ele alınabilir. Alışveriş merkezindeki her bir mağaza işletmenin varlıkları olarak düşünülebilir. Güvenlik kameralarının sunduğu ekran grafiksel varlık ağacının ana ekranı olarak düşünülebilir. Bakım şefi grafiksel varlık ağacını kontrol ederek hangi varlık için açık iş emri olduğunu görebilir. Bu iş emrinin sorumlusunu aynı ekrandan seçerek o iş emrine bakımcı atar ve bakımcı varlık yanına giderek işi yapıp iş emrini kapatır. Bakım şefine kapanan iş emri bilgisi gelmiş olur. Böylece telsizle haberleşme işlemi de sistem üzerinden iş emrinin açılması ve kapatılması aşamasında bilgilendirme yoluyla bakımcı ve bakım şefine iletilmiş olur.



Şekil 4.36. Grafiksel varlık ağacı dizaynı.

Şekil 4.36’da sağ alt kısımdaki yönetim panelinin üst kısmında işletme içerisindeki tüm varlıkların üst varlık, kısım, sarfyeri, işletme şeklinde kırımlarıyla birlikte listelenmesi sağlandı. Bu kırım yapısına varlık ağacı denilmektedir. Varlık ağacının alt kısmında tarih filtresi yer almaktadır. Bu filtreye göre açık iş emri olan varlıklar seçildiğinde ekrandaki varlıklardan hangilerine ait açık iş emirleri varsa o iş emirleri kırmızı renkli olarak haritada gösterilmektedir. Periyodik bakım iş emri olan varlıklar işaretlendiğinde, periyodik bakımı oluşmuş iş emirlerine sahip varlıklar mavi renkli olarak gösterilmektedir. Duruşlu bakım iş emri olan varlıklar işaretlendiğinde, Üretimi durdurdum işaretli iş emrine sahip olan varlıklar haritada siyah renkle gösterilmektedir. Yönetim panelinde filtrenin alt kısmında bulunan izleme, değiştirme, çizme butonları yardımıyla varlıklar harita üzerine ilk harita resmi yüklendikten sonra çizilerek yerleştirilmektedir. Yerleştirilen her varlık yeşil kutu olarak gösterilmektedir. Aynı şekilde bu varlıkların yerleri değiştirilebilmektedir.



Şekil 4.37. Grafiksel varlık ağacı izleme modu.

Şekil 4.37’de grafiksel varlık ağacı yönetim panelinde izleme butonu işaretlendiğinde varlıkları temsil eden yeşil kutuların üzerine gelindiğinde varlık bilgileri gösterilmektedir. Üzerine tıklandığında ise bu varlığa iş talebi ve iş emri oluşturma fonksiyonları aktifleşmektedir. Devam eden işler gösterilebilmektedir. Bu varlığa ait tarihçe gösterilmesi için varlık tarihçe raporuna yönlendirme yapılmaktadır. Varlık resmi gösterilmektedir.

4.5.1. Grafiksel Varlık Ağacı Modülü Geliştirilmesi Sonucu Kazanımlar

Geliştirme sonucunda bakım şefi ya da yönetici tek ekrandan bütün işletmesine, sarf yerine ya da kısmına ait tüm varlıklarını yönetebilmektedir. Varlıkların şu an duruş yaşıyor yaşamadıklarını, üzerlerinde açık iş emirlerinin olup olmadığını anlık

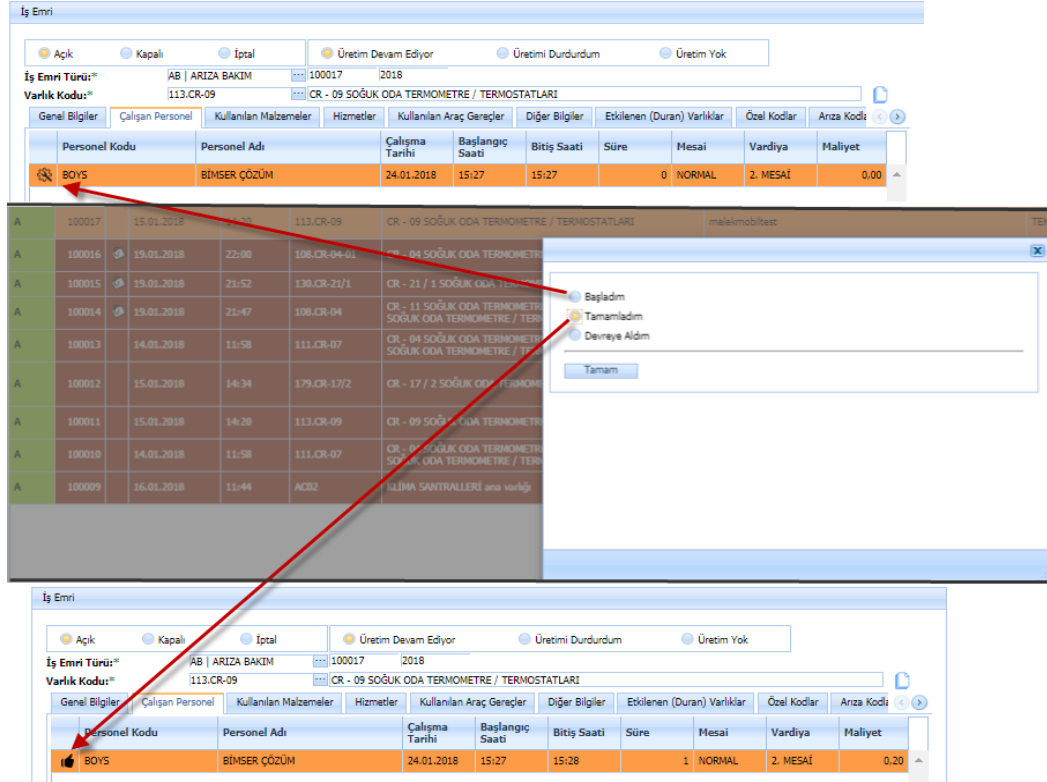
görebilmektedirler. Ayrıca sorun gördükleri varlıklara aynı ekran üzerinden iş emri açarak bakımcıların o varlığa yönelmelerini sağlayabilmektedirler.

4.6. İŞ EMRİ BAKIM DURUMLARI SEÇİMİ GELİŞTİRMESİ

Bakım personelleri kendine açılan iş emri içerisine girerek çalışan personeller altına kendini eklemekte ve hangi tarihlerde ne kadar süre çalıştığı, hangi mesaide çalıştığı, kaç dakika süre harcadığı bilgileri girmektedir. İş tamamladıktan sonra iş emrini kapatarak yaptıkları çalışmaları yapılan işin açıklaması alanlarına eklemekte ve iş emri performansı gibi raporlar ile performans kriterlerine göre performanslarını raporlayabilmektedir.

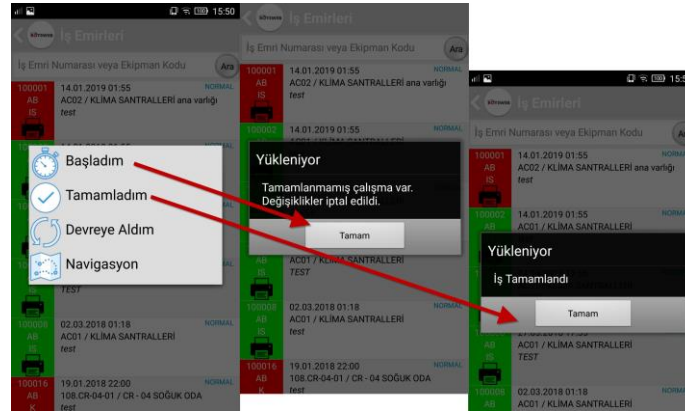
Çalışacak bakımcıların bu aşamada işlerini kolaylaştırmak adına fonksiyonel ihtiyaçları vardır. Kendilerine ait her iş için iş emri içerisine girerek çalışan personellere her seferinde kendilerini eklemek yerine direk iş emri listesi üzerinden kendilerini eklemek istemektedir.

İhtiyaçları gerçekleştirmek adına iş emri listesi - ek özellikler butonu altında açılan fonksiyon listesi içerisine bakım durumu fonksiyonu eklenmiştir.



Şekil 4.38. İş emri bakım durumu seçim işlemi.

Şekil 4.38’de iş emri bakım listesi altında ek özellikler butonu kullanılarak bakım durumu fonksiyonu açılmıştır. Buradan bakım personeli sorumlu olduğu iş emrini seçer. İş emri içerisine girmeden başladım özelliğini seçerek iş emri çalışan personellere otomatik olarak kendini atamış olur. Daha sonra işi yapıp tamamladığında iş emri listesi içerisinde aynı işi seçerek tamamladım özelliğini seçer. Böylece çalışan personeller içerisinde çalıştığı süre, bitiş tarihi, maliyet bilgileri sistem saati üzerinden hesaplanarak aktarılmaktadır. İş emri içerisinde başka yapılacak iş kalmamışsa, sorun çözülmüş ise devreye aldım özelliğini seçerek iş emrini devreye almaktadır.



Şekil 4.39. İş emri bakım durumu özelliğinin mobil versiyona uyarlanması.

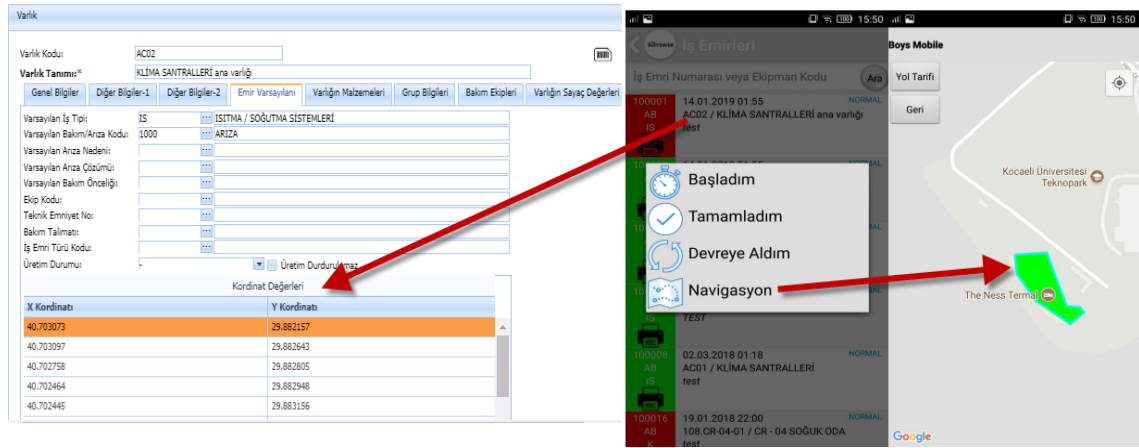
İş emri bakım durumunu sahada çalışan personeller de mobil cihazları üzerinden anlık kullanmak istemektedirler. Bu noktada bakımcılar akıllı cihazlarına gelen bildirimlere göre iş emrini görerek ilgili işi gerçekleştirmek üzere sahaya gittiklerinde mobil cihaz üzerinde iş emrine başladıkları tamamladıkları bilgilerini Şekil 4.39’daki gibi kullanabilmektedirler. Bunun için mobil cihaz üzerinde iş emirlerinin listelendiği liste sayfasına menü eklenmiştir. İş emri üzerine uzun basıldığında tetiklenen fonksiyon aracılığı ile işe başladıkları, tamamladıkları, devreye aldıkları restfull servis aracılığı ile veritabanına aktarılmaktadır. Böylece web uygulama ile mobil uygulama senkronize çalışmaktadır. Mobil üzerinde bakım personeli işe başladım dediğinde web uygulamada ilgili personelin o iş üzerinde çalıştığı bilgisine varılabilir. Tamamladım, Devreye aldım fonksiyonları içinde aynı durum geçerlidir. İş Emri Bakım Durumu Geliştirmesi Sonucu Elde Edilen Kazanımlar

İş Emri ekranında iş emrini açmadan liste üzerinden bakımcıların çalışma durumlarını seçebilme özelliği sayesinde zaman ve iş gücü kazanımı olmuştur. Mobil versiyona bu özellik kazandırıldığında ilgili personel sahada işe başladığında tek tuşla bu işte ben şu

saat ve dakikadan itibaren çalışıyorum bilgisini tüm işletmeye duyurmuş olmaktadır. Daha sonra işi tamamladığında anlık olarak bu işi tamamladım. Bu iş ile ilgili işlemleri gerçekleştirdim ve şu kadar maliyet ortaya çıktı bilgisini tek tuş ile tüm işletmeye mobil cihazı yardımıyla duyurmaktadır. Böylece işletme üzerinde binlerce bakımcı farklı işler içerisinde anlık olarak çalışmakta ve anlık olarak işleri tamamladıkları bilgisini bildirmektedirler.

4.7. VARLIK KORDİNAT BİLGİLERİ GELİŞTİRMESİ

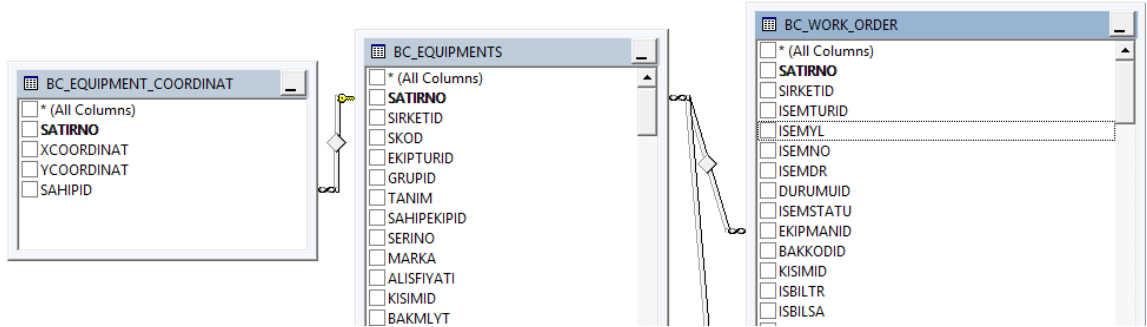
Varlık üzerine eklenen liste ekranından varlığın dünya haritasında yer alan koordinat bilgilerinin girilmesi istenmektedir. Koordinat eklenirken varlıkların harita üzerinde bir noktada gösterilmesi ya da haritada bir alanda gösterilmesi istenebilmektedir. Şekil 4.40'da çoklu X ve Y koordinatı girilmesi sağlanarak birden fazla X ve Y koordinatı girildiğinde mobil cihaz üzerinden girilen koordinatlar harita üzerinde işaretlenmekte ve bu noktalar birleştirilerek harita üzerinde bir alan işaretlenmektedir. Bu alan ilgili varlığı temsil etmektedir.



Şekil 4.40. Varlık navigasyon bilgisinin mobil ekranda gösterilmesi.

Web uygulama üzerinde emir varsayılanları altına varlık koordinat bilgileri alanı eklenmiştir. Buraya eklenen grid sayesinde X ve Y koordinatları girilebilmektedir. Bu varlık kullanılarak iş emri oluşturulduğunda mobil uygulama üzerinden iş emri listesi açılmaktadır. Açılan iş emri listesinde ilgili varlığın olduğu kayıt üzerine uzun tıklandığında menü açılması sağlanmaktadır. Açılan menüye Navigasyon isiminde bir item eklenmiştir. Navigasyon seçildiğinde mobil harita ekrana basılarak varlık üzerindeki X ve Y navigasyon bilgilerini veritabanından restfull servis ile çekerek json şeklinde dönen

veriler harita üzerine her bir X ve Y kordinatı bir noktayı temsil edecek şekilde aktarılır. Aktarılan bu noktalar birleştirilerek o varlığa ait alan harita üzerinden şekil 4.41’deki gibi gösterilir.



Şekil 4.41. Varlık koordinat bilgileri ilişkisel veritabanı.

Varlık koordinatlarının birden fazla eklenebilmesi için yeni BC_EQUIPMENT_COORDINAT isminde tablo eklenmiştir. Bu tabloya SATIRNO alanı birincil anahtar olarak otomatik artan şekilde belirtilmiştir. SAHIPID alanı ise ikincil anahtar olarak varlıkların tanımlı olduğu BC_EQUIPMENTS tablosu üzerindeki SATIRNO birincil anahtarına bağlanmıştır. 1 – N ilişki sağlanmıştır. Ekipmanın iş emri üzerinde seçildiğinde ilgili ekipmana ait bilgilere iş emri üzerinden ulaşılabilmesi için BC_EQUIPMENTS tablosunun SATIRNO birincil anahtarına iş emri tablosu olan BC_WORK_ORDER tablosunun EKİPMANİD ikincil anahtarı bağlanmıştır. 1 – 1 ilişki vardır. Böylece mobil uygulama üzerinde iş emri listesi sayfasında navigasyon bilgisi alanı seçildiğinde iş emri üzerinden ekipmana geçilir. Ekipman üzerinden ekipman kordinatları tablosuna geçilir. Ekipman kordinatları tablosu üzerindeki X ve Y bilgilerine sahip olan liste restfull servis yardımı ile veritabanından alınarak json veri türünde mobil uygulamaya aktarılır. Mobil uygulama bu listeyi dönüştürerek harita üzerine aktarır. Böylece varlık koordinat bilgilerinin mobil uygulama üzerinden haritaya taşınması sağlanmış olur. Harita üzerinde varlık konumları son kullanıcıya gösterilir.

4.8. VARLIK TAŞIMA MODÜLÜNÜN GELİŞTİRİLMESİ

İşletmeler bünyelerinde yer alan varlıkların bakımlarını Boysweb üzerinden takip edebilmektedirler. Öncelikle sistemlerine tüm varlıklarını tanımlarlar. Daha sonra periyodik bakım tanımları üzerinden ya da iş talepleri yardımı ile tanımladıkları varlıklara iş emri oluşturmaktadırlar. Böylece periyodik bakım zamanı gelen varlıklara ya da arıza

oluşan varlıklara bakım personellerin müdahale edebilmeleri için sistemde iş emri adına çağrılar oluşturulmuş olur. Bu iş emirleri bakımçı personeller tarafından bakım yapılarak kapatılmaktadır. Bazı durumlarda işletmelerin sahip oldukları ekipmanların yerleri değişebilmektedir. Örneğin bir bilgisayar sisteme tanımlansın. Daha sonra bir ekranda bilgisayara bağlı olarak sisteme tanımlansın. Bu ekran bilgisayarın alt ekipmanı olarak sistemde seçilsin. Daha sonra bu ekranın bozulması durumunda bu ekranın yerinden alınarak yeni ekranın bu ekran yerine getirilmesi gerekir. Çıkarılan ekranında hurdaların içerisine atılması gerekir. Bu senaryonun Boysweb üzerinde işletilebilmesi için varlık taşıma modülü geliştirilmiştir.

The screenshot displays the 'Varlık Taşıma' (Asset Transfer) interface. At the top, there's a toolbar with navigation and action icons. Below it, a table lists assets with columns for Transfer Number, Asset Code, Asset Name, Old Owner, New Owner, Old Part, New Part, Old Quantity, New Quantity, Old Location, New Location, Operation Code, and Operation Name. Two rows are visible: Transfer 72 (Asset AC02, Monitor 1) and Transfer 73 (Asset 109.CR-05, MONİTÖR 2). Below the table, two detailed views are shown for Transfer 72 and Transfer 73. Each view includes fields for Transfer Number, Asset Code, Asset Name, Old Owner, New Owner, Old Part, New Part, Old Quantity, New Quantity, Old Location, New Location, Operation Code, Operation Name, Date, Time, and a description. The bottom of each view has a 'Tamam' (Finish) button.

Transfer Numarası	Varlık Kodu	Varlık Tanımı	Eski Sahip Ekipman Kodu	Eski Sahip Ekipman Tanımı	Yeni Sahip Ekipman Kodu	Yeni Sahip Ekipman Tanımı	Eski Kısım Kodu	Eski Kısım Tanımı	Yeni Kısım Kodu	Yeni Kısım Tanımı	İşlemi Yapan Kodu	İşlemi Yapan Tanımı
72	AC02	Monitör 1			108.CR-04	HURDA AMBARI	10.00	TESİSAT ALANLARI	10.00	TESİSAT ALANLARI	08705	Mustafa Çebi
73	109.CR-05	MONİTÖR 2			AC01	NOTEBOOK 1	10.00	TESİSAT ALANLARI	10.00	TESİSAT ALANLARI	08705	Mustafa Çebi

Varlık Taşıma

Transfer Numarası: 72

Varlık: AC02 Monitör 1

Bulunduğu Kısım: 10.00 TESİSAT ALANLARI

Bağlı Olduğu Varlık:

Yeni Sahip Varlık: 108.CR-04 HURDA AMBARI

Yeni Kısım: 10.00 TESİSAT ALANLARI

Yeni Gider Merkezi:

İşlemi Yapan: 08705 Mustafa Çebi

Tarih: 24.01.2018 17:03

Açıklama: Mobil Değişim Bilgileri Üzerinden Gerçekleştirilen Giden Transfer Kaydıdır

İş Emri : 100001-2019

Tamam

Varlık Taşıma

Transfer Numarası: 73

Varlık: 109.CR-05 MONİTÖR 2

Bulunduğu Kısım: 10.00 TESİSAT ALANLARI

Bağlı Olduğu Varlık:

Yeni Sahip Varlık: AC01 NOTEBOOK 1

Yeni Kısım: 10.00 TESİSAT ALANLARI

Yeni Gider Merkezi:

İşlemi Yapan: 08705 Mustafa Çebi

Tarih: 24.01.2018 17:03

Açıklama: Mobil Değişim Bilgileri Üzerinden Gerçekleştirilen Gelen Transfer Kaydıdır

İş Emri : 100001-2019

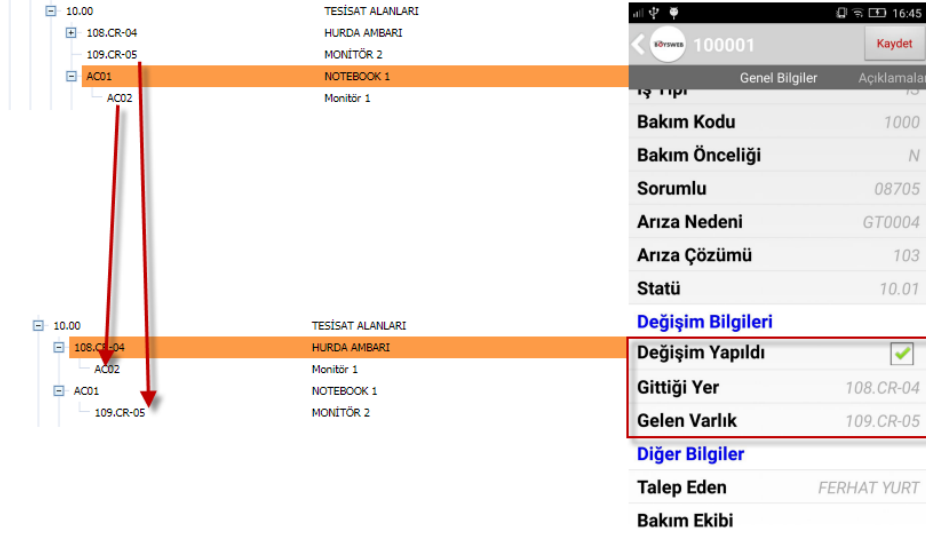
Tamam

Şekil 4.42. Varlık taşıma ekranları.

Varlık taşıma modülünün gerçekleştirebilmesi için 2 adet sayfanın tasarlanması gerekir. Şekil 4.42'de tasarlanan 2 sayfanın durumu gösterilmektedir. Varlık sayımlarının listelenmesi için varliktasima.aspx sayfası tasarlanmıştır. Bu liste sayfasında yeni eklenen BC_EQUIPMENT_TRANSPORT tablosu üzerindeki verinin çekilerek listelenmesi sağlanmıştır. Varlık taşıma modülünde ana taşıma fonksiyonlarının gerçekleştirilmesi için varliktasima_v.aspx sayfası geliştirilmiştir. Bu sayfa içerisinde taşınması istenen varlık, bu varlığa ait kısım, bu varlığın bağlı olduğu üst varlık var ise onun bilgisi yer almaktadır. Bu varlığın gideceği yeni kısım, hangi varlık altına bağlanacak ise yeni sahip varlık bilgileri yer almaktadır. İşlemi yapan personel, tarih saat bilgileri ve bu varlık taşıma işlemi bir iş emrine bağlı ise o iş emrinin numarası ve o iş emrinin yılı bilgileri de bu sayfa içerisinde yer almaktadır.

Taşınması istenen varlık seçildikten sonra taşınacak varlığın nereye taşınacağına göre

yeni kısmı seçildiğinde bu sayfa üzerinden varlık taşıma işlemi gerçekleştirilerek BC_EQUIPMENT_TRANSPORT üzerine yeni bir kayıt atılmaktadır. Aynı zamanda varlığın ilgili kısım ile bağlantısı kırılarak yeni kısım ile yeni bağlantı kurulur. Sahip varlıkları üzerinde de sahip varlıkları var ise bağlantısı kırılarak yeni sahip varlığı var ise yeni bağlantı kurulur.



Şekil 4.43. Varlık taşıma fonksiyonunun mobil uygulama üzerinde geliştirilmesi.

Şirketler varlık taşıma işlemlerini mobil uygulama üzerinden de gerçekleştirmek istemektedirler. Şekil 4.43’de yetkiye sahip bakım personeli mobil uygulama üzerinden iş emrine girdiğinde eğer iş emri içerisindeki varlık hurdaya ayrılması gerekiyor ise bunu nasıl gerçekleştirebileceği gösterilmiştir. Mobil uygulama üzerinde değişim yapıldı tiki işaretlenir. Gittiği yer alanına iş emri varlığının değiştiğinde nereye gideceği bilgisi girilir. Gelen varlık alanına ise bu iş emrindeki varlık gittiğinde onun yerine hangi varlık geleceği bilgisi seçilir. Kayıt işleminden sonra varlık taşıma işlemi gerçekleştirilmiş olur. Şekil 4.43’ün sağ kısmında web uygulama üzerinde yer alan varlıkların varlık ağacı gösterilmektedir. İşletmenin elinde Notebook 1 varlığı ve bu varlığa ait Monitör 1 varlığı vardır. Monitör 1 varlığı arızalanması sonucu bu varlığa iş emri açılmıştır. Bakımcı personel iş emrine istinaden Monitör 1 varlığına giderek kontrol etmiş ve tamir edilemeyecek durumda olduğunu anlamıştır. Varlığı hurda ambarına atarak yerine yeni bir Monitör varlığı takılması gerektiği kanaatine varmıştır. Bu işlemi yaptıktan sonra mobil uygulama üzerinden yaptığı işlemi Boysweb uygulamasına aktarmak istemektedir. Bunun için mobil uygulama üzerinden iş emrine girmektedir. İş emrinde Monitör 1 varlığı mevcuttur. Değişim yapıldı tikini işaretler. Gittiği yer alanına Hurda ambarını seçer.

Gelen varlık alanına Monitör 2 varlığını yeni varlık olarak seçer. Kayıt işleminden sonra Şekil 4.45’deki gibi varlık ağacında Monitör 1 varlığı hurda ambarı altına gider. Monitör 2 varlığı Notebook 1 varlığı altına gelir.

BC_EQUIPMENT_TRANSPORT tablosu altına da 2 adet kayıt atılır. İlk kayıt giden varlığa aittir. 2. Kayıt gelen varlığa aittir. Bu geliştirme yapılırken iş emri içerisine GIDENID ve GELENID alanları eklenmiştir. BC_EQUIPMENT_TRANSPORT tablosu oluşturulmuştur. Bu tablo üzerine ISEMRIID alanı eklenmiştir. ISEMRIID alanı aracılığıyla ikincil anahtar olarak 1- 1 ilişki ile varlık taşıma tablosu ile iş emri tablosu kayıtları eşleştirmek için bağlanmıştır. GIDENID GELENID alanları ise ekipman tablosuyla ikincil anahtar olarak bağlanmıştır. Böylece giden varlık ile gelen varlık bilgileri bu alanlarda tutulmaktadır.

The screenshot shows the design view of a report in Visual Studio. The report is titled 'Ekipman Taşıma Raporu'. It includes a header section with the following fields:

- Tarih / Saat : 24 Ocak 2018 Çarşamba 17:12
- Kullanıcı Adı / Kodu : [Parameters.USER]
- Şirket Kodu / Adı : [Parameters.COMPANY]
- Filtre : [Parameters.FILTRE]

Below the header is a table with the following columns:

- Ekipman Kodu / Tanım
- [Ekipman Kodu]
- [Ekipman Tanım]
- Transfer No
- Tarih
- Saat
- Eski Sahip Ekipman Kodu / Tanımı
- Eski Sarfyeri Kodu / Tanımı
- Eski Kısım Kodu / Tanımı
- Yeni Sahip Ekipman Kodu / Tanımı

The table is grouped by Ekipman Kodu / Tanım. The design also includes a page header and a page footer.

Şekil 4.44. Varlık taşıma raporunun dizayn edilmesi.

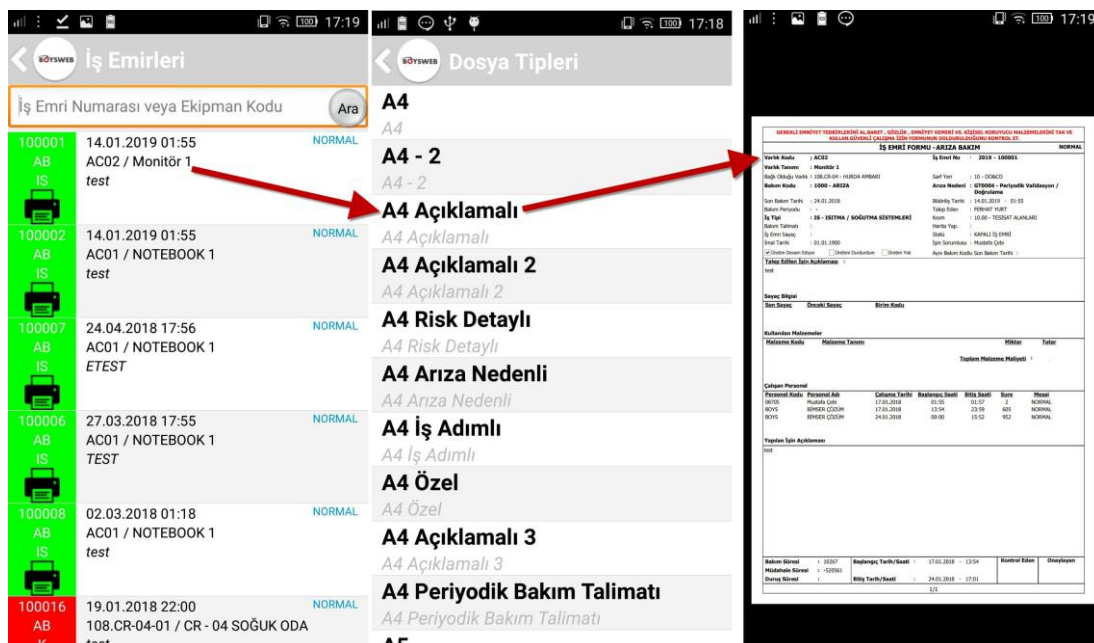
İşletmeler hangi varlıkların hurdaya ayrıldığını, yerine hangi varlıkların aktif olarak kullanıma geçtiklerini, hurdaya ayrılan varlıkların hangi iş emirlerine istinaden hurdaya ayrıldıklarını görmek istemektedirler. Bu isteği gerçekleştirebilmek için Visual Studio üzerine eklenen Devexpress uygulaması kullanılarak varlık taşıma raporunu Şekil 4.44’de dizayn edilmiştir. Bu rapor için BC_EQUIPMENT_TRANSPORT tablosun kullanılarak bir dataset oluşturulmuştur.

4.8.1. Varlık Taşıma Modülünün Geliştirilmesi Sonucu Elde Edilen Kazanımlar

Bakımları yapılamayan kullanılamayacak duruma gelen varlıkların aktif kullanımdan çekilerek yerine yeni varlıkların getirilmesiyle alakalı fonksiyonel yapılan geliştirmeler neticesinde kullanıcılar mobil uygulama veya web uygulaması üzerinden kolayca varlıklarını hurda ambarlarına taşıyıp yerine yeni varlıkları ekleyebilmektedir. Her taşınan varlık için istenildiğinde nereye nasıl taşındığı bilgileri varlık taşıma raporu ile üst yöneticilere sunulmuştur. Böylece işletme kullanılmayan varlıkların hurdaya ayrılma süreçlerini Boysweb uygulaması üzerinden yönetebilmektedirler.

4.9. İŞ EMRİ FORMLARININ MOBİL ÜZERİNDEN ALINMASI

İşletmeler varolan iş emirlerine ait bilgilerin çıktılarını alıp dosyalayabilmek için ve sorumlu kişilere imzalatarak fiziksel olarak kayıt altına alabilmek için iş emirlerini yazdırmak istemektedirler. Yazdırma işlemlerinde farklı formatlar mevcuttur. Formlar A4 ya da A5 kağıt boyutu türünde çıktılar alabilmek üzere dizayn edilmiştir. İş emri üzerindeki açıklama alanına göre, iş adımlarına göre, malzemelere göre formlar farklılık göstermektedir. İş emri üzerinde yazdır butonu kullanılarak form tiplerinin listelendiği ekrana ulaşılır. Buradan hangi form türü seçilirse ona göre yazdırma işlemi gerçekleştirilir. Bu işlemi işletmeler mobil uygulama üzerinden de gerçekleştirmek istemektedirler.



Sekil 4.45. Mobil iş emri form yazdırma özelliği.

Şekil 4.45’de iş emirleri listesi üzerinden form tipi seçilerek mobil uygulamaya formun indirilmesi ve yazıcıya gönderilmesi işlemi gösterilmektedir. Burada iş emri listesi Restfull servis aracılığıyla bir listeye alınarak json veri türünde mobil uygulamaya gönderilmektedir. Kullanıcı iş emri listesi üzerinden yazıcı ikonuna tıkladığında form tiplerinin listelendiği sayfa tetiklenerek açılır. Daha sonra kullanıcı hangi form tipinde çıktı almak istiyor ise o form tipini tıklar.(A4, A4 Açıklamalı vb.) Restfull servis ile ilgili iş emrine ait veriler mobile gönderilir. Mobilde bu veriler dosya gönderim formatlarına göre iletilmiş olan forma aktarılarak telefona indirilir. Kullanıcı isterse bu formu telefon üzerinden bluetooth aracılığı ile yazıcıya bağlanarak yazıcıya gönderebilir.

4.10. ELEKTRONİK İMZA UYGULAMASININ ENTEGRE EDİLMESİ

4.10.1. Elektronik İmza ve Sayısal İmza

Elektronik imza, klasik imzaya tanınan işlevleri de kapsayan ve bir veri mesajında bulunan veya ona eklenen ya da mesaj ile mantıksal bağlantısı kurulabilen, bireyin kimliğini tanıtan ve bireyin, mesajın içeriğini onayladığını gösteren elektronik formattaki imzadır [21].

Sayısal imza, açık anahtar altyapısını kullanarak gerek imzalama gerek şifreleme işlemlerini güvenli bir şekilde yerine getirmektedir. Açık anahtar altyapısı mimarisinde biri imza doğrulama işlevi gerçekleştiren “açık anahtar [22]”, diğeri imza oluşturmaya yarayan “özel anahtar” olmak üzere iki anahtar kullanılmaktadır [23].

Sayısal imza, ıslak imzanın sahip olduğu özellikleri bilgisayar ortamında gerçekleştiren matematiksel formül ve şifreleme algoritmalarına sahiptir. Bugün için sayısal imza, internet işlem güvenliği sağlamak için en çok tercih edilen bir elektronik imza çeşididir [24].

4.10.2. Elektronik İmza Prosedürü

İmzalamak istenen dosya hazır olduğunda imzala butonu ve imza anahtarını yerleştir adımları gerçekleştirilerek imzalama işlemi tamamlanır. İmzalama aşamasında bir imza makamından alınan çip kart, kart okuyucuya yerleştirilir. Bilgisayara tanımlanan algoritmalar işletilerek imzanın tasdik edilmesini sağlanır.

Islak imzayla imzalanan dokümanlardaki gibi elektronik imza ile imzalanan dosyaların altında kişisel bilgilerin içerisinde imzayı atan kişiye özel ek bir veri ile belgenin

tamamlanması sağlanmaktadır.

Elektronik imza kullanılan dokümanları imzalayan kişinin kolay bulunması, imzanın bilgisayar algoritmalarına göre özel olarak tanımlandığından değişme ihtimalinin olmaması, belgenin bilgisayar ortamında saklanması, kolay transferi göz önüne alındığında daha güvenlidir.

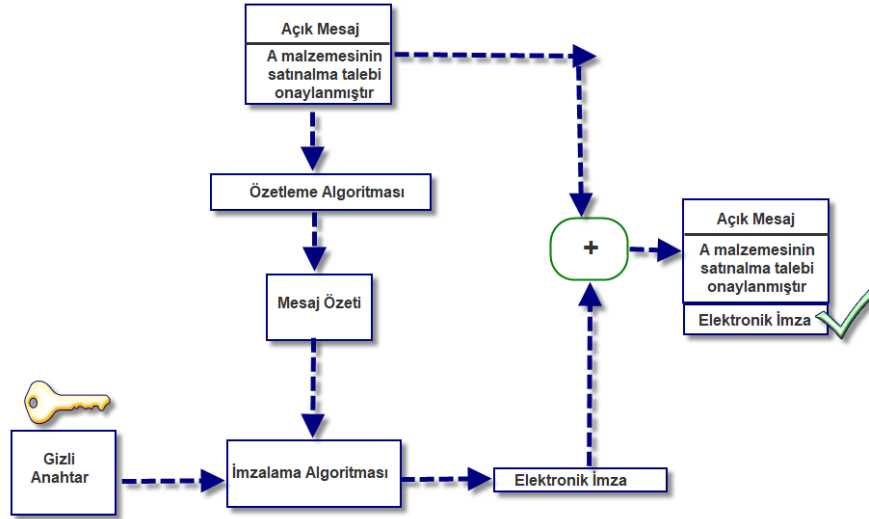
Islak imzalı belgelerin bazı kısımlarında sonradan veri değişikliği yapılabilir. Yeni eklenen veriler ayrıca imzalanmadıklarından imza sahibi bu verileri ben eklemedim derse hukuken bu belge geçersiz sayılabilir. Elektronik imzalı bir belge üzerinde sonradan değişiklik yapıldığı takdirde elektronik imza kontrol edilerek belgenin değiştirilip değiştirilmediği anlaşılabılır. Her değişimde iki tarafında onayı ile değişiklik gerçekleştirilmiş ise elektronik imza tekrar atılarak belgenin son hali muhafaza edilir. Belgenin en son halinin hangisi olduğu bilgisine zaman damgası ile ulaşılabilir.

Elektronik imza sayesinde imzayı atan kişinin başka kişiyle karıştırılmasının önüne geçilmiştir, imza atacak kişiyi değiştirilmeyecek şekilde tekillemektedir [25].

Elektronik imzanın riskli olduğu kısımlarda vardır. Elektronik imza aygıtı ve üzerine takılan çip kaybolabilir başkasının eline geçebilir. Yada bilgisayara takılı unutulmuş çip diğer kişiler tarafından sahibinin izni olmadan imzalama işlemleri için kullanılabilir. Bu gibi durumlarda elektronik imza sahibi kişi kredi kartı çalınması sonucunda müşteri hizmetlerinden kartı iptal ettirdiği gibi çipide iptal ettirebilmektedir.

4.10.3. Elektronik İmza Altyapısı

Elektronik ortamlardan gönderilen mesajlar e-imza açık anahtar şifreleme yöntemleri ve algoritmaları kullanılarak gönderilir.



Şekil 4.46. Elektronik imza algoritması.

Şekil 4.46’da elektronik imza atılması gereken bir dokümanın süreci gösterilmiştir. Burada doküman açık mesaj olarak özetleme algoritmasına gider. Özetleme algoritması kendi içerisinde yarattığı metotlar ile açık mesajı mesaj özetine çevirir. Sisteme takılan soket ve buna ait benzersiz kart ile elde edilen gizli anahtar içerisinde yer alan imzalama algoritması ile mesaj özeti imzalanır. İmzalanan mesaj özeti açık mesaj olarak tekrar dokümana çevrilir. Yeni dokümanda imzası atılmış olarak elde edilir.

4.10.3.1. Kriptografi

Kriptografik algoritmalar dijital imzanın hazırlanmasında, şifrelemede ve kontrol edilmesinde kullanılır. Şifreleme açık olan metni gizli metin haline getirir. Gönderilen şifre alıcı tarafından çözülerek tekrar açık metin haline getirilir.

Simetrik Şifreleme Yöntemi, şifreleme ve çözme işlemlerinde aynı şifre kullanılır. Ortada bir şifre vardır ve iki kişi arasında data alışverişinde bu şifre kullanılmaktadır. Bu alışveriş içerisine 3. Kişi ya da daha fazla kişi katıldığında tüm katılımcıların şifreyi bilmesi gerekmektedir. Şifrenin değişmesi gerektiğinde yeni şifre toplantı tutanağı, e-posta yolu ile dağıtılabilir.

Asimetrik Şifreleme Yöntemi, Birbiri ile haberleşen iki kişiden birisi açık diğeri gizli ve yöndeş iki şifreye sahiptirler. Bir kişi diğer kişiye açık metin göndermek istediğinde diğer kişinin açık şifresini kullanacaktır. Diğer kişi şifreli mesajı almak için kendi gizli şifresini kullanmak zorundadır. Burada kritik konu herkesin kendi açık şifresini ulaşılabilir durumda tutmasıdır. Meydana gelen problem açık şifrenin belirli bir kişiye ait olup olmadığı bilgisinin kesin olmamasıdır. Güvenilirliği gizli olan şifrenin iyi saklanması ve

anahtar uzunluğunun uygun olması ile mümkündür [26].

4.10.3.2. Açık Anahtar Şifreleme

Açık anahtar şifreleme yönetiminde doğrulanmış bir kanal vardır. Ancak simetrik şifreleme yönteminin tersine bu kanalın gizli tutulma şartı söz konusu değildir. Metin alıp verecek her kişi kendi üzerinde bir çift anahtara sahiptir. Bu anahtar çiftlerinden birisi açık diğeri gizli anahtardır. Bir kişi diğeri kişiye metin göndermek istediğinde karşısındaki kişinin açık anahtarının kopyasını kullanarak açık anahtar şifreleme algoritması ile şifreli gizli bir metin elde eder. Metni alacak kişi kendi bünyesindeki gizli anahtarı kullanarak bu metni deşifre algoritmasını kullanarak açar ve açık metni elde etmiş olur. Bu algoritma tam tersi yönde de çalışır. Şifreleme ve deşifreleme işlemleri bu şekilde yapılarak gizlilik ilkesi sağlanmış olur.

4.10.3.3. Güvenli Elektronik İmza Oluşturma ve Doğrulama Araçları

Elektronik imza oluşturma aracı gizli ve güvenli şekilde elektronik imza oluşturan yazılımı içinde barındıran donanıma verilen isimdir. İmza oluşturma araçları donanım bazlı akıllı kartlar, USB Token'lar, bilgisayarlar, el terminalleri, cep telefonları ile yazılım bazlı olarak bilgisayar programları, smartcardlar, işletim sistemleri, özel yazılımlar olabilir. Mevzuatın getirdiği zorunluluklar sonucu elektronik imza oluşturma araçları genel olarak akıllı çubuk (Token) ve ya akıllı kartlardır (Smart Card).



Şekil 4.47. Elektronik imza doğrulama aracı.

Şekil 4.47'de Boysweb üzerine entegre edilen elektronik imza uygulamasının kullanılabilmesi adına bilgisayara tanıtılması gereken yazılımın içerisinde yer aldığı akıllık çubuk ve bu çubuğun bilgisayarı tanınması için gerekli olan ArkSigner e-İmza Sihirbazı ara yazılımı gösterilmektedir.

ArkSigner; ofis, metin, video gibi her türlü dosya ve doküman imzalanması ve kimlik doğrulama işlemlerinde kullanılan CadES(CMS Advanced Electronic Signature), PDF dosyaların imzalanmasında kullanılan PadES(PDF Advanced Elektronik Signature) ve e-Fatura, e-Reçete gibi XML formatlı doküman imzalama için kullanılan XAdES(XML Advanced Electronic Signature) imza formatlarını desteklemektedir. Bahse konu imza formatlarını mevzuatta belirtildiği şekilde ETSI TS 101 733 (CADES), ETS TS 101 903(XAdES), ETSI 102 778(PAdES) standartlarına uygun olarak gerçekleştirmektedir [27].

4.10.4. Elektronik İmza Uygulamasının Boysweb Üzerine Eklenmesi

ArkSigner uygulaması kurulduğunda bilgisayar üzerinde elektronik imza ayarlarını gerçekleştirir. İmzalama sırasında istenilen ekranı ve görseli kullanıcı oluşturur. Sistem üzerinde yüklü olan akıllı kartların listesini verir. İmzalama aşamasında güvenlik için bilgisayar dışına veri gönderimi gerçekleştirmez. ArkSigner uygulaması kurulduktan sonra Boysweb uygulaması üzerinde imzalama uygulamasının geliştirilmesi aşamasında ArkSignerJS javascript dosyası Boysweb uygulaması üzerine eklenerek senkronizasyon sağlanır ve Boysweb uygulaması tüm imzalama ve doğrulama işlemlerini gerçekleştirebilir.

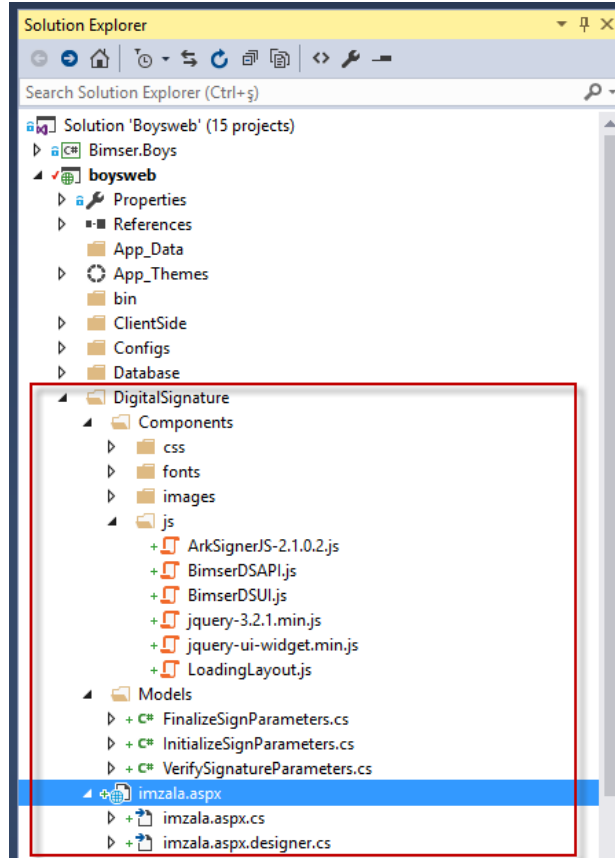
```
<script>
$(document).ready(function () {
    var container = $("#UIContainer");
    $("<div>").BimserDSUI({
        useAsAuthenticator: false,
        description: "Elektronik imza ile giriş yaptıktan sonra sistemde yapacağınız işlem",
        license: "cXq7G34gj5RDqapcFGj755pBAmL+yE7qxr2ptwYbSLbvj4LqFbOH067z7LHFUV+00167nmV",
        signatureFormat: BimserDSAPI.SignatureFormat.PadES,
        signatureType: BimserDSAPI.SignatureType.BES,
        addTimeStamp: false,
        embeddedSignature: true,
        initializeSigningUrl: "imzala.aspx/InitializeSigning",
        finalizeSigningUrl: "imzala.aspx/FinalizeSigning",
        verifySignatureUrl: "api/Sign/VerifySignature",
        onSignCompleted: function () {
            console.log("Sign Completed.");
        },
        onAuthenticated: function () {
            console.log("Authentication Completed.");
        },
        onError: function (ex) {
            console.log("onErrorCallback", ex);
        }
    });
});
```

Şekil 4.48. Elektronik imza doğrulama aracı.

Şekil 4.48’de Boysweb uygulaması içerisinde elektronik imzanın atılmasını sağlayan ekrandaki javascript dosyası içerisinde ;

signatureFormat: BimserDSAPI.SignatureFormat.PadES

parametresi ile ArgSigner uygulamasına PDF formatındaki dosyaların elektronik imza işlemi yapılabileceği değeri dönülmüştür.



Şekil 4.49. Boysweb uygulaması içerisine elektronik imzanın eklenmesi.

Şekil 4.49’da boysweb dizini altında elektronik imzanın eklenmesi sırasında oluşturulan yeni dosya (DigitalSignature) içerisinde javascript dosyalarının bulunduğu dizinde ArkSignerjs-2.1.0.2.js dosyası eklenmiştir. Böylece Boysweb uygulaması ile ArgSigner uygulaması uyumlu hale getirilmiştir.

DigitalSignature dosyası içerisine Components klasörü altında arayüze ait dosyalar yer almaktadır. Models klasörü altında proje içerisinde kullanılacak olan nesnelerin oluşturulduğu kısımdır. İmzanın boyu, tipi, bilgileri gibi nesneler models altında yer alır. İmzala.aspx sayfası içerisinde kullanıcının gördüğü arayüz yer almaktadır. Bu bir web sayfasıdır. Proje yapısına göre farklı tasarımlarda yapılabilir.

localhost/BOYSWEB214/

localhost/BOYSWEB214/DigitalSignature/imzala.aspx

Parola

Kart Okuyucu

ACS CCID USB Reader 0

Sertifika

TR 11111111111 Test Bir Kullar

Sertifika Sahibi

Test Bir Kullanıcı

Kimlik No

11111111111

Sertifikayı Veren

CN=Kamu Elektronik Sertifika Hizmet Sağlayıcısı - TEST,OU=BİLGEM,O=Türkiye Bilimsel ve Teknolojik Araştırma Kurumu - TÜBİTAK,L=Gebze - Kocaeli,C=TR

Sertifika Seri No

71126857417485

Geçerlilik

Geçerli

Elektronik imza ile giriş yaptıktan sonra sistemde yapacağınız işlemler, ıslak imza ile yapılan işlemlerle aynı hukuki sonucu doğurmaktadır.

Taahhütname tarihi: 12.01.2018 14:37

Şekil 4.50. Boysweb uygulaması imzalama arayüzü.

Boysweb uygulaması içerisinde imzalama fonksiyonu tetiklendiğinde Şekil 4.50'deki ekran açılmaktadır. Bu ekran içerisinde akıllı çubuk bilgisayara takılıp tanımlandığında kart okuyucu ve sertifika bilgileri akıllı çubuk içerisindeki yazılım üzerinden çekilerek buraya aktarılır. Akıllı çubuk Kamu Elektronik Sertifika Hizmet Sağlayıcısı tarafından sağlanmıştır. Kullanıcı kendisine ait parolayı girerek imzala butonuna tıklandığında belgenin elektronik imzası atılır. Yeni belge Boysweb üzerinden seçilen klasöre kaydedilir.

Satınalma Talep Formu

Talep Tarihi	: 01.02.2018 / 14:43	Aciliyet	: 1
Talep Numarası	: 162	Belge Türü	: ÖRNEK
Talep Eden	: BİHŞER ÇÖZÜM	Masraf Yürü	: 1
Talep Edinin Görevi	: ELEKTRİK TEKNİSYENİ / KONTROLÖRÜ	Teslim Ambar	: MERKEZ AMBAR
Talep Eden Kısım	: BULASIDHANE / POTWASH	Önerilen Firma	: AKIŞ PAZARLAMA TAAHHÜT İNŞAAT
Masraf Yeri	: DORCO		
Varlık Tanımı	:		
Açıklama	: notlar		

Sıra No	Malzeme Kodu	Malzeme Tanımı	Birim	Talep Miktarı	Ambar Miktarı	Teslim Ambar	Malzeme Açıklaması
1	10052	KANYON LASTİĞİ	ADET	1,00	0,00	MERKEZ AMBAR	
2	1021	ŞAŞE CİVATALARI-somun	ADET	1,00	0,00	MERKEZ AMBAR	
3	1027	Kasa yan kapı malakası	ADET	1,00	0,00	MERKEZ AMBAR	
4	1062	İNDERNE BUTONU	ADET	1,00	1,00	MERKEZ AMBAR	
5	1092	Kapı yayı	ADET	1,00	0,00	MERKEZ AMBAR	

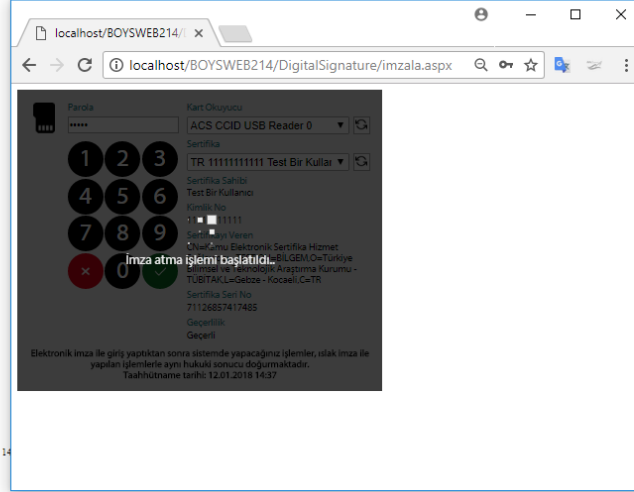
Onay Bilgileri

Onaylayan
BOYS BOYS
BOYS BOYS

Onaylanan Açıklaması

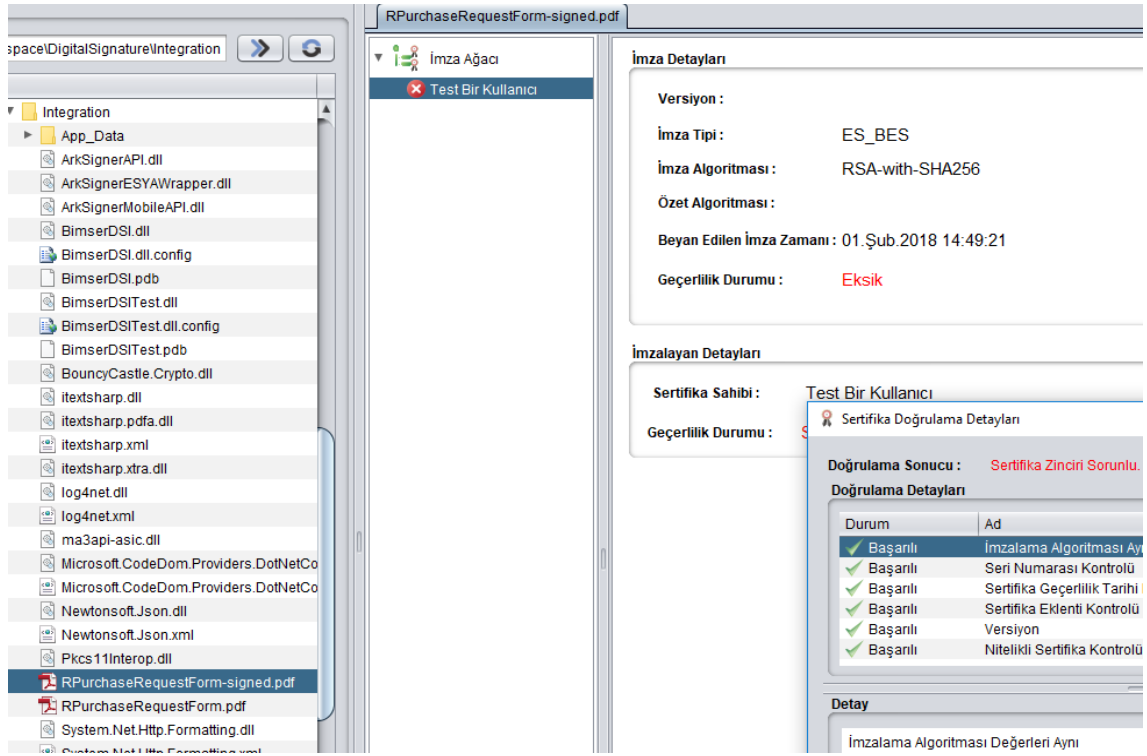
Onay Durumu
Onaylandı
Onaylandı

Onay Tarihi
01.02.2018 14:45
01.02.2018 14:45



Şekil 4.51. Boysweb satınalma talep formunun imzalanması.

Şekil 4.51’de Boysweb uygulaması içerisinde yer alan satınalma talep formunun elektronik olarak imzalanma aşaması gösterilmektedir. Satın alınacak malzemeler için sistemde satınalma talebi oluşturulmaktadır. Üst yöneticiler satınalma talep formunu inceleyerek onaylamaktadırlar. Onaylanan satınalma taleplerinin çıktılarının alınıp imzalanarak dosyalanması süreci başlar. Boysweb uygulaması kağıtsız ofis mantığını gerçekleştirebilmek adına satınalma talep formlarının çıktılarının alınıp ıslak imzalı saklanması önüne geçmek için satınalma talep formu açıldığında bir buton aracılığı ile elektronik imza ekranını tetiklemektedir. Açılan ekranda akıllı çubuk üzerindeki yazılım içerisinde yer alan imzalama algoritmaları ile satınalma talep formu imzalanır. Açılan satınalma karar formu imzala dendiğinde form PDF olarak Boysweb uygulamasının klasörü altına indirilir. İndirilen klasör imzalama işlemi gerçekleştirildikten sonra imzalı hali tekrar PDF dosya türüne çevrilerek aynı klasör altına –signed.pdf olarak kaydedilir. Böylece elektronik olarak imzalanan dosyalar sanal ortamda saklanabilmektedir.



Şekil 4.52. Elektronik imzası atılan dosyanın imzalama durumunun gösterilmesi.

Elektronik imzalı belgelerin imza çözümleri ve imzalı belgelerin takibi onayı gibi işlemleri kullanıcının takip etmesi zordur. Kullanıcıyı bu süreçten kurtaran basit ve standartlarla uyumlu bir elektronik imzalı doküman oluşturma ve doğrulama aracı önemli bir ihtiyaç haline gelmiştir. İmzager uygulaması her türlü elektronik belgenin ETSI TS 101 733 CADES ve ETSI TS 101 903 XADES standartlarında tanımlı basit ve gelişmiş imza formatlarında imzalanmasını ve imzalı dokümanların imza bilgilerinin görüntülenip doğrulanabilmelerini sağlar [28].

İmzager uygulaması yardımı ile Boysweb uygulaması üzerinden imzalanan satınalma talep formu dosyası bulunarak açılır ve imzanın başarılı olup olmadığı kontrol edilir. İmzanın başarılı olduğu bilgisi Şekil 4.52’de gösterilmiştir.

4.10.5. Elektronik İmza Uygulaması Kazanımları

Boysweb uygulamasına elektronik imzanın eklenmesi ile imzalanması gereken dosyaların sanal ortamda imzalanarak saklanması sağlanmıştır. Excel formlarda bakımlarını takip eden şirketlerin daha kolay ve kullanılabilir bir ortama taşınması için Boysweb uygulamasına geçerek bakımlarında hız, güvenlik ve kalite sağlamalarının yanında aynı zamanda elektronik imza uygulaması eklenerek kağıtsız evrak yönetimi sağlamışlardır.

4.11. GELİŞTİRİLEN UYGULAMANIN KULLANIMI

Bu bölümde de işletmeler bakım prosedürlerini ve işleyişlerini uygulama üzerinden yönetirlerken ihtiyaçları doğrultusunda geliştirilen yeni modüllerin ve fonksiyonların nasıl kullanıldığına ilişkin bilgilerin gösterimi gerçekleştirilmiştir.

Periyodik Kontrol Bilgileri

Şablon Numarası: 1

Bakım / Arıza Kodu*: 01 KORUYUCU BAKIM

İş Tipi Kodu*: A ATÖLYE

Bakım Periyodu*: 1 Periyod Birimi*: G-GÜN

Bakım Yapılacak Tarih: 21.02.2018

Varlıklar Bakım Planı Ölçüm Paketleri Açıklama

Plan Kodu	Plan Tanımı	Açıklama	Aktif	İş Tipi Gruplu Oluştur	Bakım Süresi
10000	AKÜLÜ FORKLİFT 3 AYLIK BAKIM PLANI		☒	☒	8

Kaydet Vazgeç

Şekil 4.53. Periyodik (Günlük) kontrol şablon ekranı.

Şekil 4.53’de işletmeler günlük olarak gerçekleştirecekleri işlerin sisteme tanımlamasını yapmaktadırlar. Ekran üzerinden ekipmanlar, bu ekipmanlara uygulanacak bakım planları ve ölçüm paketleri seçilir. Şablonun hangi periyotlarda forma dönüşmesi gerekiyorsa seçilir.

Periyodik Kontrol Formu Bilgileri					
Genel Bilgiler		Kontrol Yapılan Varlıklar	Çalışan Personel	İş Adımları	Ölçüm Paketleri
İş Adımları			TKM01		
SNO	İş Adımı Kodu	İş Adımı Tanımı	Yapıldı mı?	Sorun Var	Süre
2	1070	EMNİYET SİVİCİLERİ KONTROLÜ	☐	☐	0
4	1130	HİDROLİK SEVİYE VE KAÇAK KONTROLÜ	☐	☐	0
3	1183	KORUYUCU MUHAFAZALARIN KONTROLÜ	☐	☐	0
1	1299	GÖSTERGE KUMANDA PANELİ KONTROLÜ	☐	☐	0
5	3500	SON TEKNİK KONTROL VE TEMİZLİK YAPILDI	☐	☐	0

Şekil 4.54. Periyodik (Günlük) kontrol form ekranı.

Şekil 4.54’de şablon, periyodik kontrol formuna dönüştüğünde kullanıcı önüne gelen ekran yer almaktadır. Bu ekranda kullanıcılar kontrol yapılacak varlıkları görmektedir. Bu varlıklara günlük kontrolleri gerçekleştirecek personelleri seçmektedir. Yapacakları kontrollere göre varlıklara uygulanan iş adımları ve ölçüm değerleri verileri kontrol edilip işaretlenir. Sorun olan iş adımı ya da ölçüm değeri için sorun var seçildiğinde sorunlu olan varlık için iş adımı ya da ölçüm paketide içerisine alınarak iş emri oluşmaktadır.

Şekil 4.55. Malzeme sayımı modülü ana ekranı.

Şekil 4.55’de malzeme sayımı işlemini gerçekleştirmek için sayım yap fonksiyonu kullanılır. Yapılan sayımların listesinin görüntülenmesi için sayım listesi ya da bugün oluşturulan fonksiyonları kullanılmaktadır.



Şekil 4.56. Malzeme sayımı modülü sayım ekranı.

Şekil 4.56’da sayıma başlanıldığında malzemelerin barkod okuyucu yardımı ile cihaz kamerasıyla ya da malzeme listesi üzerinden seçilip miktar ve sayaç numaraları seçilerek sayımların liste ekranına aktarılması aşaması yer almaktadır.



Şekil 4.57. Malzeme sayımı modülü liste ekranı.

Şekil 4.57’de sayımı tamamlanan malzemeler listeye alınır. Kayıt işlemi gerçekleştirildiğinde malzemeler Restfull servis yardımı ile sunucuya bağlanarak JSON

veri formatında sunucuya aktarılır. Sunucu üzerinde BC_ITEM_INVENTORY tablosuna kayıt işlemi gerçekleştirilir. Malzeme sayımı sonucunda istenilen sayım numarasındaki sonuçlar malzeme hareketine uygulama üzerinden dönüştürülebilir.

Bakım Talimatı

Filtre

Talep Ettiklerim

Onay Vereceklerim

Hepsi

Onay Bekleyen

Onaylanmış

Reddedilmiş

Hepsi

1

1

10

3

Drag a column header here to group by that column

Bakım Talimatı Kodu	Bakım Talimatı Tanımı	Onay	Bakım Talimatı Durumu
CR-01	EA-04/02 , EURAMET CG-13 , 7.6.1	Onay Bekliyor	Açık
DKK-01	7.6.1	Onay Bekliyor	Açık
test1	test 1	Onay Bekliyor	Açık

Şekil 4.58. Bakım talimatı ekranı.

Şekil 4.58’de bakım talimatı ana ekranı yer almaktadır. Bu ekran üzerinde hangi talimatlarda değişiklik olduğu ve onay beklediği bilgisi ile statü bilgisi yer almaktadır.

Bakım Talimatı						
Kayıt Kullanımda ☒						
Bakım Talimat Kodu:* CR-01						
Bakım Talimatı Tanımı: EA-04/02 , EURAMET CG-13 , 7.6.1						
Talimat Tipi: Bakım Talimatı						
Statü: Açık						
Genel Bilgiler Talimat Bakım Kodları Bağlı Dokümanlar Onay Bilgileri						
Onaylayanın Kodu	Onaylayan	Onay Seviyesi	Onaya Gönderilme Tarihi	Onaylanma Tarihi	Onay Durumu	
BOYS	BOYS BOYS	1	19.01.2018			
	BOYS BOYS	2	19.01.2018		Onay Bekliyor	
Onay Reddet Vazgeç						

Şekil 4.59. Bakım talimatı onay bilgileri.

Şekil 4.59’da onay bekleyen bakım talimatı içerisinde talimatın genel bilgileri, statüsü, bakım kodları, talimat açıklaması, onay bilgileri yer almaktadır. Onaylama işlemi

gerçekleştirildiğinde bakım talimatı tekrar kullanıma açılır. Kullanıma açılan bakım talimatı bakım planına eklenir. Bakım planı periyodik bakım tanımlarına eklenerek oluşacak iş emirleri içerisine atanıp kullanıcılara sunulur. Talimatta meydana gelen yeni değişiklikler için onay döngüsü tekrar devreye girer.



Şekil 4.60. Grafiksel varlık ağacı çizimi.

Şekil 4.60’da grafiksel varlık ağacı ekranı yer almaktadır. Mode bölümünde izleme, çizme, değiştirme fonksiyonları yer almaktadır. Varlık ağacı üzerinden fabrika, sarfyeri, kısım, varlık, alt varlık birimlerine resimler ve krokiler eklenebilmektedir. Çizme fonksiyonu ile kroki üzerine varlıklar yerleştirilerek varlıkların yerleri belirlenebilmektedir. İzleme fonksiyonu ile açık iş emirlerine sahip varlıklar kırmızı renkli, periyodik iş emirlerine sahip varlıklar mavi renkli, duruşlu iş emirlerine sahip varlıklar siyah renkli olarak kroki üzerinde gösterilebilmektedir.



Şekil 4.64. Elektronik imzalı dökümanların saklanması.

Şekil 4.64’de elektronik imza aşamasından geçen satınalma talepleri Döküman Yönetim Sistemi uygulaması altında BOYSWEB2 klasörü içerisinde DigitalSignature alt klasörü içerisinde tutulmaktadır. Buraya BOYSWEB uygulaması üzerinden otomatik atanan imzalı dökümanların oluşturulma ve değiştirilme tarihleri, kim tarafından oluşturuldukları bilgileri de yer almaktadır.

4.12. GELİŞTİRİLEN UYGULAMANIN TEST DURUMLARI

Yapılan geliştirmeler firmalara iletilmeden önce fonksiyonel testlerden ve performans testlerinden geçmektedir. Yapılan geliştirmelerin son kullanıcılardan önce geliştirme tamamlandıktan sonra kullanıcıya açılmadan önce test edilebilmesi adına test adımları tanımlanmıştır. Tanımlanan test adımlarının gerçekleşmesi durumunda elde edilecek sonuçlar belirlenmiştir.

4.12.1. Periyodik (Günlük) Kontrol Modülü Test Senaryosu

Çizelge 4.1’de periyodik (günlük) kontrol modülü içerisinde şablon oluşturabilmek için gerçekleştirilmesi gereken adımların yer aldığı test senaryosu örneği yer almaktadır.

Çizelge 4.1. Günlük kontrol şablonu oluşturma test senaryosu.

1. Senaryo Ön Koşulu	
No	Ön Koşul
1.1	En az bir tane Bakım/Arıza Kodu tanımı bulunması gerekmektedir.
1.2	En az bir tane Periyot Birimi tanımı bulunması gerekmektedir.
1.3	En az bir tane Varlık tanımı bulunması gerekmektedir.
1.4	En az bir tane Bakım Planı tanımı bulunması gerekmektedir.
1.5	En az bir tane Ölçüm Paketi tanımı bulunması gerekmektedir.

Çizelge 4.1.(Devam) Günlük kontrol şablonu oluşturma test senaryosu.

2. Test Adımları					
No	İşlem Türü	İşlem	Beklenen Sonuç	Gerçekleşen Sonuç	Statü
2.1	Listeleme	Menü üzerinde “Bakım - > Periyodik Kontroller - > Periyodik Kontrol Şablonu” tıklanır.	Sistemde Periyodik Kontrol Şablonları görüntülenir.		
2.2	Düzenleme	Ekle butonuna tıklanır.	Periyodik Kontrol Şablonu kayıt sayfası görüntülenir.		
2.3	Düzenleme	Bakım/Arıza Kodu seçilir.	Seçilen Bakım/Arıza Kodu görüntülenir.		
2.3	Düzenleme	Bakım Periyodu numerik olarak girilir.	Girilen Bakım Periyodu görüntülenir.		
2.4	Düzenleme	Bakım Periyot Birimi seçilir.	Seçilen Bakım Periyot Birimi görüntülenir.		
2.5	Düzenleme	“Varlık” sekmesinden “+” butonuna tıklanır.	Varlık seçim satırı aktif olur.		
2.6	Düzenleme	Varlık seçim satırından “Varlık Kodu” sütununda bulunan “...” butonuna tıklanır.	Varlıklar sayfası açılır.		
2.7	Düzenleme	Açılan “Varlıklar” sayfasından ilgili varlıklar sol çift tıkla seçilir.	Seçilen varlıklar tanımlarıyla birlikte görüntülenir.		
2.8	Düzenleme	“Bakım Planı” sekmesinden “+” butonuna tıklanır.	Bakım Planı seçim satırı aktif olur.		

Çizelge 4.1.(Devam) Günlük kontrol şablonu oluşturma test senaryosu.

2.9	Düzenleme	Bakım Planı seçim satırından ‘‘Plan Kodu’’ sütununda bulunan ‘‘...’’ butonuna tıklanır.	Bakım Planı seçim sayfası açılır.		
2.10	Düzenleme	Açılan ‘‘Bakım Planı’’ sayfasından ilgili Bakım Planı sol çift tıkla seçilir.	Seçilen Bakım Planı tanımlarıyla birlikte görüntülenir.		
2.11	Düzenleme	‘‘Ölçüm Paketi’’ sekmesinden ‘‘+’’ butonuna tıklanır.	Ölçüm Paketi seçim satırı aktif olur.		
2.12	Düzenleme	Ölçüm Paketi seçim satırından ‘‘Ölçüm Paketi Kodu’’ sütununda bulunan ‘‘...’’ butonuna tıklanır.	Ölçüm Paketi seçim sayfası açılır.		
2.13	Düzenleme	Açılan ‘‘Ölçüm Paketi’’ sayfasından ilgili Ölçüm Paketi sol çift tıkla seçilir.	Seçilen Ölçüm Paketi tanımlarıyla birlikte görüntülenir.		
2.14	Düzenleme	‘‘Kaydet’’ butonuna tıklanır.	Periyodik Kontrol Şablonu sayfasında tanımladığımız şablon görüntülenir.		
3. Veri Doğrulama					
No	Doğrulama				
3.1	Bakım / Arıza Kodu seçilmeden kaydetme işlemi gerçekleştirilemez.				
3.2	Bakım Periyodu seçilmeden kaydetme işlemi gerçekleştirilemez.				
3.3	Varlık seçilmeden kaydetme işlemi gerçekleştirilemez.				
3.4	Bakım Periyot Birimi seçilmeden kaydetme işlemi gerçekleştirilemez.				
3.5	Bakım Planı Kodu seçilmeden veya Ölçüm paketi seçilmeden kaydetme işlemi gerçekleştirilemez.				

Çizelge 4.2’de periyodik (günlük) kontrol modülü içerisinde form oluşturabilmek için gerçekleştirilmesi gereken adımların yer aldığı test senaryosu örneği yer almaktadır.

Çizelge 4.2. Günlük kontrol formu oluşturma test senaryosu.

1. Senaryo Ön Koşulu					
No	Ön Koşul				
1.1	En az bir tane Periyodik Kontrol Formu Şablonu bulunması gerekmektedir.				
2. Test Adımları					
No	İşlem Türü	İşlem	Beklenen Sonuç	Gerçekleşen Sonuç	Statü
2.1	Listeleme	Menü üzerinde “Bakım - > Periyodik Kontroller - > Periyodik Kontrol Şablonu “ tıklanır.	Sistemde Periyodik Kontrol Şablonları görüntülenir.		
2.2	Düzenleme	Yeni Kontrol Formu Oluştur butonuna tıklanır.	Yeni bir Periyodik Kontrol formu oluşturması beklenir.		
3. Veri Doğrulama					
No	Doğrulama				
3.1	Otomatik olarak yeni form numarası ile Periyodik Kontrol Şablonunda bulunan verilere göre Periyodik Kontrol Formu oluşturmalıdır.				

Çizelge 4.3’de periyodik (günlük) kontrol modülü içerisinde oluşturulan formu tamamlayıp kapalı statüsüne almak için gerçekleştirilmesi gereken adımların yer aldığı test senaryosu örneği yer almaktadır.

Çizelge 4.3. Günlük kontrol formu kapatma test senaryosu.

1. Senaryo Ön Koşulu	
No	Ön Koşul
1.1	En az bir tane Periyodik Kontrol Formu oluşturulmuş olmalıdır.
1.2	En az bir tane Personel tanımlanmış olmalıdır.
1.3	Bakım Planı içerisinde bulunan İş Adımlarının Varsayılan İş Tipleri Ve Bakım/Arıza Kodları tanımlanmış olmalıdır.
1.4	Ölçüm Paketi içerisinde bulunan İş Emri Bilgileri Alanı doldurulmuş olmalıdır.

Çizelge 4.3.(Devam) Günlük kontrol formu kapatma test senaryosu.

2. Test Adımları					
No	İşlem Türü	İşlem	Beklenen Sonuç	Gerçekleşen Sonuç	Statü
2.1	Listeleme	Menü üzerinde “Bakım - > Periyodik Kontroller - > Periyodik Kontrol Formu“ tıklanır.	Sistemde Periyodik Kontrol Formları görüntülenir.		
2.2	Düzenleme	Kapatılmak istenen Periyodik Kontrol Formu çift tıkla açılır.	Periyodik Kontrol Formu sayfasında Genel Bilgiler sekmesi görüntülenir.		
2.3	Düzenleme	Başlangıç Tarihi ve Saati girilir.	Giriş yapılan tarihler GGAAYY saatlar 24 saat dilimi olarak görüntülenir.		
2.4	Düzenleme	Bitiş Tarihi ve Saati girilir.	Giriş yapılan tarihler GGAAYY saatlar 24 saat dilimi olarak görüntülenir.		
2.5	Düzenleme	Yapılan İşin Açıklaması alanı doldurulur.	Numara ve Yazı formatında yazılan veriler görüntülenir.		
2.6	Düzenleme	“Çalışan Personel ” sekmesinden “+” butonuna tıklanır.	Personel seçim satırı aktif olur.		
2.7	Düzenleme	Personel seçim satırından “Personel Kodu” sütununda bulunan “...” butonuna tıklanır.	Kaynaklar sayfası açılır.		

Çizelge 4.3.(Devam) Günlük kontrol formu kapatma test senaryosu.

2.8	Düzenleme	Açılan “Kaynaklar” sayfasından ilgili personeller sol çift tıklara seçilir ve Kaynak Çalışma Süresi girilir.	Seçilen Personeller çalışma süreleri tanımlarıyla birlikte görüntülenir.		
2.9	Düzenleme	“İş Adımları” sekmesine tıklanır.	Bakım Planı içerisinde yer alan İş adımları seçilen Varlıklar ile birlikte görüntülenir.		
2.10	Düzenleme	Kontrolü yapılan Varlık ilgili İş Adımı için “Yapıldı mı?” alanı işaretlenir.	“Yapıldı mı?” Alanında tik işareti görüntülenir.		
2.11	Düzenleme	Kontrolü yapılan Varlık ilgili İş Adımı için “Sorun Var mı?” alanı yapılan kontrol de sorun varsa işaretlenir.	“Sorun Var mı?” Alanında tik işareti görüntülenir.		
2.12	Düzenleme	Kontrolü yapılan Varlık ilgili İş Adımı için “Süre” alanı gerçekleşen süre miktarı ile güncellenir.	“Süre” alanında girilen değer görüntülenir.		
2.13	Düzenleme	“Ölçüm Paketleri” sekmesine tıklanır.	ÖlçümPaketlerinin standart değerleri ve data girişinin yapılacağı alanlar görüntülenir.		
2.14	Düzenleme	Ölçümü yapılan Varlık ilgili Ölçüm Paketi için “Ölçüm Değeri” alanına veri girişi yapılır.	Ölçüm Değeri girildiğinde ilgili alan yeşil renk almalıdır.		
2.15	Düzenleme	Genel Bilgiler sekmesinden “Açık” durumu “Kapalı” ya çevirilerek Kaydet Butonuna tıklanır.	Periyodik Kontrol Formu sayfasında, kapalı duruma çevrilen formun Periyodik Kontrol Formu Durumu yeşil renk almalıdır.		

Çizelge 4.3.(Devam) Günlük kontrol formu kapatma test senaryosu.

3. Veri Doğrulama	
No	Doğrulama
3.1	Başlangıç Tarihi ve Saati girilmeden kapatılamaz.
3.2	Bitiş Tarihi ve Saati girilmeden kapatılamaz.
3.3	Yapılan İşin Açıklaması girilmeden kapatılamaz.
3.4	İş Adımları Yapıldı olarak işaretlenmeden kapatılamaz.
3.5	Ölçüm Değeri girilmeden kapatılamaz.
3.6	Çalışan Personel girilmeden kapatılamaz.
3.7	İş Adımlarına Sorun Var işaretlendi ise; İlgili İş Adımının üzerinde Tanımlı İş Tipi ve Bakım/Arıza Koduna göre kontrolü yapılan varlık için otomatik İş Emri oluşturmalıdır.
3.8	Ölçüm Değeri, Ölçüm Paketi üzerinde tanımlanan Alt Kabul, Üst Kabul ve Tolerans değerlerini geçecek bir değer girilmiş ise; Ölçüm Paketi üzerinde Tanımlanan İş Emri Bilgilerine göre ölçümü yapılan Varlık için otomatik İş Emri oluşturmalıdır.

4.12.2. Bakım Talimatı Geliştirmesi Test Senaryosu

Çizelge 4.4’de bakım talimatı oluşturabilmek için gerçekleştirilmesi gereken adımların yer aldığı test senaryosu örneği yer almaktadır.

Çizelge 4.4. Bakım talimatı oluşturma test senaryosu.

1. Senaryo Ön Koşulu					
No	Ön Koşul				
1.1	En az bir İş Emri Türü tanımlı olmalıdır.				
1.2	En az bir İş Tipi tanımlı olmalıdır				
1.3	En az bir Bakım Arıza Kodu tanımlı olmalıdır.				
1.4	En az bir Kaynak tanımlı olmalıdır.				
2. Test Adımları					
No	İşlem Türü	İşlem	Beklenen Sonuç	Gerçekleşen Sonuç	Statü
2.1	Listeleme	Menü üzerinde “Bakım - >Periyodik Bakım- >Bakım Talimatları” tıklanır.	Sistemde kayıtlı olan Bakım Talimatları listelenir.		
2.2	Görüntüleme	Ekle butonu tıklanır.	Bakım Talimatı kayıt sayfası görüntülenir.		

Çizelge 4.4.(Devam) Bakım talimatı oluşturma test senaryosu.

2.3	Düzenleme	Bakım Talimatı Kodu yazılır.	Yazılan Bakım Talimatı Kodu görüntülenir.		
2.4	Düzenleme	Bakım Talimatı Tipi seçilir.	Seçilen Bakım Talimatı Tipi görüntülenir.		
2.5	Düzenleme	İş Tipi seçilir.	Seçilen İş Tipi görüntülenir.		
2.6	Düzenleme	İş Emri Türü seçilir.	Seçilen İş Emri Türü görüntülenir.		
2.7	Düzenleme	Hazırlayan seçilir.	Seçilen Kaynak görüntülenir. Uygulamaya giriş yapan kullanıcının kaynağı mevcutsa otomatik dolu olarak gelir.		
2.8	Düzenleme	Koordinatör seçilir.	Seçilen Kaynak görüntülenir.		
2.9	Düzenleme	Yürürlük tarihi seçilir.	Seçilen tarih görüntülenir.		
2.10	Düzenleme	Dağıtım yazılır.	Yazılan dağıtım görüntülenir.		
2.11	Düzenleme	Ait Olduğu birim yazılır.	Yazılan birim görüntülenir.		
2.12	Düzenleme	Revizyon Numarası yazılır.(Nümerik)	Yazılan revizyon numarası görüntülenir.		
2.13	Düzenleme	Revizyon Tarihi seçilir.	Seçilen Tarih görüntülenir.		
2.14	Düzenleme	Revize Eden seçilir.	Seçilen Kaynak görüntülenir.		
2.15	Düzenleme	Revizyon Nedeni yazılır.	Revizyon nedeni görüntülenir.		
2.16	Düzenleme	Referans No yazılır.	Yazılan Referans no görüntülenir.		
2.17	Düzenleme	Talimat sekmesinden Bakım talimatı tanımı yazılır.	Yazılan bakım talimatı tanımı görüntülenir.		
2.18	Düzenleme	Bakım kodları sekmesine tıklanır.	Bakım talimatına bağlı bakım arıza kodları görüntülenir.		
2.19	Düzenleme	+ Butonuna basılarak, bakım arıza kodu seçilir.	Seçilen bakım arıza kodu görüntülenir.		

Çizelge 4.4.(Devam) Bakım talimatı oluşturma test senaryosu.

2.20	Düzenleme	Değişiklikleri uygulamak için yeşil tik butonuna basılır.	Seçilen bakım arıza kodu kaydedilir. Tekrar + butonuna basılarak yeni bakım kodları seçilebilir.		
2.21	Düzenleme	Dökümanlar sekmesine basılır.	Bakım talimatına bağlı dökümanlar görüntülenir.		
2.22	Düzenleme	+ Butonuna basılarak, bakım arıza kodu seçilir.	Seçilen Döküman görüntülenir.		
2.23	Düzenleme	Değişiklikleri uygulamak için yeşil tik butonuna basılır.	Seçilen Döküman kaydedilir. Tekrar + butonuna basılarak yeni ları seçilebilir.		
2.24	Kaydetme	Kaydet butonu tıklanır.	Bakım Talimatlarına yönlendirilir. Yönlendirilen sayfada kaydedilen Bakım Talimatları görüntülenir.		
3. Veri Doğrulama					
No	Doğrulama				
3.1	Bakım Talimatı Kodu yazılmadan kaydetme işlemi gerçekleştirilemez.				
3.2	Bakım Talimatı Tanımı yazılmadan kaydetme işlemi gerçekleştirilemez.				
3.3	Hazırlayan yazılmadan kaydetme işlemi gerçekleştirmez.				

4.12.3. Malzeme Hareketi Test Senaryosu

Çizelge 4.5’de malzeme giriş hareketi oluşturabilmek için gerçekleştirilmesi gereken adımların yer aldığı test senaryosu örneği yer almaktadır.

Çizelge 4.5. Malzeme giriş hareketi oluşturma test senaryosu.

1. Senaryo Ön Koşulu	
No	Ön Koşul
1.1	Malzeme Giriş Hareket Tanımı olmalıdır
1.2	Ambar Tanımı olmalıdır.
1.3	Malzeme tanımlanmalıdır.

Çizelge 4.5.(Devam) Malzeme giriş hareketi oluşturma test senaryosu.

2. Test Adımları					
No	İşlem Türü	İşlem	Beklenen Sonuç	Gerçekleşen Sonuç	Statü
2.1	Listeleme	Menü üzerinde “Malzeme - > Malzeme Hareketleri” tıklanır.	Sistemde kayıtlı olan Malzeme Hareketleri listelenir.		
2.2	Görüntüleme	Ekle butonu tıklanır, Malzeme Girişi seçilir.	Malzeme Malzeme Hareketleri kayıt sayfası görüntülenir.		
2.3	Düzenleme	Fiş No yazılır.	Yazılan Fiş No görüntülenir.		
2.4	Düzenleme	Fiş Tarihi yazılır.	Yazılan Fiş Tarihi görüntülenir.		
2.5	Düzenleme	Fiş Saati yazılır.	Yazılan Fiş Saati görüntülenir.		
2.6	Düzenleme	Ambar seçilir.	Seçilen Ambar görüntülenir. Eğer ambar üzerinde Kısım seçili ise kısım ve sarfyeri alanları otomatik olarak dolar.		
2.7	Düzenleme	Kısım seçilir.	Seçilen Kısım görüntülenir. Sarfyeri alanı otomatik olarak dolar.		
2.8	Düzenleme	Sarfyeri seçilir.	Seçilen Sarfyeri görüntülenir.		
2.9	Düzenleme	‘+’ butonu tıklanır, malzeme kodu seçilir.	Seçilen Malzeme görüntülenir.		
2.10	Düzenleme	Seçilen Malzemenin Miktarı girilir.	Seçilen Malzeme Miktarı görüntülenir.		
2.11	Düzenleme	Yeşil tik butonun tıklanır	Seçilen Malzeme kaydedilir.		

Çizelge 4.5.(Devam) Malzeme giriş hareketi oluşturma test senaryosu.

3. Veri Doğrulama	
No	Doğrulama
3.1	Fiş No otomatik olarak sistem tarafından oluşturulur.
3.2	Fiş Tarihi yazılmadan kaydetme işlemi gerçekleştirilemez.
3.3	Fiş Saati yazılmadan kaydetme işlemi gerçekleştirilemez.
3.4	Ambar seçilmeden kaydetme işlemi gerçekleştirilemez.
3.5	En az bir tane malzeme kaydı girilmeden kaydetme işlemi gerçekleştirilemez.
3.6	Malzeme giriş hareketi oluşturulduktan sonra toplam malzeme miktarı, hareket kaydında girilen malzeme miktarı kadar artar.

Çizelge 4.6’da malzeme çıkış hareketi oluşturabilmek için gerçekleştirilmesi gereken adımların yer aldığı test senaryosu örneği yer almaktadır.

Çizelge 4.6. Malzeme çıkış hareketi oluşturma test senaryosu.

1. Senaryo Ön Koşulu					
No	Ön Koşul				
1.1	Malzeme Çıkış Hareket Tanımı olmalıdır				
1.2	Ambar Tanımı olmalıdır.				
1.3	Malzeme tanımlanmalıdır.				
2. Test Adımları					
No	İşlem Türü	İşlem	Beklenen Sonuç	Gerçekleşen Sonuç	Statü
2.1	Listeleme	Menü üzerinde “Malzeme - > Malzeme Hareketleri” tıklanır.	Sistemde kayıtlı olan Malzeme Hareketleri listelenir.		
2.2	Görüntüleme	Ekle butonu tıklanır, Malzeme Çıkışı seçilir.	Malzeme Malzeme Hareketleri kayıt sayfası görüntülenir.		
2.3	Düzenleme	Fiş No yazılır.	Yazılan Fiş No görüntülenir.		
2.4	Düzenleme	Fiş Tarihi yazılır.	Yazılan Fiş Tarihi görüntülenir.		
2.5	Düzenleme	Fiş Saati yazılır.	Yazılan Fiş Saati görüntülenir.		

Çizelge 4.6.(Devam) Malzeme çıkış hareketi oluşturma test senaryosu.

2.6	Düzenleme	Ambar seçilir.	Seçilen Ambar görüntülenir. Eğer ambar üzerinde Kısım seçili ise kısım ve sarfyeri alanları otomatik olarak dolar.		
2.7	Düzenleme	Kısım seçilir.	Seçilen Kısım görüntülenir. Sarfyeri alanı otomatik olarak dolar.		
2.8	Düzenleme	Sarfyeri seçilir.	Seçilen Sarfyeri görüntülenir.		
2.9	Düzenleme	‘+’ butonu tıklanır, malzeme kodu seçilir.	Seçilen Malzeme görüntülenir.		
2.10	Düzenleme	Seçilen Malzemenin Miktarı girilir.	Seçilen Malzeme Miktarı görüntülenir.		
2.11	Düzenleme	Yeşil tik butonun tıklanır	Seçilen Malzeme kaydedilir.		
2.12	Kaydetme	Kaydet butonu tıklanır.	Malzeme Hareketlerine yönlendirilir. Yönlendirilen sayfada kaydedilen Malzeme Hareketleri görüntülenir.		
3. Veri Doğrulama					
No	Doğrulama				
3.1	Fiş No yazılmadan kaydetme işlemi gerçekleştirilemez.				
3.2	Fiş Tarihi yazılmadan kaydetme işlemi gerçekleştirilemez.				
3.3	Fiş Saati yazılmadan kaydetme işlemi gerçekleştirilemez.				
3.4	Ambar seçilmeden kaydetme işlemi gerçekleştirilemez.				
3.5	En az bir tane malzeme kaydı seçilmeden kaydetme işlemi gerçekleştirilemez.				
3.6	Malzeme çıkış hareketi oluşturulduktan sonra toplam malzeme miktarı, hareket kaydında girilen malzeme miktarı kadar azalır.				

4.12.4. Varlık Taşıma Test Senaryosu

Çizelge 4.7’de varlık taşımak için gerçekleştirilmesi gereken adımların yer aldığı test senaryosu örneği yer almaktadır.

Çizelge 4.7. Varlık taşıma test senaryosu

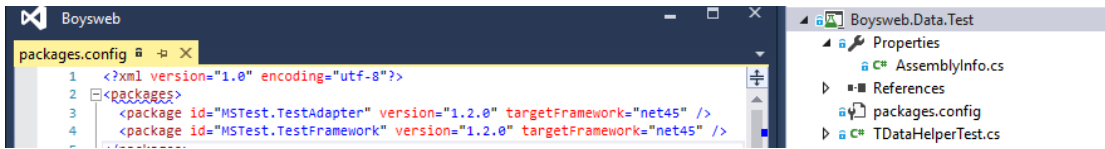
1. Senaryo Ön Koşulu					
No	Ön Koşul				
1.1	En az iki tane varlık kaydı bulunmalıdır.				
1.2	En az bir tane kısım bulunmalıdır.				
1.3	En az bir tane kaynak bulunmalıdır.				
2. Test Adımları					
No	İşlem Türü	İşlem	Beklenen Sonuç	Gerçekleşen Sonuç	Statü
2.1	Listeleme	Menü üzerinde “Varlık - > Varlık Taşıma” tıklanır.	Sistemde kayıtlı olan İşletmeler listelenir.		
2.2	Görüntüleme	Ekle butonu tıklanır.	Varlık taşıma kayıt sayfası görüntülenir.		
2.3	Düzenleme	Varlık seçilir.	Seçilen varlık görüntülenir.		
2.4	Düzenleme	Bulunduğu kısım seçilir.	Seçilen kısım görüntülenir.		
2.5	Düzenleme	Yeni sahip varlık seçilir.	Seçilen yeni sahip varlık görüntülenir.		
2.6	Düzenleme	Yeni kısım seçilir.	Seçilen yeni kısım görüntülenir.		
2.7	Düzenleme	İşlemi yapan seçilir.	Seçilen işlemi yapan görüntülenir.		
2.8	Düzenleme	Tarih seçilir.	Seçilen tarih görüntülenir.		
2.9	Düzenleme	Açıklama yazılır.	Yazılan açıklama görüntülenir.		
2.10	Kaydetme	Kaydet butonuna tıklanır.	Varlık taşıma listesi sayfasına yönlendirilir.Yönlendirilen sayfada kaydedilen varlık taşıma kayıtları görüntülenir.		

Çizelge 4.7.(Devam) Varlık taşıma test senaryosu

3. Veri Doğrulama	
No	Doğrulama
3.1	Transfer numarası otomatik artar, manuel girilmez.
3.2	Varlık seçilmeden kaydetme işlemi gerçekleştirilemez.
3.3	Varlık seçildiğinde bulunduğu kısım otomatik olarak gelir.
3.4	Yeni kısım seçilmeden kaydetme işlemi gerçekleştirilemez.
3.5	Tarih alanına sadece kısa tarih formatında bir değer girilebilir.
3.6	Saat alanına sadece 24 saat diliminde değer girilmelidir.
3.7	Tarih ve saat yazılmadan kaydetme işlemi gerçekleştirilemez.
3.8	İşlemi yapan seçilmeden kaydetme işlemi gerçekleştirilemez.
3.9	Varlık ağacında ring oluşacağı durumda kayıt işlemi gerçekleştirilmez.

4.12.5. Birim Test Metodları

Uygulamanın geliştirilme aşamasında yazılan metodların istenilen değerleri alıp işledikten sonra istenilen değerleri geri döndürdüğünü kontrol edebilmek için metodların test metodları yazılmaktadır. Yazılan test metodlarına ve bu metodların çıktılarına birim test denilmektedir. Geliştirilen uygulamada birim testlerin yazılabilmesi için Boysweb.Data.Test projesi oluşturulmuştur. Oluşturulan proje Boysweb kod dizilimi içerisinde proje kütüphaneleri arasına eklenmiştir.



Şekil 4.65. Birim test proje yapısı.

Şekil 4.65’de Boysweb.Data.Test projesi ve bu projenin config dosyası içerisine eklenen TestAdapter ve TestFramework paketleri yer almaktadır. Bu paketler içerisinde yazılan metodların test sonuçlarının kontrolünü sağlayan sınıflar yer almaktadır.

Çizelge 4.8’de uygulama içerisinde kullanılan örnek birim test metodları ve dönen sonuçları yer almaktadır.

Çizelge 4.8. Birim test metodları.

Fonksiyon Adı	Fonksiyon Açıklaması	Test Kodu	Dönen Sonuç
ToString	Toplam Ek sayısını çevirmek için kullanılır	<pre>[TestMethod()] public void ToString() { Dagitim target = new Dagitim(); string actual = ""; string expected = ""; actual = target.ToString(14, false); Assert.AreEqual(expected, actual); }</pre>	true
DateReplace	Tarih ayracını . - / değiştirmek için kullanılır.	<pre>[TestMethod()] public void DateReplaceTest() { eYazisma target = new eYazisma(); string date = "27.07.2016"; string actual = ""; string expected = "27/07/2016"; actual = target.DateReplace(date); Assert.AreEqual(expected, actual); }</pre>	true

Çizelge 4.8.(Devam) Birim test metodları.

Olustur	Toplam sayfa sayısını bulmak ve numaralandırmak için önce template olarak oluşturulur.	<pre>[TestMethod()] public void OlusturTest1() { eBAFormYazisma target = new eBAFormYazisma(); try { target.Olustur(151517, stream, true, "192.168.42.56", "", false,true); Assert.IsTrue(true); } catch { Assert.IsTrue(false); } }</pre>	true
BelgeSayiGetir	Belge ID ile Belge Sayısı getirir.	<pre>[TestMethod] public void BelgeSayiGetir() { int belgeId = 1559930; int gercekDeger = 28613; int beklenenDeger = BELGE.BelgeSayiGetir(belgeId); Assert.AreEqual(beklenenDeger, gercekDeger); }</pre>	true

Çizelge 4.8.(Devam) Birim test metodları.

BelgePostaBilgi leriKaydet	Posta Gönderi Detayları döner	[TestMethod] public void BelgePostaBilgileriKaydet() { int bdgtId = 1; int bdgtHgtId = 1; string gonderiDetay = "Test Gönderi detay"; string beklenenDeger = BELGE.PostaBilgileriKaydet(bdgtId, bdgtHgtId, gonderiDetay); Assert.AreEqual(beklenenDeger, ""); }	true
GetFormat	eYazisma classından gelen Tarihayrac değeri 1 ise / değil ise . Formatta çevrilir.	[TestMethod() public void GetFormatTest() { eBAFormYazisma target = new eBAFormYazisma(); char expected = '/'; char actual = ' '; actual = target.GetFormat(eyazisma); Assert.AreEqual(expected, actual); }	true

Çizelge 4.8.(Devam) Birim test metodları.

SonEkiGetir	Devam eden cümleden son ekinin ne ile tamamlanacağı nı getirmek için kullanılır.	<pre>[TestMethod()] public void SonEkiGetirTest() { Ilgi target = new Ilgi(DateTime.Now, "11515", "", "", 1, ' '); string actual = target.SonEkiGetir("a", "redis kurulum ve konfigurasyon"); string expected = "ın"; Assert.AreEqual(expected, actual); }</pre>	true
-------------	--	--	------

Çizelge 4.8.(Devam) Birim test metodları.

SonUnluKarakterBul	dökümandaki son ünlü karakteri geri döndüren fonksiyondur.	<pre>[TestMethod()] public void SonUnluKarakterBulTest() { Ilgi target = new Ilgi(DateTime.Now, "11515", "", "", 1, ' '); string actual = target.SonUnluKarakterBul("redis kurulum ve konfigurasyon",27, "on"); string expected = "o"; Assert.AreEqual(expected, actual); }</pre>	true
Imzalimi	Path i parametre geçilen dökümanın imzalı olup olmadığını gösterir.	<pre>[TestMethod()] public void ImzalimiTest() { string yol = "C:\\Users\\kkocer\\Desktop\\SPICE_19 062017\\Projeler\\EBYS_Taslak\\Rapor lar\\02.001.01 Proje Başlama Raporu"; eBAFormYazisma target = new eBAFormYazisma(); bool actual = target.Imzalimi(yol); bool expected = false; Assert.AreEqual(expected,actual); }</pre>	true

Çizelge 4.8.(Devam) Birim test metodları.

ImzaEkle	Dökümana imza eklemeyi sağlayan fonksiyon	<pre>[TestMethod()] public void ImzaEkleTest() { eBAFormYazisma target = new eBAFormYazisma(); eBAProject project = new eBAProject("ebys", "eBAFormYazismaOlusturucu"); eBAFormData.eBAForm imzaForm = new eBAFormData.eBAForm(1,"process","f orm"); eYazisma eyazisma = new eYazisma(); try { target.ImzaEkle(imzaForm, eyazisma); Assert.IsTrue(true); } catch { Assert.IsTrue(false); } }</pre>	true
----------	--	--	------

Çizelge 4.8.(Devam) Birim test metodları.

EkListesineEkle riEkle	Oluşturulan ek listeye ekleri ekler	<pre>[TestMethod()] public void EkListesineEkleriEkleTest() { WordDocument doc = new WordDocument(); eYazisma target = new eYazisma(); try { target.EkListesineEkleriEkle(doc); Assert.IsTrue(true); } catch { Assert.IsTrue(false); } }</pre>	true
---------------------------	---	--	------

5. SONUÇLAR VE ÖNERİLER

Yapılan geliştirme ile firmaların bakım yönetimi iş süreçleri iyileştirilmiştir. Geliştirilen iş süreçleri ile bakım yönetiminde bir standartlaşmaya gidilmiştir. Günlük Kontrol modülü ile standart süreçler tanımlanarak bakım ekipleri ve operatörler arasındaki iletişim problemi çözülmüş olur. Bu çalışma sonucunda hangi ekipmanların kritik ekipman olduğu belirlenir. Bu ekipmanlara düzenli olarak hangi iş adımları doğrultusunda bakım yapılacağı bilgisi tanımlanır. Ayrıca bu ekipmanların bakımında hangi tür ölçümlerin hangi noktalara yapılacağı bilgisi kayıt altına alınmıştır. Böylece bu bilgiler bir sistematik yapı altına toplanmıştır. Bakımdan sorumlu olan her personelin herhangi bir varlık hakkında istediği anda istediği bilgiye hızlı ve doğru bir şekilde erişimi sağlanmaktadır. Ayrıca iş adımları, bakım planları, ölçüm standartları, ölçüm paketleri, periyodik bakım tanımlarının da standartlaştırılıp sisteme aktarılmasıyla günlük bakım tanımları ortaya çıkmış olur. Arıza bildirilirken de operasyon ve teknik ekip ile ortak bir dil yaratılmış olur. Tolerans değerine göre zamanı gelen planlı bakımın daha önceden sorumlulara otomatik olarak mail bildirimi yapması sağlanmıştır. Planlı bakımın zamanı geldiğinde otomatik olarak iş emri oluşturması sağlanmıştır.

Yönetim sistemi tarafından otomatik olarak planlı bakım sayaç bilgisi iş emri üzerine aktarılması ile planlı bakımda zor olan sayaç yönetimi kolay bir şekilde yönetilebilir hale gelmiştir.

Bakım arıza kodu ve arıza nedeni bazında alınabilen maliyet analizleri ile hangi arıza nedeninden dolayı işletmede en çok maliyet oluştuğu konusu belirlenir. Böylece işletme için hayati önem arz eden maliyet hesabı açığa kavuşmuş olur.

Sistemden alınacak varlık tarihçe raporu ile ilgili varlığa daha önce hangi günlük kontroller ve bakım planları uygulandığı bilgisine ulaşılır. Hangi ölçüm paketi uygulandığı ve hangi ölçüm noktalarında değerlerin yüksek çıktığı verisi toplanır. Hangi iş adımlarında sorun belirlendiği verisinin dökümü alınabilir. Böylece herhangi bir ekipman için arıza nedenine göre maliyet hesabı yapılır. Oluşan arızanın ne kadar sürede giderildiği hesaplanır. Arızanın giderilmesinde izlenen yola ulaşıldığında iyileştirme

alıřmaları iin nemli detay bilgiler raporlanabilir hale gelmiřtir.

Periyodik bakımların gerekleřtirilmesi iin iř adımlarına alternatif olarak geliřtirilen bakım talimatları zelliğinin bir onay sistemine baėlanması ile ekipmanların dzenli olarak deėiřen bakım talimatlarının periyodik bakımlara uygulanması ve BOYSWEB sisteminde gncel tutulması saėlanmıřtır. Deėiřecek olan bakım talimatlarının st yneticilerin onayından sonra kullanılabilir hale gelmesi ile yanlıř talimatların bakım sırasında ekipmanlara uygulanmasının nne geilerek son versiyon bakım talimatları ile ekipmanların bakımlarının gerekleřmesi saėlanmıřtır. Bakım talimatlarına eklenen tarihe zelliėi ile hangi bakım talimatlarının kim tarafından hangi tarihlerde deėiřtirildiėi anlık olarak kayıt altına alınarak yanlıř bakım talimatlarının ekipmana uygulanarak ekipmanın hurdaya ayrılması gibi istenmeyen durumların oluřmasının nne geilmeye alıřılmıřtır.

Grafiksel varlık aėacı modlnn BOYSWEB sistemine kazanılması ile iřletmeler sahip oldukları tm ekipmanları tek bir sayfadan takip edebilme isteklerine kavuřmuřlardır. İřletmede hangi ekipmanın nerede yer aldıėı, anlık olarak bu ekipmana aılmıř bir iřin olup olmadıėı, anlık olarak bu ekipmanın duruř yařayıp yařamadıėı gibi bilgileri tek bir ekrandan gvenlik kamerasında izlercesine grafiksel varlık aėacından kontrol edebilmektedirler. Bylece iřletmeler iin hayati nem tařıyan duruř yařanmasının minimuma indirilmesi ve duruřtan dolayı kaybedilen zamanda kaybedilen maliyetlerin nne geilmesi hedeflenmiřtir. Bu hedef doėrultusunda gerekleřtirilen grafiksel varlık aėacı ile iřletmeler gemiře ynelik kayıtlara da aynı sayfa zerinden gidebilmektedirler. Ekranda yer alan ekipmanlardan herhangi birine anlık olarak iř talebi, iř emri aabilmektedirler. Aynı řekilde bu ekipmanlara nceden hangi iřlerin aıldıėı ve kimlerin bu iřleri yaptıėı bilgilerine de bu ekrandan ulařabilmektedirler. Bu modl ile iřletmeler tm ekipmanlarının gemiře ynelik bilgilerine tek sayfadan ulařabilmekte ve geleceėe ynelik iřlemlerinide tek sayfadan ynetebilmektedirler.

Ekipman tařıma uygulamasının BOYSWEB sistemine kazandırılması ile iřletmeler hurdaya ayıracakları ekipmanları ve yerine gelecek yeni ekipmanları tek bir iř emri ierisinden seerek hızlı bir řekilde transfer edebilme imkanı kazanmıřlardır. Bu uygulamanın mobil uygulama zerine de eklenmesi ile sahada alıřan bakımcılar yada yneticiler hurdaya ayıracakları ekipmanları mobil cihazları zerinden iki alanı doldurarak hurdaya ayırmıř olmaktadır. Bu uygulama iřletmelere hız kazandırmaktadır. Zaman alacaėı dřnlerek hurdaya ayrılan ekipmanların iřlemleri

bittikten sonra aylık olarak sisteme girilmesinin önüne geçilmiştir. Hurda olarak görülen ekipmanlar değiştirilme anında mobil cihazları üzerinden de değiştirilerek BOYSWEB sisteminde en güncel ekipman listesinin tutulması sağlanmıştır.

Elektronik imza uygulamasının BOYSWEB' e kazandırılması ile formların çıktılarının alınıp ıslak imza atılarak dosyalarda fiziksel olarak saklanması önüne geçilmiştir. Formlar elektronik olarak imzalanarak sistem üzerinde tutulmaktadır. Böylece denetimlerde ve resmi süreçlerde sanal ortamda kayıtları yer alan elektronik imzalı dokümanlar üzerinden işlemler yapılarak ıslak imza, fiziksel saklama, kağıt kullanımı gibi zahmetli durumların önüne geçilmiştir.

Firmanın kaç tane ambarı var, hangi ambarda hangi malzemeleri var, bu malzemelerin miktarları nelerdir, hangi malzeme kaç adet kullanılmış, kaç adet kalmış gibi verileri mobil cihaz üzerinden görüp yönetilebilmesi için malzeme sayımı modülü geliştirilmiştir. Bu modüllerin eklenmesi ile kullanıcılar mobil cihaz üzerinden kameralarını barkod okuyucu şeklinde kullanabilirler. Ambara girip malzeme barkodlarını okutarak ambar sayımını hızlı bir şekilde gerçekleştirebilirler. Mobil cihaz üzerinden iş emrine malzeme aktarımı yapabilirler. Azalan malzemeler için malzeme giriş hareketi, iş emrine gönderilecek malzemeler için malzeme çıkış hareketi yapabilirler. Ambarlar arasında malzeme aktarımı için malzeme transferi hareketlerini gerçekleştirebilirler.

BOYSWEB uygulaması işletmelerin bakım sistemlerini kapsarken ileriye yönelik bazı dönüşümlere ihtiyaç duymaktadır. 10.000 veri sonrası sayfaların yüklenme hızlarında azalmalar meydana gelmektedir. Sayfalara yerleştirilen Devexpress GridView bileşenleri üzerine kayıtlar çekilmekte ve GridView birleşeni üzerinde sayfalama işlemleri gerçekleştirilmektedir. BOYSWEB uygulamasının tüm ana sayfalarında işleyiş bu şekildedir. Ancak bu durum varlık sayfası gibi 10.000 kayıt üstü kayıtların sunucudan çekilip istemciye yüklenmesi sırasında bekleme süresini uzatmaktadır. Bu durumun önüne geçilebilmesi için verinin sunucudan çekilip GridView' de listelenmesi ekran üzerinde sayfanın yerine, ana ekranda her bir sayfa değişikliğinde GridView' den bir istek gönderilerek sunucudan çekilen veri önbelleğe alınmalı ve her sayfa değişiminde önbelleğe istek gönderilerek veri önbellekten çekilmeli. Ancak bu durumda önbelleğin şişmesine ve tarayıcının isteğe cevap verme gücünü kaybetmesine sebep olabilir. Olumsuz durumun önüne geçebilecek asıl geliştirme GridView'deki her bir sayfadan istek geldiğinde sunucuya gidip o sayfa verisinin çekilerek istemciye aktarılması olacaktır.

BOYSWEB uygulamasının mobil versiyonunda internet bağlantısının olmadığı ortamlarda kullanıcıların işlemlerini gerçekleştirmesi ve değişen verilerin bir alanda çevrimdışı tutularak internete bağlanılan ilk ortamda sunucuya gönderilmesi sağlanması gerekmektedir. İşletmeler malzeme sayımlarını çevrimdışı ortamlarda gerçekleştirip ambarlarından çıktıktan sonra internetin çektiği ortamlarda sayılan malzeme miktarlarının sunucuya aktarılmasının büyük kolaylık sağlayacağını bildirmektedirler. Çevrimdışı çalışan uygulamaların en bilineni Google Maps uygulamasıdır. Bu uygulama maksimum 2 GB boyutundaki haritaları çevrimiçi durumda indirerek çevrimdışı olduğunuz durumlarda indirdiği veri üzerinden yol tarifini gerçekleştirmektedir. 30 gün boyunca internet bağlantısı olmadığı durumda cihaz haritayı silmektedir. BOYSWEB mobil uygulamasına benzer şekilde çevrimiçi iken belirli bir boyutta verinin (malzemeler, ambarlar, mevcut malzeme miktarları vb.) indirilmesi sağlanabilir. Çevrimdışı durumda bu veriler üzerinden malzeme sayımı işlemleri gerçekleştirilerek indirilen veriler üzerindeki değişiklikler mobil cihaz üzerinde tutulabilir. İnternet bağlantısı sağlandığı durumlarda indirilen ve değiştirilen bu veriler sunucu üzerindeki veriler ile karşılaştırılarak sunucu verileri üzerinde güncelleme işlemleri gerçekleştirilebilir.

BOYSWEB uygulaması içerisinde koordinat bilgileri girilen varlıkların kordinat bilgileri mobil uygulamaya Restfull servis aracılığıyla aktarılmaktadır. Mobil uygulama bu kordinatları baz alarak varlıkların konumlarına göre bakımçıların varlıklara ulaşımını sağlamaktadır. Dağınık şekilde farklı lokasyonlarda, farklı illerde yer alan işletme bölümlerine ulaşım bu şekilde sağlanmaktadır. Bu geliştirme üzerinde yapılacak yeni değişiklikler ile kordinatları doğrultusunda açılan varlıkların üzerinde yapılacak işler ya da duruş yaşanıp yaşanmadığı bilgileri görülebilir. Grafiksel varlık ağacı gibi mobil cihaz navigasyon ekranından varlıkların durumları anlık izlenerek müdahale süresi hızlandırılabilir.

BOYSWEB uygulamasında veriler veritabanında tablolarda yer almaktadır. Verilerin sayfalara ve raporlara çekilmesi esnasında SQL sorguları kullanılarak tablolar birleştirilmekte ve SQL View yapıları sayesinde birleştirilen tablolar tek bir isim altında toplanarak sunucudan istemciye SQL View'lerin dataları aktarılmaktadır. Aktarılan datalar program içerisinde veri kaynaklarına ayrılarak sayfalara yüklenmektedir. İş Emri, Malzeme Hareketi gibi çok sayıda veri ve fonksiyona sahip sayfalarda SQL View'lerin boyutları büyümektedir. Tek bir İş Emri sayfası için 25 tablo birleştirilerek V_Work_Order viewi oluşturulmaktadır. Bu View içerisinde her bir tablodaki kolonların

bilgisi yer almaktadır. Oluşan veri yumağının sunucudan istemciye alınması aşamasında süre kaybı yaşanmaktadır. Bu durumun önüne geçebilmek için XML formatlarında tablo kolon isimleri program içerisine kaydedilebilir. Oluşturulacak olan SQL kütüphaneleri ile XML formattaki veriler ile veritabanı tablo kayıtları eşleştirilir. Sayfalarda ve raporlarda Viewlerin yüklenmesi yerine XML yapıdaki veriler SELECT içerisine alınarak hangi kolonlar gerekli ise o kolonların SQL kütüphanelerinden istemci ekranlarına gönderilmesi sağlanabilir.

BOYSWEB uygulaması yeni modülleri ve fonksiyonları ile firmaların bakım süreçlerine ait tüm işlemleri tek bir sistem üzerinden sorunsuz olarak yönetebilmeleri için gelişmelere devam edilmektedir.



6. KAYNAKÇA

- [1]. A. C. Marquez, “The maintenance management framework: A practical view to maintenance management”, Ph.D. dissertation, Department Industrial Management, University of Seville, Seville, Spain, 2009.
- [2]. M. F. Taşkın, “Önleyici bakım politikası altında optimum stok miktarının bulanık mantık yöntemiyle belirlenmesi”, Yüksek lisans tezi, Endüstri Mühendisliği Bölümü, Sakarya Üniversitesi, Sakarya, Türkiye, 2006.
- [3]. A. Bal, “Üretim tesisleri için RFID destekli bakım yönetimi”, Yüksek lisans tezi, Endüstri Mühendisliği Bölümü, İstanbul Teknik Üniversitesi, İstanbul, Türkiye, 2013.
- [4]. S. Doğan, “Bilgisayar destekli bakım yönetim sisteminin Kardemir Ray Profil Haddehanesinde uygulanabilirliği”, Yüksek lisans tezi, Makine Mühendisliği Bölümü, Karabük Üniversitesi, Karabük, Türkiye, 2014.
- [5]. M. M. Ayrancı, “Bilgisayar destekli bakım yöntemleri ve gemilerde bakım yönetimi”, Yüksek lisans tezi, Gemi Mühendisliği Bölümü, İstanbul Teknik Üniversitesi, İstanbul, Türkiye, 1997.
- [6]. W. S. Chaneski, “Total productive maintenance an effective technique”, *Modern Machine Shop*, vol. 75, pp. 46-47, 2002.
- [7]. İstanbul Teknik Üniversitesi Bilişim ve Teknoloji Bölümü. (2013,11 Eylül). “Service Oriented Architecture(SOA)” [Online]. Erişim: [http://bidb.itu.edu.tr/sevirdefteri/blog/2013/09/06/servis-y%C3%B6nelimli-mimari-\(service-oriented-architecture-soa\)](http://bidb.itu.edu.tr/sevirdefteri/blog/2013/09/06/servis-y%C3%B6nelimli-mimari-(service-oriented-architecture-soa)).
- [8]. G. Acar, “Otomotiv sektöründe kullanılan kestirimci bakım yöntemleri ve uygulamaları”, Yüksek lisans tezi, Makine Mühendisliği Bölümü, Kocaeli Üniversitesi, Kocaeli, Türkiye, 2014.
- [9]. B. B. Küçük, “Orman ürünleri endüstrisinde toplam verimli bakıma yönelik sektörel altyapının belirlenmesi, İnegöl örneği”, Yüksek lisans tezi, Orman Endüstri Mühendisliği, Düzce Üniversitesi, Düzce, Türkiye, 2016.
- [10]. Fatik Boy, (2010, 19 Mart) “Appfabric Önbellek Mimarisi C#Nedir?Com”, [Online]. Erişim: <http://www.csharpnadir.com/articles/read/?id=1066>.
- [11]. O. H. Şenferah, “Yüksek lisans öğrenci ön kabul otomasyon sistemi için arayüzlerin tasarlanması ve uygulanması”, Yüksek lisans tezi, Elektrik Elektronik Mühendisliği, Dumlupınar Üniversitesi, Kütahya, Türkiye, 2012.
- [12]. M. H. Akündüz, “Dağıtık mimari tasarımı ve yönetim sistemi geliştirmesi”, Yüksek lisans tezi, Bilgisayar Mühendisliği Bilimleri-Bilgisayar ve Kontrol, İstanbul Teknik Üniversitesi, İstanbul, Türkiye, 2015.
- [13]. Yavuz Taşçı, (2011, 4 Aralık). “Windows Server AppFabric Kurulumu ve Genel

- Yapılandırma Ayarları, Çözüm Park*", [Online]. Erişim: http://www.cozumpark.com/blogs/windows_server/archive/2011/12/04/windows-server-appfabric-kurulumu-ve-genel-yapilandirma-ayarlari.aspx.
- [14]. O. Akkaya, "Savaş gemilerinde RAM analizi", Yüksek lisans tezi, Gemi Mühendisliği, İstanbul Teknik Üniversitesi, İstanbul, Türkiye, 2013.
- [15]. M. Kılıçaslan, "E-ders içeriklerinde kullanılabilir çevrimiçi çoklu video aktarımı için örnek bir uygulama", Yüksek lisans tezi, Bilgisayar Mühendisliği Bilimleri-Bilgisayar ve Kontrol, Süleyman Demirel Üniversitesi, Isparta, Türkiye, 2013.
- [16]. L. Juval. *Mastering WCF and the Azure AppFabric Service Bus*, 3rd ed., Boston, USA: O'Reilly Media, 2010, pp. 30-40.
- [17]. C. L. Josiah. *Redis In Action*, Shelter Island, USA: Mannig Publications, Salvatore Sanfilippo, 2013, pp. 200-240.
- [18]. E.C. Yılmaz, R. Oktaş, B. Karabulut, "Web application development with docker", *Journal of Global Engineering Studies*, vol. 3, pp. 12-15, 2016.
- [19]. Chamila Nishantha, (2012, 5 Mayıs) "*Concatenate report pages into one report (XtraReport)*", [Online]. Erişim: <https://www.codeproject.com/Tips/379448/Concatenate-Report-Pages-into-one-Report-XtraReport> .
- [20]. W. Kory, "Microsoft, SAP, and DevExpress reporting services", *University of Wisconsin*, vol.1, pp.5-10, 2015.
- [21]. S. Canbaz, "Türkiye' de elektronik ticaret uygulamasında karşılaşılan muhasebe sorunları", Yüksek lisans tezi, Sosyal Bilimler Enstitüsü, Trakya Üniversitesi, Edirne, Türkiye, 2006.
- [22]. Elektronik İmza Kanunu, TC Resmi Gazete, Sayı: 5070, 23.01.2004, Madde-3/f.
- [23]. B. Yalçınkaya, "Elektronik imzalı belgelerin yönetilmesi ve arşivlenmesi", Yüksek lisans tezi, Türkiyat Araştırmaları Enstitüsü, Marmara Üniversitesi, İstanbul, Türkiye, 2008.
- [24]. K. Berber, (2016, 25 Ekim), Elektronik İmzanın Düzenlenmesi Hakkında Kanun Tasarısı Hükümlerinin Değerlendirilmesi, Erişim: http://www.hukukcu.com/bilimsel/kitaplar/e_imza_tasarilestiri.htm.
- [25]. M. G. Ahi, (2003, 27 Ocak), Sayısal İmza Nedir?, Erişim: <https://gokhanahi.com/2003/01/27/sayisal-imza-elektronik-imza-nedir/>.
- [26]. K. Berber, "İnternet Üzerinden yapılan İşlemlerde Elektronik Para ve Dijital İmzalama", Yetkin, Ankara, ss. 108-111, 2002.
- [27]. ArkSigner, (2018, 1 Şubat), Ark Dan.Bilisim Teknolojileri Mek. San. Tic. Ltd. Sti., Erişim: <https://www.arksigner.com/arksigner-eimza>.
- [28]. İmzager, (2018, 1 Şubat), Kamu Sertifikasyon Merkezi, Erişim: <https://yazilim.kamusm.gov.tr/?q=node/5>.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Kenan KOÇER
Doğum Tarihi ve Yeri : 10 Haziran 1990 / Tokat
Yabancı Dili : İngilizce
E-posta : kenankocer90@gmail.com

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Y. Lisans	Elektrik Elektronik ve Bilgisayar Müh.	Düzce Üniversitesi	2018
Lisans	Bilgisayar Müh.	Dumlupınar Üniversitesi	2013
Lise	Fen Bilimleri	Gümüşhane Fen Lisesi	2009