KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

DOKTORA TEZİ

PROJE YAŞAM DÖNGÜSÜ PARAMETRELERİ İLE PROJE PERFORMANSLARININ ÖLÇÜLMESİNE YÖNELİK YAZILIM SİSTEMİNİN GELİŞTİRİLMESİ

ÖZGÜR GÜN

KOCAELİ 2020

KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

DOKTORA TEZİ

PROJE YAŞAM DÖNGÜSÜ PARAMETRELERİ İLE PROJE PERFORMANSLARININ ÖLÇÜLMESİNE YÖNELİK YAZILIM SİSTEMİNİN GELİŞTİRİLMESİ

ÖZGÜR GÜN

Dr. Öğr. Üyesi Pınar Yıldız KUMRU **Danışman, Kocaeli Üniversitesi**	.....
Prof.Dr. Zerrin ALADAĞ **Jüri Üyesi, Kocaeli Üniversitesi**	.....
Dr. Öğr. Üyesi Didem Yılmaz ÇAPKUR **Jüri Üyesi, İstanbul Gelişim Üniversitesi**	.....
Doç. Dr. Özer UYGUN **Jüri Üyesi, Sakarya Üniversitesi**	.....
Dr.Öğr. Üyesi Celal ÖZKALE **Jüri Üyesi, Kocaeli Üniversitesi**	.....

Tezin Savunulduğu Tarih: 16.07.2020

ÖNSÖZ VE TEŞEKKÜR

Öncelikle, doktora çalışmamdaki uzmanlığı, anlayışı, rehberliği ile sabrı ve desteği için Dr. Öğr. Üyesi Pınar Yıldız Kumru'ya en derin saygılarımı ve minnettarlığımı sunuyorum.

Tez çalışmamın tüm aşamalarında bilgi ve destekleriyle katkıda bulunan, çalışmalarıma yön veren Prof. Dr. Zerrin Aladağ ve Dr. Öğr. Üyesi Didem Yılmaz'a anlayışları ve geri bildirimleri için sonsuz teşekkürlerimi sunuyorum.

Örnek uygulamaları sırasında yardımları ve katkıları için Soner Körük, Süleyman Yaman ve Resul Furkan Demirkaya'ya minnettarım. Ayrıca iş yerimdeki meslektaşlarıma gösterdikleri neşe ve destek için minnettarım.

Tez çalışmamda gösterdiği anlayış ve destek için sayın Prof. Dr. Ercan Öztemel'e teşekkürlerimi sunarım.

Sevgili eşim Hale'ye, kalıcı sabrı, desteği ve en önemlisi hayatı benimle paylaştığı için en derin duygulara sahibim.

Bu uzun süreli dönemin zevklerini ve zorluklarını anladığım ve bu çalışmaların bir sonuca ulaşacağına olan inancımı kaybetmediğim için kendimi takdir ediyorum.

Son olarak, en iyi dileklerimi çocuklarım Enes Taha ve Ali Emir'e gönderiyorum ve bu tezi onlara ithaf ediyorum.

Temmuz – 2020 Özgür GÜN

İÇİNDEKİLER

ÖNSÖZ VE TEŞEKKÜR i
İÇİNDEKİLER ii
ŞEKİLLER DİZİNİ v
TABLOLAR DİZİNİ viii
SİMGELER VE KISALTMALAR DİZİNİ xi
ÖZET xiii
ABSTRACT xiv
GİRİŞ 1
1. LİTERATÜR ARAŞTIRMASI 4
2. MODEL OLUŞTURMA ÇALIŞMASI 14
2.1. Araştırma İş Akışı 14
2.2. Analitik Hiyerarşi Süreci 15
2.2.1. Hiyerarşinin oluşturulması 16
2.2.2. İkili karşılaştırma matrisinin oluşturulması 17
2.2.3. Öz değer ve öz vektörlerin hesaplanması 18
2.2.4. Tüm karşılaştırma matrislerinin tutarlılığının test edilmesi 20
2.2.5. Alternatiflerin nihai önceliklerinin oluşturulması 21
2.2.6. AHS'de grup kararı 22
2.2.7. Delphi yöntemi 22
2.2.8. Nominal grup yöntemi 22
2.2.9. Duyarlılık analizi 23
2.2.10. AHS'nin avantajları ve dezavantajları 23
2.3. İstatistiksel Süreç Kontrolü Teknikleri 24
2.3.1. Genel neden ve özel nedenler 26
2.3.2. X ve R kontrol grafikleri 32
2.3.3. X ve S kontrol grafikleri 34
2.3.4. Bireysel gözlem değerleri için kontrol kartları 36
2.3.5. Kontrol grafiklerindeki desenlerin analizleri 37
2.3.6. X ve R grafiklerinin yorumlanması 40
2.3.7. Neden-sonuç diyagramı 44
2.4. Veri Toplama Yöntemi 46
2.5. Verilerin Toplandığı Araçlar 48
2.6. Yazılımda Ölçme Süreç ve Standartları 58
2.6.1. Bütünleşik yetenek olgunluk modeli 58
2.6.2. ISO 15504 yazılım süreç değerlendirme 67
2.6.3. IEEE 15939 ölçme süreci 69
2.7. Yazılım Geliştirme Yaşam Döngüsü 75
2.8. Yazılım Geliştirme Yaşam Döngüsü Süreç Metrikleri 80
3. YAZILIM YAŞAM DÖNGÜSÜ SÜREÇ METRİKLERİ İLE PROJE PERFORMANSI ÖLÇME VE PROJE BAŞARI PUANI HESAPLAMA MODELİ 109
3.1. Model Bileşenleri 109

3.2. Proje Performans Değerlendirme ve Proje Başarı Puanı Hesaplama Süreci İş Akışı 111
4. MODELİN UYGULAMASI 113
4.1. Süreç Metrik Kriterleri Hiyerarşisi 113
4.2. AHS Çalışması ile Süreç Metriklerinin Ağırlıklandırılması 115
4.3. Kontrol Grafiklerinin Oluşturulması 124
4.3.1 Müşteri tarafından raporlanan hata açık kalma süresi metriği kontrol grafiği 124
4.3.2. Müşteri tarafından raporlanan iyileştirme açık kalma süresi metriği kontrol grafiği 127
4.3.3. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma süresi metriği kontrol grafiği 130
4.3.4. Müşteri tarafından raporlanan hata açık kalma oranı metriği kontrol grafiği 132
4.3.5. Müşteri tarafından raporlanan iyileştirme açık kalma oranı metriği kontrol grafiği 133
4.3.6. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı metriği kontrol grafiği 135
4.3.7. Hata yoğunluğu metriği kontrol grafiği 136
4.3.8. Kilometre taşı sapması metriği kontrol grafiği 137
4.3.9. Efor sapması metriği kontrol grafiği 139
4.3.10. Proje başına ortalama gereksinim değişikliği sayısı metriği kontrol grafiği 140
4.3.11. Gözden geçirme hazırlık hızı metriği kontrol grafiği 141
4.3.12. Bulgu yoğunluğu metriği kontrol grafiği 142
4.3.13. Gözden geçirme hazırlık verimliliği metriği kontrol grafiği 143
4.3.14. Hata yakalama yüzdesi metriği kontrol grafiği 145
4.3.15. Hata tespit yüzdesi metriği kontrol grafiği 146
4.3.16. Süreç denetim planlamasından sapma metriği kontrol grafiği 147
4.3.17. Proje uygunsuzluklarının ortalama açık kalma süresi metriği kontrol grafiği 149
4.3.18. Konfigürasyon denetim planlamasından sapma metriği kontrol grafiği 151
4.3.19. Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı metriği kontrol grafiği 152
4.3.20. Konfigürasyon uygunsuzluklarının ortalama açık kalma süresi metriği kontrol grafiği 154
4.3.21. Konfigürasyon uygunsuzluklarının açık kalma oranı metriği kontrol grafiği 155
4.3.22. Risk gözden geçirme oranı metriği kontrol grafiği 157
4.3.23. Üst ve alt kontrol limitleri özet tablosu 158
4.4. Neden-Sonuç Analizleri 159
4.4.1. Proje yönetimi süreci neden-sonuç diyagramı 160
4.4.2. Gereksinim yönetimi süreci neden-sonuç diyagramı 161
4.4.3. Geliştirme süreci neden-sonuç diyagramı 163
4.4.4. Doğrulama ve geçerleme süreci neden-sonuç diyagramı 166
4.4.5. Kalite güvence süreci neden-sonuç diyagramı 167
4.4.6. Konfigürasyon yönetimi süreci neden-sonuç diyagramı 169
4.4.7. Risk yönetimi süreci neden-sonuç diyagramı 170

4.5. Örnek Bir Projenin Proje Başarı Puanının Hesaplanması 171
4.6. Hata Kapatma Süresi Tahmini 173
4.6.1. Basit doğrusal regresyon ve korelasyon 174
4.6.2. Çoklu doğrusal regresyon ve korelasyon 177
4.6.3. Grafik analizler, korelasyon ve regresyon analizleri 180
5. MODEL İÇİN GELİŞTİRİLEN YAZILIM 187
5.1. Yazılım Gereksinimleri Belirtimi 187
5.2. Yazılım Mimari Tasarım 190
5.3. Yazılım Detaylı Tasarım 191
5.4. Yazılımda Kullanılan Veri Tabanı Tasarımı 192
5.5. Yazılımda Ara Yüz Ekranlar 193
5.6. Yazılım Kodları 196
6. SONUÇLAR VE ÖNERİLER 197
KAYNAKLAR 200
EKLER 208
KİŞİSEL YAYIN VE ESERLER 220
ÖZGEÇMİŞ 221

ŞEKİLLER DİZİNİ

Şekil 2.1. Araştırma iş akışı 15
Şekil 2.2. AHS'de üç seviyeli tipik hiyerarşi yapısı 17
Şekil 2.3. Genel neden ve özel neden değişkenliği 27
Şekil 2.4. Tipik bir kontrol grafiği 28
Şekil 2.5. Hem süreç ortalamasını hem de süreç değişkenliğini kontrol etme ihtiyacı. (a) Nominal seviyelerde ortalama ve standart sapma. (b) Süreç ortalaması $\mu 1 > \mu 0$. (c) Süreç standart sapması $\sigma 1 > \sigma 2$ 32
Şekil 2.6. Örnek bir kontrol kartı 38
Şekil 2.7. Western Electric kuralları-son dört nokta kural 3 ihlali 39
Şekil 2.8. Kontrol grafiğinde döngüler 41
Şekil 2.9. Karışım deseni 42
Şekil 2.10. Süreç seviyesinde değişim 42
Şekil 2.11. Süreç seviyesinde trend 43
Şekil 2.12. Katmanlaşma deseni 43
Şekil 2.13. Neden–sonuç diyagramı- aşama 1 44
Şekil 2.14. Neden-sonuç diyagramı-aşama 2 45
Şekil 2.15. Neden-sonuç diyagramı-aşama 3 45
Şekil 2.16. İş aktivite formu 51
Şekil 2.17. Değişiklik istek formu 52
Şekil 2.18. Hata bildirim formu 53
Şekil 2.19. Risk formu 53
Şekil 2.20. Toplantı tutanağı formu 54
Şekil 2.21. Süreç denetim formu 55
Şekil 2.22. İş ürünü konfigürasyon denetim formu 56
Şekil 2.23. Gözden geçirme formu 57
Şekil 2.24. Düzeltici faaliyet formu 57
Şekil 2.25. Öğrenilen dersler formu 58
Şekil 2.26. IEEE 15939 ölçme süreç modeli 70
Şekil 2.27. Ölçüm bilgi modelindeki temel ilişkiler 71
Şekil 2.28. Yazılım yaşam döngüsü süreç etkileşim haritası 77
Şekil 3.1. Proje başarı puanı hesaplama modeli 109
Şekil 3.2. Proje performans değerlendirme ve proje başarı puanı hesaplama süreç iş akışı 112
Şekil 4.1. Süreç metrik kriterleri hiyerarşisi 114
Şekil 4.2. Süreç metrik kriter ağırlıkları 121
Şekil 4.3. MTR hataların açık kalma süresi – ham veri 126
Şekil 4.4. MTR hataların açık kalma süresi – birinci iterasyon 127
Şekil 4.5. MTR iyileştirme açık kalma süresi – ham veri 129
Şekil 4.6. MTR iyileştirme açık kalma süresi kontrol grafiği – birinci iterasyon 130
Şekil 4.7. MTR gereksinim değişikliği açık kalma süresi kontrol grafiği 131
Şekil 4.8. MTR hata açık kalma oranı kontrol grafiği 133

Şekil 4.9. MTR iyileştirme açık kalma oranı kontrol grafiği 134
Şekil 4.10. MTR gereksinim değişikliği açık kalma oranı kontrol grafiği 136
Şekil 4.11. Hata yoğunluğu kontrol grafiği 137
Şekil 4.12. Kilometre taşı sapması kontrol grafiği 138
Şekil 4.13. Efor sapması kontrol grafiği 139
Şekil 4.14. Proje başına ortalama gereksinim değişikliği sayısı kontrol grafiği 141
Şekil 4.15. Gözden geçirme hazırlık hızı kontrol grafiği 142
Şekil 4.16. Bulgu yoğunluğu kontrol grafiği 143
Şekil 4.17. Gözden geçirme hazırlık verimliliği kontrol grafiği 144
Şekil 4.18. Hata yakalama yüzdesi kontrol grafiği 146
Şekil 4.19. Hata tespit yüzdesi kontrol grafiği 147
Şekil 4.20. Süreç denetim planlamasından sapma kontrol grafiği 149
Şekil 4.21. Proje uygunsuzluklarının ortalama açık kalma süresi kontrol grafiği 150
Şekil 4.22. Konfigürasyon denetim planlamasından sapma kontrol grafiği 152
Şekil 4.23. Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı kontrol grafiği 153
Şekil 4.24. Konfigürasyon uygunsuzluklarının ortalama açık kalma süresi kontrol grafiği 155
Şekil 4.25. Konfigürasyon uygunsuzluklarının açık kalma oranı kontrol grafiği 156
Şekil 4.26. Risk gözden geçirme oranı kontrol grafiği 158
Şekil 4.27. Proje yönetimine ait metriklerin neden sonuç diyagramı 160
Şekil 4.28. Gereksinim yönetimine ait metriklerin neden sonuç diyagramı 162
Şekil 4.29. Geliştirme sürecine ait metriklerin neden sonuç diyagramı 164
Şekil 4.30. Geliştirme süreci hata yoğunluğu metriği neden sonuç diyagramı 165
Şekil 4.31. Doğrulama ve geçerleme süreci metrikleri neden sonuç diyagramı 167
Şekil 4.32. Kalite güvence süreci metrikleri neden sonuç diyagramı 168
Şekil 4.33. Konfigürasyon yönetimi süreci metrikleri neden sonuç diyagramı 169
Şekil 4.34. Risk yönetimi süreci metrikleri neden sonuç diyagramı 171
Şekil 4.35. Hata kapatma süresi regresyon analizi – hata terimleri grafiği 186
Şekil 4.36. Hata kapatma süresi regresyon analizi – hata terimleri olasılık grafiği 186
Şekil 5.1. Yazılım mimari tasarım şeması 190
Şekil 5.2. Yazılım sınıf diyagramı 192
Şekil 5.3. Yazılımda kullanılan veri tabanı tasarım gösterimi 193
Şekil 5.4. Yazılım ana sayfa ekranı 194
Şekil 5.5. Yazılım metrik tanımlama ekranı 194
Şekil 5.6. Yazılım metrik tanımlama detay ekranı 195
Şekil 5.7. Yazılım proje tanımlama ekranı 195
Şekil 5.8. Yazılım ölçüm sonuç ekranı 195
Şekil E.1. Hataların ortalama açık kalma süresi hesaplama program kodu 216
Şekil E.2. Gereksinim değişikliği açık kalma süresi hesaplama program kodu 217

Şekil E.3. Konfigürasyon uygunsuzlukları açık kalma süresi hesaplama program kodu 218
Şekil E.4. Konfigürasyon uygunsuzlukları açık kalma oranı hesaplama program kodu ……..... 219

TABLOLAR DİZİNİ

Tablo 1.1. Referanslarla en sık kullanılan yöntemler 10
Tablo 1.2. Referanslarla en sık kullanılan proje başarı kriterleri 11
Tablo 2.1. AHS için kullanılan 1-9 temel ölçeği 18
Tablo 2.2. Ortalama rasgele indeks 21
Tablo 2.3. Shewhart kontrol grafikleri için bazı hassaslaştırıcı kurallar 39
Tablo 2.4. Yazılım yaşam döngüsü süreç metrikleri 47
Tablo 2.5. Proje geliştirme destek araçları 49
Tablo 2.6. Proje süreci iş akışları ve ilgili araçlar 50
Tablo 2.7. CMMI Süreç alanları ve olgunluk seviyeleri 63
Tablo 2.8. CMMI Ölçme ve analiz süreç alanı - hedefler ve pratikler 64
Tablo 2.9. CMMI Nicel proje yönetimi süreç alanı - hedefler ve pratikler 65
Tablo 2.10. ISO / IEC TR 15504'teki yetenek seviyeleri ve süreç özellikleri 67
Tablo 2.11. Kilometre taşı sapması ölçüm bilgi formu 81
Tablo 2.12. Efor sapması ölçüm bilgi formu 82
Tablo 2.13. Gelir/gider endeksi ölçüm bilgi formu 83
Tablo 2.14. Proje başına ortalama gereksinim değişikliği sayısı ölçüm bilgi formu 84
Tablo 2.15. Müşteri tarafından raporlanan hataların açık kalma süresi ölçüm bilgi formu 85
Tablo 2.16. Müşteri tarafından raporlanan iyileştirmelerin açık kalma süresi ölçüm bilgi formu 86
Tablo 2.17. Müşteri tarafından raporlanan gereksinim değişikliklerinin açık kalma süresi ölçüm bilgi formu 88
Tablo 2.18. Müşteri tarafından raporlanan hataların açık kalma oranı ölçüm bilgi formu 89
Tablo 2.19. Müşteri tarafından raporlanan iyileştirmelerin açık kalma oranı ölçüm bilgi formu 90
Tablo 2.20. Müşteri tarafından raporlanan gereksinim değişikliklerinin açık kalma oranı ölçüm bilgi formu 91
Tablo 2.21. Hata yoğunluğu bilgi formu 92
Tablo 2.22. Gözden geçirme hazırlık hızı bilgi formu 93
Tablo 2.23. Bulgu yoğunluğu bilgi formu 95
Tablo 2.24. Gözden geçirme hazırlık verimliliği bilgi formu 96
Tablo 2.25. Hata yakalama yüzdesi ölçüm bilgi formu 97
Tablo 2.26. Hata tespit yüzdesi ölçüm bilgi formu 98
Tablo 2.27. Süreç denetim planından sapma ölçüm bilgi formu 100
Tablo 2.28. Proje uygunsuzlukları ortalama açık kalma süresi ölçüm bilgi formu 102
Tablo 2.29. Konfigürasyon denetim planından sapma ölçüm bilgi formu 103
Tablo 2.30. Konfigürasyon denetimi başına düşen ortalama uygunsuzluk ölçüm bilgi formu 104
Tablo 2.31. Konfigürasyon uygunsuzluklarının ortalama açık kalma süresi ölçüm bilgi formu 105

Tablo 2.32. Konfigürasyon uygunsuzluklarının ortalama açık kalma oranı ölçüm bilgi formu 106
Tablo 2.33. Risk gözden geçirme oranı ölçüm bilgi formu 107
Tablo 3.1. Başarı puanı hesaplama modeline girecek metrikler 110
Tablo 4.1. Ana kriter ağırlıkları 115
Tablo 4.2. Proje yönetimi süreci alt kriter ağırlıkları 116
Tablo 4.3. Geliştirme süreci alt kriter ağırlıkları 116
Tablo 4.4. Doğrulama ve geçerleme süreci alt kriter ağırlıkları 116
Tablo 4.5. Kalite güvence süreci alt kriter ağırlıkları 117
Tablo 4.6. Konfigürasyon yönetimi süreci alt kriterler 117
Tablo 4.7. Ana kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu 118
Tablo 4.8. Proje yönetimi süreci alt kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu 118
Tablo 4.9. Geliştirme süreci alt kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu 118
Tablo 4.10. Doğrulama ve geçerleme süreci alt kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu 119
Tablo 4.11. Kalite güvence süreci alt kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu 119
Tablo 4.12. Konfigürasyon yönetimi süreci alt kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu 119
Tablo 4.13. Ana kriterleri ağırlık değerleri 119
Tablo 4.14. Süreç metrik kriterleri sıralama tablosu 122
Tablo 4.15. Müşteri tarafından raporlanan hata açık kalma süresi metriği ölçüm verileri 124
Tablo 4.16. Müşteri tarafından raporlanan iyileştirme açık kalma süresi metriği ölçüm verileri 128
Tablo 4.17. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma süresi metriği ölçüm verileri 131
Tablo 4.18. Müşteri tarafından raporlanan hata açık kalma oranı metriği ölçüm verileri 132
Tablo 4.19. Müşteri tarafından raporlanan iyileştirme açık kalma oranı metriği ölçüm verileri 134
Tablo 4.20. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı metriği ölçüm verileri 135
Tablo 4.21. Hata yoğunluğu metriği ölçüm verileri 136
Tablo 4.22. Kilometre taşı sapması metriği ölçüm verileri 138
Tablo 4.23. Efor sapması metriği ölçüm verileri 139
Tablo 4.24. Proje başı ortalama gereksinim değişiklik sayısı metrik ölçüm verileri 140
Tablo 4.25. Gözden geçirme hazırlık hızı metriği ölçüm verileri 141
Tablo 4.26. Bulgu yoğunluğu metriği ölçüm verileri 143
Tablo 4.27. Gözden geçirme hazırlık verimliliği metriği ölçüm verileri 144
Tablo 4.28. Hata yakalama yüzdesi metriği ölçüm verileri 145
Tablo 4.29. Hata tespit yüzdesi metriği ölçüm verileri 146
Tablo 4.30. Süreç denetim planlamasından sapma metriği ölçüm verileri 148
Tablo 4.31. Proje uygunsuzluklarının ortalama açık kalma süresi metriği ölçüm verileri 149

Tablo 4.32. Konfigürasyon denetim planlamasından sapma metriği ölçüm verileri 151
Tablo 4.33. Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı metriği ölçüm verileri 152
Tablo 4.34. Konfigürasyon uygunsuzluklarının ortalama açık kalma süresi metriği ölçüm verileri..... 154
Tablo 4.35. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı metriği ölçüm verileri..... 155
Tablo 4.36. Risk gözden geçirme oranı metriği ölçüm verileri..... 157
Tablo 4.37. Süreç metrikleri üst ve alt kontrol limit değerleri 158
Tablo 4.38. Kabul aşamasındaki bir projenin ölçülen metrik değerleri 172
Tablo 4.39. Başarı puanı hesaplama sonuçları 173
Tablo 4.40. Regresyon modeli bağımlı ve bağımsız değişkenler..... 180
Tablo 4.41. Bağımsız değişkenler ve alabileceği değerler..... 181
Tablo 4.42. SPSS aracında oluşturulan geçici değişkenler 182
Tablo 4.43. Test hatalarına ait betimleyici istatistikler 182
Tablo 4.44. Regresyon analizi belirlilik katsayısı 183
Tablo 4.45. Regresyon analizi ANOVA tablosu..... 184
Tablo 4.46. Hata kapatma süresi regresyon analizi..... 184
Tablo A.1. Kontrol limitleri hesaplamaları için katsayılar tablosu 209
Tablo B.1. Kritik T değerleri tablosu..... 210
Tablo C.1. Kritik F değerleri tablosu ($\alpha = 0,05$) 211
Tablo D.1. Hata kapatma süresi tahminleme regresyon analizi çalışmasında kullanılan veriler..... 212

SİMGELER VE KISALTMALAR DİZİNİ

σ	:	Sigma, standart sapma
$\hat{\sigma}$	:	Sigmanın tarafsız tahmini
σ^2	:	Dağılımın değişkenliği
s	:	Örnek kütlenin standart sapması
s^2	:	Örnek kütlenin değişkenliği
μ	:	Dağılımın ortalaması
n	:	Örnek büyüklüğü
$\bar{x}$	:	Örnek ortalaması (x veri seti için)

Kısaltmalar

AHS	: Analitik Hiyerarşi Süreci
AİBB	: Arayüz İsterleri Belirtimi Belgesi
AKL	: Alt Kontrol Limiti
Ar-Ge	: Araştırma ve Geliştirme
ATTB	: Arayüz Tasarım Tanımlama Belgesi
BY	: Bulgu Yoğunluğu
CDR	: Critical Design Review (Kritik Tasarım Gözden Geçirme)
CI	: Consistency Index (Tutarlılık İndeksi)
CMMI	: Capability Maturity Model Integration (Bütünleşik Yetenek Olgunluk Modeli)
CR	: Consistency Ratio (Tutarlılık Oranı)
ÇKKV	: Çok Kriterli Karar Verme
DG	: Doğrulama ve Geçerleme
DGP	: Doğrulama ve Geçerleme Planı
DİBB	: Donanım İsterleri Belirtimi Belgesi
DİF	: Değişiklik İstek Formu
DTSR	: Donanım Test Sonuç Raporu
DTTB	: Donanım Tasarım Tanımlama Belgesi
DTeTB	: Donanım Test Tanımlama Belgesi
ES	: Efor Sapması
EVM	: Earned Value Management (Kazanılmış Değer Yönetimi)
HAKO	: Hata Açık Kalma Oranı
HAKS	: Hata Açık Kalma Süresi
HTY	: Hata Tespit Yüzdesi
HY	: Hata Yoğunluğu
HYY	: Hata Yakalama Yüzdesi
GAKO	: Gereksinim Değişikliklerinin Açık Kalma Oranı
GAKS	: Gereksinim Değişikliklerinin Açık Kalma Süresi
GEL	: Geliştirme
GGH	: Gözden Geçirme Hazırlık Hızı
GGHV	: Gözden Geçirme Hazırlık Verimliliği
GGS	: Gelir Gider Sapması

GY : Gereksinim Yönetimi
İAKO : İyileştirme Açık Kalma Oranı
İAKS : İyileştirme Açık Kalma Süresi
İGG : İster Gözden Geçirme
İKA : İş Kırınım Ağacı
IEEE : The Institute of Electrical and Electronics Engineers (Elektrik ve Elektronik Mühendisleri Enstitüsü)
KDOUS : Konfigürasyon Denetimi Başına Düşen Ortalama Uygunsuzluk Sayısı
KDPS : Konfigürasyon Denetim Plan Sapması
KG : Kalite Güvence
KGP : Kalite Güvence Planı
KS : Kilometretaşı Sapması
KTeTB : Kabul Test Tanımlama Belgesi
KTSR : Kabul Test Sonuç Raporu
KUAKO : Konfigürasyon Uygunsuzluk Ortalama Açık Kalma Oranı
KUAKS : Konfigürasyon Uygunsuzluk Ortalama Açık Kalma Süresi
KY : Konfigürasyon Yönetimi
KYP : Konfigürasyon Yönetim Planı
MACBETH : Measuring Attractiveness by a Categorical Based Evaluation Technique (Kategorik tabanlı bir değerlendirme tekniği)
MİBB : Müşteri İsterleri Belirtimi Belgesi
MR : Moving Range (Hareketli Aralık)
OÇ : Orta Çizgi
OGDS : Ortalama Gereksinim Değişikliği Sayısı
PDR : Preliminary Design Review (Ön Tasarım Gözden Geçirme)
PİR : Proje İlerleme Raporu
PMO : Project Management Office (Proje Yönetim Ofisi)
PUAKS : Proje Uygunsuzluk Açık Kalma Süresi
PY : Proje Yönetimi
PYP : Proje Yönetim Planı
RGGO : Risk Gözden Geçirme Oranı
RY : Risk Yönetimi
RYP : Risk Yönetim Planı
SDPS : Süreç Denetim Plan Sapması
SİBB : Sistem İsterleri Belirtimi Belgesi
SMYP : Sistem Mühendisliği Yönetim Planı
SP : Specific Practice (Spesifik Hedef)
SPC : Statistical Process Control (İstatistiksel Süreç Kontrolü)
STTB : Sistem Tasarım Tanımlama Belgesi
STeTB : Sistem Test Tanımlama Belgesi
TRR : Test Readiness Review (Test Hazırlık Gözden Geçirmesi)
TAP : Test Ana Planı
ÜKL : Üst Kontrol Limiti
YGP : Yazılım Geliştirme Planı
YİBB : Yazılım İsterleri Belirtimi Belgesi
YMTTB : Yazılım Mimari Tasarım Tanımlama Belgesi
YTSR : Yazılım Test Sonuç Raporu
YTTB : Yazılım Tasarım Tanımlama Belgesi
YTeTB : Yazılım Test Tanımlama Belgesi

PROJE YAŞAM DÖNGÜSÜ PARAMETRELERİ İLE PROJE PERFORMANSLARININ ÖLÇÜLMESİNE YÖNELİK YAZILIM SİSTEMİNİN GELİŞTİRİLMESİ

ÖZET

Yazılım dünyasındaki süreç ve araç altyapısındaki gelişmelere rağmen, proje başarısı önemli ölçüde iyileşmemiştir. Yazılım projelerinde, proje başarısının tanımı, müşterinin planlanan efor, zaman ve bütçede istediği ürünleri üretmek anlamına gelir. Bu amaca ulaşmak için planlama, analiz, tasarım, kodlama, entegrasyon, test ve teslim süreçleri bir yazılım projesinin başından sonuna kadar etkileşimli olarak yürütülmektedir. Bu süreçlerin metrikleri yazılım projelerinin performansını ölçmek için kullanılır. Literatür taraması, proje performansının ölçülmesinde bütçe, efor, takvim, müşteri memnuniyeti, ürün kalitesi gibi proje yönetimi süreç ölçütlerinin kullanıldığını gösterdiğinden, proje performanslarının ölçülmesinde daha kapsamlı ve etkili kriterlerin tanımlanması ve uygulanması gerekmektedir. Proje performans değerlendirmesinin önemi nedeniyle, bu çalışmada genel bir değerlendirme modeli oluşturulmuştur. Önerilen model yazılım endüstrisinde kullanılmak üzere tasarlanmıştır. Proje performansı açısından proje yönetimi, gereksinim, risk, kalite ve konfigürasyon, geliştirme, doğrulama ve geçerleme süreçlerine odaklanan bir model geliştirilmiştir. Bu tez çalışmasının amacı, yazılım projelerinin performansını değerlendiren ve proje başarısını sayısal bir değerle ifade eden bir model sunmaktır. Analitik hiyerarşi süreci, performans değerlendirmesine girdi sağlayan her süreç metrik kriteri ve alt kriterlerinin göreceli önemini hesaplamak için kullanılmıştır. Proje performansının değerlendirilmesinde ve proje başarı puanının hesaplanmasında istatistiksel süreç kontrol yöntemi kullanılmıştır. Bu tez çalışmasında proje geliştirme destek araçlarından proje süreç metrik ölçümlerini istenilen tarihlerde otomatik olarak toplayan ve proje başarı puanını hesaplayan bir yazılım uygulaması geliştirilmiştir. Önerilen modeli doğrulamak için bir Ar-Ge kuruluşunda model işletilmiş ve müşteriye teslim aşamasındaki bir projenin gösterdiği performans geliştirilen yazılım uygulaması ile ölçülmüştür. Bu çalışmada sunulan modelin, proje durumunu izleyen yöneticilerin proje performansını değerlendirmesine ve projeler arasındaki performansın sayısal karşılaştırmasını yapmasına yardımcı olacağı düşünülmektedir.

Anahtar Kelimeler: Analitik Hiyerarşi Süreci, İstatistiksel Süreç Kontrolü, Proje Başarı Puanı, Proje Performans Değerlendirme.

DEVELOPING A SOFTWARE FOR MEASURING PROJECT PERFORMANCES WITH PROJECT LIFE CYCLE PARAMETERS

ABSTRACT

Despite the developments in the process and tool infrastructure in the software world, project success has not significantly improved. In software projects, the definition of project success means to produce products that the customer desires in the planned effort, time and budget. To achieve this goal, planning, analysis, design, coding, integration, testing and delivery processes are operated interactively from the beginning to the end of a software project. Metrics of these processes are used to measure the performance of software projects. Since the literature review shows that project management process metrics such as budget, effort, schedule, customer satisfaction, product quality are used in measuring project performance, more comprehensive and effective criteria are needed to be defined and applied in measuring project performances. Due to the importance of the project performance evaluation, a general evaluation model has been created in this study. The proposed model is designed for use in the software industry. In terms of project performance, a model has been developed that focuses on management of project, requirement, risk, quality and configuration, development, verification and validation processes. The purpose of this thesis is to present a model that evaluates the performance of software projects and expresses project success with a numerical value. Analytical hierarchy process has been used to calculate the relative importance of each process metric criterion and sub-criteria that provide input to the performance evaluation. Statistical process control method has been used in the evaluation of project performance and calculation of the project success score. In this thesis, a software application that automatically collects project process metric measurements from the project development support tools and calculates the project success score has been developed. In order to verify the proposed model, the model has been operated in an R&D organization and the performance of a project in acceptance test phase has been measured with the software application developed. It is thought that the model presented in this study will help the managers, who monitor the project status, to evaluate project performance, as well as provide the numerical comparison of performance between projects.

Keywords: Analytical Hierarchy Process, Statistical Process Control, Project Success Score, Project Performance Evaluation.

GİRİŞ

Ülkemizde enerji, sağlık, havacılık, uzay, otomotiv, raylı sistemler, bilişim, savunma, bilgi güvenliği ve kamu dijital dönüşüm sektörleri gibi öncelikli sektörlerde teknolojik ürün yatırımlarının ve kümelenme çalışmalarının desteklenmesi yönünde politikalar oluşturulmuştur [1-3]. Bu alanlarda başlatılan yazılım proje sayısında büyük bir artış vardır. Ülkemiz için önem arz eden bu projelerin başarıyla sonuçlandırılması da büyük önem arz etmektedir.

Modern proje yönetiminin başlangıcı 1950'li yıllara dayanmaktadır. Son 50 yılda süreçler, metodolojiler, standartlar, iyi uygulamalar ve birçok araç ve teknik proje yönetimi alanına katkıda bulunmuştur [4]. Yazılım dünyasındaki süreç ve araç altyapısındaki gelişmelere rağmen başarısız olan proje sayısı yüksektir [5]. Proje başarısızlıklarının en büyük nedenlerinden biri, yazılım yaşam döngüsü süreçlerinin tam ve doğru bir şekilde işletilmemesidir. Yazılım yaşam döngüsü süreçleri, yazılım projelerinin tedarik, planlama, gereksinim (ihtiyaç) analizi, tasarım, kodlama, entegrasyon, gözden geçirme, test etme, teslim etme ve elden çıkarma aşamalarına yönelik faaliyetleri ve talimatları tanımlayan etkileşimli bir yönetim sistemidir [6]. Bu süreçler analistlerin, yazılım geliştiricilerin, entegratörlerin, operatörlerin, bakım görevlilerinin, yöneticilerin, kalite güvence yöneticilerinin ve yazılımı kullanacak olan son kullanıcıların işleteceği süreç adımlarını ve sorumluluklarını tanımlar. Yazılım yaşam döngüsü önce planlama aşamasıyla başlar. Bu aşamada, bir yazılım projesinin işletmesi gereken tüm süreçler planlanır. İkinci aşama olan analiz aşamasında, geliştirilecek yazılımdan beklenen fonksiyonlar ve gereksinimler tanımlanır. Tasarım aşamasında, analiz aşamasında belirlenen ihtiyaçlara çözüm olacak yazılımın ilk ve temel versiyonu oluşturulur. Bu aşamada kodlama aktivitesi yapılmaz. Uygulama aşaması kodlama, birim testi, kod gözden geçirme ve entegrasyon olarak detaylandırılabilir. Proje dokümanları ve yazılım bileşenleri hakkındaki gözden geçirmeler doğrulama aşamasında yapılır. Yazılım gereksinimleri geçerleme aşamasında test edilir. Test sırasında yazılımdaki hatalar düzeltilir. Proje sözleşmesinde tanımlanan yazılım müşteriye teslim edilir ve bakım aşaması başlatılır.

Bakım aşamasında, müşteri ortamında kurulu yazılımda sorun giderme ve güncelleme gibi işlemler yapılır [6].

Bu tezde kullanılan metrikler, yazılım yaşam döngüsü süreçlerinin performansını ölçmek için kullanılmıştır. Proje yönetimi sürecinde proje süresi, bütçe ve efor metrikleri kullanılmıştır. Gereksinim yönetimi sürecinde, gereksinimlerin değişkenliğine ilişkin metrik kullanılmıştır. Geliştirme sürecinde hatalar, iyileştirmeler, gereksinim değişikliklerine ait metriklerin ortalama açık kalma süreleri ve oranları kullanılmıştır. Doğrulama ve doğrulama süreci, gözden geçirme ve test olmak üzere iki alt sürece ayrılmıştır. Gözden geçirme sürecinin etkinliği, verimliliği ve bulgu yoğunluğu; test sürecinde ise hataların yakalama ve tespit yüzdesi metrik olarak kullanılmıştır. Kalite ve konfigürasyon süreçlerinin amaçları farklı olsa da, metrikleri benzerdir. Her iki süreçte de denetim planından sapma ve denetimlerde bulunan uygunsuzlukların ortalama açık kalma süresi ve oranları metrik olarak kullanılmıştır. Risk yönetimi sürecinde risk gözden geçirme oranı metrik olarak belirlenmiştir.

Bu çalışmanın amacı, yazılım projelerinin performans değerlendirmesi için yeni bir model sunmaktır. Performans değerlendirmesi sonucunda projenin başarısı sayısal bir değerle gösterilmektedir. Proje başarısını etkileyen süreç metriklerini (kriterlerini) belirlemek için uzmanlarla yapılan yapılandırılmış görüşmeler kullanılmıştır. Belirlenen kriterlerin proje başarısına katkısını bulmak ve kriterlerin göreceli önemini bulmak için Analitik Hiyerarşi Süreci (AHS) kullanılmıştır. Projelerden toplanan metrik ölçüm sonuçlarını analiz etmek için İstatistiksel Süreç Kontrolü yöntemleri kullanılmıştır.

Yeterli karar verme kapasitesine sahip olmak için, projenin başarısını etkileyen kriterlerin göreceli önemini belirlemek gerekir [7]. Bu çalışmada, bir Ar-Ge kurumunda çalışan uzman proje yöneticileri ve araştırmacılardan değerlendirmeler toplanmıştır. Ağırlıkların, her bir kriterin göreceli önemini yansıtması bir kuraldır. Bu nedenle uzmanların yargılarına uygun olarak kriterlerin ve alt kriterlerin göreceli önemini hesaplamak için AHS yöntemi seçilmiştir. Bu tez çalışmasında alternatifler arasında seçim yapma işlemi tez kapsamında bulunmamaktadır. AHS yöntemi sadece kriterleri ağırlıklandırmak için kullanılmıştır.

Bu tez çalışması kapsamında geliştirilen yazılım, projelerin kullandığı araçlardan süreç metriklerini otomatik olarak toplamaktadır. Böylece proje süreç performanslarını değerlendirmek için her bir projenin süreç metriklerini ayrı ayrı toplamak gibi uzun süren işlemler çok kısa bir sürede gerçekleştirilebilmektedir. Süreç metriklerinin araçlardan otomatik olarak toplanması ölçüm sonuç değerlerinin kişiye bağımlı olması ve hata yapılma olasılığını ortadan kaldırmaktadır. Geliştirilen bu yazılımın proje performans değerlendirme alanında literatüre önemli katkı yapacağı değerlendirilmektedir.

Bu tez çalışmasında, proje süreç performanslarının belirlenmesinde istatistiksel süreç kontrol yöntemleri kullanılmıştır. Proje süreçlerine ait alt ve üst kontrol limitleri belirlenmiştir. Böylece daha sonra başlayan projelerin süreç performansları, belirlenen alt ve üst kontrol limitleri ile karşılaştırılarak proje süreçlerinin istatistiksel olarak kontrol altında olup olmadığına karar verilmiştir.

Bu çalışmada proje performans değerlendirmesinde kullanılacak proje başarı faktörlerinin bir listesi oluşturulmuştur. Bu liste, proje yöneticilerinin proje başarısı için önemli olan temel süreç başarı kriterlerini anlamalarına yardımcı olacaktır. Bu tez çalışmasında ortaya konan yazılım projelerinin performansını değerlendiren ve proje başarı puanını hesaplayan modelin proje yöneticilerine ve projeleri değerlendiren üst yöneticilere yardımcı olacağı değerlendirilmektedir.

Bu tez aşağıdaki gibi düzenlenmiştir. Bir sonraki bölüm, proje performans değerlendirmesi ve proje başarı faktörü hakkında literatür taraması sunmaktadır. Bölüm 2 modellemede kullanılan yöntemleri açıklar. 3. Bölüm proje performans değerlendirme ve proje başarı puanı modelini sunar. Bölüm 4 modelin uygulaması kapsamında proje metrik kriterleri hiyerarşisi ile kriterlerin önem derecelerinin belirlenmesi, kontrol grafikleri ile süreç performans dayanaklarının belirlenmesi, neden-sonuç analizleri, hata kapatma süresi tahmini için çoklu regresyon analizi konularında bilgi verir. Bölüm 5 model için geliştirilen yazılımın analizi, tasarımı, kodları ve ekran görüntüleri hakkında bilgi verir. Son olarak, sonuçlar ve öneriler Bölüm 6'da sunulmaktadır.

1. LİTERATÜR ARAŞTIRMASI

Bu bölümde, proje başarı faktörleri, proje performans değerlendirme modelleri, performans değerlendirme modellerinde kullanılan yöntemler konusunda daha önce yapılan çalışmalar üzerine literatür tarama sonuçları sunulmaktadır.

Literatürde, süreç metrikleri kullanılarak proje başarı faktörlerinin belirlenmesi ve proje performansının değerlendirilmesi ile ilgili çalışmalar mevcuttur. Bu bölümde bu çalışmalardan bahsedilecektir. Tablo 1.1 referanslarla en sık kullanılan yöntemleri özetlemektedir. Öte yandan, Tablo 1.2 literatürde kullanılan proje başarı kriterleri setini vermektedir.

Farklı ölçek ve özelliklere sahip yazılım kuruluşlarında proje başarısını etkileyen parametreleri belirlemek için yapılan bir çalışmada; projedeki çalışan sayısı, proje yöneticisinin tecrübesi ve mezun olunan üniversitesi, gereksinimlere ve test sürecine tahsis edilen kaynak oranı, işten ayrılan çalışanların oranı, gereksinimdeki kelime sayısı ve yazılım kod satır sayısı projenin başarısına etki eden faktörler olarak belirlenmiştir [8].

Yapılan bir başka çalışmada yazılım karmaşıklığı, işlev puanı, tekrar kullanılabilirlik, proje bütçesi, kullanılan teknoloji, çalışan yeterliliği, çalışma ortamı özellikleri, verimlilik durumu ve ürün özellikleri proje başarısını etkileyen metrikler olarak tanımlamıştır [9].

Proje başarı faktörleri için İş Kırınım Ağacı (İKA) şeklinde bir yapı geliştirildiği bir çalışmada proje başarı faktörleri üç gruba ayrılmıştır: proje yönetimi yeterliliği, organizasyon yapısı ve metodolojisi. İKA'da proje yönetimi yeterliliği; proje yöneticisi yeterliliği, proje ekibi yeterliliği ve koordinasyon faktörlerinden oluşmaktadır. Organizasyon faktörü; organizasyonun yapısı, kültürü, atmosferi ve yeterliliği alt faktörlerinden oluşur. Metodoloji faktörü; kullanılan yöntemler, proje yönetim yazılımı, proje yönetim araçları, karar verme teknikleri, risk değerlendirme araçları ve bilgi iletişim destek araçları olarak alt faktörlere ayrılmıştır [4].

Korelasyon ve regresyon analizi yöntemleri kullanılarak çok düzeyli bir perspektiften proje başarısının kilit yönleri tartışılmıştır. Proje büyüklüğü, süre, erteleme, proje ekibi büyüklüğü, ekibin katılımı ve proje yöneticisinin resmi gücü başarı faktörleri olarak belirlenmiştir [10].

Korelasyon analizi ve çoklu regresyon analizi yöntemlerinin kullanıldığı proje yönetimi performansı ile proje başarısı arasındaki ilişkiyi test eden bir araştırmada; proje yönetimi performansının, proje yönetimi anahtar performans göstergelerinin ve proje yönetimi personel yetkinliğinin proje başarısına en çok etki eden faktörler olduğunu belirtilmiştir. Proje başarısının proje verimliliği, ekip ve müşteri etkisi, iş başarısı veya geleceğe hazırlanma gibi farklı perspektiflerden ölçülebileceği belirtilmiştir [11].

Korelasyon ve regresyon analizini kullanılarak kritik başarı faktörlerine ve performans göstergelerine dayanan bir proje başarı analizi çerçevesi sunma çalışması yapılmıştır. Sunulan proje başarı analizi çerçevesi; kritik başarı faktörlerinin tanımı, anahtar performans göstergelerinin tanımı, tanımlanmış anahtar performans göstergelerine göre proje başarısının ölçülmesi ve proje başarı ölçümünün belgelenmesi ve nihai proje raporunun oluşturulması elemanlarını içermektedir. Regresyon analizi sonuçları, sunulan proje çerçevesinin tüm önemli unsurlarının, proje başarı analizi söz konusu olduğunda, proje ortamındaki bilgi edinme ve bilgi transferine pozitif etki yaptığını göstermiştir [12].

Proje başarısını ve başarısızlığını etkileyen faktörlerin belirlenmesi için yapılan bir başka çalışmada; müşterinin proje yöneticisine ve proje ekibine olan güveni, müşterinin gerçekçi beklentileri, gereksinimlerin iyi ve yeterli derecede tanımlanmış olması, proje zaman ve bütçe kestiriminin iyi yapılmış olması, takvim gecikmesi yaşanmaması için iş gücü kaynağı eklenmesi, proje yöneticisinin yeteneği ve proje ekibi ile iletişim yeteneği, proje aşamalarına/süreçlerine yeterli zamanın ayrılması, proje ekibinin motivasyonu ve ekip olarak çalışma seviyesi ile proje çalışma altyapısı yeterliliği faktörlerinin proje sonucuna anlamlı etki ettiği ileri sürülmüştür. Gerçekleştirilen çalışmada korelasyon analizi ve lojistik regresyon analizi yöntemleri kullanılmıştır [13].

Yazılım proje başarısı ile kritik başarı faktörleri arasındaki ilişkinin korelasyon analizi yöntemi ile araştırıldığı çalışmada; Türkiye yazılım endüstrisindeki 101 projeden anket yöntemi ile toplanan veriler analiz edilmiştir. Analiz sonuçlarına göre proje başarısını en çok etkileyen faktörler; proje ekibinin proje geliştirme metodolojilerindeki deneyimi, proje ekibinin alan uzmanlığı ve proje izleme ve kontrol olarak sıralanmıştır [14].

İnşaat sektöründeki projeler için kurumsal karne (balanced scorecard) ve AHS yöntemleri birlikte kullanılarak çok boyutlu bir performans değerlendirme modeli önerilmiştir. Altı perspektiften oluşan yeni bir genişletilmiş kurumsal karne modeli geliştirmek için kapsamlı literatür taraması ve ön anket yaklaşımı kullanılmıştır. Bu altı perspektif sırasıyla; müşteri, finansal, dâhili iş süreçleri, proje ekibi, sağlık/emniyet/sosyo-çevre ile yenilik/öğrenme/büyüme olarak belirlenmiştir. Çalışmada, kurumsal karnedeki proje performans göstergeleri için yapılandırılmış anket yöntemi ile veri toplanmış ve AHS yöntemi ile göstergelerin önem derecesi belirlenmiştir. AHS sonuçları, müşteri ve finansal perspektiflerin modeldeki diğer perspektiflerden nispeten iki kat daha önemli seviyeye sahip olduğunu, yenilik, öğrenme ve büyüme perspektifinden üç kat daha önemli olduğunu ortaya koymuştur [15].

Proje yönetimi performansını değerlendiren ve sayısal bir performans değeri hesaplayan bir modelin geliştirildiği çalışmada; proje yönetimi performansı hesaplama işleminde başarı kriterleri için maliyet performans endeksi, faturalandırma performans endeksi, takvim performans endeksi, emniyet performans endeksi, kalite performans endeksi ve çevre performans endeksi parametreleri kullanılmıştır. Başarı faktörlerinin göreceli önemini belirlemek için AHS yöntemi kullanılmıştır [16].

Proje başarısını değerlendirmek için iş kırınım ağacı yöntemi kullanılarak proje başarısına etki eden faktörler gruplara ayrılmıştır. Başarı faktörleri müşteri, proje ekibi ve toplum açısından detaylandırılmıştır. Müşteri açısından proje kapsamı, kalite, takvim, maliyet, proje ekibi morali ve müşteri memnuniyeti proje başarı faktörleri olarak belirlenirken diğer yandan proje ekibi açısından bu faktörlere ek olarak emniyet, entegrasyon, raporlama, risk yönetimi, sözleşme yönetimi, iletişim ve alt yüklenici

faktörleri proje başarı faktörü olarak belirlenmiştir. Bu faktörler daha sonra ilgili paydaşlar için toplam puana ulaşacak şekilde ağırlıklandırılmıştır [17].

İstatistiksel süreç kontrolü tekniği kullanılarak geleneksel Kazanılmış Değer Yönetimi'nin (Earned Value Management) performansının geliştirildiği bir çalışma yapılmıştır. Yapılan çalışmada bireysel kontrol grafikleri, proje performans verilerini izlemek için araç olarak kullanılmıştır. Böylece olumsuz değişikliklerin zamanında tespit edilebildiği ileri sürülmüştür. Yeni yaklaşımın etkinliğini belirlemek için geleneksel kazanılmış değer yönetimi ile yeni yaklaşım arasında karşılaştırmalı bir analiz yapılmıştır. Sonuç olarak istatistiksel süreç kontrolü tabanlı kazanılmış değer yönetimi modelinin birden fazla yararı olduğunu ileri sürmüşlerdir: 1. Proje yöneticilerinin proje sürecinde ilgili performans değişikliklerinden haberdar olabileceği bir ortam sağlanması, 2. Performans değişikliklerinin nedenleri sistematik olarak analiz edilebilmesi ve performansın kontrol altına alınması veya olumsuz eğilimler için düzeltici önlemler alınabilmesi, 3. Karakteristik tabanlı bir örüntü performans korelasyon modeli ile proje yöneticilerine veya planlamacılarına, ilk proje aşamasında proje performans örüntülerinin olasılığını değerlendirmek için somut bir araç sağlanması [18].

Bir proje sürecinin istatistiksel kontrol altında olup olmadığını belirlemek için istatistiksel proje kontrol aracı geliştirmiş ve uygulaması açıklamıştır. Geliştirien araç bir dizi değiştirilmiş Shewart istatistiksel süreç kontrol grafiğinden oluşmaktadır. Geliştirilen araç, projenin takvim, maliyet ve teknik performansla ilgili proje parametrelerini izleyen grafikler içermektedir. Sunulan bu araç ile proje yöneticilerinin gelişmekte olan durumlara karşı uyarılabileceği ve projede hata oluşmadan düzeltici önlemler alınabileceği ileri sürülmüştür [19].

Kazanılmış değer yönetiminin proje performans değerlendirme tekniği olarak sınırlı benimsenmesinin nedenleri araştırılmıştır. Proje yönetimi uygulayıcılarına faydalı olacak bu tekniğe yeni uzantılar önerilmiştir. Yapılan çok yönlü araştırma aşağıdaki unsurları içermektedir: kazanılmış değer yönetimi hakkında önceki ve güncel literatürün gözden geçirilmesi, proje yönetimi uygulayıcılarının kazanılmış değer yönetimine yönelik uygulamaları ve tutumları üzerine bir anket, kazanılmış değer yönetimi tekniğinin bilinen zorluklarının analizi, kazanılmış değer yönetimi

zorluklarını ele almak ve çözmek için tekniklerin geliştirilmesi, bu tekniklerin tek bir çerçeve ve uygulama modelinde birleştirilmesi ve bu çerçevenin ve modelin birden çok yöntemle doğrulanması [20].

Büyük ölçekli projeler için önem arz eden yazılım projesi faktörlerinin belirlenmesine yardımcı olan nicel bir araştırmada; akademik ve endüstriyel çaptaki profesyonelleri içeren 176 katılımcı tarafından sağlanan veriler üzerinde korelasyon ve güvenilirlik analizi gerçekleştirilmiştir. Sonuç olarak proje başarı faktörleri listesi ve anketin sonucunu desteklemek için istatistiksel kanıtlar sunulmuştur. Maliyet, zaman, proje ekip büyüklüğü, kaynaklar, gereksinim değişikliği, yeniden kullanılabilir kod, kalite ve risk yönetimi başarı faktörleri olarak tanımlanmıştır [21].

Çok kriterli karar verme yöntemleri kullanılarak proje yönetim ofisleri (PYO) için bir performans değerlendirme modeli önerilmiştir. 17 kritere bölünmüş 79 performans göstergesinin bir listesine dayanarak, MACBETH tekniği kullanılarak PYO'ların değerlendirilmesi için bir süreç tanımlanmıştır. Sonuçların, PYO performansının güçlü ve zayıf yanlarının yanı sıra, tüm paydaşların ortak bir vizyonuna göre her birini iyileştirme önceliğinin net bir resmini gösterdiği ileri sürülmüştür [22].

Proje başarısının değerlendirilmesi için yeni bir uzman karar verme bulanık modeli önerilmiştir. Kazanılmış değer yönetimi metrikleri (zaman performans endeksi, bütçe performans endeksi), proje risklerinin (alt risklerin toplam sayısı ve alt risklerin toplam risk değeri) toplam değeri ve proje kalitesinin (proje gereksinimlerine uyumluluk derecesi ve proje ürünlerinin amaçlandığı şekilde kullanılabilirliği) toplam değeri önerilen bulanık modelin girdileri olarak belirlenmiştir. Sonuç olarak modele giren parametrelerin aldığı değerler ile hesaplama yapılarak bir proje başarı puanı üretilmektedir. Proje başarısı çok büyük, büyük, orta, düşük ve çok düşük şeklinde kategorik olarak ifade edilmiştir [23].

Maliyet, zaman ve geliştirici sayısı gibi kaynakların, proje yöneticisi tarafından geliştirme süreci boyunca tahsis edildiği şekilde projenin başarılı bir şekilde tamamlanmasına yönelik bir etki analizi gerçekleştirilmiştir. Çalışmada ayrıca, endeks değeri 0'dan 1'e değişen yeni bir niteliksel metrik olan Başarı Düzeyi Endeksi Metriği (Success Level Index - SLI) tanımlanmıştır. SLI, şirketin başarı açısından olgunluk derecesini gösteren bir olgunluk faktörü olarak ifade edilmiştir. SLI metriği,

gerçekleşen kaynak değerleri ile planlanan kaynak değerlerinin oranını ifade etmektedir. Bu çalışma sonucunda şirketlerin kaynakları tahsis ederken daha fazla dikkat göstermeleri, planlanan ve gerçekleşen kaynak farklarını en aza indirmelerinin çok önem arz ettiği belirtilmiştir [24].

Proje yatırım başarısını sağlamak için proje yönetimi ve faydalar yönetimi (benefits management) uygulamalarının gerekli olduğu ileri sürülen bir çalışma yapılmıştır. Başka bir deyişle, proje yönetimi ve fayda yönetimini projeleri yönetmek için birleştiren kuruluşların, yalnızca proje yönetimi ve fayda yönetimi uygulayan diğer kuruluşlardan önemli ölçüde daha yüksek bir başarı seviyesine ulaşabildiği ifade edilmiştir. Başarılı proje çıktıları (geçerli ve güvenilir) sağlamak için planların sürekli gözden geçirilmesi, paydaş beklentilerinin ve gereksinimlerin yakından yönetilmesi gerektiği ifade edilmiştir [25].

Faydalar yönetiminin projeler için hayati bir itici güç haline geldiğini ve proje başarısını geleneksel zaman, maliyet ve kapsam ölçütleri aracılığıyla değerlendirmek yerine, faydalarına ulaşarak değerlendirmenin artık daha yaygın olduğunu ifade eden bir çalışma yapılmıştır. Program düzeyinde ve diğer uygulamalar ile sistemler arasında proje düzeyinde fayda yönetimi uygulamalarında kayda değer bir eksiklik olduğu belirtilmiştir. Bu kavramı etkinleştirmek ve faydaları etkin bir şekilde yönetmek için proje yöneticisi rolünün iyi tanımlandığı güçlü bir süreç tanımına ihtiyaç olduğu vurgulanmıştır [26].

Faydalar yönetimi ve proje yönetiminin kurumların daha başarılı projelere ulaşmasına nasıl yardım ettiğinin cevabının araştırıldığı bir çalışma yapılmıştır. Orta ölçekli bir kurumda farklı rollerde yirmi üç kişi ile yarı yapılandırılmış bir anket çalışması yapılmış ve anket sonuçları analiz edilmiştir. Analiz yöntemi olarak fayda yönetimi yaklaşımı ve fayda bağımlılık ağı (Benefits Management Approach and Benefits Dependency Network) metodolojisi kullanılmıştır. Analiz sonuçlarına göre; zaman, maliyet, teknik gereksinimler, müşteri memnuniyeti ve hedef başarımı proje başarı faktörleri olarak belirlenmiştir. Diğer yandan kapsam kontrolü, üst yönetim desteği, ekip katılımı, kaynak yeterliliği, risk yönetimi, iş fırsatı, pazar etkisi, finansal kaynaklar ise proje başarısına ulaşmada takip edilecek kritik başarı faktörleri olarak belirlenmiştir [27].

Yüklenici ve müşteri açısından proje başarısına etki eden faktörlerin hangilerinin önemli olduğuna dair deneysel bir çalışma yapılmıştır. Sonuçlar, yüklenicilerin proje maliyetini ve süresini minimize etmeye daha fazla önem verirken, müşterilerin diğer paydaşların ihtiyaçlarını karşılamaya daha fazla önem verdiğini göstermiştir [28].

Proje başarı kriterlerini belirleme ve proje performans değerlendirme konusundan en sık kullanılan yöntemler referanslarla yıllara göre Tablo 1.1'de özetlenmektedir.

Tablo 1.1. Referanslarla en sık kullanılan yöntemler

Yöntem	**Referans**	**Yazar Adı ve Yılı**	**Ana Konular**
İstatistiksel Süreç Kontrolü, Kazanılmış Değer Yönetimi	[16] [18] [19]	Ilbeigi ve diğ.. (2009), Sen Leu ve diğ. (2008), Bauch ve diğ. (2001)	Kritik ve etkin endeksler, proje yönetimi performansının değerlendirilebileceği bir model geliştirilerek tanımlanmıştır.
Kazanılmış Değer Yönetimi	[20]	Bower ve diğ. (2009)	Proje performans değerlendirmesinde kazanılmış değer yönetimi kriterlerinin kullanımı.
Analitik Hiyerarşi Süreci, Kurumsal Karne	[15] [16]	Wadugodapitiya ve diğ. (2010), Ilbeigi ve diğ. (2009),	Kurumsal karne ve Analitik hiyerarşi yöntemlerini entegre ederek proje performans değerlendirmesi için çok boyutlu performans ölçüm modeli.
İş kırınım ağacı değerlendirmesi	[4] [17]	Radujković ve Sjekavica (2017), Koelmans (2004)	Göstergeler, başarı göstergelerini göstermek için iş kırınım ağacına benzer bir grafik haline getirilir.
Korelasyon ve Lojistik Regresyon Analizi	[13] [14]	Garousi ve diğ. (2019), Verner ve Cerpa (2007)	Başarısızlığı tahmin etmek için proje değişkenleri ile proje çıktıları ve lojistik modellerin korelasyon çalışması.
Korelasyon ve Regresyon Analizi	[8] [10] [11] [12]	Sanchez, ve diğ. (2017), Badewi (2016), Todorović (2015), Mir ve diğ. (2014), Akyol (2014)	Proje yönetimi performansını ve proje başarısını ölçmek için çok boyutlu çerçevelerin kullanılması.

Tablo 1.1.(Devam) Referanslarla en sık kullanılan yöntemler

Yöntem	Referans	Yazar Adı ve Yılı	Ana Konular
Ampirik analiz, Veri analizi	[24] [28]	Shashi ve diğ. (2014), Bryde ve Robinson (2005)	Maliyet, zaman ve geliştirici sayısı gibi kaynakların etki analizi.
Korelasyon analizi ve güvenilirlik analizi	[21] [27]	Gomes ve Romao (2016), Shahzad ve diğ. (2014)	Proje başarı kriterlerinin tanımlanmasının istatistiksel analizi.
MACBETH - Çok kriterli karar verme yöntemi	[22]	Viglioni ve diğ. (2016)	Kategorik tabanlı bir değerlendirme tekniği ile çekiciliği ölçen performans değerlendirme modeli.
Bulanık modelleme	[23]	Doskočil ve diğ. (2016)	Proje başarısının değerlendirilmesinde bulanık mantık kullanımı.

Tablo 1.2'de literatürde en sık kullanılan olan proje başarı faktörleri referanslarla yıllara göre özetlenmektedir.

Tablo 1.2. Referanslarla en sık kullanılan proje başarı kriterleri

Kriterler	Referans	Yazar ve Yayınlanma Yılı
Maliyet	[10, 14, 17, 22, 23, 25, 26, 27, 28, 29, 30]	Garousi (2019), Sanchez (2017), Badewi (2016), Mossalam (2016), Gomes (2016), Viglioni (2016), Doskočil (2016), Bryde (2005), Jugdev (2005), Koelmans (2004), Baccarini (1999)
Zaman	[10,14, 17, 22, 25, 27, 28, 29, 30]	Garousi (2019), Sanchez (2017), Badewi (2016), Gomes (2016), Viglioni (2016), Bryde (2005), Jugdev (2005), Koelmans (2004), Baccarini (1999)
Kalite	[10, 14, 17, 22, 25, 28, 29, 30]	Garousi (2019), Sanchez (2017), Badewi (2016), Viglioni (2016), Bryde (2005), Jugdev (2005), Koelmans (2004), Baccarini (1999)
Kapsam	[10, 17, 22, 23, 26, 27, 29, 31, 32]	Sanchez (2017), Gomes (2016), Mossalam (2016), Viglioni (2016), Doskočil (2016), Besner (2006), [Koelmans (2004), Jugdev (2005), Lim (1999)
İnsan kaynakları	[4, 11, 14, 22, 29, 32, 33, 34]	Garousi (2019), Radujković (2017), Viglioni (2016), Radujković (2014), Mir (2014), Jugdev (2005), Cooke-Davies (2002), Lim (1999)
Paydaş memnuniyeti	[14, 15, 17, 22, 23, 27, 35, 36]	Garousi (2019), Gomes (2016), Viglioni (2016), Doskočil (2016), Wadugodapitiya (2010), Müller (2012), Koelmans (2004), Bryde (2003)

Tablo 1.2.(Devam) Referanslarla en sık kullanılan proje başarı kriterleri

Kriterler	Referans	Yazar ve Yayınlanma Yılı
Kurumsal faydalar	[32, 37]	Chou (2013), Lim (1999)
Teknik, finansal, eğitimsel, sosyal, profesyonel elemanlar	[27, 38]	Gomes (2016), Ellatar (2009)
Proje yönetim araçları ve teknikleri	[4, 14, 37, 39]	Garousi (2019), Radujković (2017), Chou (2014), PricewaterCuppers (2012)
Kurumsal yapı	[4, 14, 33, 40]	Garousi (2019), Radujković (2017), Radujković (2014), Feger (2012)
Risk değerleme	[4, 27]	Radujković (2017), Gomes (2016)
Gereksinimler	[13, 23, 27, 41, 42]	Gomes (2016), Doskočil (2016), Keil (2012), Verner (2007), Taylor (2006),
Liderlik, personel, politika ve strateji, ortaklık ve kaynaklar	[27, 36]	Gomes (2016), Bryde (2003)
Proje yaşam döngüsü yönetimi süreçleri	[36, 45]	Bryde (2008), Bryde (2003)
Getiri ve kar, Pazar payı, saygınlık	[27, 43, 44]	Gomes (2016), Al-Tmeemy (2011), Blindenbach-Driessen (2006)
Rekabet avantajı	[43, 44]	Al-Tmeemy (2011), Blindenbach-Driessen (2006)
Proje yönetimi standartları	[4, 37, 46]	Radujković (2017), Nahod (2013), Chou (2013)
Proje yöneticisinin toplam yetkinlikleri	[4, 13, 33, 40, 47, 48, 49, 50]	Radujković (2017),Radujković (2014), Ramazani (2015), Feger (2011), Turner (2009), Ika (2009), Verner (2007), Radujković (2000)
Kurumsal kültür	[4, 14, 51, 52]	Garousi (2019), Radujković (2017), Westerveld (2003), Skulmoski (2001)
Proje yöneticilerinin duygusal zekâsı	[46, 53]	Nahod (2013), Yang (2011)
Amaç	[23, 26]	Mossalam (2016), Doskočil (2016)
Proje ekibi faktörleri	[4, 11, 13, 14, 17, 22, 27, 33]	Garousi (2019), Radujković (2017), Gomes (2016), Viglioni (2016), Mir (2014), Radujković (2014), Verner (2007), Koelmans (2004)
Kazanılmış değer yönetimi temel performans göstergeleri	[12, 16, 22, 23]	Viglioni (2016), Doskočil (2016), Todorović (2015), Ilbeigi (2009)
Tahminler	[13]	Verner (2007)
Müşteri faktörleri	[13, 14, 22]	Garousi (2019), Viglioni (2016), Verner (2007)

Literatürde, projenin başarısını etkileyen kriterlerin belirlenmesi ve bu kriterlerin girdi oluşturduğu proje performans değerlendirme modelleri ile ilgili çalışmalar görülmüştür. Literatür incelendiğinde konfigürasyon yönetimi, kalite yönetimi, doğrulama ve geçerleme süreci ile geliştirme süreci metriklerinin proje başarı kriteri olarak değerlendirilip bir proje performans modeli oluşturulmadığı görülmektedir. Bu çalışmada, proje başarı faktörü olarak yazılım yaşam döngüsü süreç metriklerinin kullanılmasının ve bu faktörlerin proje performans değerlendirmesinde kullanılmasının literatüre yenilik getirerek katkı sağlayacağı düşünülmektedir.

2. MODEL OLUŞTURMA ÇALIŞMASI

Bu bölümde araştırma iş akışı, modelde kullanılan Analitik Hiyerarşi Süreci ve İstatistiksel Süreç Kontrolü teknikleri, model oluşturma çalışmasında kullanılan veri toplama yöntemi, verilerin toplandığı araçlar, literatürdeki ölçme süreç ve standartları anlatılmaktadır.

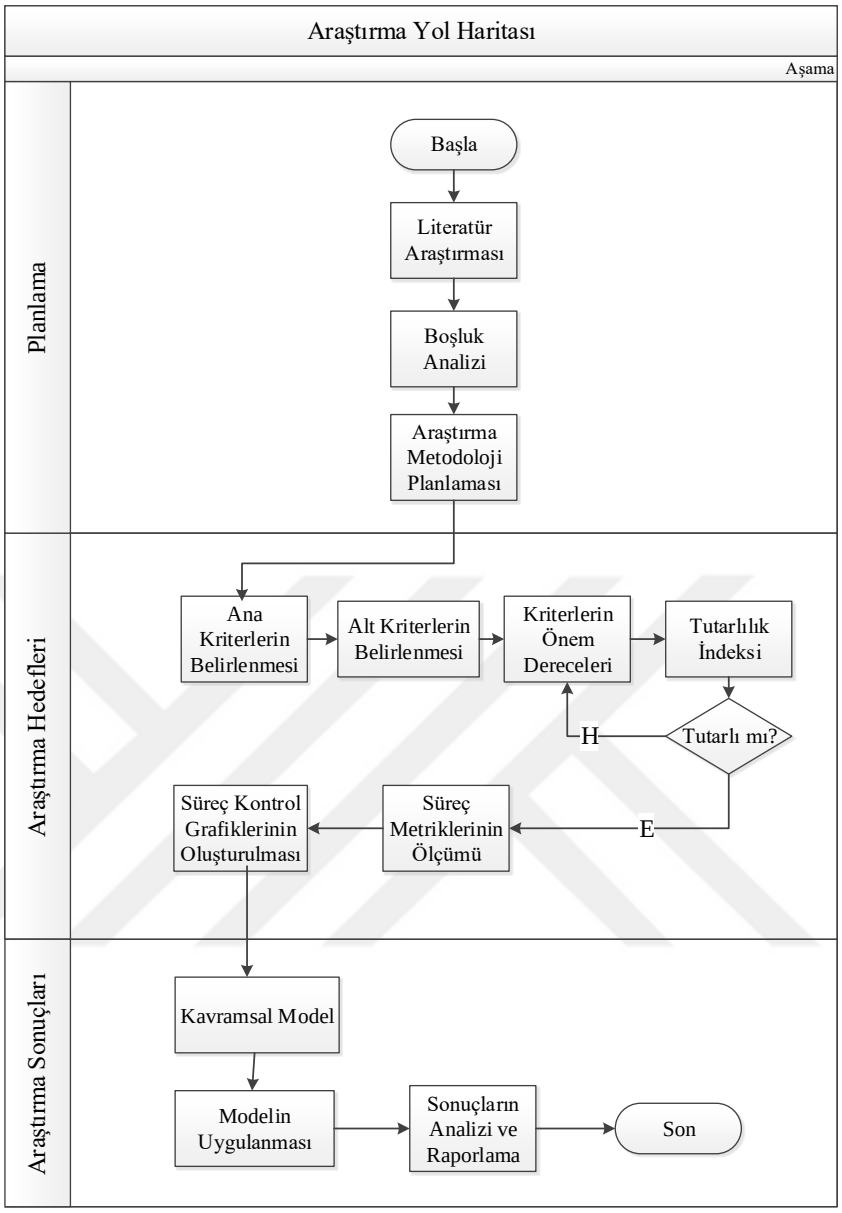
2.1. Araştırma İş Akışı

Bu tez çalışmasında bilimsel ilkeler çerçevesinde literatür araştırması, ölçütlerin belirlenmesi, modelin önerilmesi ve modelin uygulanması faaliyetleri yapılmıştır. Bu faaliyetler iş akışı olarak Şekil 2.1'de belirtilmiştir.

Tez çalışmasına literatür de kullanılmış olan proje başarı metriklerinin ve proje performans değerlendirmede kullanılan yöntemlerin araştırılması ile başlanmıştır. Hedeflenen performans değerlendirme modeli ve proje başarı kriterleri ile literatürdeki yöntem ve başarı kriterleri arasında boşluk analizi yapılmıştır. Araştırma metodoloji planlama aşamasında çalışmanın yapılacağı Ar-Ge kurumunun seçilmesi, süreç metrik verilerinin toplanacağı yazılım projelerinin belirlenmesi, verilerin toplanacağı proje destek araçları altyapısının incelenmesi, ölçüm toplama yöntemlerinin belirlenmesi ve toplanan verilerin analiz edilme yöntemlerinin belirlenmesi faaliyetleri yapılmıştır.

Araştırma hedefleri olarak proje başarısına etki eden ana kriterlerin belirlenmesinin ardından alt kriterlerin belirlenmesi, ana ve alt kriterlerin önem derecelerinin belirlenmesi ve tutarlılık oranlarının kontrol edilmesi işlemleri yapılarak AHS çıktıları kayıt altına alınmıştır. Sonraki aşamada süreç metrik ölçümleri yapılarak her bir metriğin alt ve üst limitleri belirlenmiştir.

Araştırma çıktısı olarak proje performans değerlendirme modeli ortaya konmuş, önerilen model bir Ar-Ge kurumunda uygulanmış ve sonuçların analizi yapılarak raporlama yapılmıştır.



Şekil 2.1. Araştırma iş akışı

2.2. Analitik Hiyerarşi Süreci

Çok Kriterli Karar Verme (ÇKKV), karmaşık problemler için karar vermeyi kolaylaştırmak için kullanılan resmi adımları tanımlayan bir yaklaşımdır. ÇKKV, karmaşık problemlerde öznelliği daha görünür hale getirmeye çalışarak ölçüm süreçlerini değer yargılarıyla birleştirmeyi amaçlamaktadır [54]. ÇKKV, bir dizi kriterin değerlendirileceği seçimler listesinden en iyi alternatife karar veren ilişkiye dayanır.

Literatürde çok kriterli karar verme konusunda çeşitli yöntemler vardır. Bu yöntemler AHS, ANP, TOPSİS, ELECTRE, PROMETHEE vb. olarak örneklendirilebilir. Analitik Hiyerarşi Süreci literatürdeki en pratik yöntemlerden biridir. Bu tez çalışmasında süreç metriklerinin ağırlıklandırma çalışması yapılması kapsamında oransal ölçek kullanan AHS yönteminin kullanılmasına karar verilmiştir. AHS, Saaty [55] tarafından geliştirilen bir karar verme yöntemidir. Birincil kullanımı, çok değişkenli ortamlarda karar ve tahmin problemlerine çözümler sunmaktır. AHS hedefleri ve kriterleri düzenleyerek alternatifler için öncelikli ağırlıklar kurar ve alt ölçütlerin hiyerarşik bir yapıda kurulmasını sağlar. ÇKKV problemlerini çözmek için birçok teknik mevcut olsa da, Saaty tarafından önerilen AHS, karmaşıklığı aşmak için daha pratik bir tekniktir.

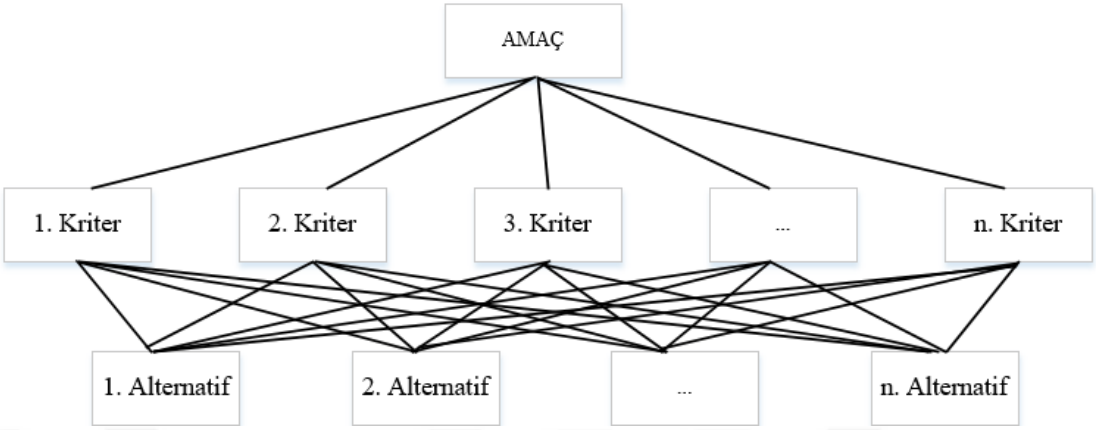
AHS karar vermede temel bir yaklaşımdır. Çeşitli ölçütlere göre değerlendirilen bir dizi alternatif arasından en iyisini seçmek için tasarlanmıştır. Bu süreçte karar verici, daha sonra alternatifleri sıralamak ve genel öncelikler geliştirmek için kullanılan basit ikili karşılaştırmalar yapar. AHS hem kararlarda tutarsızlığa izin verir hem de tutarlılığı artırmak için bir araç sağlar [56]. AHS'nin tipik adımları aşağıdaki beş adımı içerir:

1. Amacın tanımlanması ve hiyerarşinin oluşturulması
2. İkili karşılaştırma matrisinin oluşturulması
3. Öz değerlerin ve öz vektörlerin hesaplanması
4. Tüm karşılaştırma matrislerinin tutarlılığının test edilmesi
5. Karar vermek için alternatiflerin nihai önceliklerinin oluşturulması

2.2.1. Hiyerarşinin oluşturulması

AHS uygulamasının ilk aşaması, karmaşık problemi alt problemlere ayırarak bir hiyerarşi geliştirmektir. Amaçlar, kriterler ve alternatifler hiyerarşinin üç temel seviyesidir [57]. En baştan başlayarak, kararın amacı, orta düzeyde kriterleri ve en düşük düzeyde alternatifleri hiyerarşik olarak yapılandırır [58]. Hiyerarşik yapı, karar vericilere kriterleri ve alternatifleri etkili bir şekilde karşılaştırma olanağı sağlar [59]. Saaty, insanın bilişsel yeteneklerinin sınırlarını göz önünde bulundurarak, her hiyerarşi seviyesindeki eleman sayısını 7 ± 2 olarak önermektedir [60]. AHS'de, daha düşük seviyeli elemanlardan daha yüksek seviyeli elemanlara olan ilişkinin bağımsız olduğu

ve bir seviyedeki elemanların da bağımsız olduğu varsayılmaktadır. Üç seviyeli tipik AHS hiyerarşi yapısı Şekil 2.2'de gösterilmiştir.



Şekil 2.2. AHS'de üç seviyeli tipik hiyerarşi yapısı

2.2.2. İkili karşılaştırma matrisinin oluşturulması

Başlangıçta Thurstone (1927) tarafından önerilen ikili karşılaştırma tekniği, iyi yapılandırılmış bir tekniktir ve belirli bir ölçüt açısından iki özniteliği veya alternatifi ikili olarak karşılaştırmak için AHS'de [55, 58, 61] yaygın olarak kullanılmaktadır. İkili karşılaştırma matrisi, sayısal ölçekte ifade edilen unsurlardan oluşur. Bu unsurların değerleri, nitel özellikleri veya kriterleri ölçülebilir sayılara dönüştürülmesi için deneyimleri ve uzmanlıkları temelinde karar vericiler tarafından verilir. Soyut özellikler, bir elementin diğerine ne kadar daha hâkim olduğunu temsil eden bir mutlak yargılama ölçeği ile ölçülür. Kriterlere göre iki unsuru karşılaştırmak için Saaty [62] tarafından önerilen 1-9 temel ölçeği Tablo 2.1'de gösterilmiştir. Bir karşılaştırma matrisini tamamlamak için n (n- 1) / 2 adet karşılaştırma gereklidir. Tüm ikili karşılaştırmalar $A = (a_{ij})_{nxn}$ şeklinde bir matris olarak ifade edilir. İkili karşılaştırma yapılacak n adet kriter olduğu varsayıldığında karşılaştırmalar sonucu oluşan A matrisi aşağıdaki gibidir [63]:

$$A = \begin{bmatrix} 1 & a_{12} & \cdots & a_{1n} \\ 1/a_{12} & 1 & \cdots & a_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ 1/a_{1n} & 1/a_{2n} & \cdots & 1 \end{bmatrix} \quad (2.1)$$

A matrisinde köşegen üzerindeki matris değerleri, aynı kriterlerin karşılaştırılmasından dolayı 1 değerini alır. Matristeki diğer değerler için $a_{ij} = \frac{1}{a_{ji}}$ kuralı geçerlidir.

Tablo 2.1. AHS için kullanılan 1-9 temel ölçeği [62]

Önem Dereceleri	Tanım	Açıklama
1	Eşit önemli	İki faaliyet amaca eşit düzeyde katkıda bulunur.
3	Biraz Daha Fazla Önemli	Tecrübe ve yargı ile bir faaliyet diğerine göre fazla derecede tercih edilir.
5	Kuvvetli Derecede Önemli	Tecrübe ve yargı ile bir faaliyet diğerine göre kuvvetli derecede tercih edilir.
7	Çok Kuvvetli Derecede Önemli	Bir faaliyet çok kuvvetli bir biçimde tercih edilir ve baskınlığı uygulamada rahatlıkla görülür.
9	Aşırı Derecede Önemli	Bir faaliyet diğerine tercih edilmesine ilişkin kanıtlar çok büyük bir güvenirliliğe sahiptir.
2, 4, 6, 8	İki ardışık karar arasındaki ara değerler	Uzlaşma gerektiğinde kullanmak üzere iki ardışık karar arasındaki ara değerler.

2.2.3. Öz değer ve öz vektörlerin hesaplanması

Günümüzde, bir karşılaştırma matrisinden alternatiflerin öncelikli ağırlıklarını elde etmek için kullanılabilecek 20'den fazla farklı yöntem vardır. Örnek olarak Normalleştirilmiş Sütun Toplamı Yöntemi ve Normalize Edilmiş Sütunlar Yönteminin Aritmetik Ortalaması [55], Özvektör Yöntemi [60], Doğrudan En Küçük Kareler Yöntemi ve Ağırlıklı En Küçük Kareler Yöntemi [64], Logaritmik En Küçük Kareler Yöntemi [65], Geometrik Ortalama Çözümü [66], Logaritmik Hedef Programlama Yöntemi [67] sayılabilir.

İkili karşılaştırmalar tamamlandıktan sonra kriterlerin önceliklerinin belirlenmesi aşamasına geçilir. Öncelikle A matrisindeki değerlerin (a_{ij}) normalleştirilmesi gereklidir. En yaygın kullanılan yöntemde, her bir sütun elemanı bulunduğu sütunun

toplam değerine bölünür. Elde edilen matris normalize edilmiş matristir. Bu işlem adımları aşağıdaki Denklem (2.2) ve Denklem (2.3)'te verilmiştir.

$$B_i = \begin{bmatrix} b_{11} \\ b_{21} \\ . \\ . \\ . \\ b_{n1} \end{bmatrix} \qquad (i = 1,2,3,\ldots,n) \tag{2.2}$$

$$b_{ij} = \frac{a_{ij}}{\sum_{i=1}^{n} a_{ij}} \qquad (i,j = 1,2,3, \ldots, n) \tag{2.3}$$

Bu işlem A matrisindeki tüm sütunlara uygulandığında n kriter sayısı olmak üzere, nxB adet sütun vektörleri elde edilir. Bu vektörler bir araya getirildiğinde Denklem (2.4)'te verilen C matrisi oluşturulur. C matrisi normalleştirilmiş matristir. C matrisinin her bir satırındaki elemanların aritmetik ortalaması alındığında kriter öncelik sıralamasına yönelik W ağırlık vektörü elde edilir ve bu vektörün değerleri toplamı 1 e eşittir.

$$C = \begin{bmatrix} c_{11} & c_{12} & . & . & c_{1n} \\ c_{12} & c_{22} & . & . & c_{2n} \\ . & . & . & . & c_{3n} \\ c_{n1} & c_{n2} & . & . & c_{nn} \end{bmatrix} \quad (i,j = 1,2,3,\ldots,n) \tag{2.4}$$

W sütun vektörü elemanlarını hesaplama formülü Denklem (2.5)'te verilmiştir. W ağırlık vektörü ise Denklem (2.6)'da gösterilmiştir.

$$w_i = \frac{\sum_{j=1}^{n} c_{ij}}{n} \qquad (i,j = 1,2,3,\ldots,n) \tag{2.5}$$

$$W = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \tag{2.6}$$

w_i = i. elemanın önem derecesi

c_{ij} = i. elemanın j. elemana göre önem derecesi

n karşılaştırılan elaman sayısı

Tüm bu yöntemlerin ortak özelliği karşılaştırma matrisinde normalleştirme işleminin yapılıyor olmasıdır. Böylece amacı başarmak için her bir kriterin amaca göre önem dereceleri ve her bir alternatifin ilgili kritere göre önem dereceleri belirlenmiş olmaktadır [68].

2.2.4. Tüm karşılaştırma matrislerinin tutarlılığının test edilmesi

Dördüncü adımda, tüm karşılaştırma matrislerinin tutarlılığı, karar vermek için kullanılmadan önce test edilir. Karar verme sürecinde, farklı nitelikler veya kriterler karşılaştırılırken, karar sorunları doğada karmaşık olduğundan, tutarsızlık sorunu ortaya çıkacaktır. Örneğin, A niteliğinin B niteliğinden 2 kat önemli olduğu ve B niteliğinin C niteliğinden 3 kat önemli olduğu, ancak A niteliğinin C niteliğinden 6 kez yerine yalnızca 4 kat önemli olduğu varsayılsın. Benzer şekilde, A değerleri B'den büyük, B değeri C'den büyük, ancak C değeri A'dan büyük, yani A> B; B> C; ancak C> A varsayılsın. Bu sorunların her ikisine de tutarsızlık denir [69]. Bu nedenle, karşılaştırma matrisinin öncelik vektörleri hesaplanmadan önce karşılaştırma matrisi için tutarlılık testi gereklidir. Karşılaştırma matrisi için tutarlılık testi başarısız olursa, karşılaştırma matrisindeki tutarsız elemanların gözden geçirilmesi gerekir, aksi takdirde karar analizi sürecinin sonucu anlamsızdır. Tutarlılığı test etmek için popüler yöntem tutarlılık oranıdır (Consistency Ratio - CR) [69]. Tutarlılık oranı Denklem (2.7)'de verilmiştir.

$$\mathrm{CR} = \frac{\mathrm{CI}}{\mathrm{RI}} < 0.1 \tag{2.7}$$

Denklem (2.7)'de CI ile ifade edilen değişken tutarlılık indeksidir ve Denklem (2.8) ile ifade edilir:

$$\mathrm{CI} = \frac{\lambda_{\max} - \mathrm{n}}{\mathrm{n}-1} \tag{2.8}$$

$\lambda_{\max}$, A matrisinin en büyük özdeğeri ve n ise A matrisinin derecesidir. Karşılaştırma matrisi yalnızca CR değeri 0.1'den düşükse tutarlıdır.

$$\lambda = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} w_i}{w_i} \tag{2.9}$$

$\lambda_{\max}$ değeri λ değerlerinin en büyüğüdür. λ'nin formülü Denklem (2.9)'da verilmiştir.

a_{ij} ikili karşılaştırma matrisindeki i. ve j. elemandır.

w_i ikili karşılaştırma matrisine ait göreceli öncelik (ağırlık) vektörüdür.

n karşılaştırılan kriter (elaman)sayısıdır.

CI tutarlılık indeksidir.

RI, Tablo 2.2'de gösterilen Matris Boyutuna dayanan ortalama rastgele indeksidir.

Tablo 2.2. Ortalama rasgele indeks

n	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.52	0.89	1.11	1.25	1.35	1.4	1.45	1.49

2.2.5. Alternatiflerin nihai önceliklerinin oluşturulması

Son aşamada, her seviyedeki karşılaştırma matrislerinden hesaplanan öncelik vektörleri, bir sonraki seviyedeki öncelikleri ağırlıklandırmak için kullanılır. Bu işlem her element için tekrarlanır, ardından alt düzeydeki alternatiflerin nihai öncelikleri elde edilene kadar kriter ve alternatiflerin tüm öncelikleri sentezlenir [55]. Bu aşamada, her alternatif her bir kriter açısından ikili karşılaştırmaya tabi tutulur. Değerlendirmede Tablo 2.1'deki 1-9 temel ölçeği kullanılır. İkili karşılaştırmalar sonucunda elde edilen matris, daha önce kriterlere yapıldığı gibi, normalleştirilir. Normalleştirilmiş matrisin satır ortalamaları alınarak ilgili kriterlere göre alternatiflerin yüzde ağırlıkları bulunur. Ardından her bir kritere göre hesaplanmış ağırlık sütun vektörü kullanılarak alternatifler için tutarlılık oranları hesaplanır. CR > 0.1'den büyük olduğu durumlarda alternatiflerin ikili karşılaştırma önem dereceleri gözden geçirilir.

AHS'de karar verirken son olarak sürecin aşamalarında elde edilen ağırlıklar kullanılarak genel amaç açısından alternatiflerin görece önem değerleri hesaplanır. Bu hesaplamada; önce her bir alternatifin kriterlere göre önem dereceleri ile kriterlerin kendi aralarındaki karşılaştırmalarından elde edilen ağırlıkları ile birebir çarpılarak yapılır, sonrasında her bir alternatife ait bu çarpım değerleri toplanarak alternatiflerin genel amaca göre önem dereceleri elde edilmiş olur. En büyük görece önem değerine sahip olan alternatifin seçilmesi kararı alınır.

2.2.6. AHS'de grup kararı

Bazı durumlarda karar vericiler birden fazla olmaktadır. Bu durumda grup kararı verme süreci işletilir. Grup ile karar verme amacıyla kullanılan tekniklerden en yaygın olanları Delphi [70] ve Nominal Group [71] teknikleridir.

2.2.7. Delphi yöntemi

Bu tekniğin amacı uzlaşma yolu ile grup kararı oluşturmaktır. Yöntem aşağıdaki adımlardan oluşur [70]:

1. Anket hazırlanır.
2. Uzmanlara gönderilir.
3. Anket yanıtları gruplanır.
4. Kimin hangi cevabı verdiği gizli kalacak şekilde tüm katılımcılara (uzmanlar) diğer katılımcıların yanıtları gösterilir.
5. Katılımcılar, bu bilgiler ışığında önceki yanıtlarını gözden geçirir ve gerekirse yanıtlarını değiştirir.
6. Bu işleme birkaç iterasyon devam edilir. Her iterasyonda sorulara verilen yanıtların standart sapmasının daha az olması beklenir.
7. Aritmetik ortalama ya da medyan ortalama alınarak son karar elde edilir.

Bu yöntem yüz yüze gerçekleştirilmez. Grup tartışması yoktur ve beyin fırtınası olanaklarından yoksundur.

2.2.8. Nominal grup yöntemi

Yöntem aşağıdaki adımlardan oluşur [71]:

1. Sessizce (tartışmadan) herkes fikir üretir ve önerilerini/görüşünü yazar.
2. Yazılan fikirler aynı anda en çok bir tane olmak üzere grubun her üyesinden alınır.
3. Fikirler gruplanır ve benzer olanlar elenir.
4. Fikirler tartışmaya açılır.
5. Her üye her fikir için oyunu kullanır (örneğin 0-10 arası puan verir).
6. Üyelerden gelen puanlar toplanır.

Tartışma bölümü, yararlarına rağmen, herkesin eşit düzeyde bilgisinin eşit olmadığı konularda birkaç kişinin görüşünün tüm grup üzerinde hâkim olmasına yol açabilir. Bu nedenle bu yöntem dikkatle kullanılmalıdır.

AHS'de de karar alma sürecindeki katılımcılar eşit öneme sahip ise geometrik ortalama yaygın olarak kullanılır ve Denklem (2.10)'daki formül ile hesaplanır:

$$(a_{ij}^{1} * a_{ij}^{2} * \dots * a_{ij}^{N})^{1/N} \tag{2.10}$$

$a_{ij}^{1}, a_{ij}^{2}, \dots, a_{ij}^{N}$ grup üyelerinin bireysel yargıları

N gruptaki katılımcı sayısını ifade etmektedir.

2.2.9. Duyarlılık analizi

Duyarlılık analizi ikili karşılaştırmalardaki yanıtların katılımcılara göre farklılık gösterebileceği ve katılımcıların görüşlerinin zamanla değişebileceği varsayımına dayanmaktadır [72]. Duyarlılık analizi kurulan modelin hangi kriter ya da kriterlere ne derece bağlı olduğunu belirler [73]. Duyarlılık analizi, alternatiflerin kriterlerin öneminde yapılacak değişikliklere ne kadar duyarlı olduklarını gösterir [57].

2.2.10. AHS'nin avantajları ve dezavantajları

Hiyerarşik yapısı ve geri bildirim sistemi ile karmaşık problemlerin analizinde kolaylık ve basitlik sağlar [74].

AHS'nin avantajları aşağıdaki gibi sıralanabilir [61,75]:

1. Sorunun çözümü için hiyerarşik bir karar yapısının oluşturulması,
2. Karmaşık problemleri kolaylaştırması,
3. Hem nitelik hem sayısal verilerin sürece dâhil edilebilmesi,
4. Çoklu uzman görüşlerinin değerlendirilmesini mümkün kılma,
5. İkili karşılaştırma avantajı,
6. Karar verme sürecinde katılımcıların uzun tartışmalara girmeyi gerektirmemesi,
7. Uzman kararlarındaki tutarsızlığı ele alma becerisi,
8. Duyarlılık analizi ile son kararların esnekliğinin analiz edilebilmesi.

Öte yandan dezavantajları aşağıdaki gibi sıralanabilir [75-78].

1. AHS sürecine yeni kriterler eklendiğinde sürecin çok zaman alması,
2. Herhangi bir alternatif eklendiğinde karar seçenekleri sıralamasının değişmesi,
3. Modelleme öznel olması,
4. Kriter sayısının artması ikili karşılaştırma sayısını artırması,
5. Tek yönlü modelleme hiyerarşisine sahip olması nedeniyle kriterler arası ilişkileri göz önüne alamaması,
6. Kriterlerin her aşamada doğru seçilememesinin sonuçların geçerliliğinin sorgulanmasına neden olması.

2.3. İstatistiksel Süreç Kontrolü Teknikleri

İstatistiksel süreç kontrolü, süreç kontrolü için istatistiki yöntemleri uygulayan bir araçtır. İstatistiksel süreç kontrolü için farklı yöntemler vardır: serpilme diyagramları, çalışma grafikleri, neden ve sonuç diyagramları, histogramlar, çubuk grafikler, pareto grafikleri ve kontrol grafikleri [18]. İstatistiksel süreç kontrolü kontrol grafikleri imalat ve yazılım süreç iyileştirmesi için başarılı bir araç olarak kullanılmaktadır [79]**.** Bu tez çalışmasında, süreç ölçümlerinin alt ve üst kontrol limitlerini belirlemek için kontrol grafikleri kullanılmıştır.

Bir süreçteki varyasyon genel nedenlerden veya özel nedenlerden kaynaklanıyor olabilir. Genel neden varyasyonu insanların, makinelerin, çevrenin ve yöntemlerin normal etkileşimlerinin sonucu olarak normal bir varyasyondur. Özel neden varyasyonu normal sürecin parçası olmayan olaylardan kaynaklanır. Bu nedenler bazen ortaya çıkar ve önlenebilir. Tüm süreç verileri kontrol limitleri içinde ve herhangi bir özel eğilim yok ise, süreç kontrol altında kabul edilir [80]. Bölüm 2.3.1'de genel ve özel nedenler hakkında detaylı bilgi verilmektedir.

İstatistiksel süreç kontrolünde kontrol grafikleri, süreci izlemek ve kontrol etmek için yaygın olarak kullanılmaktadır. İstatistiklerde farklı kontrol şeması türleri vardır [81-83]. Yazılım süreçlerinde, veri noktaları genellikle imalat süreçlerinde olduğu kadar sık değildir [84]. Her veri noktası ayrı ayrı çizilir ve değerlendirilir. Bu nedenle, yazılım mühendisliği alanında en çok kullanılan kontrol grafiği Bireysel (X) ve Hareketli Aralık (MR) kontrol grafikleridir.

XMR kontrol grafikleri iki temel özellikten oluşur; orta çizgisi ve sigma değeri. Orta çizgi veri noktalarının ortalama değeridir ve sigma bu veri noktalarının standart sapmasıdır. X grafiklerinde veri noktaları grafiklerde ayrı değerlerdir, MR grafiklerinde ise veriler birbirini takip eden iki değer arasındaki mutlak fark hesaplanarak elde edilir. Bu kontrol grafiklerinde, üst ve alt kontrol limitleri genellikle tüm verilerin% 99'unu kapsayan orta çizgiden üç sigma değeriyle hesaplanır [79]. Bölüm 2.3.4'te bireysel gözlem değerleri için kullanılan kontrol grafiği hakkında detaylı bilgi verilmektedir. Bu tez çalışmasında süreç metrik ölçümleri için bireysel gözlem değerleri için kontrol grafiği kullanılmıştır.

Bir ürün müşteri beklentilerini karşılıyor ya da aşıyorsa bu ürün kontrol altında olan bir süreç tarafından üretilmiş olmalıdır. Bir başka deyişle, süreç ürünün kalite özelliklerinin hedef boyutları etrafında çok az değişkenlik gösterir. İstatistiksel Süreç Kontrolü değişkenliği azaltarak süreç kararlılığını sağlamada ve süreç kapasitesini iyileştirmede güçlü bir problem çözme araç koleksiyonudur. İstatistiksel süreç kontrolü yirminci yüzyılın en büyük teknolojik gelişmelerinden biridir çünkü temel ilkelere dayanır, kullanımı kolaydır, önemli etkisi vardır ve herhangi bir sürece uygulanır. İstatistiksel süreç kontrolünün yedi ana aracı vardır:

1. Histogram ya da dal yaprak grafiği
2. Kontrol sayfası
3. Pareto kartı
4. Neden sonuç diyagramı
5. Hata konsantrasyon diyagramı
6. Serpilme (scatter) diyagramı
7. Kontrol grafiği

İstatistiksel süreç kontrolünün uygun şekilde işletilmesi, bir kuruluştaki tüm bireylerin kaliteyi ve üretkenliği sürekli iyileştirmeye çalıştığı bir ortam oluşturulmasına yardımcı olur. Bu ortam en iyi, yönetim işin içine dâhil olduğunda gelişir.

Yedi araçtan Shewhart kontrol grafiği en teknik olarak en gelişmiş araçtır. 1920'lerde Walter A. Shewhart tarafından Bell Laboratuvarlarında geliştirilmiştir. İstatistiksel süreç kontrolünün temelini oluşturan istatistiksel kavramların daha iyi anlaşılması için aşağıdaki bölümde önce Shewhart'ın değişkenlik teorisini hakkında bilgi verilmiştir.

2.3.1. Genel neden ve özel nedenler

Herhangi bir üretim sürecinde, ne kadar iyi tasarlandığına veya dikkatlice idame edildiğine bakılmaksızın, belirli bir doğal değişkenlik miktarı her zaman mevcuttur. Bu doğal değişkenlik veya "Arka plan gürültüsü" birçok küçük ve kaçınılmaz nedenin birikmiş etkisidir. İstatistiksel kalite kontrol çerçevesinde, bu doğal değişkenliğe genellikle istikrarlı şans nedenleri sistemi denir. Şans değişkenliğine sahip bir sürece istatistiksel olarak kontrol altında denilir. Günümüzde şans nedeni terminolojisi yerine genel neden ifadesi de kullanılmaktadır.

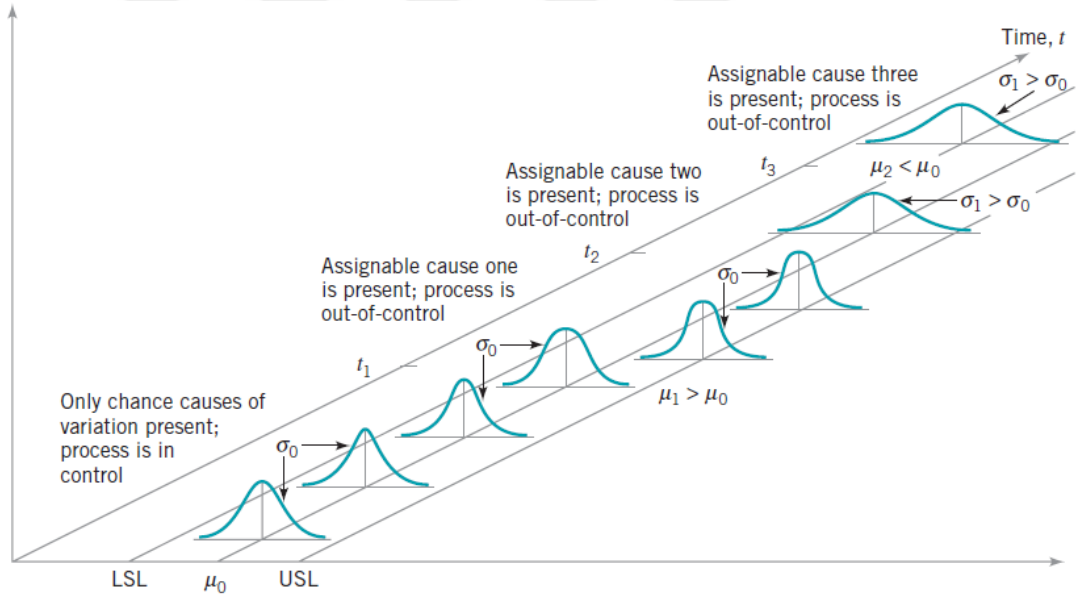
Bir sürecin çıktısında ara sıra farklı tipte değişkenlikler olabilir. Bu anahtar kalite özelliklerindeki değişkenlik genellikle üç kaynaktan ortaya çıkar: yanlış ayarlanmış veya kontrol edilen makineler, operatör hataları veya kusurlu hammadde. Böyle değişkenlik arka plan gürültüsü ile karşılaştırıldığında genellikle büyüktür ve genellikle kabul edilemez süreç performansı düzeyini temsil eder. Şans nedeni örüntüsüne sahip olmayan bu değişkenliğe özel değişkenlik nedeni denir. Özel nedenlere sahip olan bir sürece kontrol dışı süreç denir.

Bu şans ve özel varyasyon nedenleri Şekil 2.3'de gösterilmiştir. t_1 anına kadar süreç kontrol altındadır; yani, sadece varyasyonun şans nedenleri mevcuttur. Sonuç olarak, sürecin hem ortalama hem de standart sapması kendi kontrol değerlerindedir (μ_0, σ_0). t_1 anında özel bir neden oluşmaktadır. Şekil 2.3'de gösterildiği gibi, bu özel neden süreç ortalamasını yeni bir değere kaydırmıştır $(\mu_1 > \mu_0)$. t_2 anında başka bir özel neden oluşmuştur. Bu aşamada $\mu = \mu_0$ olmuştur fakat sürecin standart sapması daha büyük bir değere ulaşmıştır $(\sigma_1 > \sigma_0)$. t_3 anında hem süreç ortalamasını hem de süreç standart sapmasını değiştiren bir başka özel neden oluşmuştur. t_1 anından itibaren süreçte meydana gelen özel nedenler sürecin kontrol dışına çıkmasına neden olmuştur.

Süreçler genellikle nispeten uzun süreler boyunca kontrol durumunda çalışır. Bununla birlikte, hiçbir süreç sonsuza kadar kararlı değildir ve sonunda rasgele olarak özel nedenler ortaya çıkar. Bu durum sürecin kontrol dışına çıkmasına ve sürecin büyük bir orandaki çıktısının gereksinimlerle uyumsuz olmasına neden olur. Örneğin, Şekil 2.3'de süreç kontrol altındayken, üretimin çoğu alt ve üst spesifikasyon sınırları

(sırasıyla LSL ve USL) arasındadır. Süreç kontrolden çıktığında, süreç çıktılarının daha yüksek bir oranı bu spesifikasyonların dışındadır.

İstatistiksel süreç kontrolünün temel amacı, süreç kaymalarının özel nedenlerini çabucak bulmaktır, böylece sürecin araştırılması ve düzeltici eylem birçok uygun olmayan birim üretilmeden önce gerçekleştirilebilir. Kontrol grafiği bu amaçla yaygın olarak kullanılan çevrimiçi bir süreç izleme tekniğidir. Kontrol grafikleri bir üretim sürecinin parametrelerini tahmin etmek ve bu bilgiler aracılığıyla süreç kapasitesini belirlemek için de kullanılabilir. Kontrol grafiği ayrıca süreç iyileştirme faaliyetlerinde yararlı bilgiler de sağlayabilir. İstatistiksel süreç kontrolünün nihai amacı süreçteki değişkenliğin giderilmesidir. Değişkenliğin ortadan kaldırılması tamamen mümkün olmayabilir, ancak kontrol grafiği değişkenliği olabildiğince azaltmada etkili bir araçtır.

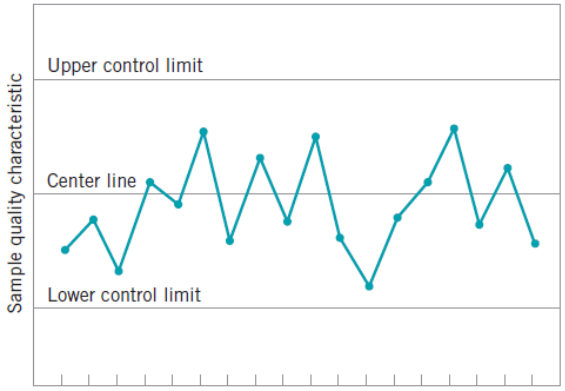


Şekil 2.3. Genel neden ve özel neden değişkenliği [85]

Kontrol kartlarının istatistiksel temeli

Tipik bir kontrol grafiği Şekil 2.4'te gösterilmektedir. Şekildeki grafik, kontrol altındaki bir süreçteki kalite karakteristiğinin ortalama değerini temsil eden bir orta çizgi (OÇ) içermektedir. Üst kontrol limiti (ÜKL) ve alt kontrol limiti (AKL) olarak adlandırılan diğer iki yatay çizgi de grafikte gösterilmiştir. Noktalar kontrol sınırları içinde kaldığı sürece, süreç kontrol altında kabul edilir ve hiçbir eylem gerekli değildir. Ancak, kontrol sınırları dışında kalan bir nokta sürecin kontrolden çıkması şeklinde

yorumlanır ve bu davranışa neden olan nedenleri ortadan kaldırmak için inceleme ve düzeltici faaliyet gereklidir. Kontrol grafiğindeki noktaları düz çizgi ile bağlamak alışılagelmiştir. Böylece nokta dizisinin zaman içinde nasıl geliştiğini görselleştirmek daha kolay olmaktadır.



Şekil 2.4. Tipik bir kontrol grafiği [85]

Tüm noktalar kontrol sınırları içinde çizilse bile, sistematik veya rastgele olmayan bir şekilde davranıyorlarsa, bu sürecin kontrolden çıkmış olduğunun bir göstergesi olabilmektedir. Örneğin, son 20 noktanın 18'i orta çizgi ile üst kontrol limitinin arasındaysa ve bu noktalardan sadece ikisi orta çizgi ile alt kontrol limitinin arasındaysa, bir şeylerin yanlış olduğundan şüphe edilebilir. Genellikle, kontrol grafiğinde belirli bir rasgele olmayan desenin görünmesinin bir nedeni vardır ve bu neden bulunup ortadan kaldırılabilirse süreç performansı iyileştirilebilir.

Kontrol grafikleri ile hipotez testleri arasında yakın bir bağlantı vardır. Bu bağlantıyı göstermek için Şekil 2.4'teki dikey eksenin örnek ortalaması $\bar{x}$ olduğunu varsayılsın. $\bar{x}$ in mevcut değeri kontrol limitleri arasındaysa süreç ortalaması kontrol altındadır sonucuna varılır. Bu durumda süreç ortalaması μ_0 değerine eşittir. Diğer taraftan, $\bar{x}$ her iki kontrol limitini de aşarsa süreç ortalamasının kontrolden çıktığı sonucuna varılır. Bu durumda sürecin yeni ortalaması μ_0 değerinden farklı bir değer alacaktır ($\mu_1 \neq \mu_0$). Bir bakıma, kontrol grafiği, sürecin istatistiksel olarak kontrol altında olduğu hipotezinin bir testidir. Kontrol limitleri içinde bir nokta çizilmesi, istatistiksel kontrol hipotezini reddedememek ve kontrol limitleri dışında bir nokta çizmek istatistiksel kontrol hipotezini reddetmeye eşdeğerdir.

Hipotez testi çerçevesi birçok açıdan yararlıdır, ancak kontrol grafikleri ve hipotez testleri arasındaki bakış açısından bazı farklılıklar vardır. Örneğin, istatistiksel testlerde hipotezler, genellikle varsayımların geçerliliğini kontrol ederken, kontrol grafikleri varsayılan istatistiksel kontrol durumundan sapmaları tespit eder. Değişkenliği azaltmak ve istatistiksel kontrolü başarmak amacıyla bir sürece kontrol grafikleri uygulanırken dağılımın şekli ve bağımsızlık gibi varsayımlar üzerinde çok endişe etmeye gerek yoktur. Özel bir neden süreç parametrelerinde birçok farklı tipte değişimlere neden olabilir. Örneğin, süreç ortalaması anlık olarak yeni bir değere kayabilir ve orada kalabilir (bu duruma bazen sürekli değişim denir); ya da süreç ortalaması aniden kayabilir; ancak özel neden kısa ömürlü olabilir ve ortalama daha sonra nominal veya kontrol değerine geri dönebilir; ya da özel neden, süreç ortalamasının değerinde sabit bir kaymaya veya eğilime neden olabilir. Sadece sürekli değişim, normal istatistiksel hipotez test modeline çok iyi uymaktadır.

Kontrol grafiği için genel bir model verebilir. W, ilgilenilen bazı kalite karakteristiklerini ölçen bir örnek istatistik olarak kabul edildiğinde ve w ortalamasının μ_w, w'nin standart sapmasının $\sigma_w$olduğunda kontrol grafiğinin orta çizgi, üst kontrol limiti ve alt kontrol limiti aşağıdaki gibi ifade edilebilir:

$$\text{ÜKL} = \mu_w + L\sigma_w$$

$$\text{OÇ} = \mu_w \qquad (2.11)$$

$$\text{AKL} = \mu_w - L\sigma_w$$

burada L, standart sapma birimlerinde ifade edilen kontrol limitlerinin orta çizgiden "mesafesi" dir. Bu genel kontrol çizelgeleri teorisi ilk olarak Walter A. Shewhart tarafından önerilmiştir ve bu ilkelere göre geliştirilen kontrol grafiklerine Shewhart kontrol grafikleri denir.

Kontrol grafiği, istatistiksel kontrol ile kast edileni tam olarak tanımlamak için bir araçtır. Bu nedenle çeşitli şekillerde kullanılabilir. Birçok uygulamada, çevrimiçi süreç gözetimi için kullanılır. Yani, örnek veriler toplanır ve kontrol grafiği oluşturmak için kullanılır ve $\bar{x}$ örnek değerleri kontrol limitleri içindeyse ve sistematik bir desen göstermiyorsa, süreç grafik tarafından belirtilen seviyede kontrol altındadır.

Süreç parametrelerinin nokta tahmini

Genel olarak, bir sürecin parametreleri bilinmemektedir; ayrıca, bu parametreler genellikle zamanla değişirler. Bu nedenle, olasılık dağılımlarının parametrelerini tahmin etmek için prosedürlere ihtiyaç vardır. Parametre tahmini ve hipotez testlerinin standart istatistiksel teknikleri bu açıdan yarlıdır. Bu teknikler, istatistiksel kalite kontrol metodolojilerinin çoğu için temel oluşturur. İstatistiksel çıkarımın amacı, popülasyondan seçilen bir örneği temel alarak ilgili popülasyon hakkında sonuçlar çıkarmak veya kararlar vermektir. İstatistiksel kalite kontrolünde, olasılık dağılımı bir ürünün kritik boyutu veya üretim sürecinin arızalı fraksiyonu gibi bazı kalite özelliklerini tanımlar veya modeller. Bu nedenle, olasılık dağılımlarının parametreleri hakkında çıkarımlar yapılır. Parametreler genel olarak bilinmediğinden, bunları örnek verilerden tahmin etmek için prosedürlere ihtiyaç vardır.

Bilinmeyen bir parametrenin tahmincisi parametreye karşılık gelen bir istatistik olarak tanımlanabilir. Örnek veriden hesaplanan bir tahmin edicinin belirli bir sayısal değerine tahmin denir. Nokta tahmincisi, bilinmeyen parametrenin tahmini olarak tek bir sayısal değer üreten bir istatistiktir.

Örnek ortalaması ve varyansı $\bar{x}$ ve s^2, sırasıyla popülasyon ortalama ve varyans μ ve σ^2 parametrelerinin tarafsız tahmincileridir ve Denklem (2.12)'deki gibi ifade edilir.

$$E(\bar{x}) = \mu \text{ ve } E(s^2) = \sigma^2 \tag{2.12}$$

Örnek standart sapma s, popülasyon standart sapmasının σ tarafsız bir tahmincisi değildir. Tahminci Denklem (2.13)'teki gibi hesaplanır.

$$E(s) = c_4\sigma \tag{2.13}$$

Ek-A'daki tablo örnek boyutları $2 \leq n \leq 25$ için c_4 değerlerini verir. Standart sapmanın tarafsız bir tahmini Denklem (2.14)'ten elde edebilir:

$$\hat{\sigma} = \frac{s}{c_4} \tag{2.14}$$

İstatistiklerin kalite mühendisliği problemlerine birçok uygulamasında, standart sapmayı aralık yöntemiyle tahmin etmek uygundur. $x_1, x_2, \ldots, x_n$, ortalaması μ,

varyansı σ^2 olan bir normal dağılımdan çekilen rastgele bir örneği n gözlemleri olmak kaydıyla

$$R = \max(x_i) - \min(x_i) \tag{2.15}$$

Yani, R sadece en büyük ve en küçük örnek gözlemler arasındaki farktır. Rasgele değişken $W = R/\sigma$ göreli aralık denir. W dağılımının ortalaması alınan örneklemin büyüklüğüne bağlı olan d_2 sabitidir. Yani, $E(W) = d_2$. Bu nedenle, normal dağılımın standart sapmasının tarafsız bir tahmincisi Denklem (2.16)'daki formül ile hesaplanır.

$$\hat{\sigma} = \frac{R}{d_2} \tag{2.16}$$

Örnek boyutları $2 \leq n \leq 25$ için d_2 değerleri Ek-A'da verilmiştir.

Kontrol grafikleri

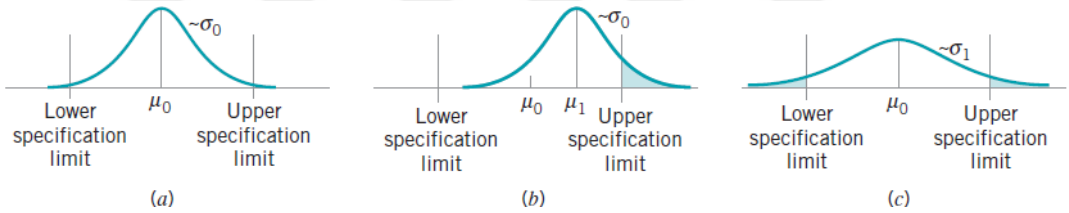
Kontrol çizelgeleri iki genel tipe ayrılabilir. Birçok kalite özellikleri ölçülebilir ve sürekli ölçekte sayılar olarak ifade edilebilir. Bu gibi durumlarda, kalite karakteristiğini merkezi eğilim ölçüsü ve değişkenlik ölçüsü ile tanımlamak uygundur. Merkezi eğilim ve değişkenlik için olan kontrol grafiklerine topluca "nicel kontrol grafikleri" denir. $\overline{X}$ grafiği, merkezi eğilimi izlemek için en yaygın kullanılan grafiktir ve örnek aralığına veya örnek standart sapmasına dayanan grafikler süreç değişkenliğini kontrol etmek için kullanılır. Birçok kalite özelliği sürekli bir ölçekte veya nicel ölçekte bile ölçülmez. Bu durumlarda, her bir ürün birimi belirli niteliklere sahip olup olmadığına bağlı olarak uygun veya uygun olmayan olarak değerlendirebilir veya bir ürün biriminde görünen uygunsuzlukların (kusurların) sayısı sayılabilir. Bu kalite karakteristikleri için kontrol grafiklerine "nitel kontrol çizelgeleri" denir. Nicel kontrol grafikleri; ortalama ($\overline{X}$) ve aralık (R), ortalama ve standart sapma (S), bireysel gözlem değerleri (I) ve hareketli aralık (MR); nitel kontrol grafikleri ise; kusurlu oranı (p), kusurlu sayısı (np), kusur sayısı(c), birim başına kusur sayısı (u) kontrol grafiklerini içerir.

Nicel kontrol grafikleri

Değişken bir kalite özelliği ile uğraşırken, genellikle hem kalite karakteristiğinin ortalama değerini hem de değişkenliğini izlemek gerekir. Süreç ortalamasının veya

ortalama kalite seviyesinin kontrolü genellikle ortalamalar için kontrol grafiği veya X kontrol grafiği ile yapılır. Süreç değişkenliği, standart sapmayı kullanan s kontrol grafiği ile ya da aralık kullanan R kontrol grafiği ile izlenebilir. R grafiği daha yaygın olarak kullanılmaktadır. Genellikle, her bir kalite karakteristiği için ayrı $\bar{x}$ ve R grafikleri korunur. $\bar{x}$ ve R (veya s) grafikleri en önemli ve kullanışlı çevrimiçi istatistiksel süreç izleme ve kontrol teknikleri arasındadır.

Hem süreç ortalaması hem de süreç değişkenliği üzerinde kontrol sağlamak önemlidir. Şekil 2.5 bir üretim sürecinin çıktısını göstermektedir. Şekil 2.5a hem ortalama μ hem de standart sapma σ nominal değerlerinde kontrol altındadır (örneğin, μ_0 ve σ_0); sonuç olarak, işlem çıktısının çoğu spesifikasyon sınırları içine düşer. Ancak, Şekil 2.5b ortalama, uygun olmayan ürünün daha yüksek bir fraksiyonuna neden olan bir $\mu_1 > \mu_0$ değerine kaymıştır. Şekil 2.5c süreç standart sapması $\sigma_1 > \sigma_0$ değerine kaymıştır. Bu aynı zamanda, süreç ortalamasının hala nominal değerde olmasına rağmen, daha yüksek süreç uygunsuzluğu (ıskarta ürün) ile sonuçlanır.



Şekil 2.5. Hem süreç ortalamasını hem de süreç değişkenliğini kontrol etme ihtiyacı. (a) Nominal seviyelerde ortalama ve standart sapma. (b) Süreç ortalaması $\mu_1 > \mu_0$. (c) Süreç standart sapması $\sigma_1 > \sigma_2$. [85]

2.3.2. $\bar{X}$ ve R kontrol grafikleri

Bir ölçüm olarak ifade edilebilen bir kalite karakteristiği ile uğraşırken, hem kalite karakteristiğinin ortalama değerini hem de değişkenliğini izlemek gelenekseldir. Ortalama kalite üzerindeki kontrol, genellikle X grafiği olarak adlandırılan ortalamalar için kontrol grafiği tarafından uygulanır. Süreç değişkenliği, popülasyon standart sapmasının nasıl tahmin edildiğine bağlı olarak bir aralık grafiği (R grafiği) veya bir standart sapma grafiği (S grafiği) tarafından kontrol edilebilir.

Bir kalite karakteristiğinin ortalaması μ ve standart sapması σ bilindiğinde ve normal dağılıma uyduğu kabul edildiğinde n örnek büyüklüğüne sahip örnek ortalaması Denklem (2.17)'deki gibidir:

$$\bar{x} = \frac{x_1 + x_2 + \ldots + x_n}{n} \tag{2.17}$$

$\bar{x}$, μ ortalama ve $\sigma_{\bar{x}} = \sigma/\sqrt{n}$ standart sapmasına sahip normal dağılıma uymaktadır. Ayrıca, herhangi bir örnek ortalamanın $1 - \alpha$ olasılıkla düşeceği üst ve alt limit değerleri Denklem (2.18)'de verilmiştir:

$$\mu + Z_{\alpha/2}\sigma_{\bar{x}} = \mu + Z_{\alpha/2}\frac{\sigma}{\sqrt{n}} \text{ ve } \mu - Z_{\frac{\alpha}{2}}\sigma_{\bar{x}} = \mu - Z_{\alpha/2}\frac{\sigma}{\sqrt{n}} \tag{2.18}$$

Bu nedenle μ ve σ biliniyorsa, Denklem (2.18), örnek ortalamaları için bir kontrol grafiğinde üst ve alt kontrol sınırları olarak kullanılabilir. $Z_{\alpha/2}$ değerini 3 ile değiştirmek gelenekseldir, böylece üç sigma limitleri kullanılır. Örnek bir ortalama bu sınırların dışına düşerse, süreç ortalamasının artık m'ye eşit olmadığını gösterir.

Uygulamada, genellikle μ ve σ bilinmez. Bu nedenle μ ve σ değerleri, sürecin kontrol altında olduğu düşünüldüğünde alınan ön örneklerden veya alt gruplardan tahmin edilmelidir. Bu tahminler genellikle en az 20 ila 25 örneklere dayanmalıdır. Her biri n adet gözleme sahip m adet örnek olduğu varsayıldığında $\overline{x_1}, \overline{x_2}, \ldots, \overline{x_m}$ her örneğin ortalaması iken süreç ortalaması yani ortalamaların ortalaması Denklem (2.19)'da verilmiştir:

$$\bar{\bar{x}} = \frac{\overline{x_1} + \overline{x_2} + \ldots + \overline{x_m}}{m} \tag{2.19}$$

$\bar{\bar{x}}$, $\bar{x}$ grafiğinin merkez çizgisi olarak kullanılır. Kontrol sınırlarını oluşturmak için, standart sapma σ tahminine ihtiyaç vardır. σ, standart sapmalardan veya örneklerin değişim aralığından hesaplanabilir. $x_1, x_2, \ldots, x_n$ n büyüklüğünde bir örnek olduğunda örnekteki en büyük ve en küçük gözlem değerlerinin farkı örneğin değişim aralığını verir.

$$R = x_{ma,x} - x_{min} \tag{2.20}$$

$R_1, R_2, \ldots, R_m$ m adet örneğin değişim aralıkları iken ortalama aralık değeri Denklem (2.21) ile hesaplanır.

$$\overline{R} = \frac{R_1 + R_2 + \cdots + R_m}{m} \tag{2.21}$$

$\hat{\sigma} = \frac{\bar{R}}{d_2}$, σ nın tarafsız tahminidir. $\bar{X}$ kontrol grafiği üzerinde kontrol sınırlarını oluşturmak için Denklem (2.22)'deki formüller kullanılır:

$$\text{ÜKL} = \bar{\bar{x}} + A_2\bar{R}$$

$$\text{OÇ} = \bar{\bar{x}} \qquad (2.22)$$

$$\text{AKL} = \bar{\bar{x}} - A_2\bar{R}$$

A_2 ve d_2 sabitleri Ek-A'da çeşitli örnek boyutları için verilmiştir.

Süreç değişkenliği, bir kontrol grafiğinde R örnek aralıklarının değerlerini çizerek izlenebilir. R grafiğinin orta çizgisi ve kontrol sınırları Denklem (2.23)'teki gibidir:

$$\text{ÜKL} = D_4\bar{R}$$

$$\text{OÇ} = \bar{R} \qquad (2.23)$$

$$\text{AKL} = D_3\bar{R}$$

D_3ve D_4 sabitleri Ek-A'da verilmiştir.

2.3.3. $\bar{X}$ ve S kontrol grafikleri

$\bar{x}$ ve R grafikleri yaygın olarak kullanılsa da, bazen R aralığının kullanımı yoluyla dolaylı olarak değil, doğrudan standart sapmayı tahmin etmek arzu edilir. Bu durumda $\bar{x}$ ve s kontrol grafikleri kullanılır, burada s örnek standart sapmadır. Genellikle aşağıdaki durumlarda $\bar{x}$ ve s grafikleri $\bar{x}$ ve R grafiklerine tercih edilir:

1. Örneklem büyüklüğü n orta derecede olduğu durumlarda, örneğin n > 10 veya 12 (σ'yı tahmin etmek için kullanılan aralık yöntemi orta ila büyük örnekler için istatistiksel verimliliği kaybettiği kaybetmektedir),
2. Örneklem büyüklüğü n değişken olduğunda.

$\bar{x}$ ve s için kontrol grafiklerinin oluşturulması ve çalıştırılması, her örnek için örnek ortalama $\bar{x}$ ve örnek standart sapması s'yi hesaplamamız dışında, $\bar{x}$ ve R grafikleri için olduğu gibi aynı adım dizisini gerektirir. Eğer σ^2 bir olasılık dağılımının bilinmeyen

varyansı ise, örneklem varyansı s^2, σ^2'nin tarafsız bir tahmincisidir ve Denklem (2.24)'teki gibi ifade edilir:

$$s^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1} \tag{2.24}$$

Bununla birlikte, örnek standart sapma s, σ nın tarafsız bir tahmincisi değildir. Tarafsızlığı sağlamak için c_4 katsayısı kullanılır. c_4 örneklem büyüklüğü n'e bağlı bir sabittir. Ayrıca, s standart sapması $\sigma\sqrt{1 - c_4^2}$'dir. Bu bilgi $\bar{x}$ ve s kontrol grafiklerini oluşturmak için kullanılabilir. Sonuç olarak, verilen σ için standart bir değere sahip s grafiğinin parametreleri Denklem (2.25)'te verilmiştir:

$$\begin{aligned} \text{ÜKL} &= B_6\sigma \\ \text{OÇ} &= c_4\sigma \\ \text{AKL} &= B_5\sigma \end{aligned} \tag{2.25}$$

B_5 ve B_6 değerleri ek Tablo VI çeşitli örnek boyutları için verilmiştir.

σ bilindiği durumda $\bar{x}$ grafiğinin parametreleri ise Denklem (2.26)'daki gibidir:

$$\begin{aligned} \text{ÜKL} &= \mu + A\sigma \\ \text{OÇ} &= \mu \\ \text{AKL} &= \mu - A\sigma \end{aligned} \tag{2.26}$$

Burada $A = 3/\sqrt{n}$ olmak üzere n değişkenine bağlı olarak Ek-A'da tanımlanmıştır.

σ bilinmediği durumlarda, geçmiş veriler analiz edilerek tahmin edilmelidir. Örneklerin her biri n boyutunda olduğu varsayıldığında ve s_i'nin i. örneğin standart sapması olduğu kabul edildiğinde m adet standart sapmanın ortalaması Denklem (2.27)'deki gibidir:

$$\bar{s} = \frac{1}{m}\sum_{i=1}^{m} s_i \tag{2.27}$$

$\bar{s}/c_4$, σ nın tarafsız tahminidir. Sonuç olarak, s grafiğinin parametreleri Denklem (2.28)'deki gibidir:

$$\text{ÜKL} = B_4\bar{s}$$

$$OÇ = \bar{s} \tag{2.28}$$

$$AKL = B_3\bar{s}$$

σ bilinmediği durumda $\bar{x}$ grafiğinin parametreleri ise Denklem (2.29)'daki gibidir:

$$ÜKL = \bar{\bar{x}} + A_3\bar{s}$$

$$OÇ = \bar{\bar{x}} \tag{2.29}$$

$$AKL = \bar{\bar{x}} - A_3\bar{s}$$

2.3.4. Bireysel gözlem değerleri için kontrol kartları

Birçok durumda, süreç kontrolü için kullanılan örnek boyutu n = 1'dir; diğer bir deyişle, örnek tek bir üniteden oluşur. Bu durumların bazı örnekleri aşağıdaki gibidir:

1. Otomatik muayene ve ölçüm teknolojisi kullanılması ve üretilen her birimin analiz edilmesi,
2. Üretim hızının çok yavaş olması ve analiz edilmeden önce n > 1 örnek büyüklüğünün birikmesine izin vermenin elverişsiz olması,
3. Süreçteki tekrar ölçümlerinin birçok kimyasal işlemde olduğu gibi sadece laboratuvar veya analiz hatası nedeniyle farklılık göstermesi.

Bu gibi durumlarda, bireysel kontrol grafiği yararlıdır. Bireyler için kontrol grafiği, süreç değişkenliğini tahmin etmek için birbirini izleyen iki gözlemin hareketli aralığını kullanır. Hareketli aralık $MR_i = |x_i - x_{i-1}|$ olarak tanımlanır ve m adet gözlem değeri için ortalama hareketli aralık Denklem (2.30)'daki gibi hesaplanır:

$$\overline{MR} = \frac{1}{m-1}\sum_{i=2}^{m}|X_i - X_{i-1}| \tag{2.30}$$

σ nın tahmini Denklem (2.31)'de verilmiştir:

$$\hat{\sigma} = \frac{\overline{MR}}{d_2} = \frac{\overline{MR}}{1.128} \tag{2.31}$$

d_2 değeri n = 2'ye karşılık gelir, çünkü her hareketli aralık iki ardışık gözlem arasındaki aralıktır. Sadece m – 1 adet hareketli aralık vardır. n = 2 için D_3 ve D_4 sabitlerini kullanarak hareketli aralık için bir kontrol grafiği oluşturmak da mümkündür. Bireyler için kontrol grafiği orta çizgisi, üst ve alt kontrol limitleri Denklem (2.31)'deki gibidir:

$$\text{ÜKL} = \bar{x} + 3\frac{\overline{MR}}{d_2} = \bar{x} + 3\frac{\overline{MR}}{1.128} = \bar{x} + 2.66\overline{MR}$$

$$\text{OÇ} = \bar{x} \tag{2.32}$$

$$\text{AKL} = \bar{x} - 3\frac{\overline{MR}}{d_2} = \bar{x} - 3\frac{\overline{MR}}{1.128} = \bar{x} - 2.66\overline{MR}$$

Hareketli aralık için Denklem (2.33)'de verilmiştir:

$$\text{ÜKL} = D_4\overline{MR} = 3.267\overline{MR}$$

$$\text{OÇ} = \overline{MR} \tag{2.33}$$

$$\text{AKL} = D_3\overline{MR} = 0$$

Bu hareketli aralık grafiği için AKL her zaman sıfırdır, çünkü n = 2 için D_3 = 0'dır.

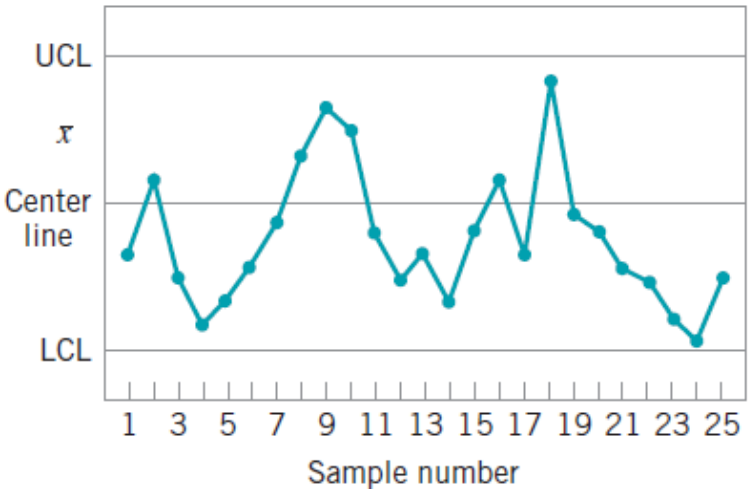
2.3.5. Kontrol grafiklerindeki desenlerin analizleri

Bir kontrol grafiği, bir veya daha fazla nokta kontrol limitlerinin ötesine geçtiğinde veya çizilen noktalar rastgele olmayan bir davranış modeli gösterdiğinde kontrol dışı bir durumu gösterebilir.

Örneğin, Şekil 2.6'teki grafikte 25 noktanın da kontrol sınırları dâhilinde olmasına rağmen, bu noktalar istatistiksel kontrolü göstermemektedir, çünkü desenleri görünüşte çok rastgele değildir. Özellikle, 25 noktanın 19'unun orta çizginin altına, sadece 6'sının yukarıdadır. Noktaların gerçekten rastgele olması, merkez çizgisinin üstünde ve altında daha eşit bir dağılım olmasını gerektirir.

Ayrıca dördüncü noktadan sonra arka arkaya beş noktanın büyüklüğünde bir artış vardır. Bu nokta düzenlemesine "koşu" denir. Benzer şekilde, azalan nokta dizisi de olabilir.

Kontrol grafiklerindeki desenlerin yorumlanması hakkında Bölüm 2.3.6'da daha ayrıntılı bilgi verilmiştir.



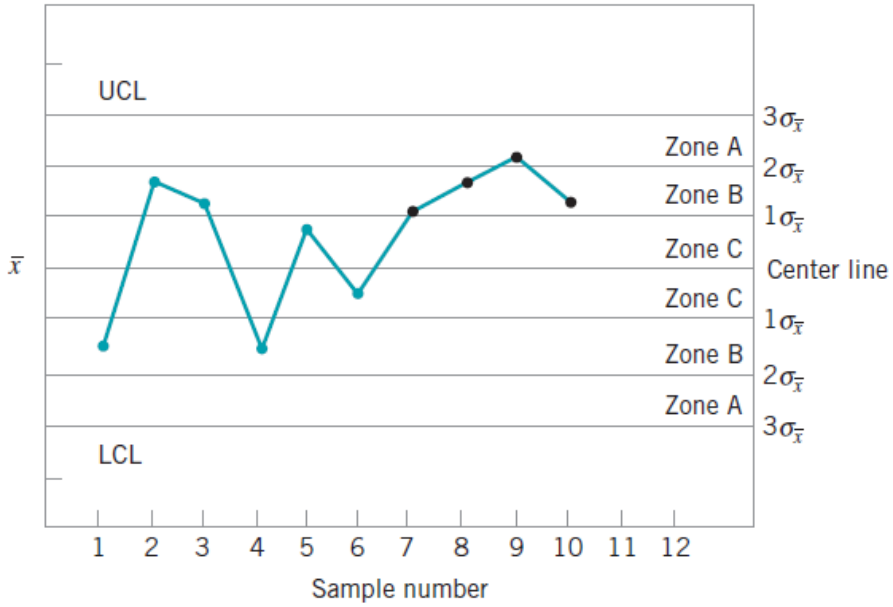
Şekil 2.6. Örnek bir kontrol kartı [85]

Western Elektrik El Kitabı [86], kontrol grafiklerinde rastgele olmayan kalıpları tespit etmek için bir dizi karar kuralı önerir. Özellikle, aşağıdaki durumlarda sürecin kontrol dışı olduğu sonucuna varılmasını önerir:

1. Üç sigma kontrol limitlerinin dışındaki bir nokta,
2. Art arda üç noktadan ikisinin, iki sigma uyarı limitlerinin ötesinde bulunması,
3. Birbirini izleyen beş noktadan dördünün orta çizgiden tek sigma veya daha uzak mesafede olması,
4. Orta çizginin bir tarafına ardışık sekiz nokta bulunması.

Bu kurallar her seferinde orta çizginin bir tarafı için geçerlidir. Bu nedenle, üst uyarı limitinin üstündeki bir nokta ve hemen alt uyarı limitinin altındaki bir nokta kontrol dışı bir alarmı işaret etmez. Bunlar genellikle pratikte kontrol grafiklerinin hassasiyetini arttırmak için kullanılır. Bu kuralların kullanılması, genel üç sigma kontrol limiti ihlali olması durumuna göre daha küçük süreç kaymalarının daha hızlı tespit edilmesine izin verebilir.

Şekil 2.7, Western Elektrik El Kitabında kullanılan tek sigma, iki sigma ve üç sigma limitlerine sahip bir $\bar{x}$ kontrol grafiğini göstermektedir. Bu limitler, kontrol grafiğini orta çizginin her iki tarafında üç bölgeye (A, B ve C) bölmektedir. Sonuç olarak, Western Elektrik kurallarına bazen kontrol grafikleri için bölge kuralları denir. Son dört nokta B bölgesinde veya ötesindedir. Birbirini takip eden beş noktanın dördü bir sigma uzaklığından fazla olduğu için Western Elektrik prosedürü, desenin rastgele olmadığı ve sürecin kontrol dışı olduğu sonucuna varır.



Şekil 2.7. Western Electric kuralları-son dört nokta kural 3 ihlali [85]

Sürecin kontrol dışı olup olmadığını belirlemek için bir kontrol grafiğine aynı anda birkaç kriter uygulanabilir. Temel kriter kontrol sınırlarının dışında bir veya daha fazla noktadır. Tamamlayıcı kriterler bazen kontrol grafiklerinin küçük bir işlem kaymasına duyarlılığını arttırmak için kullanılır, böylece özel nedene daha hızlı yanıt verebilir. Uygulamada yaygın olarak kullanılan kontrol grafikleri için bazı duyarlılaştırma kuralları Tablo 2.3'te gösterilmiştir. Tablo 2.3'teki kriterlerden bir veya daha fazlasının karşılanması durumunda sürecin kontrol dışı olduğu sonucuna varılır.

Tablo 2.3. Shewhart kontrol grafikleri için bazı hassaslaştırıcı kurallar

1	Kontrol sınırları dışında bir veya daha fazla nokta
2	Arka arkaya üç ardışık noktadan ikisinin iki sigma uyarı limitleri dışında ancak yine de kontrol limitleri içerisinde olması
3	Ardışık beş noktanın dördü tek sigma sınırlarının dışında olması
4	Orta çizginin bir tarafında sekiz ardışık nokta koşusu
5	Arka arkaya altı noktanın sürekli artması veya azalması
6	C bölgesinde arka arkaya on beş nokta (orta çizginin hem üstü hem de altında)
7	Yukarı ve aşağı sırayla on dört nokta

Tablo 2.3.(Devam) Shewhart kontrol grafikleri için bazı hassaslaştırıcı kurallar

8	C bölgesinde hiç nokta olmaması kaydıyla orta çizginin her iki yanında arka arkaya sekiz nokta
9	Verilerde olağandışı veya rasgele olmayan bir desen
10	Uyarı veya kontrol sınırına yakın bir veya daha fazla nokta

Bu hassaslaştırıcı kuralların birçoğu aynı anda uygulandığında, genellikle kontrol dışı sinyallere dereceli bir yanıt kullanılır. Örneğin, bir nokta bir kontrol limitini aşarsa, hemen özel nedeni aranmaya başlanır, ancak bir veya iki ardışık nokta yalnızca iki sigma uyarı limitini aşarsa, örnekleme sıklığı artırılabilir. Bu uyarlanabilir örnekleme yanıtı, özel bir neden için tam bir arama kadar önemli olmayabilir, ancak süreç gerçekten kontrol dışıysa, bu durumu daha hızlı tespit etme olasılığını artırır.

2.3.6. $\bar{\bar{X}}$ ve R grafiklerinin yorumlanması

Çizilen noktaların deseni rastgele olmayan veya sistematik davranış gösteriyorsa, hiçbir nokta kontrol limitlerinin dışında çıkmasa da, bir kontrol grafiği kontrol dışı bir durumu gösterebilir. Çoğu durumda, çizilen noktaların deseni, işlem hakkında yararlı tanı bilgileri sağlar ve bu bilgi, değişkenliği azaltan süreç modifikasyonları yapmak için kullanılabilir (istatistiksel süreç kontrolünün amacı). Ayrıca, bu desenler Faz I'de oldukça sık görülür (geçmiş verilerin retrospektif çalışması) ve bunların ortadan kaldırılması, bir süreci kontrol altına almak için çok önemlidir.

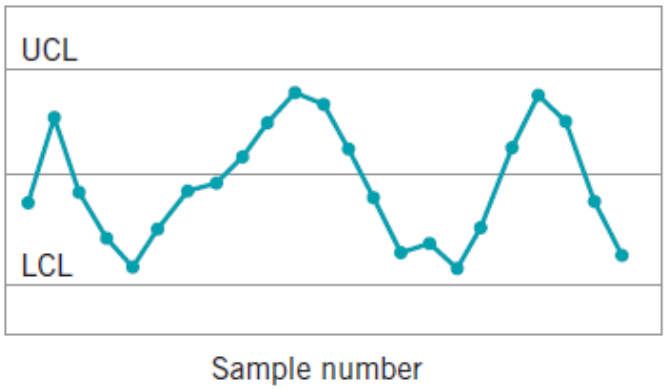
Bu bölümde $\bar{x}$ ve R grafiklerinde görülen bazı daha yaygın desenlere ve desen üretebilecek bazı süreç özelliklerine odaklanılarak kontrol grafikleri yorumlanmıştır. $\bar{x}$ ve R grafiklerinin etkili bir şekilde yorumlanabilmesi için analistin hem kontrol grafiği altında yatan istatistiksel prensiplere hem de sürecin kendisine aşina olması gerekir.

$\bar{x}$ grafiğindeki desenler yorumlanırken, önce R grafiğinin kontrol altında olup olmadığı tespit edilmelidir. Bazı özel nedenler hem $\bar{x}$ hem de R grafiklerinde görünür. Hem $\bar{x}$ hem de R grafikleri rasgele olmayan bir desen gösteriyorsa, en iyi strateji önce R grafiği özel nedenlerini ortadan kaldırmaktır. Çoğu durumda, bu otomatik olarak $\bar{x}$

grafiğinde rasgele olmayan deseni ortadan kaldırır. R grafiği bir kontrol dışı koşulu gösterdiğinde asla grafik yorumlanmamalıdır.

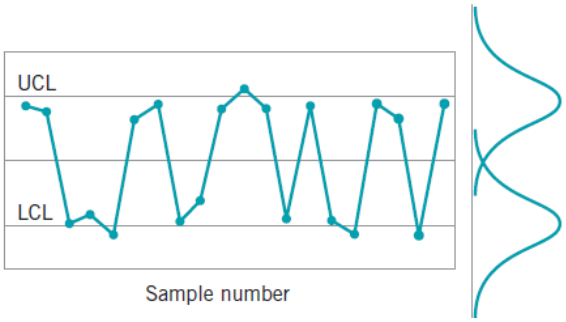
Kontrol kartlarında aşağıdaki desenler bulunabilir:

1. Döngüsel desenler: $\bar{x}$ grafiğindeki böyle bir desen sıcaklık, operatör yorgunluğu, operatörlerin ve / veya makinelerin düzenli rotasyonu veya voltaj veya basınçtaki dalgalanma veya üretim ekipmanındaki başka bir değişken gibi sistematik çevresel değişikliklerden kaynaklanabilir. R grafikleri bazen bakım programları, operatör yorgunluğu veya takım aşınması nedeniyle döngüler ortaya çıkar ve aşırı değişkenliğe neden olur. Tipik bir örnek Şekil 2.8'de gösterilmektedir.



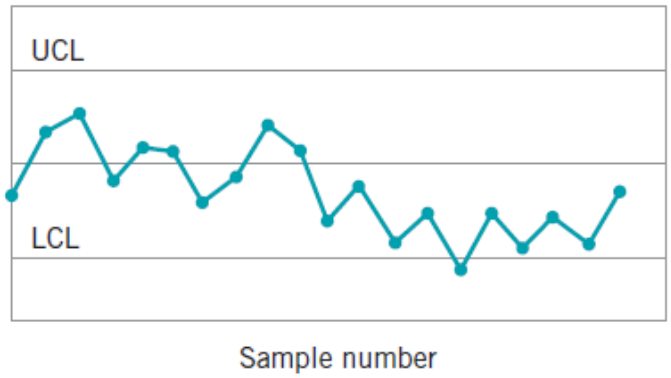
Şekil 2.8. Kontrol grafiğinde döngüler [85]

2. Karışım: Çizilen noktalar, Şekil 2.9'de gösterildiği gibi orta çizginin yakınında nispeten az nokta olacak şekilde kontrol limitlerinin yakınına veya biraz dışına düşme eğilimi gösterdiğinde bir karışım meydana gelir. Süreç çıktısını üreten iki (veya daha fazla) örtüşen dağılım tarafından bir karışım deseni oluşturulur. Bazen karışımlar, operatörlerin süreç ayarlamalarını çok sık yaptıkları ve sistematik nedenlerden ziyade çıktıdaki rastgele değişime yanıt verdiği "aşırı kontrol" den kaynaklanır. Birkaç kaynaktan (paralel makineler gibi) çıkan ürün, daha sonra süreç izleme amacıyla örnek toplanan ortak bir akıma beslendiğinde bir karışım deseni de meydana gelebilir.



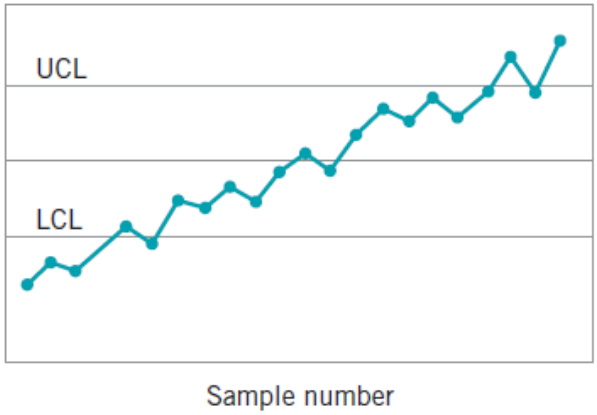
Şekil 2.9. Karışım deseni [85]

3. Süreç seviyesinde değişim: Süreç seviyesinde bir değişiklik Şekil 2.10'da gösterilmektedir. Bu değişimler yeni işçilerin getirilmesinden; yöntem, hammadde veya makinelerde değişikliklerden; denetim yönteminde veya standartlarında bir değişiklikten veya operatörlerin beceri, dikkat veya motivasyonlarındaki bir değişikliklerden kaynaklanabilir. Bazen, işçileri etkileyen motivasyon faktörleri nedeniyle, bir kontrol grafiği programının uygulanmasının ardından süreç performansında bir iyileşme kaydedilmektedir.



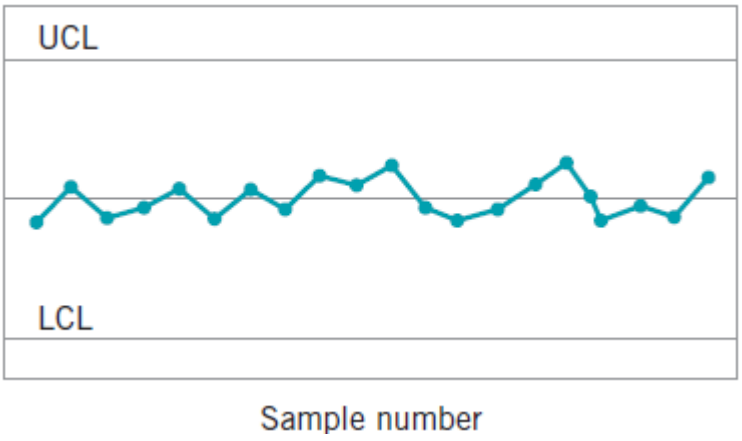
Şekil 2.10. Süreç seviyesinde değişim [85]

4. Trend: Bir eğilim veya bir yönde sürekli hareket, Şekil 2.11'daki kontrol grafiğinde gösterilmiştir. Eğilimler genellikle bir aletin veya başka bir kritik süreç bileşeninin kademeli olarak aşınması veya bozulmasından kaynaklanır. Kimyasal işlemlerde genellikle bir karışımın bileşenlerinin çökmesi veya ayrılması nedeniyle oluşurlar. Ayrıca operatör yorgunluğu veya denetimin varlığı gibi insan nedenlerinden de kaynaklanabilirler. Son olarak, eğilimler sıcaklık gibi mevsimsel etkilerden kaynaklanabilir.



Şekil 2.11. Süreç seviyesinde trend [85]

5. Katmanlaşma: Katmanlaşma veya noktaların orta çizgi etrafında yapay olarak kümelenme eğilimi Şekil 2.12'de gösterilmektedir. Bu desende belirgin bir doğal değişkenlik eksikliği vardır. Katmanlaşmanın olası nedenlerinden biri, kontrol limitlerinin yanlış hesaplanmasıdır. Bu desen her bir alt grup içindeki birkaç farklı temel dağılımdan bir veya daha fazla birim topladığında da ortaya çıkabilir.



Şekil 2.12. Katmanlaşma deseni [85]

Kontrol grafiğinin faz I ve faz II uygulamaları

Standart kontrol grafiği kullanımı, iki farklı amaçla faz I ve faz II uygulamalarını içerir. Faz I'de, bir dizi süreç verisi bir defada geriye dönük olarak analiz edilerek toplanır ve analiz edilir; gelecekteki üretimi izlemek için verilerin toplandığı süre boyunca sürecin kontrol altında olup olmadığını belirlemek ve kontrol limitlerinin güvenilir olup olmadığını görmek için deneme kontrol sınırları oluşturulur. Bu, genellikle kontrol grafikleri herhangi bir sürece uygulandığında yapılan ilk şeydir. Faz I'deki kontrol grafikleri öncelikle operasyonel personele süreci istatistiksel kontrol

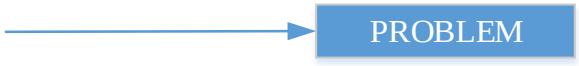
durumuna getirme konusunda yardımcı olur. Kararlı koşullar altında toplanan ve kontrol durumunda süreç performansını temsil eden bir “temiz” süreç verisi setine sahip olduktan sonra Faz II başlar. Faz II'de, süreçten alınan her ardışık örnek için oluşturulan örnek istatistikleri kontrol grafiğindeki üst ve alt kontrol limitleri ile karşılaştırılarak süreç izlemesi yapılır.

2.3.7. Neden-sonuç diyagramı

Kaoru Ishikawa tarafından 1943 yılında geliştirilen bu araca sık sık onuruna Ishikawa diyagramı denir. Amacı, bir sorunun temel nedenine ilişkin çeşitli teorilerin ilişkilerini organize etmek ve sergilemektir. Diyagram, belirli bir sorunun olası nedenlerine yapılandırılmış, sistematik bir şekilde odaklanarak, problem çözme ekibinin bu potansiyel nedenler hakkındaki düşüncelerini netleştirmesini ve ekibin gerçek kök nedenleri keşfetmesi için daha verimli çalışmasını sağlar [87]. Oluşturulan bu diyagram bir balığın kılçığını andırdığı için "Balık Kılçığı Diyagramı" olarak da bilinmektedir.

Kontrol grafiklerinde bir özel neden tespit edildikten ve ileri çalışmalar için izole edildikten sonra, bu istenmeyen etkinin potansiyel nedenleri analiz edilmelidir. Nedenlerin belirgin olmadığı durumlarda, neden-sonuç diyagramı potansiyel nedenlerin ortaya çıkarılmasında sıklıkla yararlı olan bir araçtır. Neden-sonuç diyagramını oluşturma adımları aşağıdaki gibidir:

1. Analiz edilecek problem veya etki tanımlanır. Problem belirlendikten sonra Şekil 2.13’deki gösterildiği gibi problem sağda bir kutu içine yerleştirilir ve ana omurgaya doğru kalın bir çizgi çizilir.

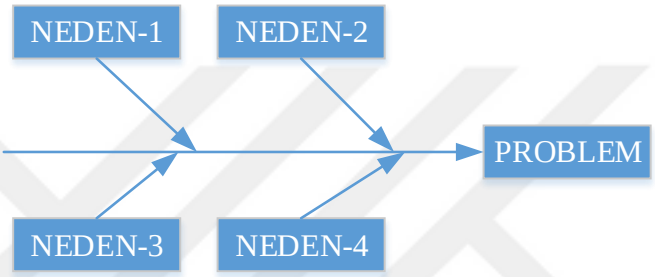


Şekil 2.13. Neden–sonuç diyagramı- aşama 1

2. Analizi yapmak için bir ekip oluşturulur. Genellikle ekip beyin fırtınası yoluyla olası nedenleri ortaya çıkarır. Beyin fırtınasına, problem ile ilgili konularda uzman personel seçilir. Seçilen personel çevresel faktörleri de gözeterek probleme etki eden nedenleri belirlerler. Nedenleri çözümüne yönelik araştırma yaparlar. Bunun için beyin fırtınası yöntemi ile probleme ait fikirler bir liste haline getirilir.

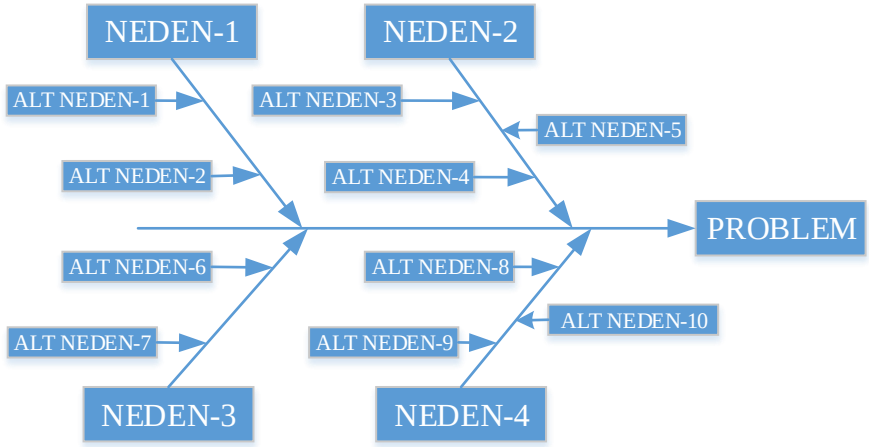
Fikirlerin ilk aşamada eleştirilmemesi en önemli gerekliliktir. Belirlenen fikirler kendi aralarında değerlendirilir ve önemli fikirler üzerine yoğunlaşılır. Nedenler balık kılçığına uygulanırken ana nedenler büyük oklarla alt nedenler ise bu oklara bağlı küçük oklarla gösterilir [88].

3. Ana potansiyel neden kategorileri belirlenir. Şekil 2.14'te gösterildiği gibi ana potansiyel nedenlerin her biri (ikiden az ve normalde beşten fazla değil) bir kutuya yerleştirilir ve yaklaşık 70 derecelik bir açılık bir çizgi ile merkezi omurgaya bağlanır.



Şekil 2.14. Neden-sonuç diyagramı-aşama 2

4. Beyin fırtınasına katılan kişiler ana nedenlerin alt nedenlerini belirler. Bu alt nedenler, ana nedenlerin çizgilerine yatay çizgilerle bağlanır. Böylece gerçek nedenlere ulaşılabilme imkânı elde edilir. Bu aşamadan sonra oluşan diyagramın durumu Şekil 2.15'te gösterilmektedir.



Şekil 2.15. Neden-sonuç diyagramı-aşama 3

5. Problemi etkileme olasılığı en yüksek olanları belirlemek için nedenler sıralanır. Belirlenen ana nedenlere bağlı alt nedenler beyin fırtınasına katılan kişiler

tarafından değerlendirilir. Kişiler alt nedenlere kendi önem sıralarına uygun puan verir. Her alt nedene verilen puanların toplamı hesaplanır. En çok puana sahip alt neden bulunur. Kök neden analizine göre problemin kaynak noktası en yüksek puana sahip alt nedendir.

6. En yüksek puana sahip alt neden belirlendikten sonra nedeni ortadan kaldırmaya yönelik düzeltici ve iyileştirici faaliyetler yapılır. Süreç ilerledikçe, diyagramın güncelleştirilmesi gerekir.

Neden-sonuç analizi son derece güçlü bir araçtır. Oldukça ayrıntılı bir neden-sonuç diyagramı etkili bir sorun giderme yardımı olarak kullanılabilmektedir.

2.4. Veri Toplama Yöntemi

Proje başarı faktörlerini belirleme çalışmaları ve proje performans değerlendirme modeli için süreç metrik ölçüm aşamaları, ISO 9001:2015 kalite yönetim sistemi ve CMMI DEV Seviye 3 kalite belgelerine sahip savunma ve sivil alanlarda sistem ve yazılım geliştiren bir kurumun proje verileri kullanılarak gerçekleştirilmiştir.

Proje başarısına etki ettiği düşünülen yazılım yaşam döngüsü süreç metriklerini tanımlama aşamasında bu kurumun proje geliştirme, test, kalite ve konfigürasyon yönetim ekipleri ile birlikte çalışılmıştır. Benzer alanlarda faaliyet gösteren diğer kurumların proje geliştirme, süreç iyileştirme, test, kalite ve konfigürasyon ekiplerinin katılımı ile bir çalıştay düzenlenmiştir. Yazılım yaşam döngüsündeki her bir süreç için çalıştayda bir alt çalışma ekibi kurulmuştur. Her ekip sorumlusu olduğu sürecin proje başarısına etki eden süreç metriklerini belirlemiştir. Belirlenen süreç metriklerinin amacı, tanımı, ölçme ve saklama yöntemi ile veri analizi yöntemi belirlenerek kayıt altına alınmıştır. Belirlenen süreç metrikleri diğer süreç ekipleri tarafından da gözden geçirilerek süreçlerin birbirleri ile uyumluluğu doğrulanmıştır. Toplam iki hafta süren çalıştay sonucunda proje başarısına etki eden süreç metrikleri ile ilgili detay bilgiler Tablo 2.4'te verilmiştir. Süreç metriklerinin ölçümleme aşamasında da aynı kurumun verileri ile çalışma sürdürülmüştür. Bu kuruma ait toplam 25 yazılım projesi kullanılarak analizler gerçekleştirilmiştir. Bu kurumun özellikleri aşağıda verilmiştir:

- Çalışan sayısı: 450

- Yazılım alanında çalışan sayısı: 390
- Kurumda CMMI DEV Seviye 3 kalite yönetim sistemi gereği süreç iyileştirme çalışmaları kapsamında her proje sonunda öğrenilmiş dersler kayıt altına alınmaktadır. Müşteriden gelen sorunlar, proje hata yönetim sistemine kaydedilmekte ve sorunlar çözülene kadar takip edilmektedir.
- Kurumda proje aşamaları çağlayan proje yaşam döngüsü modeline göre tanımlanmış, yazılım geliştirme aşamasında yinelenen artırımlı geliştirme metodolojileri kullanılmaktadır.
- Yinelenen artırımlı geliştirme metotları kapsamında proje yazılım geliştirme planında tanımlanan sürüm planına göre belirlenen tarihlerde yazılım ürünün ara sürümleri üretilmekte ve müşteriye teslim edilmektedir. Proje sonunda proje sözleşmesine uygun olarak yazılım ürününün son sürümü çıkarılarak müşterinin katılımı ile kabul testine tabi tutulmaktadır. Kabul testi laboratuvar ve/veya müşteri ortamında gerçekleştirilmektedir.
- C, C++, C#, Java, .Net ile geliştirme yapmaktadır.
- SQL Server ve Oracle platformlarını kullanmaktadır.

Tablo 2.4. Yazılım yaşam döngüsü süreç metrikleri

Süreç Adı	Metrik Adı
Proje Yönetimi - PY	PY 1.1. Kilometre taşı sapması PY 1.2. Efor sapması PY 1.3. Gelir/gider endeksi
Gereksinim Yönetimi - GY	GY 1.1. Proje başına ortalama gereksinim değişikliği sayısı
Geliştirme - GEL	GEL 1.1. Müşteri tarafından raporlanan hata açık kalma süresi GEL 1.2. Müşteri tarafından raporlanan iyileştirme açık kalma süresi GEL 1.3. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma süresi GEL 1.4. Müşteri tarafından raporlanan hata açık kalma oranı

Tablo 2.4.(Devam) Yazılım yaşam döngüsü süreç metrikleri

Süreç Adı	Metrik Adı
	GEL 1.5. Müşteri tarafından raporlanan iyileştirme açık kalma oranı
	GEL 1.6. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı
	GEL 1.7. Hata yoğunluğu
Doğrulama ve Geçerleme - DG	DG 1.1. Gözden geçirme hazırlık hızı
	DG 1.2. Bulgu yoğunluğu
	DG 1.3. Gözden geçirme hazırlık verimliliği
	DG 1.4. Hata yakalama yüzdesi
	DG 1.5. Hata tespit yüzdesi
Kalite Güvence - KG	KG 1.1. Süreç denetim planlamasından sapma
	KG 1.2. Proje uygunsuzluklarının ortalama açık kalma süresi
Konfigürasyon Yönetimi - KY	KY 1.1. Konfigürasyon denetim planlamasından sapma
	KY 1.2. Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı
	KY 1.3. Konfigürasyon Uygunsuzluklarının Açık Kalma Süresi
	KY 1.4. Konfigürasyon Uygunsuzluklarının Açık Kalma Oranı
Risk Yönetimi - RY	RY 1.1. Risk gözden geçirme oranı

Tablo 2.4'te belirtilen süreçlerin yazılım yaşam döngüsü içerisindeki sırası ve birbirleri ile etkileşim haritası Bölüm 2.7 Yazılım Yaşam Döngüsü ve Süreç Metrikleri kısmında verilmiştir.

2.5. Verilerin Toplandığı Araçlar

Süreç metrik ölçümünün gerçekleştirildiği kurumdaki projelerde kullanılan yazılım geliştirme destek araçları listesi Tablo 2.5'te verilmiştir.

Tablo 2.5. Proje geliştirme destek araçları

Süreç	Araç
Proje planlama ve iş yönetimi	Atlassian JIRA
Proje planlama ve iş yönetimi	Atlassian Confluence IBM Rational Team Concert SAP and MS Project
Gereksinim yönetimi	IBM DOORS IBM DOORS Next Generation Atlassian JIRA
Analiz ve tasarım	Enterprise Architect IBM Rational Rhapsody IBM Rational Software Architect Altium Designer
Kodlama	Eclipse, Visual Studio
Birim test	Junit Google Code Pro Quick Test Professional
Derleme ve entegrasyon	Atlassian Bamboo Jenkins Ant IBM Rational Team Concert
Test yönetimi	Rational Quality Manager
Konfigürasyon yönetimi	IBM Rational Team Concert SVN (Subversion) GIT
Aktivite yönetimi	Atlassian JIRA IBM Rational Team Concert
Gözden geçirme	SmartBear Collaborator
Yazılım kalite değerlendirme	Understand Sonar Source

Süreç metrik değerleri, proje geliştirme araçlarında iş akışlarının çalıştırılması sonucunda üretilir. Projede gerçekleştirilen süreç iş akışları bir başka deyişle proje faaliyetleri ile ilgili metrikler ve araçlar arasındaki ilişki Tablo 2.6'da verilmiştir.

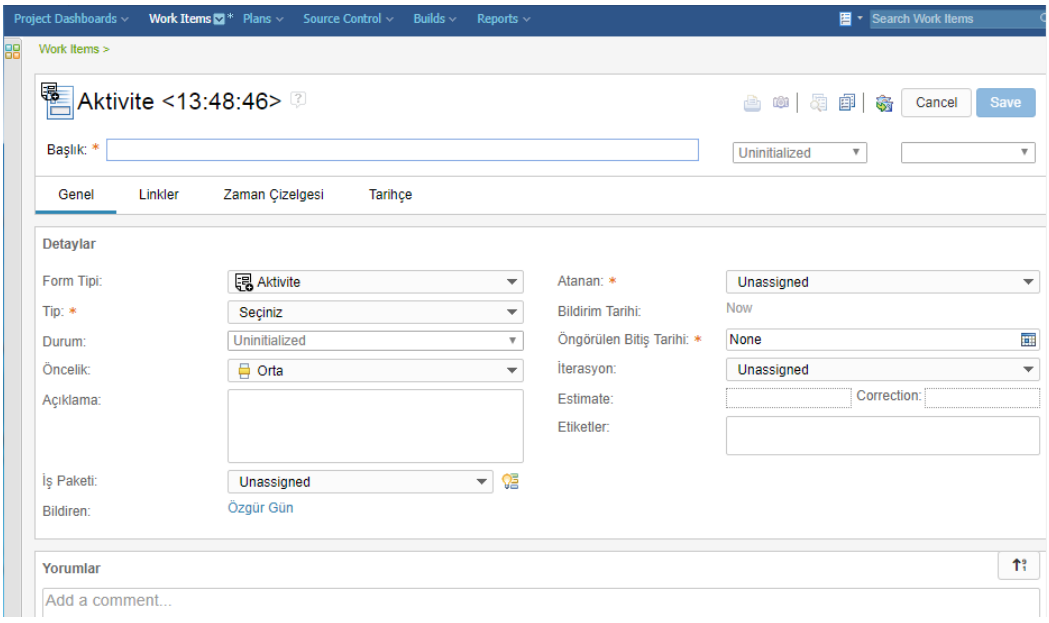
Tablo 2.6. Proje süreci iş akışları ve ilgili araçlar

İş akışı	İlgili araç	İlgili süreç metriği
İş aktiviteleri	IBM Rational Team Concert, Atlassian JIRA	Kilometre taşı sapması Efor sapması Gelir/gider endeksi
Değişiklik istekleri	IBM Rational Team Concert, Atlassian JIRA	Müşteri tarafından raporlanan gereksinim değişikliklerinin açık kalma süresi ve oranı Proje başına ortalama gereksinim değişikliği sayısı
Hatalar	IBM Rational Team Concert, Atlassian JIRA	Müşteri tarafından raporlanan hata açık kalma süresi ve oranı Müşteri tarafından raporlanan iyileştirmelerin açık kalma süresi ve oranı Hata yakalama yüzdesi Hata tespit yüzdesi Hata yoğunluğu
Riskler	IBM Rational Team Concert, Atlassian JIRA	Risk gözden geçirme oranı
Süreç denetimleri	IBM Rational Team Concert, Atlassian JIRA	Süreç denetim planlamasından sapma Konfigürasyon denetim planlamasından sapma
İş ürünü konfigürasyon denetimleri	IBM Rational Team Concert, Atlassian JIRA	İş ürünü konfigürasyon denetim planından sapma

Tablo 2.6.(Devam) Proje süreci iş akışları ve ilgili araçlar

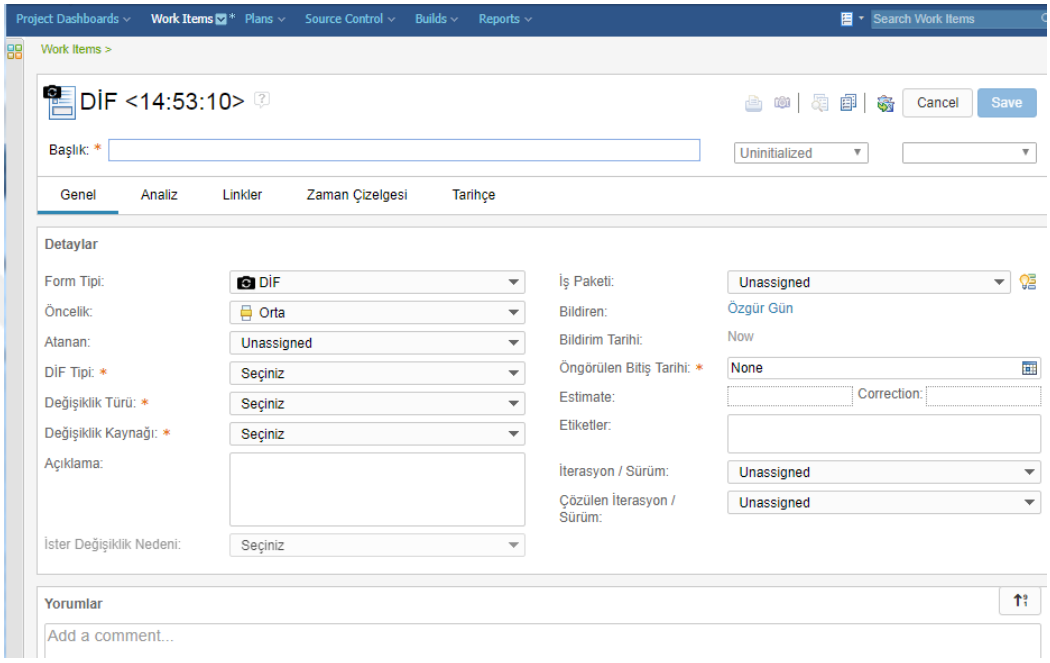
İş akışı	İlgili araç	İlgili süreç metriği
Gözden geçirmeler	SmartBear Collaborator	Gözden geçirme hazırlık hızı Bulgu yoğunluğu Hazırlık verimliliği
Düzeltici faaliyetler	IBM Rational Team Concert, Atlassian JIRA	Proje uygunsuzluklarının ortalama açık kalma süresi Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı Konfigürasyon Uygunsuzluklarının Açık Kalma Süresi Konfigürasyon Uygunsuzluklarının Açık Kalma Oranı

Tablo 2.6'da belirtilen iş akışları, projede kullanılan yazılım geliştirme destek araçları üzerinde geliştirilen formlar ile işletilmektedir. Projelerin bu süreçlere ait iş akışlarını işletmeleri sonucu ölçüm yapılacak süreç metrikleri oluşmaktadır. Çalışmanın yürütüldüğü Ar-Ge kurumunda kullanılan iş akış formları hakkında kısa bilgiler sırasıyla aşağıdaki şekillerde belirtilmiştir. Şekiller kurumda kullanılan IBM Rational Team Concert aracından alınmıştır.



Şekil 2.16. İş aktivite formu

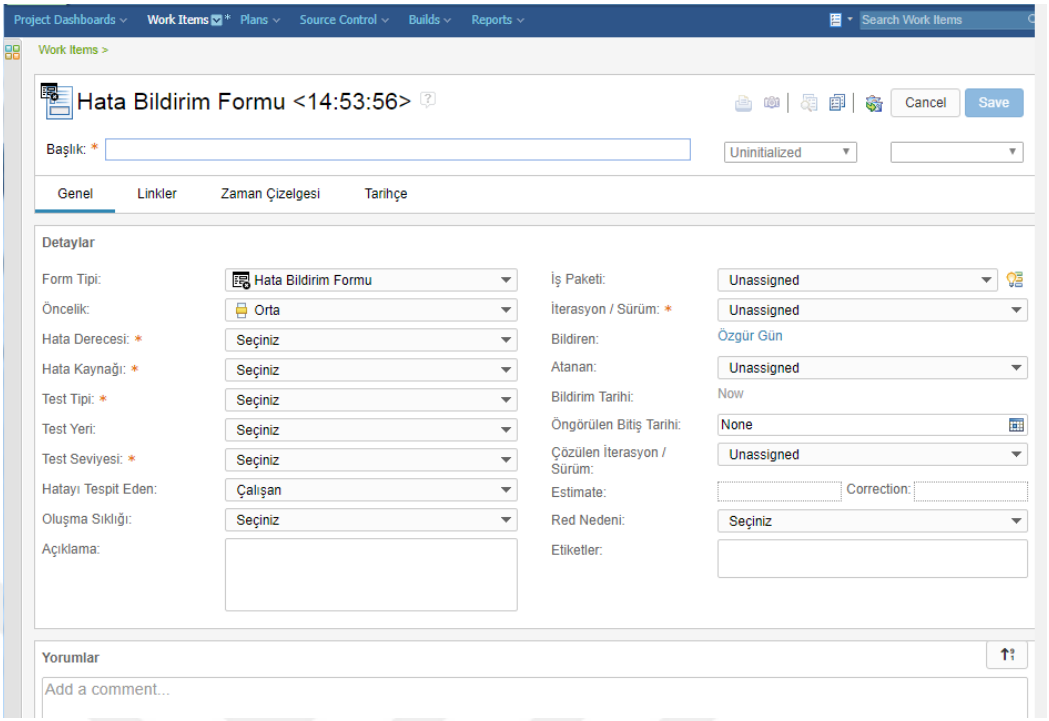
Şekil 2.16'daki aktivite formunda, projede yapılan işlerin açıklaması, işi kimin tanımladığı, işin atandığı personel bilgisi, işin ne zaman açıldığı, işin öngörülen bitiş tarihi, işin önceliği, işin bağlı bulunduğu proje iş paketi ve işin tamamlanma durumu bilgileri yer almaktadır. Projede teknik olarak yapılması gereken tüm faaliyetler aktivite formu ile tanımlanmaktadır.



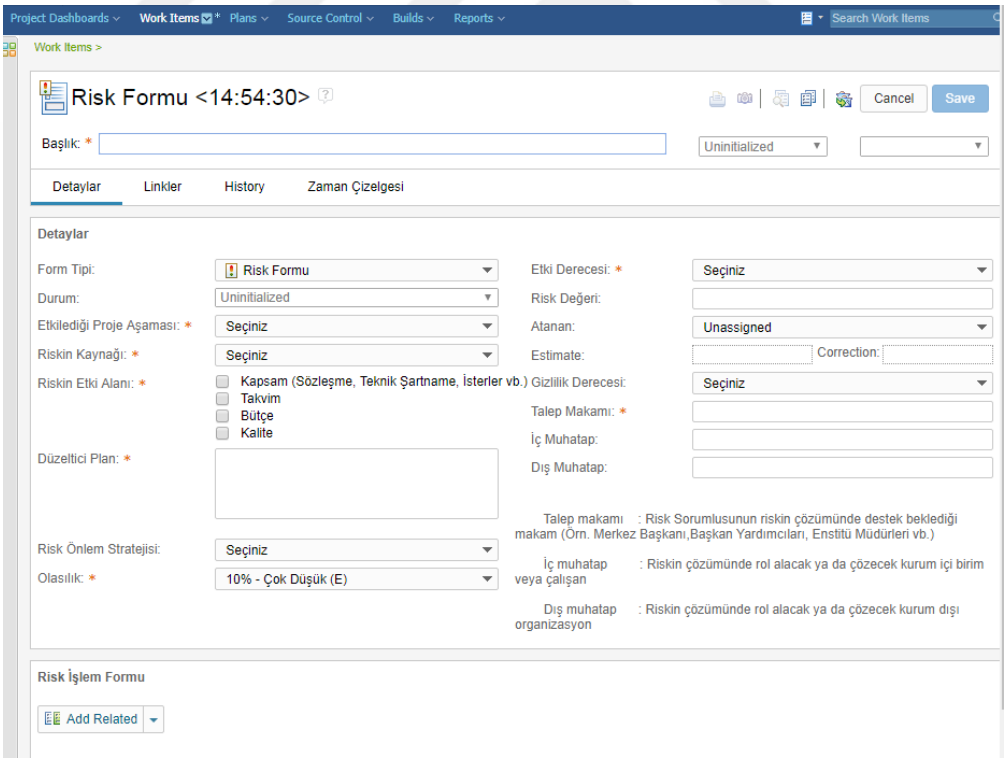
Şekil 2.17. Değişiklik istek formu

Şekil 2.17'deki değişiklik istek formu, projede dayanak alınmış/müşteriye teslim edilmiş iş ürünlerinde gelen değişiklik isteklerini kayıt altına almak için kullanılır. Formda bulunan alanlar; değişiklik isteğinin açıklaması, isteğin kimden geldiği, değişikliğin etkilediği diğer iş ürünleri bilgisi, değişiklik isteğini gerçekleştirecek personel ismi, isteğin ne zaman açıldığı, değişikliğin öngörülen bitiş tarihi, değişiklik isteğinin önceliği, değişiklik isteğinin geldiği iş ürününün bağlı olduğu iş paketi ve değişiklik isteğinin tamamlanma durumu bilgileri yer almaktadır.

Şekil 2.18'deki hata bildirim formunda, yazılım testlerinde hata bulunduğunda bu hataya ilişkin; hatanın tanımı, hatanın kimin tarafından ve ne zaman bulunduğu bilgisi, hatanın önemi, kaynağı, test tipi, test yeri, hatanın önceliği, hata oluşma sıklığı, hatanın çözülmesi için öngörülen bitiş tarihi, hatanın kapatılma durumu bilgileri yer almaktadır. Yazılım testleri birim test, bileşen testi, entegrasyon testi, performans testi, işlevsel test, ara yüz testi, saha kabul testi şeklinde farklı tiplere sahiptir.



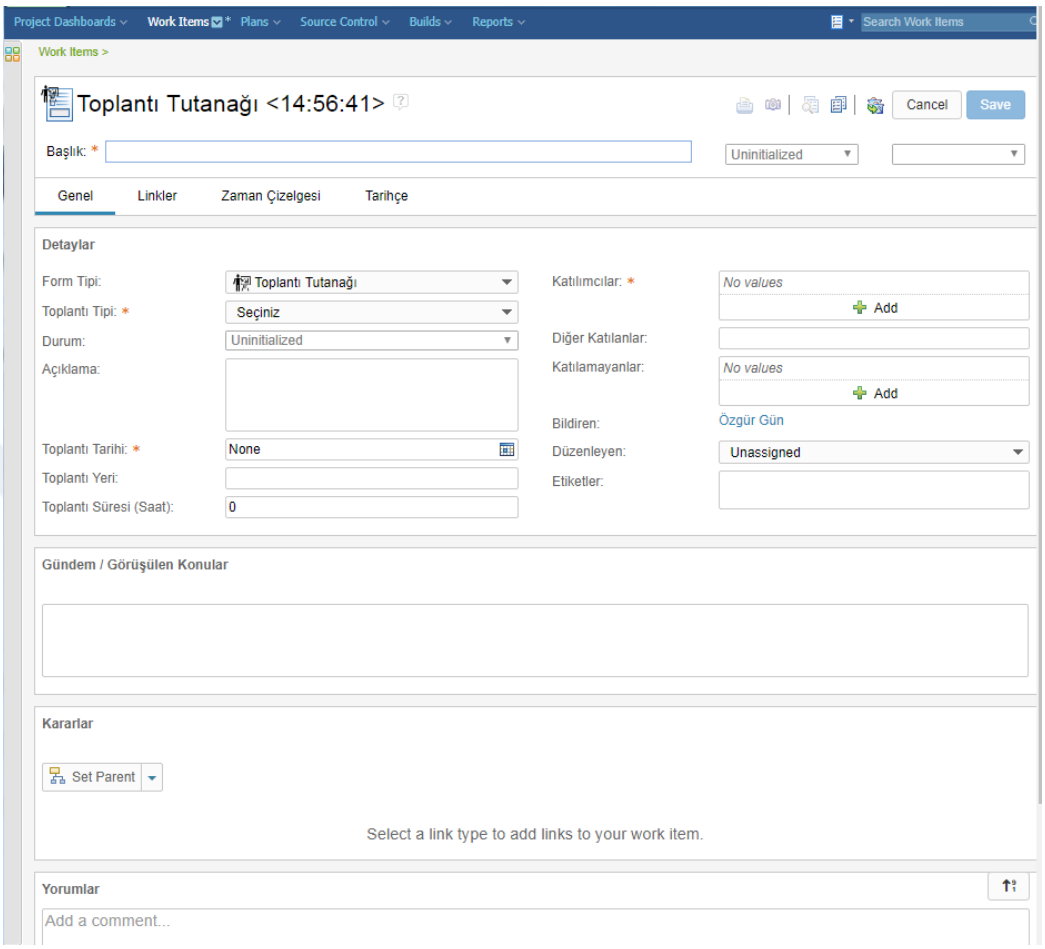
Şekil 2.18. Hata bildirim formu



Şekil 2.19. Risk formu

Şekil 2.19'daki risk formu projeyi etkileyebilecek tehdit ve fırsatların belirlenmesi ve yönetilmesi amacıyla kullanılır. Formda; riskin tanımı, riskin etkilediği proje aşaması, riskin kaynağı, etki alanı, etki ve olasılık değerleri, risk değeri, risk yanıt stratejisi,

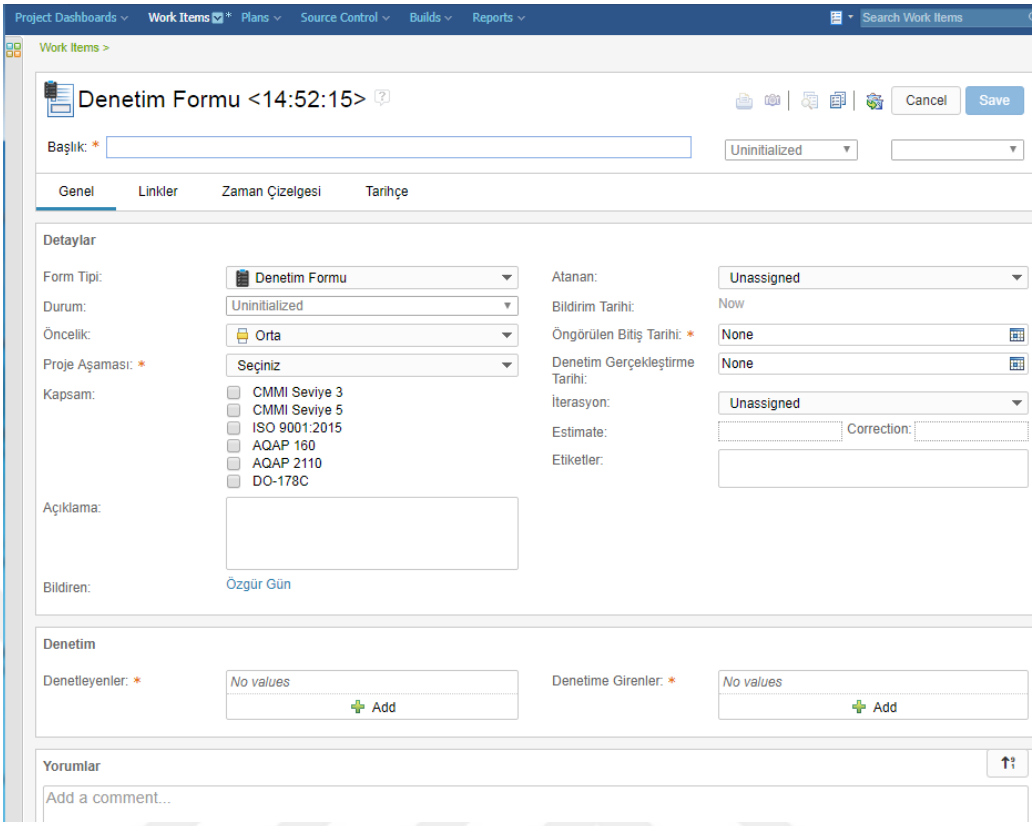
riski azaltma faaliyetleri, riskin sorumlusu, riskin etkilediği proje iş paketi ve riskin hangi aşamada olduğunu belirten durum bilgileri yer almaktadır.



Şekil 2.20. Toplantı tutanağı formu

Şekil 2.20'deki toplantı tutanağı formunda; projede yapılan toplantılarda alınan kararların kayıt altına alınması ve takip edilmesi amacıyla toplantının amacı, tipi, yeri, saati, toplantıya katılanların isimleri, görüşülen konular ve alınan kararlar ve kararların tamamlanma durumu bilgileri yer almaktadır.

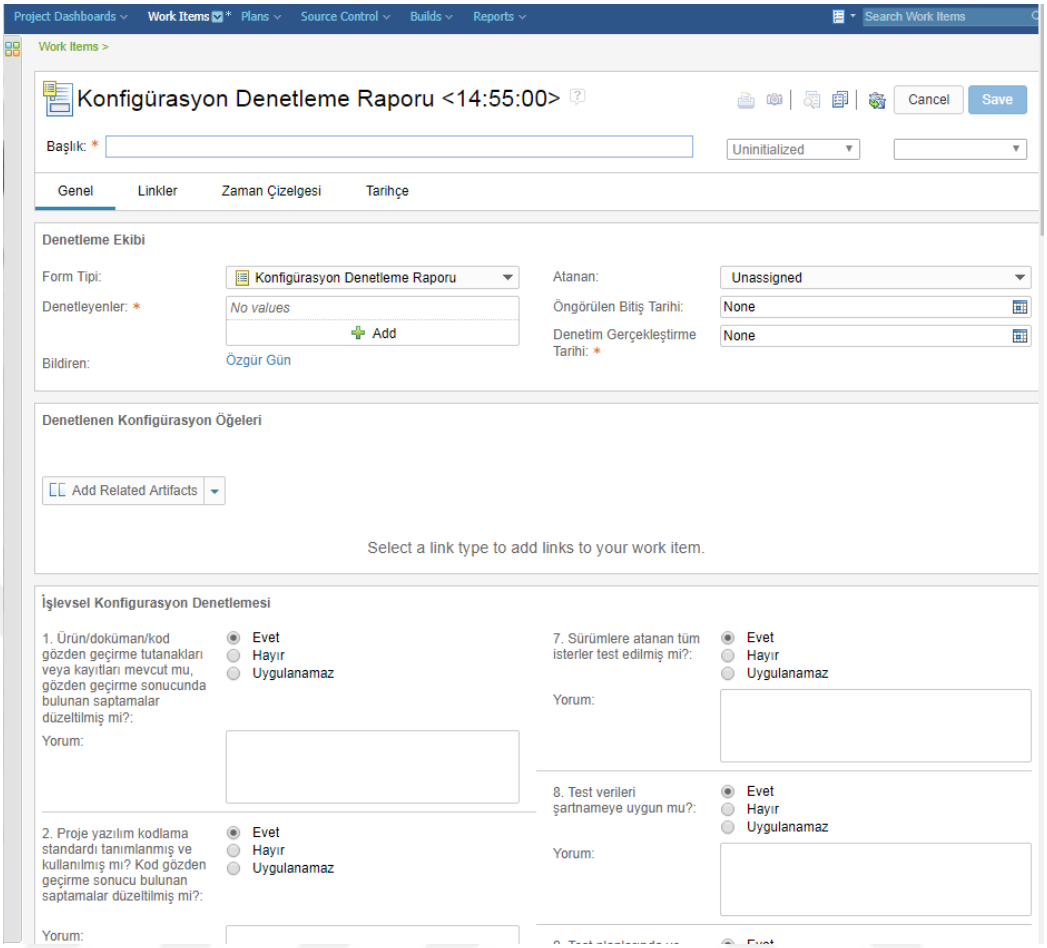
Projede farklı tipte toplantılar gerçekleştirilmektedir. Bunlar arasında sistem mühendisliği toplantıları, proje yönetimsel toplantılar, proje teknik toplantıları, risk değerlendirme toplantıları, konfigürasyon kontrol kurulu toplantıları örnek olarak sayılabilir. Projenin planlama aşamasında tüm proje boyunca gerçekleştirilecek toplantılar planlanır ve proje takviminde kayıt altına alınır. Toplantıda alınan kararlar toplantı tutanağı formu ile kayıt altına alınarak takip edilir.



Şekil 2.21. Süreç denetim formu

Şekil 2.21'deki süreç denetim formu projede işletilen süreç adımlarını denetlemek amacıyla kullanılmaktadır. Formda; denetimin adı, denetim tarihi, denetime katılan kişiler, denetimin kapsamı, denetimin planlanan ve gerçekleşen tarih bilgileri, denetimde sorulan sorular ve sorulara verilen yanıtlar, denetimde bulunan uygunsuzluklar, uygunsuzlukların kapatılması için açılan düzeltici faaliyetlerin tamamlanma durum bilgileri yer almaktadır. Süreç denetiminde ortaya çıkan uygunsuzluklar düzeltici faaliyet formu ile kayıt altına alınır ve kapanana kadar takip edilir.

Şekil 2.22'deki iş ürünü konfigürasyon denetim formunda; projede üretilen yazılım, sistem ve donanım gibi iş ürünlerini denetlemek amacıyla, denetimin adı, denetim tarihi, denetime katılan kişiler, denetimin planlanan ve gerçekleşen tarih bilgileri, denetimde sorulan sorular ve sorulara verilen yanıtlar, denetimde bulunan uygunsuzluklar, uygunsuzlukların kapatılması için açılan düzeltici faaliyetlerin tamamlanma durum bilgileri yer almaktadır. İş ürünü konfigürasyon denetiminde ortaya çıkan uygunsuzluklar düzeltici faaliyet formu ile kayıt altına alınır ve kapanana kadar takip edilir.



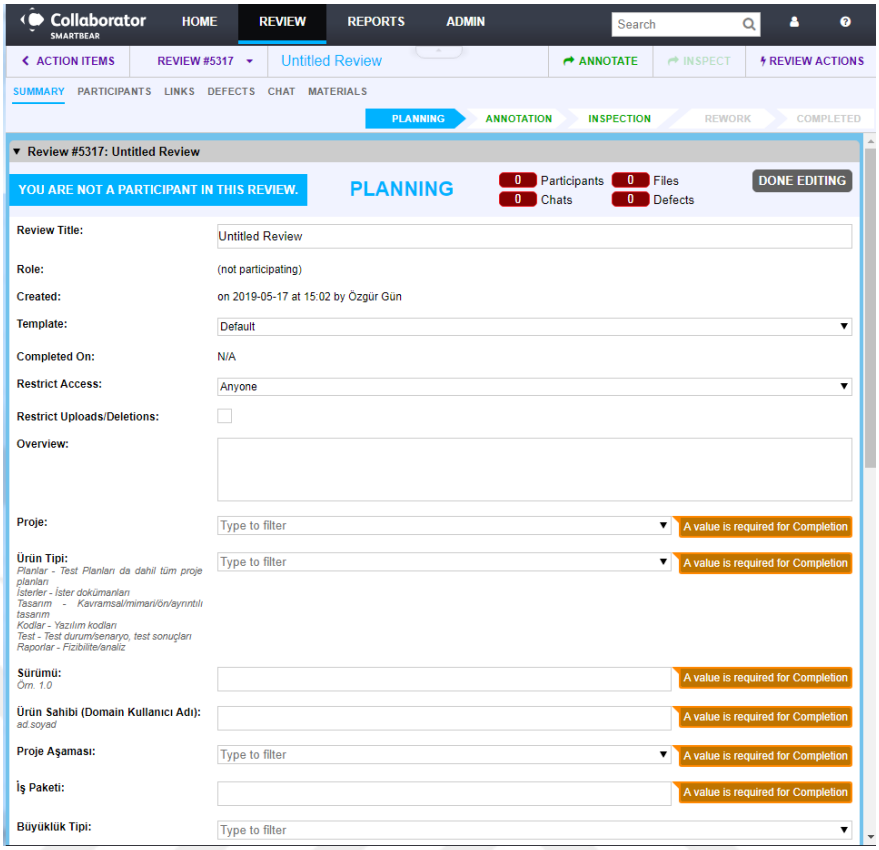
Şekil 2.22. İş ürünü konfigürasyon denetim formu

Şekil 2.23'teki gözden geçirme formu projede üretilen yazılım kodları ve doküman tipindeki iş ürünlerini gözden geçirmek amacıyla kullanılmaktadır.

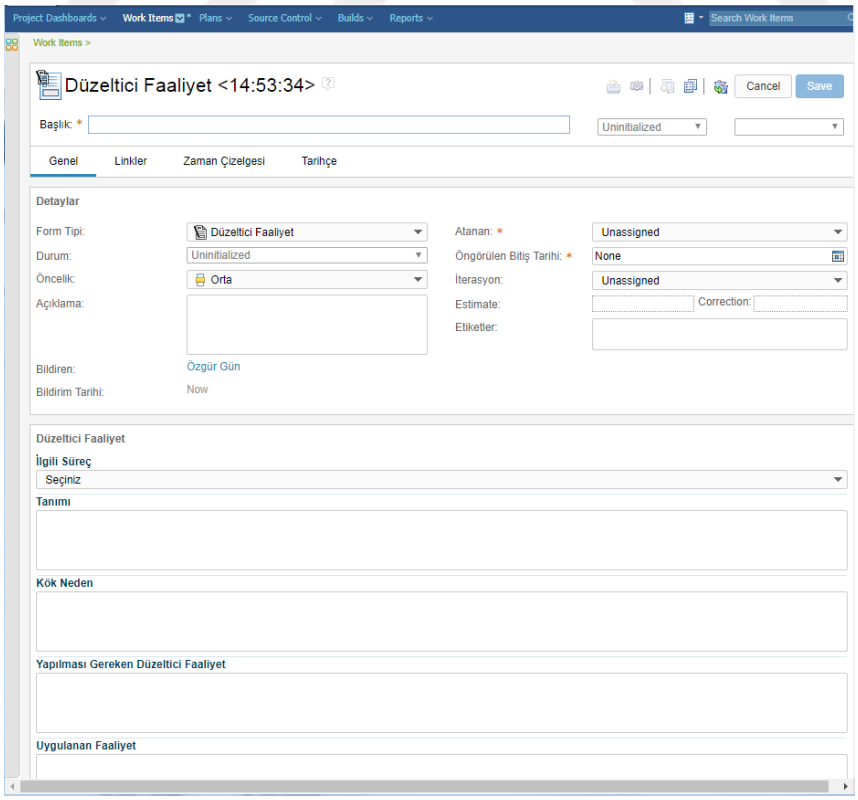
Formda; gözden geçirmenin amacı, gözden geçirilecek dokümanlar, yazılım birimleri/bileşenleri ve bunların sürüm bilgileri, gözden geçirmenin tarihi, gözden geçirmeye katılan kişiler, gözden geçirmenin tipi, gözden geçirme sonucunda bulunan uygunsuzluklar, uygunsuzlukların kapatılması için açılan düzeltici faaliyetlerin tamamlanma durum bilgileri yer almaktadır.

Konfigürasyon denetimleri, müşteriye gidecek ürün/ara ürünlerin önce tam ve doğru olmasını garanti altına almak için gerçekleştirilen kurum içi bir denetimdir.

Denetimde ortaya çıkan uygunsuzluklar kapatılmadan ürünler müşteriye gönderilmez. Konfigürasyon denetimini projenin konfigürasyon yöneticisi, kalite yöneticisi ve test mühendisleri gerçekleştirir.

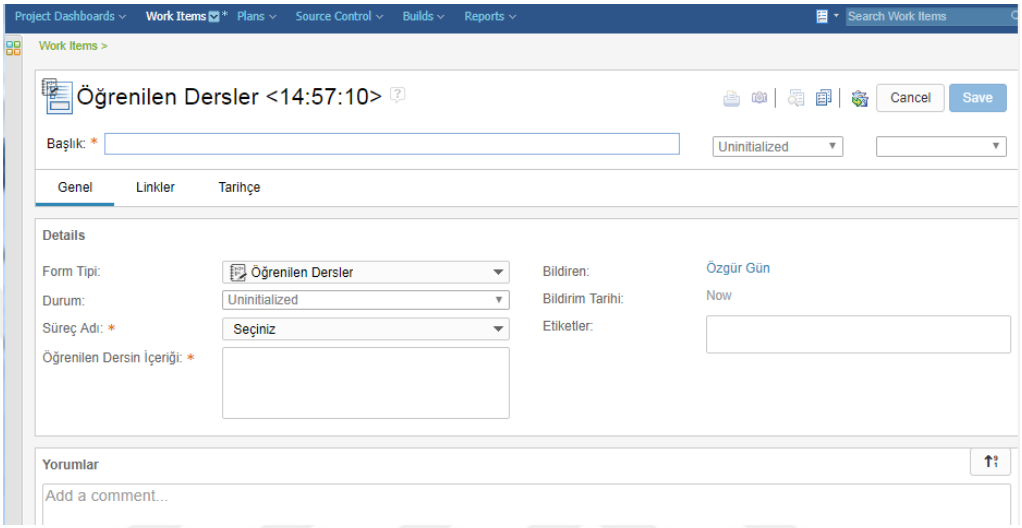


Şekil 2.23. Gözden geçirme formu



Şekil 2.24. Düzeltici faaliyet formu

Şekil 2.24’teki düzeltici faaliyet formunda; projede ortaya çıkan problemleri ve meseleleri çözmek amacıyla, düzeltici faaliyetin tanımı ve problemin kök nedeni, düzeltici faaliyetin öngörülen bitiş tarihi, düzeltici faaliyeti gerçekleştirecek kişi, yapılan düzeltici faaliyetlerin açıklamaları ve düzeltici faaliyetin tamamlanma durum bilgileri yer almaktadır.



Şekil 2.25. Öğrenilen dersler formu

Şekil 2.25’teki öğrenilen dersler formunda; projede öğrenilen faydalı bilgilerin kayıt altına alınması ve diğer projeler ile paylaşılması amacıyla öğrenilen dersin içeriği ve ilgili olduğu süreç bilgileri yer almaktadır.

2.6. Yazılımda Ölçme Süreç ve Standartları

Bu bölümde yazılım alanında ölçüm yapmak için geliştirilen literatürde en yaygın kullanılan süreç ve standartlar hakkın bilgi verilmektedir.

2.6.1. Bütünleşik yetenek olgunluk modeli

Bütünleşik Yetenek Olgunluk Modeli (Capability Maturity Model Integrated-CMMI), Amerikan Savunma Bakanlığının isteği ile Carnegie Melon Üniversitesi Yazılım Mühendisliği Enstitüsü (Software Engineering Institute) 1986 yılında geliştirilmeye başlanmıştır. Model, organizasyonların süreçlerini, altyapısını ve insan kaynaklarını sürekli iyileştirmeyi amaçlayan bir süreç çerçevesi önerir. Mart 2018 tarihinde CMMI modelinin 2.0 versiyonu yayınlanmıştır.

Bu model, etkin bir sistem ve yazılım geliştirme yaşam döngüsü süreçlerinin anahtar elemanlarını tanımlayan çerçeve modeldir. CMMI, olgun olmayan bir süreçten olgun ve disiplinli bir sürece giden bir yol çizer. Temel prensipler olarak yeteneğin olgunluğa bağlı olduğunu ve olgunluğun da zamanla gelişerek ölçülebileceğini belirtir. CMMI, temel olarak, yazılım geliştirme süreçlerinin ne kadar olgun olduğunu değerlendiren ve aynı işi yapan diğer organizasyonlarla karşılaştırma yapabilmeyi sağlayan bir araçtır.

CMMI, organizasyonların sonuç olarak performanslarını arttırmaya neden olacak etkili bir süreç iyileştirme yaklaşımıdır. CMMI, süreç iyileştirme bağlamında bir projede, bir bölümde ya da bir organizasyonunun tümünde uygulanabilir. Genel olarak CMMI, kurumların geleneksel olarak ayrı ayrı olan organizasyonel fonksiyonlarının entegre edilmesine, süreç iyileştirme çalışmalarının amaç ve önceliklerinin belirlenmesine, kalite süreçlerine rehberlik edilmesine ve mevcut süreçlerin değerlendirilmesine yönelik bir referans yaratma konularında yardımcı olur. CMMI, ne yapılması gerektiğini açıklar ancak bu gerekliliklerin nasıl yapılacağı sorusuna kesin bir cevap vermemektedir.

CMMI yaklaşımı, dünyada ve Türkiye'de genellikle bilişim sektöründe ve özellikle yazılım geliştirme konularında referans model olarak uygulanmaktadır. Sistem, donanım, hizmet ve tedarik alanlarında da uygulanan model, özellikle savunma sanayi başta olmak üzere; Ar-Ge ve yeni ürün geliştirme konusunda hizmet veren kurumların başvurduğu bir modeldir. Ülkemizde artık pek çok yazılım projelerinde, tedarik makamları yüklenicilerden CMMI konusunda en az 3. Seviye sertifikasyon talep etmektedirler.

CMMI modeli, süreç olgunluğunun bir seviyeden diğerine aşamalı olarak geliştirilebileceği teorik bir süreklilik sağlar ve 5 seviyeli bir süreç olgunluk modeli sunar.

CMMI olgunluk seviyeleri

Olgunluk seviyesi, kuruluşun genel performansını artıran önceden tanımlanmış bir dizi süreç alanı için ilgili özel ve genel pratiklerden oluşur. Bir kuruluşun olgunluk seviyesi, performansını karakterize etmek için bir yol sağlar. Deneyimler, kuruluşların

süreç iyileştirme çabalarını bir seferde yönetilebilir sayıda süreç alanına odakladıklarında ellerinden gelenin en iyisini yaptığını göstermektedir. Olgunluk seviyesi, organizasyonel süreç iyileştirme için tanımlanmış bir evrim platosudur. Her olgunluk seviyesi, kuruluşun süreçlerinin önemli bir alt kümesini olgunlaştırarak bir sonraki olgunluk seviyesine geçmeye hazırlar. Olgunluk seviyeleri, önceden tanımlanmış her bir süreç alanı seti ile ilişkili spesifik ve genel hedeflere ulaşılmasıyla ölçülür. Her biri sürekli süreç iyileştirme için temel olan beş olgunluk seviyesi 1'den 5'e kadar sayılarla belirtilir:

1. Başlangıç
2. Yönetilen
3. Tanımlı
4. Nicel olarak yönetilen
5. Sürekli gelişen

Seviye 1: Başlangıç

Olgunluk seviyesi 1'de süreçler genellikle kaotiktir. Kuruluş genellikle süreçleri desteklemek için istikrarlı bir ortam sağlamaz. Bu kuruluşlardaki başarılar, kanıtlanmış süreçlerin kullanımına değil, kuruluştaki kişilerin yeterliliğine ve kahramanlıklarına bağlıdır. Bu kargaşaya rağmen, olgunluk seviyesi 1 organizasyonları genellikle işe yarayan ürün ve hizmetler üretir, ancak planlarında belgelenen bütçeyi ve takvimi sık sık aşarlar. Olgunluk seviyesi 1 organizasyonları, aşırı kriz eğilimi, kriz anında süreçlerini terk etme ve başarılarını tekrarlayamama eğilimi ile karakterize edilir.

Seviye 2: Yönetilen

Olgunluk seviyesi 2'de projeler, süreçlerin politikaya uygun olarak planlanmasını ve yürütülmesini sağlar; projeler, kontrollü çıktılar üretmek için yeterli kaynaklara sahip nitelikli kişiler istihdam eder; projelere ilgili paydaşları dâhil edilir; projeler izlenir, kontrol edilir ve gözden geçirilir ve süreç adımlarına uymaları açısından değerlendirilir. Olgunluk seviyesi 2 ile yansıtılan süreç disiplini, mevcut pratiklerin stres dönemlerinde korunmasını sağlar. Bu pratikler yürürlüğe girdiğinde, projeler belgelenmiş planlarına göre gerçekleştirilir ve yönetilir. Ayrıca, olgunluk seviyesi 2'de, iş ürünlerinin durumu, tanımlanan noktalarda (örneğin, önemli aşamalarda,

büyük görevlerin tamamlanmasında) yönetimin erişimine açıktır. Taahhütler ilgili paydaşlar arasında oluşturulur ve gerektiğinde gözden geçirilir. İş ürünleri uygun şekilde kontrol edilir. İş ürünleri ve hizmetleri, belirtilen süreç tanımlarını, standartlarını ve prosedürlerini karşılar.

Seviye 3: Tanımlı

Olgunluk seviyesi 3'te süreçler iyi karakterize edilir, anlaşılır ve standartlar, prosedürler, araçlar ve yöntemler ile açıklanır. Organizasyonun olgunluk seviyesi 3 için temel olan standart süreçler seti, zamanla kurulmakta ve iyileştirilmektedir. Bu standart süreçler, kuruluş genelinde tutarlılık sağlamak için kullanılır. Projeler, kuruluşun standart süreç setini uyarlama kurallarına göre uyarlayarak tanımlanmış süreçlerini oluştururlar. Olgunluk seviyesi 2 ve 3 arasındaki kritik bir ayrım standartların, süreç tanımlarının ve prosedürlerin kapsamıdır. Olgunluk seviyesi 2'de standartlar, süreç tanımları ve prosedürler projeden projeye oldukça farklılık gösterebilir. Olgunluk seviyesi 3'te, bir proje veya organizasyon birimine uyacak standartlar, süreç tanımları ve prosedürler, organizasyonun standart süreç setinden uyarlanır ve bu nedenle, uyarlama kılavuzlarının izin verdiği farklılıklar dışında daha tutarlıdır.

Olgunluk seviyesi 3'te bir başka kritik ayrım, süreçlerin tipik olarak olgunluk seviyesi 2'den daha titiz bir şekilde tarif edilmesidir. Tanımlanmış bir süreç, amacı, girdileri, giriş kriterlerini, faaliyetleri, rolleri, ölçüleri, doğrulama adımlarını, çıktıları ve çıkış kriterlerini açıkça belirtir. Olgunluk seviyesi 3'te süreçler, süreç faaliyetlerinin birbirleriyle olan ilişkileri ve sürecin ayrıntılı ölçüleri, iş ürünleri ve hizmetleri göz önüne alınarak daha proaktif bir şekilde yönetilir.

Seviye 4: Nicel olarak yönetilen

Olgunluk seviyesi 4'te, organizasyon ve projeler kalite ve süreç performansı için nicel hedefler belirler ve bunları projelerin yönetiminde kriter olarak kullanır. Nicel hedefler müşterinin, son kullanıcıların, kuruluşun ve süreç uygulayıcılarının ihtiyaçlarına dayanır. Kalite ve süreç performansı istatistiksel olarak anlaşılır ve projelerin ömrü boyunca yönetilir.

Seçilen alt süreçler için, belirli süreç performansı ölçümleri toplanır ve istatistiksel olarak analiz edilir. Analizler için alt süreçleri seçerken, farklı alt süreçler arasındaki ilişkileri ve bunların kalite ve süreç performansı hedeflerine ulaşma üzerindeki etkisini anlamak önemlidir. Böyle bir yaklaşım, bir organizasyonun en çok katma değer sağlayan alt süreçlerinin istatistiksel olarak izlenmesine yardımcı olur. Süreç performans dayanakları ve modelleri, iş hedeflerine ulaşılmasına yardımcı olan kalite ve süreç performansı hedeflerinin belirlenmesine yardımcı olmak için kullanılabilir. Olgunluk seviyeleri 3 ve 4 arasındaki kritik bir ayrım, süreç performansının öngörülebilirliğidir. Olgunluk seviyesi 4'te, projelerin ve seçilen alt süreçlerin performansı, istatistiksel ve diğer nicel teknikler kullanılarak kontrol edilir ve tahminler, kısmen, iyi çözünürlükteki süreç verilerinin istatistiksel analizine dayanır.

Seviye 5: Sürekli gelişen

Olgunluk seviyesi 5'de bir organizasyon, iş hedeflerinin ve performans ihtiyaçlarının nicel olarak anlaşılmasına dayanarak süreçlerini sürekli olarak geliştirir. Organizasyon, sürecin ve süreç çıktılarının doğasında var olan değişkenliği anlamak için nicel bir yaklaşım kullanır.

Olgunluk seviyesi 5, artımlı ve yenilikçi süreç ve teknolojik iyileştirmeler yoluyla süreç performansını sürekli iyileştirmeye odaklanmaktadır. Organizasyonun kalite ve süreç performansı hedefleri oluşturulur, değişen iş hedeflerini ve kurumsal performansı yansıtacak şekilde sürekli olarak gözden geçirilir ve süreç iyileştirmeyi yönetmede kriter olarak kullanılır. Uygulamaya alınan süreç iyileştirmelerinin etkileri, istatistiksel ve diğer nicel teknikler kullanılarak ölçülür ve kalite ve süreç performansı hedefleriyle karşılaştırılır. Projenin tanımlanmış süreçleri, organizasyonun standart süreçleri ve destekleyici teknoloji, ölçülebilir iyileştirme faaliyetlerinin hedefleridir. Olgunluk seviyeleri 4 ve 5 arasındaki kritik bir ayrım, organizasyonel performansı yönetmeye ve geliştirmeye odaklanmaktadır. Olgunluk seviyesi 4'te, organizasyon ve projeler, alt süreç seviyesinde performansı anlama ve kontrol etmeye ve sonuçları projeleri yönetmek için kullanmaya odaklanır. Olgunluk seviyesi 5'te, organizasyon birden fazla projeden toplanan verileri kullanarak genel organizasyon performansı ile ilgilenir. Verilerin analizi, performanstaki eksiklikleri veya boşlukları tanımlar. Bu

boşluklar, performansta ölçülebilir gelişme sağlayan kurumsal süreç iyileştirmesini yönlendirmek için kullanılır.

CMMI süreç haritası

CMMI modelinde olgunluk seviyelerine göre süreç alanlarının sınıflandırılması Tablo 2.7'de verilmiştir.

Tablo 2.7. CMMI Süreç alanları ve olgunluk seviyeleri [89]

Seviye	**Fokus**	**Süreç Alanları**
5	Sürekli Gelişen	Neden Analizi ve Çözüm, Organizasyonel Performans Yönetimi
4	Nicel Olarak Yönetilen	Organizasyonel Süreç Performans Niceliksel Proje Yönetimi
3	Tanımlı	Karar Analizi ve Çözüm Bütünleşik Proje Yönetimi Organizasyonel Süreç Tanımlama Organizasyonel Süreç Yönetimi Organizasyonel Eğitim Sistem Tümleştirme Gereksinim Geliştirme Risk Yönetimi Teknik Çözüm
2	Yönetilen	Konfigürasyon Yönetimi Ölçme ve Analiz Proje İzleme ve Kontrol Proje Planlama Gereksinim Yönetimi Tedarik Yönetimi
1	Başlangıç	-

CMMI, ölçüm ve analiz süreç alanını olgunluk seviyesi 2'de tanımlar ve bu da yazılım projelerinin bu ihtiyaçlara hizmet edecek özel bilgi ihtiyaçlarını ve metriklerini ve ayrıca metrik veri toplama, analiz ve paylaşım prosedürlerini tanımlar. CMMI ayrıca,

projenin tanımlı sürecinin niceliksel yönetimini teklif eden 4. seviye Niceliksel Proje Yönetimi süreç alanını tanımlar. Bu iki süreç alanı, CMMI'nin seviye 2'deki ölçüm uygulamalarını teşvik etmesine rağmen, süreç kontrolünü seviye 4'e kadar ertelediğini göstermektedir.

CMMI ölçme ve analiz süreci (Seviye – 2)

Bu süreç alanının amacı, yönetim bilgi ihtiyaçlarını desteklemek için kullanılan bir ölçüm yeteneğini geliştirmek ve sürdürmektir. Ölçüm yeteneği, bireysel projelere veya diğer organizasyon fonksiyonlarına (örneğin kalite güvencesi) entegre edilebilir. Ölçüm faaliyetleri için ilk odak noktası proje düzeyindedir. Bununla birlikte, bir ölçüm kabiliyeti organizasyon seviyesinde bilgi ihtiyaçları için de yararlıdır. Süreç alanının spesifik hedef ve spesifik pratikleri (SP) Tablo 2.8'de verilmiştir. Bunlar, ölçüm ve analize özgü görevleri tanımlayan hedef ve pratiklerdir.

Tablo 2.8. CMMI Ölçme ve analiz süreç alanı - hedefler ve pratikler [89]

Hedef	**Pratikler**
Spesifik Hedef 1: Ölçme ve Analiz Faaliyetlerini Planlama	SP 1.1 Ölçüm Hedefleri Belirleme
	SP 1.2 Ölçümleri Belirleme
	SP 1.3 Veri Toplama ve Saklama Prosedürlerini Belirleme
	SP 1.4 Analiz Prosedürleri Belirleme
Spesifik Hedef 2: Ölçüm Sonuçlarını Sağlama	SP 2.1 Ölçüm Verilerini Toplama
	SP 2.2 Ölçüm Verilerini Analiz Etme
	SP 2.3 Verileri ve Sonuçları Kaydetme
	SP 2.4 Sonuçları Bildirme

CMMI ölçme ve analiz süreç alanı aşağıdaki adımları içerir:

1. Ölçme ve analizin amaçlarını belirlenmiş bilgi ihtiyaçları ve hedefleri ile uyumlu olacak şekilde belirlemek,

2. Ölçüm belirleme, veri toplama ve depolama mekanizmalarının, analiz tekniklerinin, raporlama ve geri bildirim mekanizmalarının belirlenmesi,
3. Verilerin toplanmasını, depolanmasını, analizini ve raporlanmasını uygulamak,
4. Karar verirken ve uygun düzeltici eylemler alırken kullanılabilecek objektif sonuçlar sağlamak.

Niceliksel proje yönetimi süreci (Seviye - 4)

Bu süreç alanının amacı, projenin belirlenmiş kalite ve süreç performansı hedeflerine ulaşmak için projenin tanımlanmış sürecini nicel olarak yönetmektir. Proses alanının spesifik hedefleri ve pratikleri Tablo 2.9'da verilmiştir.

Tablo 2.9. CMMI Nicel proje yönetimi süreç alanı - hedefler ve pratikler [89]

Hedef	**Pratikler**
Spesifik Hedef 1: Nicel Yönetim İçin Hazırlık	SP 1.1 Projenin Hedeflerini Belirleme
	SP 1.2 Tanımlı Süreci Oluşturma
	SP 1.3 Alt Süreçleri ve Özellikleri Seçme
	SP 1.4 Ölçüm ve Analitik Teknikleri Seçme
Spesifik Hedef 2: Projenin Nicel Olarak Yönetilmesi	SP 2.1 Seçilen Alt Süreçlerin Performansını İzleme
	SP 2.2 Proje Performansını Yönetme
	SP 2.3 Kök Neden Analizi Gerçekleştirme

Nicel proje yönetimi süreç alanı aşağıdakileri içerir:

1. Projenin kalite ve süreç performansı hedeflerinin oluşturulması ve sürdürülmesi,
2. Süreç performans taban dayanaklarında veya modellerinde bulunan geçmişe dönük yetenek verilerine dayanarak projenin tanımlanmış sürecini oluşturan uygun alt süreçlerin belirlenmesi,
3. İstatistiksel olarak yönetilecek projenin tanımlanmış sürecinin alt süreçlerini seçmek,
4. Projenin kalite ve süreç performansı hedeflerinin karşılanıp karşılanmadığını belirlemek ve uygun düzeltici eylemleri belirlemek için projeyi izlemek,

5. Seçilen alt süreçlerin istatistiksel olarak yönetilmesinde kullanılacak ölçümlerin ve analitik tekniklerin seçilmesi,
6. Seçilen ölçümleri ve analitik teknikleri kullanarak seçilen alt süreçlerin varyasyonunun anlaşılması ve sürdürülmesi,
7. Kalite ve süreç performans hedeflerini karşılayıp karşılamadıklarını belirlemek için seçilen alt süreçlerin performansını izlemek ve düzeltici eylemleri belirlemek,
8. Kuruluşun ölçüm deposuna istatistik ve kalite yönetimi verilerinin kaydedilmesi.

Alt süreçler, daha büyük bir tanımlanmış sürecin tanımlanmış bileşenleridir. Örneğin, tipik bir organizasyonun geliştirme süreci, gereksinimlerin geliştirilmesi, tasarım, kodlama, test ve gözden geçirme gibi alt süreçler olarak tanımlanabilir. Alt süreçlerin kendileri gerektiğinde diğer alt süreçlere bölünebilir. Nicel yönetimin önemli bir unsuru daha doğru tahminlerin yapılmasıdır.(örneğin, projenin kalite ve süreç performans hedeflerini ne ölçüde yerine getirebileceğini tahmin edebilmek). İstatistiksel olarak yönetilecek alt süreçler, öngörülebilir performans için belirlenen ihtiyaçlara göre seçilir.

Bu tez çalışmasında bir Ar-Ge kurumunda test süreci çıktıları analiz edilmiştir. Projelerde test alt sürecinde bulunan hataların özellikleri toplanmış ve hataların düzetilip kapatılma süresini tahmin eden regresyon modeli ortaya çıkarılmıştır. Bu konuda Bölüm 4.6'da detaylı bilgi verilmiştir.

Nicel yönetimin bir diğer önemli unsuru, süreç performansında karşılaşılan varyasyonun doğasını ve kapsamını anlamak ve projenin gerçek performansının, projenin kalite ve süreç performansı hedeflerine ulaşmak için ne zaman yeterli olmayacağını anlamaktır. İstatistiksel yönetim, istatistiksel düşünceyi ve çalışma çizelgeleri, kontrol grafikleri, güven aralıkları, tahmin aralıkları ve hipotez testleri gibi çeşitli istatistiksel tekniklerin doğru kullanımını içerir. Nicel yönetim, projenin kalite ve süreç-performans hedeflerine ulaşıp ulaşamayacağını tahmin etmesine ve hangi düzeltici önlemlerin alınması gerektiğini belirlemeye yardımcı olmak için istatistiksel yönetimden gelen verileri kullanır. Bu süreç alanı bir projenin yönetimi için geçerlidir, ancak burada bulunan kavramlar diğer grupların ve işlevlerin yönetimi için de geçerlidir. Bu kavramları, diğer grupları ve işlevleri yönetmeye uygulamak, kuruluşun iş hedeflerine ulaşılmasına katkıda bulunması zorunlu değildir, ancak bu kavramlar grupların ve işlevlerin kendi süreçlerini kontrol etmelerine yardımcı olabilir.

2.6.2. ISO 15504 yazılım süreç değerlendirme

ISO / IEC TR 15504 [90], yazılım süreçlerinin değerlendirilmesi için bir çerçeve sağlar. Değerlendirme, bir organizasyon birimi içindeki mevcut uygulamayı seçilen süreçlerin yetenekleri açısından karakterize eder. Süreç boyutu müşteri-tedarikçi, mühendislik, destek, yönetim ve organizasyon gibi temel kategoriler altındaki yaşam döngüsü süreçlerini içerir. Yetenek boyutu her biri bir veya daha fazla süreç özelliğinden oluşan 6 seviye (0 ila 5) içerir. ISO / IEC TR 15504, süreç yeteneği göstergelerini tanımlar. Göstergeler Tablo 2.10'da verilmiştir. Süreç kontrolü Seviye 4 ile geldiğinden (özellikle süreç niteliği 4.2 ile), o seviyeye kadar olan yönetim pratiklerinin göstergeleri, istatistiksel kontrole tabi olan bir sürecin özellikleri hakkında fikir verir.

Tablo 2.10. ISO / IEC TR 15504'teki yetenek seviyeleri ve süreç özellikleri [90]

Yetenek Seviyesi	Süreç Özelliği	Yönetim Pratiği
1: Gerçekleştirilmiş	1.1 Süreç değerlendirme	1.1.1: Girdi ve çıktı iş ürünlerini tanımlanması. 1.1.2: Süreç yürütülmesi ve süreç tarafından kullanılacak ve üretilecek iş ürünleri için iş kapsamının tanımlandığından emin olunması. 1.1.3: Tanımlanan süreç sonuçlarına ulaşılmasını destekleyen iş ürünleri üreterek temel pratiklerin uygulanmasını sağlama.
2: Yönetilen	2.1: Performans yönetimi	2.1.1: Sürecin gerçekleştirilmesine yönelik hedeflerin belirlenmesi (örneğin; zaman ölçeği, döngü süresi ve kaynak kullanımı). 2.1.2: Sürecin faaliyetlerini, beklenen zaman çizelgesini ve her bir faaliyet için kaynak tahsisini belirleyerek sürecin performansının belirlenen hedeflere göre planlanması. 2.1.3: Sürecin iş ürünlerini geliştirmek için sorumluluk ve yetkinin planlanması ve atanması. 2.1.4: Tanımlanan hedefleri karşılayan iş ürünleri üretmek için izleme ve yeniden planlama yapmaya devam ederek faaliyetlerin yürütülmesinin yönetilmesi. Projede izleme faaliyetleri için gerçekleştirilecek tüm toplantılar proje başlangıcında planlanır. Ayrıca proje aşama sonlarında müşteri ile gerçekleştirilecek proje izleme ve gözden geçirme toplantıları da proje takviminde belirtilir. Proje izleme faaliyetlerinde ortaya çıkan uygunsuzluklar kapatılana kadar takip edilir.

Tablo 2.10.(Devam) ISO / IEC TR 15504'teki yetenek seviyeleri ve süreç özellikleri [90]

Yetenek Seviyesi	Süreç Özelliği	Yönetim Pratiği
	2.2: İş ürünü yönetimi	2.2.1: Hem işlevsel hem de işlevsel olmayan yönler de dâhil olmak üzere iş ürünleri için gereksinimlerin belirlenmesi. 2.2.2: İş ürünlerinin belgelerini, konfigürasyon yönetimini ve değişiklik kontrolünün yönetilmesi. 2.2.3: İş ürünü bağımlılıklarının belirlenmesi. 2.2.4: Fonksiyonel ve fonksiyonel olmayan gereksinimlerini karşıladıklarından emin olmak için iş ürünlerinin kalitesinin yönetilmesi.
3: Kurulmuş	3.1: Süreç tanımı	3.1.1: Yönetilen sürecin işletimini destekleyen standart sürecin belirlenmesi. 3.1.2: Süreç bağlamına uygun tanımlı bir süreç elde etmek için standart sürecin uyarlanması. 3.1.3: Tanımlanan sürecin davranışının anlaşılabilmesi için süreç performans verilerinin toplanması. 3.1.4: Süreç performans verilerini kullanarak süreç davranışının anlaşılmasının oluşturulması ve geliştirilmesi. 3.1.5: Standart sürecin geliştirilmesi.
	3.2: Süreç kaynakları	3.2.1: Tanımlanan sürecin uygulanmasını desteklemek için gereken rolleri, sorumlulukları ve yetkinliklerin belirlenmesi ve belgelenmesi. 3.2.2: Tanımlanmış sürecin uygulanmasını desteklemek için süreç altyapısı gereksinimlerinin tanımlanması ve belgelendirilmesi. 3.2.3: Tanımlanan sürecin performansını desteklemek için kaynakların sağlanması, tahsis edilmesi ve kullanılması. 3.2.4: Tanımlanmış sürecin performansını desteklemek için yeterli bir süreç altyapısının sağlanması, tahsis edilmesi ve kullanılması.
4: Tahmin Edilebilir	4.1 Ölçme	4.1.1: İlgili iş hedeflerine ulaşılmasını destekleyen ürün ve süreç hedeflerinin ve önlemlerinin belirlenmesi. 4.1.2: Tanımlanan işlemi gerçekleştirerek belirtilen ürün ve süreç ölçümlerinin toplanması. 4.1.3: Kuruluş genelinde süreç performansındaki eğilimlerin analiz edilmesi. 4.1.4: Süreç yeteneğinin ölçülmesi ve kurum genelinde belirlenen sınırlar içinde tutulması.

Tablo 2.10.(Devam) ISO / IEC TR 15504'teki Yetenek Seviyeleri ve Süreç Özellikleri [90]

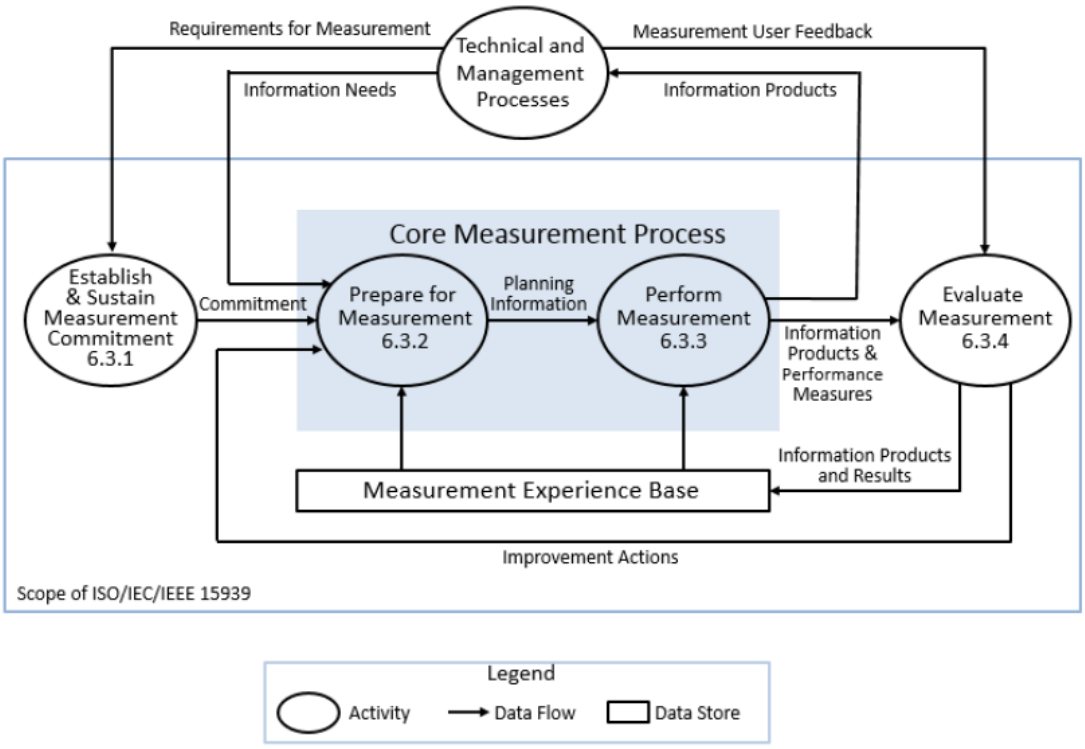
Yetenek Seviyesi	Süreç Özelliği	Yönetim Pratiği
	4.2: Süreç kontrol	4.2.1: Süreci ve ürün geliştirmeyi desteklemek için süreç bağlamına uygun ölçüm tekniklerinin tanımlanması. 4.2.2: Analiz yapmak için ölçümlerin toplanması ve süreç kontrol parametrelerinin belirlenmesi. 4.2.3: Ölçümleri analiz ederek süreç performansının kontrol edilmesi.
5: En İyileştirici	5.1: Süreç değişimi	5.1.1: Sürecin nicel olarak anlaşılması temelinde standart süreç tanımındaki değişikliklerin belirlenmesi. 5.1.2: Önerilen tüm değişikliklerin, tanımlanan ürün ve süreç hedefleri üzerindeki etkisinin değerlendirilmesi. 5.1.3: Süreç performansındaki aksaklıkların anlaşılmasını ve düzeltilmesini sağlamak için bir uygulama stratejisinin tanımlanması. 5.1.4: Etkilenen süreçte onaylanan değişikliklerin uygulama stratejisine göre uygulanması. 5.1.5: Süreç değişikliğinin etkinliğini, tanımlanan ürün, süreç ve iş hedeflerine göre performans temelinde değerlendirerek ayarlamaların yapılması.
	5.2: Süreç iyileştirme	5.2.1: Organizasyonun ilgili iş hedeflerini destekleyen süreç için süreç iyileştirme hedeflerinin tanımlanması. 5.2.2: Süreci sürekli iyileştirmek için iyileştirme fırsatlarını sistematik ve proaktif bir şekilde belirleyerek mevcut süreçteki gerçek ve potansiyel sorunların kaynağının analiz edilmesi. 5.2.3: Uygulama stratejisine göre, uyarlanmış sürecin seçilen alanlarındaki değişikliklerinin uygulanması. 5.2.4: Süreç ve iş hedeflerine karşı gerçek performans temelinde süreç değişikliği etkinliğinin doğrulanması.

2.6.3. IEEE 15939 ölçme süreci

IEEE 15939 [91] uluslararası standardı, yazılım organizasyonlarının ve projelerinin özel ihtiyaçlarını karşılayan bir yazılım ölçüm sürecini oluşturan bir dizi faaliyet ve görevi içermektedir. Şekil 2.26'da model olarak belirtilen gösterildiği gibi bu standart, genel bir proje veya organizasyonel ölçüm yapısı içinde ölçümü başarıyla belirlemek,

tanımlamak, seçmek, uygulamak ve geliştirmek için gerekli olan faaliyetleri ve görevleri tanımlar. Ayrıca, sistem ve yazılım endüstrilerinde yaygın olarak kullanılan ölçüm terimleri için tanımlar sağlar. Yazılım tedarikçileri ve edinenleri tarafından kullanılmak üzere tasarlanmıştır. Bu uluslararası standart, süreç metriklerini sınıflandırmaz ve projelere uygulanacak önerilen bir metrik kümesi sağlamaz. Belirli bilgi ihtiyaçlarını karşılayan uygun bir dizi metrik tanımlamayı destekleyen bir süreci tanımlar.

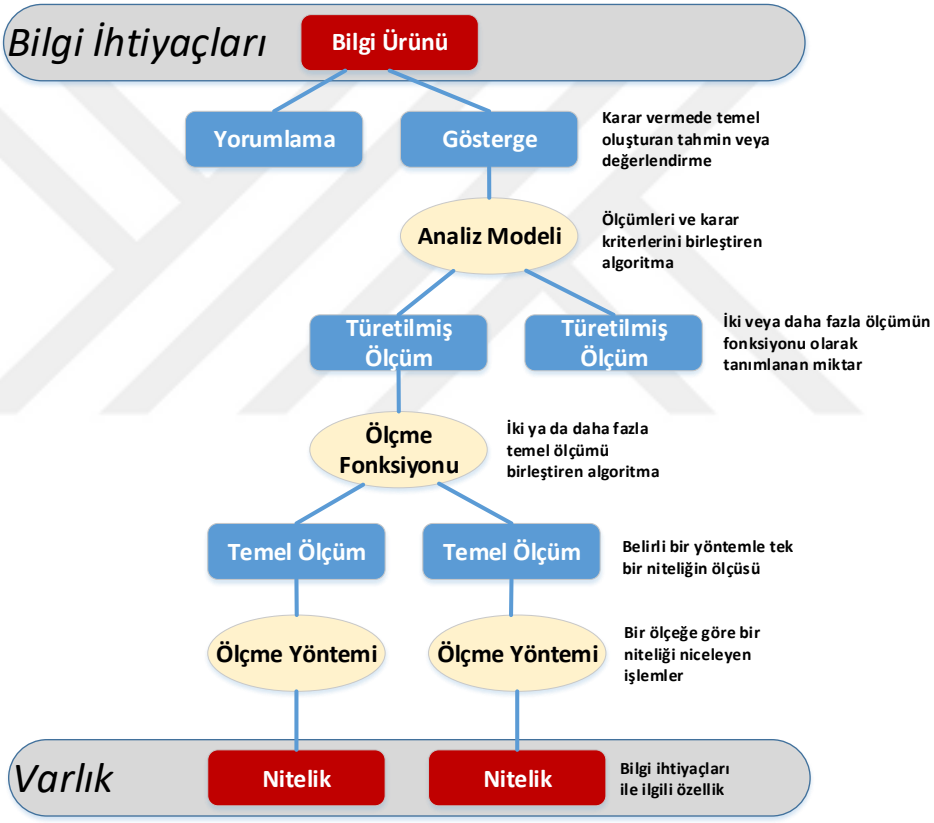
Standart, ölçüm için bir organizasyonel bir model önermemektedir. Kullanıcılar kuruluş içinde ayrı bir ölçüm fonksiyonunun gerekli olup olmadığına, ölçüm fonksiyonunun projelere entegre edilmesi gerekip gerekmediğine, mevcut organizasyonel yapıya, kültüre ve hakim kısıtlamalara dayanarak karar verirler. Ayrıca, üretilecek belgelerin adını, biçimini veya içeriğini tanımlamaz ve bu kararları kullanıcılarına bırakır [92].



Şekil 2.26. IEEE 15939 ölçme süreç modeli [91]

Bu tez çalışmasında kullanılan yazılım yaşam döngüsü süreç metriklerinin gösterimi için IEEE Std 15939:2017 standardında tanımlı olan "Ölçüm Bilgi Modeli" kullanılmıştır. Ölçüm bilgi modeli, bilgi ihtiyaçlarını ilgili ilgili varlıklara ve

niteliklere bağlayan bir yapıdır. Varlıklar süreçleri, ürünleri, projeleri ve kaynakları içerir. Ölçüm bilgi modeli, ilgili özelliklerin nasıl ölçüleceğini ve karar vermeye temel oluşturan göstergelere nasıl dönüştürüldüğünü açıklar. Bir bilgi ihtiyacını ele almak için uygun ölçümlerin seçimi veya tanımı ölçülebilir bir kavramla başlar. Bu kavram hangi ölçülebilir niteliklerin bir bilgi ihtiyacıyla ilişkili olduğu ve bunların nasıl ilişkili olduğu fikridir. Ölçüm planlayıcısı, bu nitelikleri belirli bir bilgi ihtiyacına bağlayan ölçüm yapılarını tanımlar. Şekil 2.27 ölçüm bilgi modeli temel terim ve kavramları tanımlar. Ölçüm bilgi modeli, ölçüm planlama, performans ve değerlendirme sırasında ölçüm planlayıcısının neyi belirtmesi gerektiğini belirlemeye yardımcı olur.



Şekil 2.27. Ölçüm bilgi modelindeki temel ilişkiler

Ölçüm bilgi modeli

Şekil 2.27, ölçüm bilgi modelinin temel bileşenleri arasındaki ilişkileri göstermektedir. Model üç tür ölçüm tanımlar: temel ölçümler, türetilmiş ölçümler ve göstergeler. Ölçümlerin bilgi içeriği modelde bilgi ihtiyacına yaklaştıkça artar. Ölçüm bilgi modelinin bileşenleri aşağıda açıklanmaktadır.

Varlık

Bir varlık, niteliklerinin ölçülmesi ile karakterize edilecek bir nesnedir (örneğin bir süreç, ürün, proje veya kaynak). Tipik mühendislik nesneleri, ürünler (örneğin Tasarım belgesi, ağ, kaynak kodu ve test senaryosu), süreçler (örneğin Tasarım süreci, test süreci ve gereksinim analizi süreci), projeler ve kaynaklar (örneğin sistem mühendisleri, yazılım mühendisleri, programcılar ve test mühendisleri) olarak sınıflandırılabilir. Organizasyon, bilgi ihtiyaçlarını karşılamak için bir veya daha fazla niteliğe sahip olabilir. Uygulamada, bir varlık yukarıdaki kategorilerden birden fazlasına sınıflandırılabilir.

Nitelik

Bir nitelik, bir varlığın insan veya otomatik yöntemlerle niceliksel veya niteliksel olarak ayırt edilebilen bir özelliğidir. Bir varlığın birçok özelliği olabilir, bunlardan bazıları ölçüm kapsamına dâhil edilebilir. Ölçüm bilgi modelinin belirli bir örneğini tanımlamanın ilk adımı, ölçüm kullanıcısının bilgi ihtiyaçları ile en ilgili nitelikleri seçmektir. Belirli bir nitelik, farklı bilgi ihtiyaçlarını destekleyen çoklu ölçüm yapılarına dâhil edilebilir.

Temel ölçüm

Bir ölçüm, bir nitelik ve onu niceleme yöntemi olarak tanımlanır. (Bir ölçüm, bir değerin atandığı bir değişkendir.) Temel ölçüm, diğer ölçümlerden işlevsel olarak bağımsızdır. Temel ölçüm, tek bir nitelik hakkındaki bilgileri yakalar. Veri toplama temel ölçümlere değer atanmasını içerir. Bir temel ölçümün beklenen değer aralığını veya türünü belirtmek, toplanan verilerin kalitesini doğrulamaya yardımcı olur.

Ölçme yöntemi

Bir ölçüm yöntemi, belirli bir ölçeğe göre bir niteliğin nicelleştirilmesinde kullanılan, genel olarak tarif edilen mantıksal bir işlem dizisidir. Operasyonlar, olayları saymak veya zamanın geçişini gözlemlemek gibi faaliyetleri içerebilir. Aynı ölçüm yöntemi birden fazla özelliğe uygulanabilir. Ancak, bir niteliğin ve bir yöntemin her benzersiz kombinasyonu farklı bir temel ölçü üretir. Bazı ölçüm yöntemleri çeşitli şekillerde

uygulanabilir. Bir ölçüm prosedürü, belirli bir organizasyonel bağlamda bir ölçüm yönteminin spesifik uygulamasını açıklar.

Ölçme yöntemi tipi

Ölçme yönteminin tipi, bir niteliği ölçmek için kullanılan işlemlerin doğasına bağlıdır. İki tür yöntem ayırt edilebilir:

1. Öznel - insan muhakemesini içeren niceleme,
2. Nesnel - sayma gibi sayısal kurallara dayalı niceleme.

Bu kurallar insan veya otomatik yöntemlerle uygulanabilir.

Ölçek

Ölçek, sürekli veya ayrık sıralı bir değerler kümesi veya niteliğin eşlendiği bir kategori kümesidir. Ölçüm yöntemi, ölçülen özelliğin büyüklüğünü bir ölçek üzerindeki bir değerle eşler. Bir ölçü birimi genellikle bir ölçekle ilişkilendirilir.

Ölçek tipi

Ölçek türü, ölçek üzerindeki değerler arasındaki ilişkinin doğasına bağlıdır. Dört tür ölçek yaygın olarak tanımlanmıştır:

1. Sınıflama: ölçüm değerleri kategoriktir. Birbirine benzeyen varlıkların aynı sınıfa konulmasıdır. Örneğin, insanları çocuk-genç-yetişkin kategorilerine ayırmak sınıflama ölçeğine örnektir. Bir başka örnek olarak, hataların türlerine göre sınıflandırılması kategoriler arasında bir sıralama anlamına gelmez.
2. Sıralı: varlıkları belirli bir özelliğe sahip oluş düzeyine göre ölçüm değerlerini sıralar. Örneğin, hataların önem düzeyine atanması bir sıralamadır.
3. Aralık: ölçüm değerleri, niteliğin eşit miktarlarına karşılık gelen eşit mesafelere sahiptir. Örneğin, siklomatik karmaşıklığın minimum değeri birdir, ancak her artış ek bir yolu temsil eder. Sıfır değeri mümkün değildir.
4. Oran: ölçüm değerleri, sıfır değerinin niteliğin hiçbirine karşılık gelmediği, niteliğin eşit miktarlarına karşılık gelen eşit mesafelere sahiptir. Örneğin, gereksinim sayısı cinsinden boyut bir oran ölçeğidir, çünkü sıfır değeri hiçbir gereksinime karşılık

gelmez ve tanımlanan her ek gereksinim eşit bir artımlı miktarı temsil eder. Uzunluk, ağırlık ölçüleri, saat birimleri oranlı ölçek örnekleridir.

Ölçüm birimi

Niceliklerin aynı türden diğer niceliklerle karşılaştırılmasıdır. Sadece aynı ölçü birimlerinde ifade edilen miktarlar doğrudan karşılaştırılabilir. Örneğin saat ve uzunluk ölçü birimi olarak ifade edilebilir.

Türetilmiş ölçüm

Türetilmiş bir ölçüm, temel ölçümlerin iki veya daha fazla değerinin bir fonksiyonu olarak tanımlanan bir ölçüdür. Türetilmiş ölçümler, birden fazla nitelikten veya birden fazla varlıktan aynı nitelik hakkında bilgi yakalar. Temel ölçümlerin basit dönüşümleri (örneğin, bir temel ölçümün karekökünü alarak) bilgi eklemez, dolayısıyla türetilmiş ölçümler olarak kabul edilmez.

Ölçme fonksiyonu

Fonksiyon, iki veya daha fazla temel ölçüyü birleştirmek için gerçekleştirilen bir algoritma veya hesaplamadır. Türetilmiş ölçümün ölçeği ve birimi, oluşturulduğu temel ölçümlerin ölçeklerine ve birimlerine ve bunların fonksiyonla nasıl birleştirildiklerine bağlıdır.

Gösterge

Bir gösterge, tanımlanmış bilgi ihtiyaçlarına göre bir modelden türetilen belirli özelliklerin bir tahminini veya değerlendirmesini sağlayan bir ölçüdür. Göstergeler analiz ve karar almanın temelini oluşturur. Ölçüm her zaman kusurlu bilgilere dayanır, bu nedenle göstergelerin belirsizliğini, doğruluğunu veya önemini ölçmek, gerçek gösterge değerini sunmanın temel bir bileşenidir.

Model

Model, Bir veya daha fazla temel veya türetilmiş ölçümleri ilişkili karar kriterleri ile birleştiren bir algoritma veya hesaplamadır. Bileşen ölçümleri veya zaman içindeki bileşen davranışları arasındaki beklenen ilişkinin anlaşılmasına veya varsayımlarına dayanır. Modeller, tanımlanmış bilgi ihtiyaçları ile ilgili tahminler veya

değerlendirmeler üretir. Ölçek ve ölçme yöntemi, gösterge üretmek için kullanılan analiz tekniklerinin veya modellerinin seçimini etkiler.

Karar kriteri

Karar kriterleri, eylem veya ileri araştırma ihtiyacını belirlemek veya belirli bir sonuca olan güven seviyesini tanımlamak için kullanılan sayısal eşik değerler veya hedeflerdir. Karar kriterleri ölçüm sonuçlarını yorumlamaya yardımcı olur. Karar kriterleri hesaplanabilir veya beklenen davranışın kavramsal anlayışına dayanabilir. Karar kriterleri geçmiş verilerden, planlardan ve sezgisel yöntemlerden türetilebilir veya istatistiksel kontrol sınırları veya istatistiksel güven sınırları olarak hesaplanabilir.

Ölçülebilir konsept

Ölçülebilir bir kavram, varlıkların özellikleri ile bilgi ihtiyaçları arasındaki soyut bir ilişkidir. Örneğin, bir bilgi gereksinimi, bir proje grubunun geliştirme verimliliğini bir hedef oranla karşılaştırma ihtiyacı olabilir. Bu durumda ölçülebilir kavram "geliştirme verimlilik oranı" dır. Kavramı değerlendirmek için sistemin veya yazılım ürünlerinin boyutunun ve ürünleri oluşturmak için uygulanan kaynak miktarının (seçilen verimlilik modeline bağlı olarak) ölçülmesi gerekebilir. Ölçülebilir kavramlara ek örnekler arasında kalite, risk, performans, yetenek, olgunluk ve müşteri değeri sayılabilir.

2.7. Yazılım Geliştirme Yaşam Döngüsü

Yazılım yaşam döngüsü süreçleri, yazılım projelerinin tedarik, planlama, ihtiyaç analizi, tasarım, kodlama, entegrasyon, gözden geçirme, test etme, teslim etme ve elden çıkarma aşamalarına yönelik faaliyetleri ve talimatları tanımlayan etkileşimli bir yönetim sistemidir. Bu süreçler analistlerin, yazılım geliştiricilerin, entegratörlerin, operatörlerin, bakım görevlilerinin, yöneticilerin, kalite güvence yöneticilerinin ve yazılımı kullanacak olan son kullanıcıların süreç adımlarını ve sorumluluklarını tanımlar [93]. Tablo 2.4'te belirtilen süreçlerin yazılım yaşam döngüsü içerisindeki sırası ve birbirleri ile etkileşim haritası Şekil 2.28'de gösterilmiştir. Yaşam döngüsündeki her aşama için kısa bilgiler aşağıda verilmiştir.

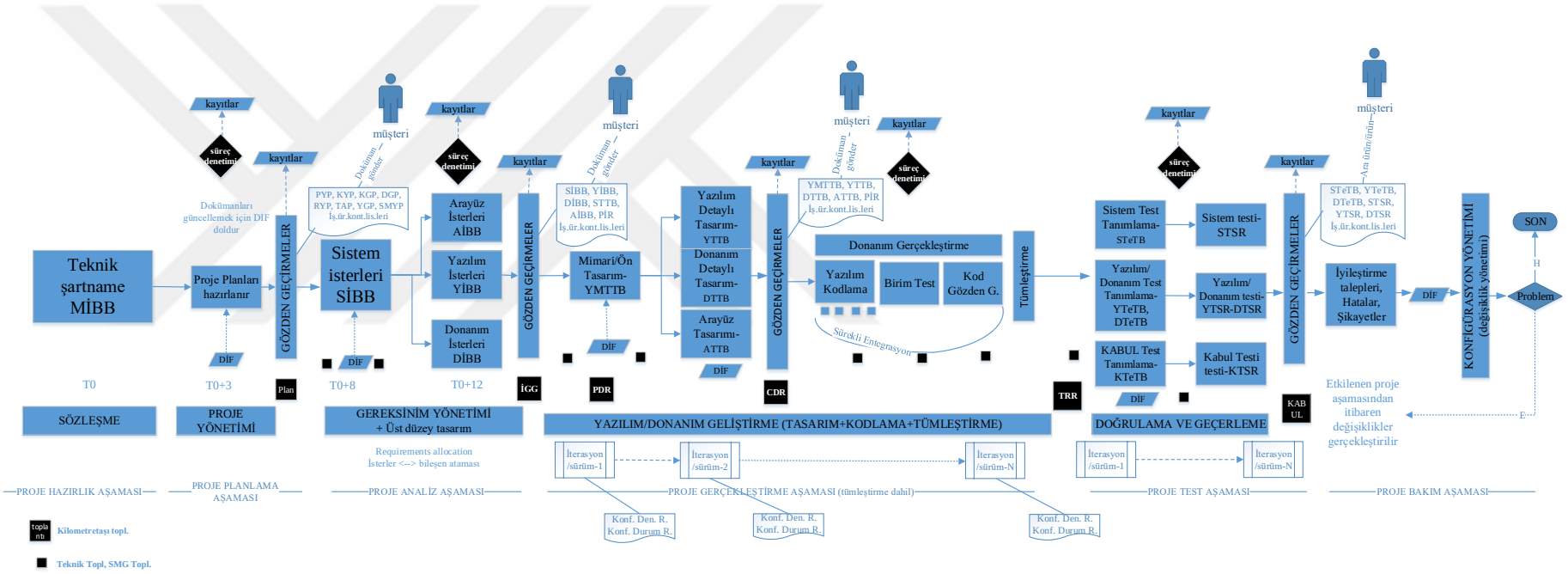
1. Proje hazırlık aşaması

Proje hazırlık aşamasında, proje teklifinin oluşturulması, gözden geçirilmesi ve değerlendirilmesi faaliyetlerini kapsayan iş geliştirme faaliyetleri gerçekleştirilir. Bu aşamada projeye teklif verip vermeme kararı verilir.

2. Proje planlama aşaması

Yazılım yaşam döngüsü önce planlama aşamasıyla başlar. Bu aşamada, bir yazılım projesinin işletmesi gereken tüm süreçler planlanır. Proje sözleşmesinin imzalanmasından sonra projenin temel hazırlık ve planlamaları yapılır. Bu safhada aşağıdaki işler yerine getirilir:

- Proje konfigürasyon yönetimi ortamının oluşturulması sağlanır.
- Proje planlama aktiviteleri (kapsam, işgücü, bütçe, takvim, izleme, risk, kalite, konfigürasyon, alt yüklenici, paydaş yönetimi vb.) yerine getirilir.
- Proje planlama aşamasında proje yönetim planı, kalite güvence planı, konfigürasyon yönetim planı, doğrulama ve geçerleme planı, test ve değerlendirme ana planı, risk yönetim planı, yazılım geliştirme planı, donanım geliştirme planı, sistem mühendisliği yönetim planı gibi bir projede işletilecek tüm süreçlerin planlamasının olduğu dokümanlar hazırlanır.
- Proje bütünleşik planlarının kurum içi gözden geçirilmesi kapsamında gözden geçirilmeye tabi tutulacak dokümanlarla ilgili proje çalışanları gözden geçirmeye katılır. Gözden geçirme sonucu bulunan bulgular doküman sahibine iletilir. Doküman sahibi gelen görüşleri analiz eder. Kabul edilen değişiklikler dokümana yansıtılarak dokümanın yeni sürümü çıkarılır. Kabul edilmeyen bulgular, bulgu sahibine gerekçesi ile birlikte bildirilir.
- Kurum içi gözden geçirme süreci tamamlanan dokümanlar müşteriye gönderilir. Müşteri tarafından gözden geçirilen dokümanlar ile ilgili bulgular proje ekibine iletilir. Proje ekibi müşteri bulgularını analiz eder. Müşteri ile mutabık kalınarak değişiklikler dokümanlara yansıtılır ve dokümanların yeni sürümleri oluşturulur. Yeni sürüm dokümanlar gözden geçirilmesi için tekrar müşteriye gönderilir. Dokümanlarda hata kalmayana kadar bu süreç devam eder. Aşama sonunda proje dokümanları müşteri ve proje yönetimi tarafından imzalanarak kayıt altına alınır.



Şekil 2.28. Yazılım yaşam döngüsü süreç etkileşim haritası

3. Proje analiz aşaması

Analizin aşamasında, geliştirilecek yazılımdan beklenen fonksiyonlar ve gereksinimler tanımlanır. Analiz faaliyetleri yürütülürken izlenecek temel adımlar aşağıda belirtilmiştir:

İhtiyaçları topla: Müşterinin istediği yazılım ürününün özellikleri müşteri ile görüşmeler yapılarak toplanır. Belirlenmiş olan gereksinim kaynaklarından ihtiyaçları toplamak için gereksinim temin etme yöntemleri kullanılır.

Müşteri gereksinimlerini geliştir: Müşterinin ihtiyaçları mühendislik dili kullanılarak müşteri gereksinimlerine dönüştürülür ve önceliklendirilir.

Müşteri gereksinimleri geçerle ve taahhüt al: Müşteri gereksinimleri dokümantasyonunun tamamlanmasının ardından proje yöneticisi, sistem mühendisliği ekibi, kalite sorumlusu ve gerekli proje çalışanlarının kurum içi teknik gözden geçirme toplantısı yapılır. Proje sözleşmesinde tanımlanmış ise, kurum içi teknik gözden geçirmesi tamamlanmış olan müşteri gereksinimleri, müşteri ile birlikte gözden geçirilir. Gözden geçirme sonunda müşteri gereksinimleri üzerinde karşılıklı mutabakata varılır ve sözleşmede belirtildi ise müşterinin imzalı onayı alınır.

Ürün (sistem/yazılım/donanım/arayüz) gereksinimlerini geliştir: Projede müşteri gereksinimlerin belirlenip kayıt altına alınmasından sonraki evre ürün gereksinimlerin geliştirilmesidir. Projelerde bu evre üç alt adımda gerçekleştirilir.

- Sistem gereksinimlerinin geliştirilmesi,
- Yazılım gereksinimlerinin geliştirilmesi,
- Donanım gereksinimlerinin geliştirilmesi.

Ürün gereksinimlerini geçerle ve taahhüt al: Ürün gereksinimleri dokümantasyonunun tamamlanmasının ardından proje yöneticisi, sistem mühendisliği ekibi, kalite sorumlusu ve gerekli proje çalışanlarının katılacağı kurum içi teknik gözden geçirme toplantısı yapılır. Proje sözleşmesinde tanımlanmış ise, kurum içi teknik gözden geçirmesi tamamlanmış olan gereksinimler, müşteri ile birlikte gözden geçirilir. Gözden geçirme sonunda ürün gereksinimleri üzerinde karşılıklı mutabakata varılır ve sözleşmede belirtildi ise müşterinin imzalı onayı alınır.

Gereksinimleri Yönet: Gereksinimlerin yönetilmesi için öncelikle gerekli araç altyapı kurulur ve işletilmeye başlanır. Gereksinimler araca kaydedilir, olabilecek potansiyel değişiklikler araçta kayıt altına alınır ve yönetilir. Gereksinimler arası izlenebilirlikler kurulur ve gereksinimlerin güncelliği, doğruluğu ve tamlığı bu araç üzerinden kontrol edilir.

4. Proje gerçekleştirme aşaması

Bu aşama tasarım, kodlama ve tümleştirme alt aşamalarından oluşur. Tasarım aşamasında, analiz aşamasında belirlenen ihtiyaçlara çözüm olacak yazılımın ilk ve temel versiyonu oluşturulur. Bu aşamada kodlama aktivitesi yapılmaz. Tasarım faaliyetleri yürütülürken izlenecek adımlar temel hatlarıyla aşağıda belirtilmiştir:

Sistem tasarımı: Sistemi oluşturacak olan yazılım ve donanım bileşenleri belirlenir. Burada detaylı tasarım anlamında bileşenler değil, eldeki dokümanlardan kolaylıkla çıkarılabilecek sistemi oluşturan temel parçalar kastedilmektedir.

Yazılım tasarımı: Yazılım tasarımı, yazılım gereksinimlerinin yazılım sistemi gösterimine dönüştürüldüğü süreçtir. Başlangıçta gösterim, yazılımın üst seviye görünüşünü tanımlar. İzleyen geliştirmelerle, kaynak koda çok yakın tasarım gösterimi elde edilir. Yazılım tasarımı süreci, mimari tasarım ve ayrıntılı tasarım olmak üzere iki adımda gerçekleştirilir.

Mimari tasarımda, yazılım gereksinimleri veri ve yazılım mimarisine dönüştürülür.

Ayrıntılı tasarımda ise yazılımın ayrıntılı veri yapısı ve algoritmik gösterimini sağlayacak mimari gösterimi geliştirilerek detaylandırılır.

Donanım tasarımı: Genel anlamda ürün bileşenleri hiyerarşisindeki üst seviye bileşenin tasarımı bir alt seviyedeki ürün bileşenlerinin belirtimi ve tanımlanması şeklinde olur. Bu adımda gerçekleştirilen faaliyetler, donanım bileşenleri için türetilmiş gereksinimlerin oluşturulmaları ve atamaları, iyileştirilmesi ve doğrulamasını da içerir.

Kodlama: Öncelikle kodlama aşamasının girdileri belirlenir, daha sonra yazılım kodlanmaya başlanır. Kodlaması tamamlanan yazılım birimlerinin birim testi

gerçekleştirilir. Birim testleri başarı ile tamamlanan yazılım birimleri, yazılım bileşenlerini oluşturmak üzere tümleştirilir.

Tümleştirme: Ürün bileşenlerinden nihai ürüne ulaşmak için yürütülecek etkinlikleri tanımlar. Tümleştirme, tek adımda ya da aşamalı olarak ürün bileşenlerinin tanımlanan sırada ve tanımlanan prosedürlere uygun olarak birleştirilmesi ile gerçekleşir. Ayrıca ürünün müşteriye ulaştırılması için kullanılacak teslimat kanalları ve ilgili prosedürler tanımlanır.

5. Proje test aşaması

Projede gerçekleştirilecek tüm gözden geçirme ve test faaliyetleri planlanır ve gerçekleştirilir. Bu kapsamda; proje dokümanları ve yazılım bileşenleri gözden geçirmeler ile doğrulanır. Yazılım gereksinimleri test faaliyetleri ile geçerlenir. Test aşamasında;

- Sistem/yazılım/donanım/kabul test faaliyetleri planlanır,
- Sistem/yazılım/donanım/kabul test durumları oluşturulur,
- Testler koşturulur ve ortaya çıkan hatalar düzeltilir,
- Test sonuçları raporlanır ve sonuçlar analiz edilir.

6. Proje bakım aşaması

Proje sözleşmesinde tanımlanan yazılım ürünü müşteriye teslim edilir ve bakım aşaması başlatılır. Bakım aşamasında, müşteri ortamına kurulan yazılımda sorun giderme ve güncelleme gibi işlemler gerçekleştirilir.

2.8. Yazılım Geliştirme Yaşam Döngüsü Süreç Metrikleri

Bu bölümde, Tablo 2.4'te belirtilen proje başarı puanı hesaplama modeline girdi olacak yazılım geliştirme yaşam döngüsü süreç metrikleri "Ölçüm Bilgi Modeli" şablonuna uygun tanımlanmıştır.

Tanımlanan yazılım geliştirme yaşam döngüsü metriklerine ait detay bilgiler tablo şeklinde verilmiştir.

Tablo 2.11. Kilometre taşı sapması ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projenin zaman planına ne kadar uyumlu devam ettiğini ölçmeyi amaçlar. Projede tanımlanan kilometre taşlarının, proje takvimine uygun olup olmadığını tespit etmek ve böylece proje planından sapmayı ölçmektir. Proje başlangıcından itibaren ölçüm yapılan döneme kadar birden fazla kilometre taşı varsa medyan ile ortalama sapma değeri bulunur.
Ölçülebilir Kavram	Kilometre taşı sapması
İlişkili Varlıklar	1. Proje yönetimi süreci 2. Proje takvimi
Nitelikler	1. Proje takviminde planlanan kilometre taşı bitiş tarihleri 2. Kilometre taşı süre girişleri
Temel Ölçümler	1. Planlanan kilometre taşı başlangıç tarihi 2. Planlanan kilometre taşı bitiş tarihi 3. Gerçekleşen kilometre taşı bitiş tarihi
Ölçme Yöntemi	1. Proje yönetimi sisteminden planlanan kilometre taşı başlama tarihi bilgisinin alınması 2. Proje yönetimi sisteminden planlanan kilometre taşı bitiş tarihi bilgisinin alınması 3. Proje yönetimi sisteminden gerçekleşen kilometre taşı bitiş tarihi bilgisinin alınması
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel 3. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı 3. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran 3. Oran
Ölçüm Birimi	1. Gün (süre) 2. Gün (süre) 3. Gün (süre)
Türetilmiş Ölçüm	Kilometre taşı sapması (KS)
Ölçme Fonksiyonu	$$\text{KS} = \frac{\text{Gerçekleşen Bitiş} - \text{Planlanan Bitiş}}{\text{Planlanan Bitiş} - \text{Planlanan Başlangıç}}$$ Eğer ölçüm yapılan tarihten önce bitmesi gereken bir kilometre taşı varsa sapma aşağıdaki formül ile hesaplanır: $$\text{KS} = \frac{\text{Ölçüm Tarihi} - \text{Planlanan Bitiş}}{\text{Planlanan Bitiş} - \text{Planlanan Başlangıç}}$$
Gösterge	Sapma Oranı

Tablo 2.11.(Devam) Kilometre taşı sapması ölçüm bilgi formu

Veri Kaynağı	Proje yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Tüm kilometre taşı sapma değerlerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın verileri toplanır. Kontrol grafiği oluşturularak alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Ölçüm yapılan proje temelinde ortalama kilometre taşı sapması için projenin başından itibaren ölçüm yapılan dönem sonuna kadar olan kilometre taşı sapmalarının medyanı alınır.

Organizasyon temelinde ortalama kilometre taşı sapmasını belirlemek için tüm projelerin kilometre taşı sapma değerlerinin medyanı alınır.

Tablo 2.12. Efor sapması ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projenin işgücü planına ne kadar uyumlu devam ettiğini ölçmeyi amaçlar. Proje işgücü kullanımının planlanan işgücü kullanımına uygun olup olmadığını tespit eder ve böylece proje planından sapmayı ölçer.
Ölçülebilir Kavram	Efor sapması
İlişkili Varlıklar	1. Proje yönetimi süreci 2. Proje yönetim sisteminde proje işgücü bilgileri
Nitelikler	1. Proje yönetim sistemine işgücü girişleri
Temel Ölçümler	1. Planlanan işgücü bilgileri 2. Gerçekleşen işgücü bilgileri
Ölçme Yöntemi	1. Proje yönetimi sisteminden planlanan işgücü bilgisinin alınması 2. Proje yönetimi sisteminden gerçekleşen işgücü bilgisinin alınması
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza gerçek sayı 2. Sıfırdan sonsuza gerçek sayı

Tablo 2.12.(Devam) Efor sapması ölçüm bilgi formu

Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Adam x ay 2. Adam x ay
Türetilmiş Ölçüm	Efor sapması (ES)
Ölçme Fonksiyonu	$\text{ES} = \frac{\text{Toplam gerçekleşen işgücü} - \text{Toplam planlanan işgücü}}{\text{Toplam planlanan işgücü}}$
Gösterge	Sapma Oranı
Veri Kaynağı	Proje yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Tüm efor sapma değerlerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın verileri toplanır. Kontrol grafiği oluşturularak alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama efor sapmasını belirlemek için tüm projelerin efor sapma değerlerinin medyanı alınır.

Tablo 2.13. Gelir/gider endeksi ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projenin ilgili dönem sonuna kadar elde ettiği gelirlerin ve gerçekleştirdiği harcamaların oranı ile proje yönetim süreci uygulama performansının eğilimini tespit etmeyi amaçlar. Bir başka deyişle fatura ve harcama dönemlerinin uygun şekilde planlanıp planlanmadığını ölçer.
Ölçülebilir Kavram	Gelir-Gider endeksi
İlişkili Varlıklar	1. Proje yönetimi süreci 2. Hak ediş ve harcama bilgileri
Nitelikler	2. Proje yönetim sistemindeki hak ediş ve harcama bilgileri
Temel Ölçümler	1. Planlanan hak ediş bilgileri 2. Gerçekleşen harcama bilgileri
Ölçme Yöntemi	1. Proje yönetimi sisteminden planlanan hak ediş bilgisinin alınması 2. Proje yönetimi sisteminden gerçekleşen harcama bilgisinin alınması
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel

Tablo 2.13.(Devam) Gelir/gider endeksi ölçüm bilgi formu

Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Bütçe 2. Bütçe
Türetilmiş Ölçüm	Gelir-gider sapması (GGS)
Ölçme Fonksiyonu	$\text{GGS} = \frac{\text{Elde edilen gelir toplamı}}{\text{Gerçekleştirilen gider toplamı}}$ Gelir ve gider toplamları, proje başlangıcından ölçümün yapıldığı döneme kadar elde edilen gelirleri ve giderleri kapsar.
Gösterge	Sapma Oranı
Veri Kaynağı	Proje yönetim araçları
Veri Toplama Sıklığı	3 Ay
Model	Tüm hak ediş ve harcama bilgilerinin kullanılarak elde edilen oranın 1'den büyük ya da küçük olmasına göre değerlendirme yapılması
Karar Kriteri	Sonuç değeri, projenin planlandığı şekilde sonuçlanması için 1'e eşit ya da büyük olmalıdır. 1'den küçük olduğu durumlarda sonuçlar kök neden analizi gerektirir.

Tablo 2.14. Proje başına ortalama gereksinim değişikliği sayısı ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Yürütülen projelerde yaşanan gereksinim değişikliklerinin merkezî eğilimini ve süreç değişkenliğini tespit etmeyi amaçlar. Projede ilgili dönem sonuna kadar gerçekleşen gereksinim değişikliklerinin organizasyonun süreç davranışına uygun olup olmadığını denetler.
Ölçülebilir Kavram	Gereksinim durumu
İlişkili Varlıklar	1. Gereksinim yönetimi süreci 2. Gereksinim değişiklikleri
Nitelikler	1. Gereksinim değişiklik kayıtları 2. Güncellenmiş gereksinim dokümanları
Temel Ölçümler	1. Gereksinim değişiklik sayısı 2. Gereksinim değişikliğine maruz kalmış proje sayısı
Ölçme Yöntemi	1. Gereksinim yönetimi aracından değişiklik sayılarının tespit edilmesi 2. Gereksinim yönetimi aracından gereksinim değişikliğine maruz kalmış projelerin sayılması
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel

Tablo 2.14.(Devam) Proje başına ortalama gereksinim değişikliği sayısı ölçüm bilgi formu

Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Değişiklik sayısı 2. Birim sayı
Türetilmiş Ölçüm	Proje Başına Ortalama Gereksinim Değişikliği Sayısı (OGDS)
Ölçme Fonksiyonu	$\text{OGDS} = \frac{\text{Gereksinim değişikliği sayısı}}{\text{Gereksinim değişikliğine maruz kalan proje sayısı}}$
Gösterge	Ortalama değişiklik
Veri Kaynağı	Gereksinim yönetim araçları
Veri Toplama Sıklığı	3 Ay
Model	Organizasyonun tüm projelerine ait gereksinim değişiklikleri kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje gereksinim değişiklik verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama gereksinim değişikliği sayısını belirlemek için tüm projelerin ortalama gereksinim değişikliği değerlerinin medyanı alınır.

Tablo 2.15. Müşteri tarafından raporlanan hataların açık kalma süresi ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede müşteri tarafından raporlanan hataların ortalama olarak ne kadar süre açık kaldığını tespit eder. Organizasyonda yürütülen projelerde müşteri tarafından iletilen hataların açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Hata durumu
İlişkili Varlıklar	1. Doğrulama ve geçerleme süreci 2. Hata kayıtları
Nitelikler	1. Hataların sahibi 2. Hataların durumları
Temel Ölçümler	1. Toplam müşteri tarafından raporlanan ve çözülerek kapatılmış hatalar 2. Hataların açılmasından kapatılmasına kadar geçen süre

Tablo 2.15.(Devam) Müşteri tarafından raporlanan hataların açık kalma süresi ölçüm bilgi formu

Ölçme Yöntemi	1. Hata yönetimi aracından müşteri hata sayılarının tespiti 2. Müşteri hatalarının toplam açık kalma sürelerinin tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza gerçek sayı 2. Sıfırdan sonsuza gerçek sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	3. Hata sayısı 4. Gün (süre)
Türetilmiş Ölçüm	Müşteri Tarafından Raporlanan Hataların Ortalama Açık Kalma Süresi (HAKS)
Ölçme Fonksiyonu	$\text{HAKS} = \begin{cases} \text{MTR Hata kapanış tarihi} - \text{MTR Hata oluşma tarihi (kapanan)} \\ \text{Ölçüm tarihi} - \text{MTR Hata oluşma tarihi (henüz açık)} \end{cases}$ Ölçüm alınan tarihte açık durumda bulunan hatalar için ölçüm alınan tarih hata kapanma tarihi olarak kabul edilir.
Gösterge	Ortalama hata süresi
Veri Kaynağı	Hata yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Projeye ait müşteri hatalarının kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje hata verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama hata açık kalma süresini belirlemek için tüm projelerin ortalama hata açık kalma süre değerlerinin medyanı alınır.

Tablo 2.16. Müşteri tarafından raporlanan iyileştirmelerin açık kalma süresi ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede müşteri tarafından raporlanan iyileştirmelerin ortalama olarak ne kadar süre açık kaldığını tespit eder. Organizasyonda yürütülen projelerde müşteri tarafından iletilen iyileştirmelerin açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	İyileştirme durumu

Tablo 2.16.(Devam) Müşteri tarafından raporlanan iyileştirmelerin açık kalma süresi ölçüm bilgi formu

İlişkili Varlıklar	1. Doğrulama ve geçerleme süreci 2. İyileştirme kayıtları
Nitelikler	1. İyileştirme taleplerinin sahibi 2. İyileştirme taleplerinin durumları
Temel Ölçümler	1. Toplam müşteri tarafından raporlanan ve çözülerek kapatılmış iyileştirmeler 2. İyileştirme taleplerinin açılmasından kapatılmasına kadar geçen süre
Ölçme Yöntemi	1. Müşteri iyileştirme talep sayısının tespiti 2. Müşteri iyileştirme taleplerinin toplam açık kalma sürelerinin tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza gerçek sayı 2. Sıfırdan sonsuza gerçek sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. İyileştirme sayısı 2. Gün (süre)
Türetilmiş Ölçüm	Müşteri Tarafından Raporlanan İyileştirmelerin Ortalama Açık Kalma Süresi (İAKS)
Ölçme Fonksiyonu	$\text{İAKS} = \begin{cases} \text{MTR İyileştirme kapanış tarih} - \text{MTR İyileştirme oluşma tarihi (kapanan)} \\ \text{Ölçüm tarihi} - \text{MTR İyileştirme oluşma tarihi (henüz açık)} \end{cases}$ Ölçüm alınan tarihte açık durumda bulunan iyileştirme için ölçüm alınan tarih iyileştirme kapanma tarihi olarak kabul edilir.
Gösterge	İyileştirme hata süresi
Veri Kaynağı	Müşterinin iyileştirme taleplerinin kayıt altına alındığı araçlar
Veri Toplama Sıklığı	Aylık
Model	Projeye ait müşteri iyileştirme talep bilgilerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje iyileştirme verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama iyileştirme açık kalma süresini belirlemek için tüm projelerin ortalama iyileştirme açık kalma süre değerlerinin medyanı alınır.

Tablo 2.17. Müşteri tarafından raporlanan gereksinim değişikliklerinin açık kalma süresi ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede müşteri tarafından raporlanan gereksinim değişiklik taleplerinin ortalama olarak ne kadar süre açık kaldığını tespit eder. Organizasyonda yürütülen projelerde müşteri tarafından iletilen gereksinim değişikliklerinin açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Gereksinim değişiklik durumu
İlişkili Varlıklar	1. Yazılım geliştirme süreci 2. Gereksinim değişiklik kayıtları
Nitelikler	1. Gereksinim değişiklik taleplerinin sahibi 2. Gereksinim değişiklik taleplerinin durumları
Temel Ölçümler	1. Toplam müşteri tarafından raporlanan ve çözülerek kapatılmış gereksinim değişiklikleri 2. Gereksinim değişiklik taleplerinin açılmasından kapatılmasına kadar geçen süre
Ölçme Yöntemi	1. Müşteri gereksinim değişiklik talep sayısının tespiti 2. Müşteri gereksinim değişiklik taleplerinin toplam açık kalma sürelerinin tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza gerçek sayı 2. Sıfırdan sonsuza gerçek sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Gereksinim değişiklikleri sayısı 2. Gün (süre)
Türetilmiş Ölçüm	Müşteri Tarafından Raporlanan Gereksinim Değişikliklerinin Ortalama Açık Kalma Süresi (GAKS)
Ölçme Fonksiyonu	$\text{GAKS} = \begin{cases} \text{MTR Gereksinim kap. tarih} - \text{MTR Gereksinim oluşma tarihi (kapanan)} \\ \text{Ölçüm tarihi} - \text{MTR Gereksinim oluşma tarihi (henüz açık)} \end{cases}$ Ölçüm alınan tarihte açık durumda bulunan gereksinim değişiklikleri için ölçüm alınan tarih gereksinim değişiklik kapanma tarihi olarak kabul edilir.
Gösterge	Gereksinim değişiklik süresi
Veri Kaynağı	Müşterinin gereksinim değişiklik taleplerinin kayıt altına alındığı araçlar
Veri Toplama Sıklığı	Aylık
Model	Projeye ait müşteri gereksinim değişiklik talep bilgilerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması

Tablo 2.17.(Devam) Müşteri tarafından raporlanan gereksinim değişikliklerinin açık kalma süresi ölçüm bilgi formu

Karar Kriteri	Son bir yılın proje gereksinim değişiklik verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama gereksinim değişiklik açık kalma süresini belirlemek için tüm projelerin ortalama gereksinim değişiklik açık kalma süre değerlerinin medyanı alınır.

Tablo 2.18. Müşteri tarafından raporlanan hataların açık kalma oranı ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede müşteri tarafından raporlanan hataların oran olarak ne kadarının kapanmadığını tespit eder. Organizasyonda yürütülen projelerde müşteri tarafından iletilen hataların açık kalma oranlarının merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Hata durumu
İlişkili Varlıklar	1. Doğrulama ve geçerleme süreci 2. Hata kayıtları
Nitelikler	1. İyileştirme taleplerinin sahibi 2. İyileştirme taleplerinin durumları
Temel Ölçümler	1. Toplam müşteri tarafından raporlanan ve halen açık iyileştirmeler 2. Müşteri tarafından raporlanan toplam iyileştirme sayısı
Ölçme Yöntemi	1. Hata yönetimi aracından kapanmamış müşteri iyileştirme sayısının tespiti 2. Müşteri hatalarının toplam sayısının tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Hata sayısı 2. Hata sayısı
Türetilmiş Ölçüm	Müşteri Tarafından Raporlanan Hataların Ortalama Açık Kalma Oranı (HAKO)
Ölçme Fonksiyonu	$\text{HAKO} = \frac{\text{Müşterinin raporladığı ve halen kapatılmamış olan geçerli hata kaydı sayısı}}{\text{Müşterinin raporladığı geçerli olan tüm hata kayıtlarının sayısı}}$

Tablo 2.18.(Devam) Müşteri tarafından raporlanan hataların açık kalma oranı ölçüm bilgi formu

Gösterge	Kapatılmamış hataların oransal gösterimi
Veri Kaynağı	Hata yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Projeye ait müşteri hatalarının kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje hata verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama hata açık kalma oranını belirlemek için tüm projelerin ortalama hata açık kalma oran değerlerinin medyanı alınır.

Tablo 2.19. Müşteri tarafından raporlanan iyileştirmelerin açık kalma oranı ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede müşteri tarafından raporlanan iyileştirmelerin oran olarak ne kadarının kapanmadığını tespit eder. Organizasyonda yürütülen projelerde müşteri tarafından iletilen iyileştirmelerin açık kalma oranlarının merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	İyileştirme durumu
İlişkili Varlıklar	1. Doğrulama ve geçerleme süreci 2. İyileştirme kayıtları
Nitelikler	1. Hataların sahibi 2. Hataların durumları
Temel Ölçümler	1. Toplam müşteri tarafından raporlanan ve halen açık iyileştirmeler 2. Müşteri tarafından raporlanan toplam iyileştirme sayısı
Ölçme Yöntemi	1. Kapanmamış müşteri iyileştirme sayısının tespiti 2. Müşteri iyileştirmelerinin toplam sayısının tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. İyileştirme sayısı 2. İyileştirme sayısı

Tablo 2.19.(Devam) Müşteri tarafından raporlanan iyileştirmelerin açık kalma oranı ölçüm bilgi formu

Türetilmiş Ölçüm	Müşteri Tarafından Raporlanan İyileştirmelerin Ortalama Açık Kalma Oranı (İAKO)
Ölçme Fonksiyonu	$\text{İAKO} = \frac{\text{Müşterinin raporladığı ve halen kapatılmamış olan geçerli iyileştirme sayısı}}{\text{Müşterinin raporladığı geçerli olan tüm iyileştirme kayıtlarının sayısı}}$
Gösterge	Kapatılmamış iyileştirme taleplerinin oransal gösterimi
Veri Kaynağı	Müşterinin iyileştirme taleplerinin kayıt altına alındığı araçlar
Veri Toplama Sıklığı	Aylık
Model	Projeye ait müşteri iyileştirme kayıtlarının kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje iyileştirme verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama iyileştirme açık kalma oranını belirlemek için tüm projelerin ortalama iyileştirme açık kalma oran değerlerinin medyanı alınır.

Tablo 2.20. Müşteri tarafından raporlanan gereksinim değişikliklerinin açık kalma oranı ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede müşteri tarafından raporlanan gereksinim değişiklik taleplerinin ortalama olarak ne kadar açık kaldığını tespit eder. Organizasyonda yürütülen projelerde müşteri tarafından iletilen gereksinim değişikliklerinin açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Gereksinim değişiklik durumu
İlişkili Varlıklar	1. Yazılım geliştirme süreci 2. Gereksinim değişiklik kayıtları
Nitelikler	1. Gereksinim değişiklik taleplerinin sahibi 2. Gereksinim değişiklik taleplerinin durumları
Temel Ölçümler	1. Toplam müşteri tarafından raporlanan ve halen açık gereksinim değişiklikleri 2. Müşteri tarafından raporlanan toplam gereksinim değişiklik sayısı
Ölçme Yöntemi	1. Kapanmamış müşteri gereksinim değişiklik sayısının tespiti 2. Müşteri gereksinim değişikliklerinin toplam sayısının tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel

Tablo 2.20.(Devam) Müşteri tarafından raporlanan gereksinim değişikliklerinin açık kalma oranı ölçüm bilgi formu

Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Gereksinim değişiklikleri sayısı 2. Gün (süre)
Türetilmiş Ölçüm	Müşteri Tarafından Raporlanan Gereksinim Değişikliklerinin Ortalama Açık Kalma Oranı (GAKO)
Ölçme Fonksiyonu	$\text{GAKO} = \frac{\text{Müşterinin raporladığı ve kapatılmamış gereksinim değişiklik sayısı}}{\text{Müşterinin raporladığı tüm gereksinim değişiklik kayıtlarının sayısı}}$
Gösterge	Kapatılmamış gereksinim değişikliklerinin oransal gösterimi
Veri Kaynağı	Müşterinin gereksinim değişiklik taleplerinin kayıt altına alındığı araçlar
Veri Toplama Sıklığı	Aylık
Model	Projeye ait müşteri gereksinim değişiklik talep bilgilerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje gereksinim değişiklik verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama gereksinim değişiklik açık kalma oranını belirlemek için tüm projelerin ortalama gereksinim değişiklik açık kalma oran değerlerinin medyanı alınır.

Tablo 2.21. Hata yoğunluğu bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede üretilen ürünlerin kalitesini ölçmeyi amaçlar. Bu değerin düşmesi, iş ürünün kalitesinin artmasını ifade eder. Organizasyonda yürütülen projelerde üretilen ürünlerin kalitesinin merkezî eğilimini tespit eder ve süreç değişkenliğini belirler. Bu ölçüm müşteriye teslim edilen yazılım ürünlerinde bulunan hataları kapsar.
Ölçülebilir Kavram	Yazılım ürün kalitesi
İlişkili Varlıklar	1. Yazılım geliştirme süreci 2. Hata kayıtları
Nitelikler	1. Hata durumları 2. Ürün büyüklüğü

Tablo 2.21.(Devam) Hata yoğunluğu bilgi formu

Temel Ölçümler	1. Dış sürümde bulunan toplam hata sayısı 2. Dış sürüme ait mantıksal kod satır sayısı Mantıksal kod satır sayısı, toplam kod satır sayısından açıklama satır sayısının çıkarılması ile elde edilir.
Ölçme Yöntemi	1. Hata yönetim aracından toplam kapatılmış hata sayısının tespiti 2. Konfigürasyon yönetim aracından test edilen yazılım ürününün satır sayısının tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Hata sayısı 2. Satır sayısı
Türetilmiş Ölçüm	Hata Yoğunluğu (HY)
Ölçme Fonksiyonu	$\text{HY} = \frac{\text{Dış sürümde bulunan toplam hata sayısı}}{\text{Dış sürüme ait mantıksal kod satır sayısı}/1000}$
Gösterge	Hata yoğunluğu
Veri Kaynağı	Hata yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Projenin müşteriye teslim edilen yazılım ürünlerine ait çözülmüş ve kapatılmış hata bilgilerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje hata verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama hata yoğunluğunu belirlemek için tüm projelerin ortalama hata yoğunluk değerlerinin medyanı alınır.

Tablo 2.22. Gözden geçirme hazırlık hızı bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede gerçekleştirilen gözden geçirme faaliyetlerinin hazırlığı için yeterli sürenin ayrılıp ayrılmadığını ölçer. Organizasyonda yürütülen projelerde gözden geçirilen ürün büyüklüklerinin merkezî eğilimini tespit eder ve süreç değişkenliğini belirler.

Tablo 2.22.(Devam) Gözden geçirme hazırlık hızı bilgi formu

Ölçülebilir Kavram	Yazılım ürün kalitesi
İlişkili Varlıklar	1. Doğrulama ve geçerleme süreci 2. Gözden geçirme kayıtları
Nitelikler	1. Bulgu bilgileri 2. Ürün büyüklüğü
Temel Ölçümler	1. Gözden geçirme için harcanan toplam hazırlık süresi 2. Gözden geçirilen ürünün büyüklüğü
Ölçme Yöntemi	1. Gözden geçirme yapılan araçtan gözden geçirme için harcanan toplam süre tespiti 2. Konfigürasyon yönetim aracından gözden geçirilen yazılım ürünün büyüklüğü ya da yazılımla ilgili dokümanların büyüklüğünün tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza gerçek sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Saat 2. Satır sayısı, sayfa sayısı
Türetilmiş Ölçüm	Gözden Geçirme Hazırlık Hızı (GGH)
Ölçme Fonksiyonu	$$\mathrm{GGH} = \frac{\mathrm{GGB}}{\mathrm{THS}}$$ GGB: Ürün yazılım ürünü ise statik kod analizi araçlarından elde edilen mantıksal kod sayısı (LLOC), doküman ise birim boyut cinsinden sayfa sayısı THS: Gözden geçirme faaliyeti için harcanan toplam hazırlık süresi (hazırlık yapan her bir kişinin yaptığı hazırlık sürelerinin toplamı)
Gösterge	Gözden geçirme verimliliği
Veri Kaynağı	Gözden geçirme araçları
Veri Toplama Sıklığı	Aylık
Model	Projede gerçekleştirilen gözden geçirmelerde ortaya çıkan bulgular kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje gözden geçirme faaliyetleri sonucu bulgu verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde gözden geçirme hazırlık hızını belirlemek için tüm projelerin ortalama gözden geçirme hazırlık hızı değerlerinin medyanı alınır.

Tablo 2.23. Bulgu yoğunluğu bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede gerçekleştirilen gözden geçirme bulgu yoğunluğunu ölçmeyi amaçlar. Organizasyonda yürütülen projelerde gözden geçirme bulgu yoğunluğunun merkezî eğilimini tespit eder ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Yazılım ürün kalitesi
İlişkili Varlıklar	1. Doğrulama ve geçerleme süreci 2. Gözden geçirme kayıtları
Nitelikler	1. Bulgu bilgileri 2. Ürün büyüklüğü
Temel Ölçümler	1. Toplam bulgu sayısı 2. Gözden geçirilen ürünün büyüklüğü
Ölçme Yöntemi	1. Gözden geçirme aracından toplam bulgu sayısının tespiti 2. Konfigürasyon yönetim aracından gözden geçirilen yazılım ürünün büyüklüğü ya da yazılımla ilgili dokümanların büyüklüğünün tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Bulgu sayısı 2. Satır sayısı, sayfa sayısı
Türetilmiş Ölçüm	Bulgu Yoğunluğu (BY)
Ölçme Fonksiyonu	$$\mathrm{BY} = \frac{\mathrm{BS}}{\mathrm{GGB}}$$ BS: Gözden geçirmelerde tespit edilen toplam bulgu sayısı GGB: Ürün yazılım ürünü ise statik kod analizi araçlarından elde edilen mantıksal kod sayısı (LLOC), doküman ise birim boyut cinsinden sayfa sayısı
Gösterge	Gözden geçirme verimliliği
Veri Kaynağı	Gözden geçirme araçları
Veri Toplama Sıklığı	Aylık

Tablo 2.23.(Devam) Bulgu yoğunluğu bilgi formu

Model	Projede gerçekleştirilen gözden geçirmelerde ortaya çıkan bulgular kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje gözden geçirme faaliyetleri sonucu bulgu verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde bulgu yoğunluğunu belirlemek için tüm projelerin bulgu yoğunluğu değerlerinin medyanı alınır.

Tablo 2.24. Gözden geçirme hazırlık verimliliği bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede gerçekleştirilen gözden geçirme faaliyetlerinin verimliliğini ölçmeyi amaçlar. Organizasyonda yürütülen projelerde gözden geçirme verimliliğinin merkezî eğilimini tespit eder ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Yazılım ürün kalitesi
İlişkili Varlıklar	1. Doğrulama ve geçerleme süreci 2. Gözden geçirme kayıtları
Nitelikler	1. Bulgu bilgileri
Temel Ölçümler	1. Toplam bulgu sayısı 2. Kişi başı ortalama gözden geçirme süresi
Ölçme Yöntemi	1. Gözden geçirme aracından toplam bulgu sayısının tespiti 2. Gözden geçirme aracından gözden geçirme için harcanan kişi başı süre tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza gerçek sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Bulgu sayısı 2. Saat
Türetilmiş Ölçüm	Gözden Geçirme Hazırlık Verimliliği (GGHV)
Ölçme Fonksiyonu	$\mathrm{GGHV} = \frac{\mathrm{BS}}{\mathrm{KBOGGS}}$ BS: Gözden geçirmelerde tespit edilen toplam bulgu sayısı KBOGGS: gözden geçirme için harcanan toplam sürenin gözden geçirmeye katılan kişi sayısına oranı

Tablo 2.24.(Devam) Gözden geçirme hazırlık verimliliği bilgi formu

Gösterge	Gözden geçirme verimliliği
Veri Kaynağı	Gözden geçirme araçları
Veri Toplama Sıklığı	Aylık
Model	Projede gerçekleştirilen gözden geçirmelerde ortaya çıkan bulgular kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje gözden geçirme faaliyetleri sonucu bulgu verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde hazırlık verimlilik değerini belirlemek için tüm projelerin hazırlık verimliliği değerlerinin medyanı alınır.

Tablo 2.25. Hata yakalama yüzdesi ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	İş ürünleri müşteri kabulüne sunulmadan önce hatanın ne kadarının geliştirme ve test faaliyetlerinde proje personeli tarafından tespit edilebildiğinin göstergesidir. Organizasyonda yürütülen projelerde iş ürünleri müşteri kabulüne sunulmadan önce hatanın ne kadarının geliştirme ve test faaliyetlerinde proje personeli tarafından tespit edilebildiğinin merkezî eğilimini tespit eder ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Yazılım ürün kalitesi
İlişkili Varlıklar	1. Doğrulama ve geçerleme süreci 2. Hata kayıtları
Nitelikler	1. Hataların sahibi 2. Hataların durumları
Temel Ölçümler	1. Toplam müşteri tarafından raporlanan ve kapatılmış hata sayısı 2. Test sürümlerinde bulunan ve kapatılan hata sayısı 3. Geliştirme sürümlerinde bulunan ve kapatılan hata sayısı
Ölçme Yöntemi	1. Hata yönetimi aracından müşteri hata sayılarının tespiti 2. Hata yönetimi aracından test sürümlerindeki hata sayılarının tespiti 3. Hata yönetimi aracından geliştirme aşamasındaki hata sayılarının tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel 3. Nesnel

Tablo 2.25.(Devam) Hata yakalama yüzdesi ölçüm bilgi formu

Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı 3. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran 3. Oran
Ölçüm Birimi	1. Hata sayısı 2. Hata sayısı 3. Hata sayısı
Türetilmiş Ölçüm	Hata Yakalama Yüzdesi (HYY)
Ölçme Fonksiyonu	$$\text{HYY} = \frac{\text{TSH} + \text{GSH}}{\text{TSH} + \text{GSH} + \text{DSH}} * 100$$ TSH: Yazılımın test sürümlerinde bulunan hata sayısı GSH: Yazılımın geliştirme sürümlerinde bulunan hata sayısı DSH: Dış sürümlerde (müşteriye teslim edilmiş yazılımlar) bulunan hata sayısı
Gösterge	Hata yakalama yüzdesi
Veri Kaynağı	Hata yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Projeye ait müşteri hatalarının kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje hata verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde hata yakalama yüzdesini belirlemek için tüm projelerin hata yakalama yüzde değerlerinin medyanı alınır.

Tablo 2.26. Hata tespit yüzdesi ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projenin çeşitli sürümlerinde (bir ana dış sürüm, o sürüme ait yama sürümleri ve geliştirilmesindeki test sürümleri) tespit edilen hataların ne kadarının proje personeli tarafından tespit edilebildiğini ölçer. Organizasyonda yürütülen projelerde bu metriğin ölçüm değerlerinin merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Yazılım ürün kalitesi
İlişkili Varlıklar	1. Doğrulama ve geçerleme süreci 2. Hata kayıtları

Tablo 2.26.(Devam) Hata tespit yüzdesi ölçüm bilgi formu

Nitelikler	1. Hataların sahibi 2. Hataların durumları
Temel Ölçümler	1. Test sürümlerinde personel tarafından bulunan ve kapatılan hata sayısı 2. Geliştirme sürümlerinde personel tarafından bulunan ve kapatılan hata sayısı 3. Dış sürümlerde müşteri tarafından raporlanan ve kapatılmış hata sayısı 4. Dış sürümlerde personel tarafından bulunan ve kapatılan hata sayısı
Ölçme Yöntemi	1. Hata yönetimi aracından test sürümlerindeki hata sayılarının tespiti 2. Hata yönetimi aracından geliştirme sürümlerindeki hata sayılarının tespiti 3. Hata yönetimi aracından müşterinin bulduğu hata sayısının tespiti 4. Hata yönetimi aracından personelin bulduğu hataların sayısının tespiti
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel 3. Nesnel 4. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı 3. Sıfırdan sonsuza tam sayı 4. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran 3. Oran 4. Oran
Ölçüm Birimi	1. Hata sayısı 2. Hata sayısı 3. Hata sayısı 4. Hata sayısı
Türetilmiş Ölçüm	Hata Tespit Yüzdesi (HTY)

Hata tespit yüzdesi hesaplamasında projede kullanılan hata bildirim formları hatayı bulan tarafa göre sınıflandırılır. Ayrıca bulunan hataların test sürümünde ya da geliştirme sürümünde yer almasına göre de bir sınıflandırma yapılır. Tablo 2.26'da yer alan ölçme fonksiyonuna göre hesaplama yapılarak hata tespit yüzdesi elde edilir.

Tablo 2.26.(Devam) Hata tespit yüzdesi ölçüm bilgi formu

Ölçme Fonksiyonu	$$\text{HTY} = \frac{\text{TSPH} + \text{GSPH} + \text{DSPH}}{\text{TSPH} + \text{GSPH} + \text{DSPH} + \text{DSMH}} * 100$$ TSPH: Yazılımın test sürümlerinde personel tarafından bulunan hata sayısı GSPH: Yazılımın geliştirme sürümlerinde personel tarafından bulunan hata sayısı DSPH: Dış sürümlerde (müşteriye teslim edilmiş yazılımlar) personel tarafından bulunan hata sayısı DSMH: Dış sürümlerde (müşteriye teslim edilmiş yazılımlar) müşteri tarafından bulunan hata sayısı
Gösterge	Hata tespit yüzdesi
Veri Kaynağı	Hata yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Projeye ait müşteri hatalarının kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje hata verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde hata tespit yüzdesini belirlemek için tüm projelerin hata tespit yüzde değerlerinin medyanı alınır.

Tablo 2.27. Süreç denetim planından sapma ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projenin süreç denetim planlamasına ne kadar uyumlu devam ettiğini ölçmeyi amaçlar. Organizasyonda kalite güvence yönetimi süreci kapsamında süreç denetimi yapılacak projeler için planlanan süreç denetimlerinden sapma miktarının merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Denetim durumu
İlişkili Varlıklar	1. Kalite güvence süreci 2. Denetim takvimi
Nitelikler	1. Proje takviminde planlanan denetleme tarihleri 2. Denetim durumları
Temel Ölçümler	1. Planlanan denetim tarihi 2. Gerçekleşen denetim tarihi

Tablo 2.27.(Devam) Süreç denetim planından sapma ölçüm bilgi formu

Ölçme Yöntemi	1. Proje takviminden planlanan denetim tarihi bilgisinin alınması 2. Proje iş yönetim sisteminden gerçekleşen denetim tarihi bilgisinin alınması
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Gün (süre) 2. Gün (süre)
Türetilmiş Ölçüm	Süreç Denetim Plan Sapması (SDPS)
Ölçme Fonksiyonu	$\text{SDPS} = \text{Gerçekleşen denetim tarihi} - \text{Planlanan denetim tarihi}$ Eğer projede ölçüm alınan dönemde birden fazla planlı denetim gerçekleştirilmiş ise sapma aşağıdaki formül ile hesaplanır: $\text{SDPS} = \text{medyan}\,(\text{Gerçekleşen denetim tarihi} - \text{Planlanan denetim tarihi})$
Gösterge	Sapma süresi
Veri Kaynağı	Proje iş yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Tüm sapma değerlerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın süreç denetim verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Ölçüm yapılan proje temelinde ortalama süreç denetim plan sapması için projenin başından itibaren ölçüm yapılan dönem sonuna kadar olan denetim plan sapma değerlerinin medyanı alınır.

Organizasyon temelinde ortalama süreç denetim plan sapmasını belirlemek için tüm projelerin süreç denetim plan sapma değerlerinin medyanı alınır.

Tablo 2.28. Proje uygunsuzlukları ortalama açık kalma süresi ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede tespit edilen uygunsuzlukların ne kadar sürede çözümlendiğini ölçmeyi amaçlar. Organizasyonda yürütülen projelerde kalite güvence süreci kapsamında projeler için açılan uygunsuzlukların ilgili dönem sonuna kadar ortalama açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Uygunsuzluk durumu
İlişkili Varlıklar	1. Kalite güvence süreci 2. Uygunsuzluklar
Nitelikler	1. Uygunsuzlukların durumları ve süre bilgileri
Temel Ölçümler	1. Uygunsuzlukların açılmasından kapatılmasına kadar geçen süre
Ölçme Yöntemi	1. Uygunsuzlukların toplam açık kalma sürelerinin tespiti
Ölçme Yöntemi Tipi	1. Nesne
Ölçek	1. Sıfırdan sonsuza gerçek sayı
Ölçek Tipi	1. Oran
Ölçüm Birimi	1. Gün (süre)
Türetilmiş Ölçüm	Proje Uygunsuzluk Ortalama Açık Kalma Süresi (PUAKS)
Ölçme Fonksiyonu	$\text{PUAKS} = \begin{cases} \text{Uygunsuzluk kapanış tarihi} - \text{Uygunsuzluk oluşma tarihi (kapanan)} \\ \text{Ölçüm tarihi} - \text{Uygunsuzluk oluşma tarihi (henüz açık)} \end{cases}$ Ölçüm alınan tarihte açık durumda bulunan uygunsuzluklar için ölçüm alınan tarih uygunsuzluk kapanma tarihi olarak kabul edilir. Eğer projede ölçüm alınan dönemde birden fazla uygunsuzluk var ise ortalama açık kalma süresi tüm değerlerin medyanı alınarak hesaplanır.
Gösterge	Ortalama uygunsuzluk süresi
Veri Kaynağı	Proje iş yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Projeye ait uygunsuzluk verilerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje uygunsuzluk verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama uygunsuzluk açık kalma süresini belirlemek için tüm projelerin ortalama uygunsuzluk açık kalma süre değerlerinin medyanı alınır.

Tablo 2.29. Konfigürasyon denetim planından sapma ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projenin konfigürasyon denetim planlamasına ne kadar uyumlu devam ettiğini ölçmeyi amaçlar. Organizasyonda konfigürasyon yönetimi süreci kapsamında konfigürasyon denetimi yapılacak projeler için planlanan süreç denetimlerinden sapma miktarının merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Denetim durumu
İlişkili Varlıklar	1. Konfigürasyon yönetimi süreci 2. Denetim takvimi
Nitelikler	1. Proje takviminde planlanan denetleme tarihleri 2. Denetim durumları
Temel Ölçümler	1. Planlanan denetim tarihi 2. Gerçekleşen denetim tarihi
Ölçme Yöntemi	1. Proje takviminden planlanan denetim tarihi bilgisinin alınması 2. Proje iş yönetim sisteminden gerçekleşen denetim tarihi bilgisinin alınması
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Gün (süre) 2. Gün (süre)
Türetilmiş Ölçüm	Konfigürasyon Denetim Plan Sapması (KDPS)
Ölçme Fonksiyonu	$\text{KDPS} = \text{Gerçekleşen denetim tarihi} - \text{Planlanan denetim tarihi}$ Eğer projede ölçüm alınan dönemde birden fazla planlı denetim gerçekleştirilmiş ise sapma aşağıdaki formül ile hesaplanır: $\text{KDPS} = \text{medyan}\,(\text{Gerçekleşen denetim tarihi} - \text{Planlanan denetim tarihi})$
Gösterge	Sapma süresi
Veri Kaynağı	Proje iş yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Tüm sapma değerlerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması

Tablo 2.29.(Devam) Konfigürasyon denetim planından sapma ölçüm bilgi formu

Karar Kriteri	Son bir yılın süreç denetim verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Ölçüm yapılan proje temelinde ortalama konfigürasyon denetim plan sapması için projenin başından itibaren ölçüm yapılan dönem sonuna kadar olan denetim plan sapma değerlerinin medyanı alınır.

Organizasyon temelinde ortalama konfigürasyon denetim plan sapmasını belirlemek için tüm projelerin konfigürasyon denetim plan sapma değerlerinin medyanı alınır.

Tablo 2.30. Konfigürasyon denetimi başına düşen ortalama uygunsuzluk ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Organizasyonda konfigürasyon yönetimi süreci kapsamında denetim yapılacak projeler için planlanan denetimlerde bulunan proje başına düşen uygunsuzluk miktarını tespit eder.
Ölçülebilir Kavram	Denetim durumu
İlişkili Varlıklar	1. Konfigürasyon yönetimi süreci 2. Denetim raporları
Nitelikler	1. Konfigürasyon uygunsuzlukları
Temel Ölçümler	1. Konfigürasyon uygunsuzluk sayıları
Ölçme Yöntemi	1. Denetim raporlarından konfigürasyon uygunsuzluk sayılarının tespit edilmesi
Ölçme Yöntemi Tipi	1. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran
Ölçüm Birimi	1. Uygunsuzluk sayısı
Türetilmiş Ölçüm	Konfigürasyon Denetimi Başına Düşen Ortalama Uygunsuzluk Sayısı (KDOUS)
Ölçme Fonksiyonu	$\text{KDOUS} = \text{medyan}\,(\text{Konfigürasyon Uygunsuzluk Sayısı})$
Gösterge	Denetim başına düşen ortalama uygunsuzluk
Veri Kaynağı	Proje iş yönetim araçları
Veri Toplama Sıklığı	Aylık

Tablo 2.30.(Devam) Konfigürasyon denetimi başına düşen ortalama uygunsuzluk ölçüm bilgi formu

Model	Tüm konfigürasyon uygunsuzluklarının kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın proje konfigürasyon denetim sonuçları toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısını belirlemek için tüm projelerin konfigürasyon denetimi başına düşen ortalama uygunsuzluk değerlerinin medyanı alınır.

Tablo 2.31. Konfigürasyon uygunsuzluklarının ortalama açık kalma süresi ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Projede konfigürasyon denetimleri kapsamında açılan konfigürasyon uygunsuzluklarının ortalama açık kalma sürelerini tespit eder. Organizasyonda yürütülen projelerde konfigürasyon uygunsuzluklarının ortalama açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Uygunsuzluk durumu
İlişkili Varlıklar	1. Konfigürasyon yönetimi süreci 2. Uygunsuzluklar
Nitelikler	1. Uygunsuzlukların durumları ve süre bilgileri
Temel Ölçümler	1. Uygunsuzlukların açılmasından kapatılmasına kadar geçen süre
Ölçme Yöntemi	1. Uygunsuzlukların toplam açık kalma sürelerinin tespiti
Ölçme Yöntemi Tipi	1. Nesnel
Ölçek	1. Sıfırdan sonsuza gerçek sayı
Ölçek Tipi	1. Oran
Ölçüm Birimi	1. Gün (süre)
Türetilmiş Ölçüm	Konfigürasyon Uygunsuzluk Ortalama Açık Kalma Süresi (KUAKS)

Ölçüm alınan tarihte henüz kapanmamış konfigürasyon uygunsuzlukları var ise uygunsuzlukların kapanış tarihi ölçüm alınan tarih olarak kabul edilir. Tablo 2.31'de yer alan ölçme fonksiyonuna göre hesaplama yapılarak konfigürasyon uygunsuzluklarının ortalama açık kalma süresi elde edilir.

Tablo 2.31.(Devam) Konfigürasyon uygunsuzluklarının ortalama açık kalma süresi ölçüm bilgi formu

Ölçme Fonksiyonu	$\text{KUAKS} = \begin{cases} \text{Uygunsuzluk kapanış tarihi} - \text{Uygunsuzluk oluşma tarihi (kapanan)} \\ \text{Ölçüm tarihi} - \text{Uygunsuzluk oluşma tarihi (henüz açık)} \end{cases}$ Ölçüm alınan tarihte açık durumda bulunan konfigürasyon uygunsuzlukları için ölçüm alınan tarih uygunsuzluk kapanma tarihi olarak kabul edilir. Eğer projede ölçüm alınan dönemde birden fazla konfigürasyon uygunsuzlukları var ise ortalama açık kalma süresi tüm değerlerin medyanı alınarak hesaplanır.
Gösterge	Ortalama uygunsuzluk süresi
Veri Kaynağı	Proje iş yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Projeye ait konfigürasyon uygunsuzluk verilerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın konfigürasyon uygunsuzluk verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde ortalama konfigürasyon uygunsuzluk açık kalma süresini belirlemek için tüm projelerin ortalama konfigürasyon uygunsuzluk açık kalma süre değerlerinin medyanı alınır.

Tablo 2.32. Konfigürasyon uygunsuzluklarının ortalama açık kalma oranı ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	İlgili dönemde açık durumda bulunan konfigürasyon uygunsuzluk sayısının toplam konfigürasyon uygunsuzluk sayısı içindeki payını tespit eder. Organizasyonda yürütülen projelerde konfigürasyon uygunsuzluklarının açık kalma oranının merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Uygunsuzluk durumu
İlişkili Varlıklar	1. Konfigürasyon yönetimi süreci 2. Uygunsuzluklar
Nitelikler	1. Uygunsuzlukların durumları
Temel Ölçümler	1. Kapatılmamış konfigürasyon uygunsuzluk sayısı
Ölçme Yöntemi	1. Proje iş yönetim aracından kapatılmamış konfigürasyon uygunsuzluk sayısının tespiti

Tablo 2.32.(Devam) Konfigürasyon uygunsuzluklarının ortalama açık kalma oranı ölçüm bilgi formu

Ölçme Yöntemi Tipi	1. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran
Ölçüm Birimi	1. Uygunsuzluk sayısı
Türetilmiş Ölçüm	Konfigürasyon Uygunsuzluk Ortalama Açık Kalma Oranı (KUAKO)
Ölçme Fonksiyonu	$\text{KUAKO} = \frac{\text{Kapatılmamış konfigürasyon uygunsuzluk sayısı}}{\text{Tüm konfigürasyon uygunsuzluk sayısı}}$
Gösterge	Kapatılmamış konfigürasyon uygunsuzluklarının oransal gösterimi
Veri Kaynağı	Proje iş yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Projeye ait konfigürasyon uygunsuzluk verilerinin kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın konfigürasyon uygunsuzluk verileri toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde kapatılmamış konfigürasyon uygunsuzluk oranını belirlemek için tüm projelerin kapatılmamış konfigürasyon uygunsuzluk oranı değerlerinin medyanı alınır.

Tablo 2.33. Risk gözden geçirme oranı ölçüm bilgi formu

Bilgi İhtiyacı (Ölçüm Amacı)	Proje başından ilgili dönem sonuna kadar planlanan risk gözden geçirme faaliyetlerinin ne kadarının gerçekleştirildiğini tespit eder. Organizasyonda yürütülen projelerde risk gözden geçirme oranının merkezî eğilimini ve süreç değişkenliğini belirler.
Ölçülebilir Kavram	Risk durumu
İlişkili Varlıklar	1. Risk yönetimi süreci 2. Riskler
Nitelikler	1. Risklerin durumları
Temel Ölçümler	1. Gözden geçirmesi yapılmış risk sayısı 2. Toplam risk sayısı
Ölçme Yöntemi	1. Proje iş yönetim aracından gözden geçirmesi yapılmış risk sayısının tespiti 2. Proje iş yönetim aracından toplam risk sayısının tespiti

Tablo 2.33.(Devam) Risk gözden geçirme oranı ölçüm bilgi formu

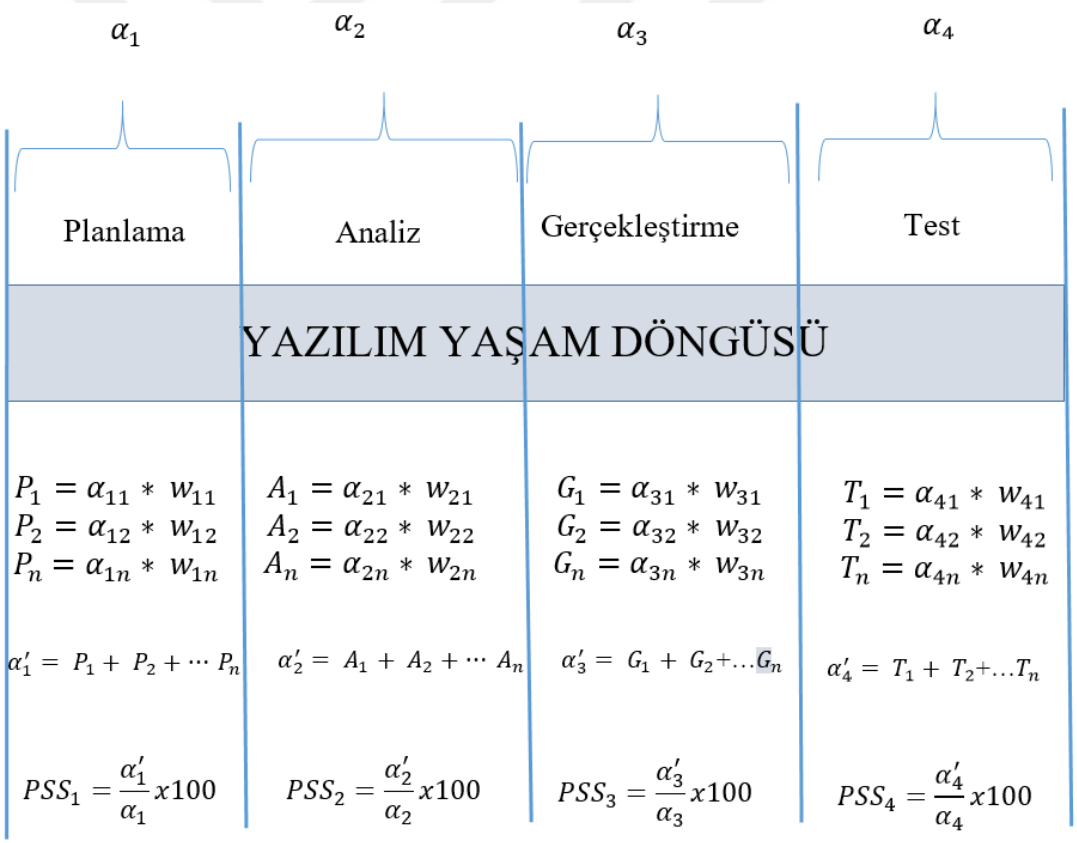
Ölçme Yöntemi Tipi	1. Nesnel 2. Nesnel
Ölçek	1. Sıfırdan sonsuza tam sayı 2. Sıfırdan sonsuza tam sayı
Ölçek Tipi	1. Oran 2. Oran
Ölçüm Birimi	1. Risk sayısı 2. Risk sayısı
Türetilmiş Ölçüm	Risk Gözden Geçirme Oranı (RGGO)
Ölçme Fonksiyonu	$\text{RGGO} = \frac{\text{Gözden geçirilen risk sayısı}}{\text{Toplam risk sayısı}}$
Gösterge	Risk durumlarının oransal gösterimi
Veri Kaynağı	Proje iş yönetim araçları
Veri Toplama Sıklığı	Aylık
Model	Projeye ait risk kayıtları kullanılarak süreç ortalama ve kontrol limitlerinin hesaplanması
Karar Kriteri	Son bir yılın risk kayıtları toplanır. Kontrol grafiği oluşturularak sürecin alt ve üst kontrol limitleri belirlenir. Sonuç değeri, projenin planlandığı şekilde sonuçlanması için kontrol limitlerinin arasında olmalıdır. Kontrol limitlerinin dışındaki sonuçlar kök neden analizi gerektirir.

Organizasyon temelinde risk gözden geçirme oranını belirlemek için tüm projelerin risk gözde geçirme oran değerlerinin medyanı alınır.

3. YAZILIM YAŞAM DÖNGÜSÜ SÜREÇ METRİKLERİ İLE PROJE PERFORMANSI ÖLÇME VE PROJE BAŞARI PUANI HESAPLAMA MODELİ

3.1. Model Bileşenleri

Şekil 3.1, süreç metriklerini kullanarak proje başarı puanı hesaplama modelini göstermektedir. Şekil 3.1'de belirtilen proje aşamaları kurumdan kuruma değişebilir. Her proje aşamasında ölçülen süreç metrikleri, kurumun iş hedeflerine ve ürün ve süreç performans hedeflerine göre değişebilir.



Şekil 3.1. Proje başarı puanı hesaplama modeli

Şekil 3.1'de gösterilen modelde projenin planlama, analiz, gerçekleştirme ve test olmak üzere dört aşamadan oluştuğu varsayılmıştır. Proje performans değerlendirmesi ve proje başarı puanı hesaplama işlemi aşama sonlarında gerçekleştirilecektir. Modelde gösterilen değişkenlerin açıklamaları aşağıdaki gibidir:

w_{ij} AHS yöntemi ile hesaplanan süreç metriklerinin ağırlık katsayısıdır.

$\alpha_1, \alpha_2, \alpha_3, \alpha_4$ beklenen başarı puanlarıdır. Beklenen başarı puanları, ilgili proje aşamasında işletilen süreçlerin metrik ağırlık toplamları ile belirlenir.

$\alpha_{1n}, \alpha_{2n}, \alpha_{3n}, \alpha_{4n}$ ölçülen süreç metrik değerleridir.

P_n, A_n, G_n, T_n projenin ilgili aşamandaki toplam ölçülen metrik başarı puanı değeridir.

PSS_n ölçüm yapılan proje aşamasındaki proje başarı puanı değeridir.

Projenin planlama, analiz, gerçekleştirme ve test aşamalarında başarı puanı hesaplama modeline girdi olacak süreç metrikleri Tablo 3.1'de verilmiştir.

Tablo 3.1. Başarı puanı hesaplama modeline girecek metrikler

Proje aşaması	Modele girecek metrikler
Planlama	PY 1.1,PY 1.2, PY1.3,KG 1.1, KG 1.2, DG 1.1, DG 1.2, DG 1.3, GY 1.1
Analiz	PY 1.1,PY 1.2, PY 1.3,GY 1.1,GEL 1.3, GEL 1.6, KG 1.1, KG 1.2, DG 1.1, DG 1.2, DG 1.3
Gerçekleştirme	Tüm süreç metrik kriterleri
Test	Tüm süreç metrik kriterleri

Proje başarı puanı, metrik ölçüm değerleri ile metrik ağırlık katsayısı çarpım sonuçlarının toplanmasıyla hesaplanır. Kontrol limitleri dışında olan süreç metrikleri, proje başarı puanına katkıda bulunmaz.

Başarı puanını hesaplamak için, metrik ölçüm değerleri ilgili süreç kontrol grafikleri ile karşılaştırılmıştır. Üst ve alt kontrol limit değerleri içindeki süreç metrikleri, proje başarı puanına kendi metrik ağırlık katsayı değeri kadar katkıda bulunur. Üst ve alt kontrol limit değerleri içinde olmayan süreç metriklerinin ağırlık katsayısı sıfır kabul edilir. Bu nedenle bu durumdaki metrikler proje başarı puanına katkı yapmazlar.

3.2. Proje Performans Değerlendirme ve Proje Başarı Puanı Hesaplama Süreci İş Akışı

Proje performans değerlendirme ve proje başarı puanı hesaplama süreci iş akışı Şekil 3.2'de verilmiştir. Süreç iş akışı modelin ve modeli işletecek yazılım oluşturulması ve modelin uygulanması olarak iki alt akıştan oluşmaktadır.

Modelin ve yazılımın oluşturulması aşamasında öncelikle proje başarısına etki eden süreç metriklerinin belirlenmesi adımı yer almaktadır. Bu adımla ilgili yöntem Bölüm 2.4'te tanımlanmıştır.

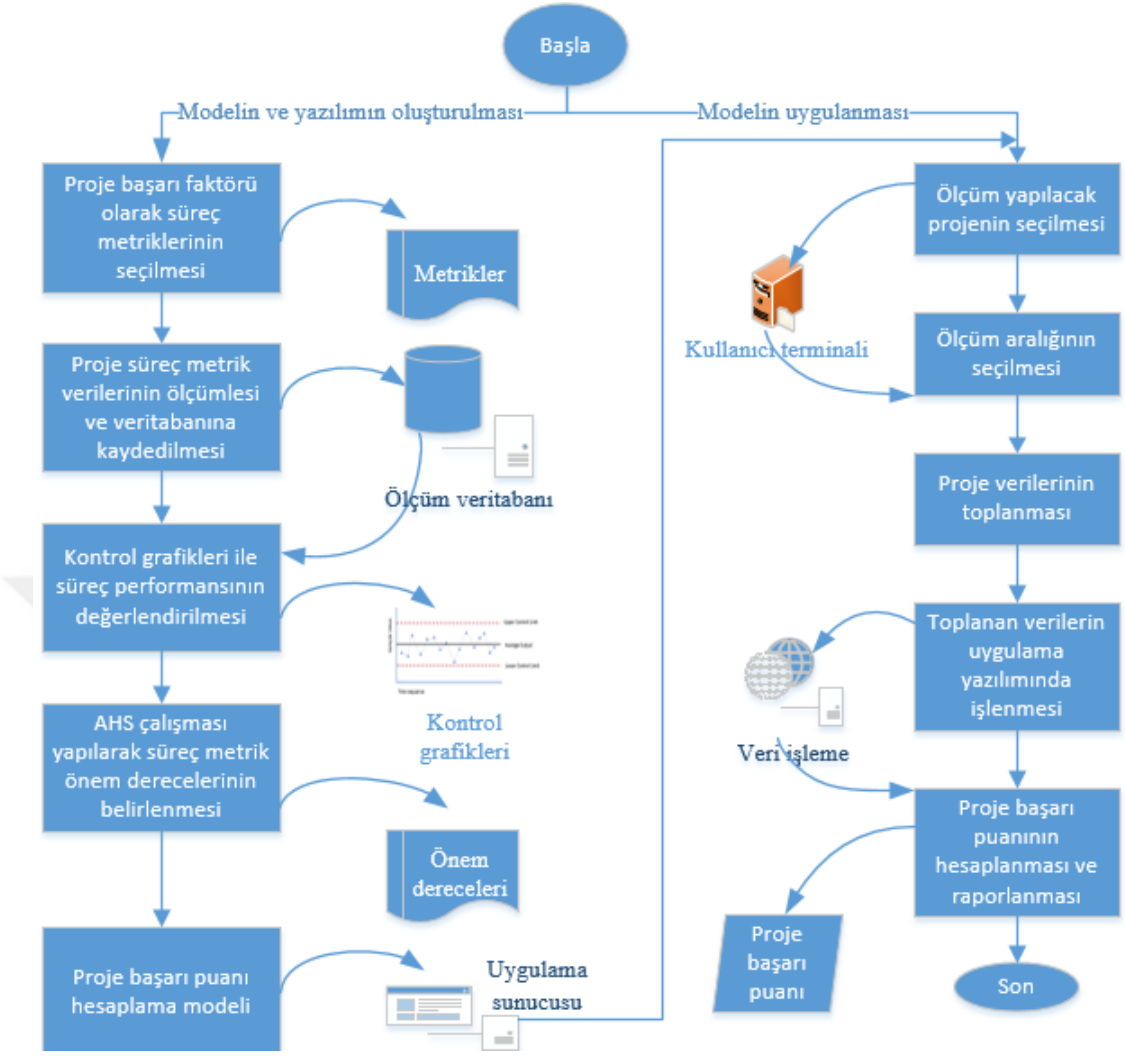
Süreç metrik listesi belirlendikten sonra süreç performans dayanaklarının oluşturulması amacıyla projelerden süreç metrik ölçümleri toplama adımı gelmektedir. Süreç metrik ölçümleri projelerin kullandığı proje destek araçlarından geliştirilen yazılım aracılığı ile otomatik toplanmaktadır.

Ölçümlerin toplanması ile her süreç metriği için bireysel gözlem değerleri için kontrol grafikleri kullanılarak süreç performans değerlendirmesi adımı gelmektedir. Her bir metrik için oluşturulan kontrol kartlarında alt ve üst kontrol limitleri belirlenmiştir. Ölçüm yapılan bir projedeki süreç metrik ölçümleri alt ve üst kontrol limit değerleri ile karşılaştırılarak proje başarısına katkı verip vermediği değerlendirilmektedir.

Bir sonraki adımda AHS yöntemi ile süreç metriklerinin proje başarısına olan etkisi açısından önem dereceleri belirleme adımı gelmektedir. AHS çalışması bir Ar-Ge kurumunda yazılım yaşam döngüsü süreçlerini yaşamış tecrübeli çalışanlar ile gerçekleştirilmiştir.

Proje başarı modeli hesaplama adımında; AHS çalışmasından gelen metrik önem katsayıları ile metrik kontrol grafiklerindeki üst ve alt kontrol limit değerleri modele girdi oluşturur ve proje başarı puanı hesaplanır.

Bu tez çalışmasında proje başarı puanını hesaplayan bir yazılım geliştirilmiştir. Bu modelin uygulaması yazılımın kullanılmasıyla gerçekleştirilmiştir.



Şekil 3.2. Proje performans değerlendirme ve proje başarı puanı hesaplama süreç iş akışı

Bu aşamada kullanıcı yazılım ara yüzünden ölçüm yapılacak projeyi ve ölçüm aralığını seçer. Seçim işlemi bittikten sonra yazılım tarafından ara işlemler gerçekleştirilir.

Bu ara işlemlerde seçilen projeye ait süreç metriklerine ait ölçümler yapılarak veriler toplanır. Toplanan veriler geliştirilen yazılım tarafından bir veri tabanına kaydedilir.

Veri tabanında bulunan ölçüm değerleri daha önce oluşturulan kontrol grafiklerindeki alt ve üst kontrol limit değerleri ile karşılaştırılır. Kontrol limit değerleri içerisinde kalan metrikler AHS çalışmasında belirlenen önem derecesi kadar proje başarı puanına katkı yapmaktadır. Projenin ölçüm yapıldığı aşamada beklenen başarının ne kadarını sağladığına dair bir başarı puanı hesaplanır ve kullanıcıya rapor verir.

4. MODELİN UYGULAMASI

4.1. Süreç Metrik Kriterleri Hiyerarşisi

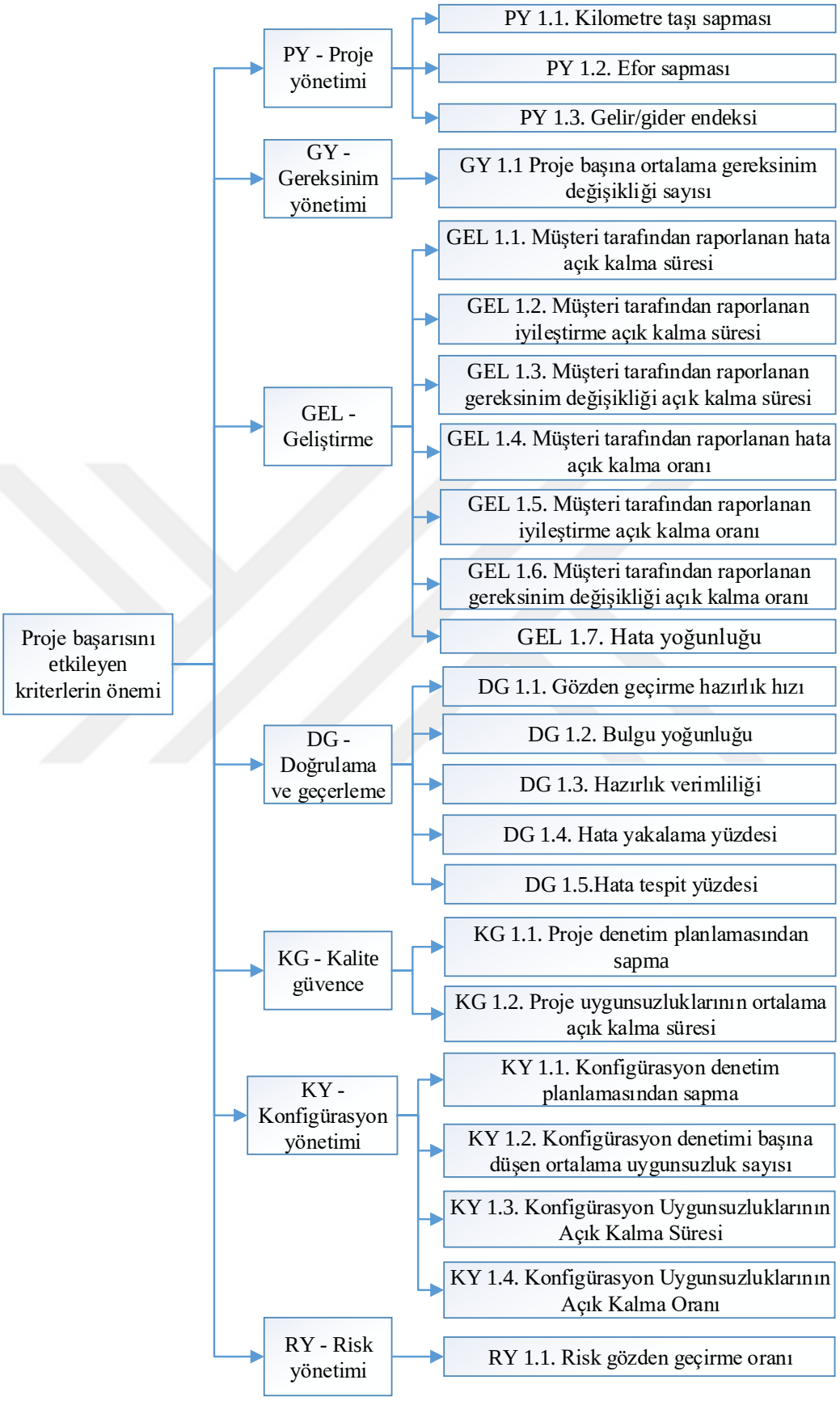
Proje başarısına etki ettiği düşünülen yazılım yaşam döngüsü süreç metrikleri, bir Ar-Ge organizasyonundaki süreç iyileştirme ekibi, kalite yöneticileri ve proje yöneticileri ile yapılan görüşmeler ve kuruluşun mevcut süreçlerinin analizi sonucunda belirlenmiştir. Metrik belirleme yöntemi Bölüm 2.4'te açıklanmıştır.

Tespit edilen metrikler bu çalışmada proje başarı kriteri olarak kabul edilmiştir. Süreç metrikleri birer kriter olarak kabul edilip AHS çalışması için ana kriter ve alt kriter olarak düzenlenmiştir. Kriter hiyerarşisi Şekil 4.1'de gösterilmektedir.

Literatür incelemesi sonucunda, yazılım yaşam döngüsünde süreç metrikleri üzerinde proje başarısını etkileyen faktörlerin belirlenmesinde yeterli çalışma olmadığı görülmüştür. Ar-Ge kurumundaki süreç iyileştirme ekibi, uzman proje yöneticileri ve kalite yöneticileri ile yapılan metrik belirleme çalışmasında, yazılım yaşam döngüsünde yürütülen süreçlerin kritik faaliyetlerinin ölçülmesinin ve analiz edilmesinin proje başarısını etkileyeceği değerlendirilmiştir.

Bu bağlamda, Şekil 4.1'de belirtilen kriterlerin projenin başarısına ağırlıklı katkısına karar verilerek önerilen modeli doğrulamak için bir uygulama yapılmıştır.

Şekil 4.1'de gösterilen hiyerarşide 1. seviyede amaç yer almaktadır. 2. seviyede 7 ana kriter ve 3. seviyede 23 alt kriter bulunmaktadır. Hiyerarşide bulunan ana ve alt kriterler bir Ar-Ge kurumunda yapılan ölçme ve analiz çalıştayları sonucunda belirlenmiştir. Belirlenen metrikler daha geniş proje ekipleri tarafından gözden geçirilerek görüşler toplanmıştır. Toplanan görüşlere göre ana ve alt kriterlerde güncellemeler yapılarak son sürümü oluşturulmuştur. Ana ve alt kriterler üzerinde çalışma yapılan Ar-Ge kurumunda geniş bir görüş birliği sağlanmıştır.



Şekil 4.1. Süreç metrik kriterleri hiyerarşisi

4.2. AHS Çalışması ile Süreç Metriklerinin Ağırlıklandırılması

Yazılım yaşam döngüsü süreç metriklerinin proje başarısı üzerindeki önem derecesini belirlemek amacıyla bir Ar-Ge kurumunda AHS çalışması gerçekleştirilmiştir. Çalışmanın gerçekleştirildiği kurum hakkında detaylı bilgi Bölüm 2.4 ve Bölüm 2.5'te verilmiştir. Bahsedilen kurumda deneyimli proje yöneticileri, bölüm yöneticileri ve kalite güvence yöneticileri ile bir dizi toplantı gerçekleştirilmiştir. Kurumda yazılım yaşam döngüsü süreçlerini işletmiş 4 çalışana yapılandırılmış bir anket verilerek AHS çalışması gerçekleştirilmiştir. Uzmanlardan süreç metriklerinin proje başarısı üzerindeki önemini değerlendirmeleri istenmiştir. Uzmanlar, Tablo 2.1'de verilen ölçeği kullanarak her bir hiyerarşi düzeyinde ikili karşılaştırmalar yapmışlardır., nihai grup kararı için bu bireysel kararları toparlamak amacıyla geometrik ortalama yöntem kullanılmıştır. Uzmanlardan alınan ikili karşılaştırma sonuçları değerlendirilmiş ve tutarlılık oranının% 10'dan fazla olduğu durumlarda, bireysel görüşlerini gözden geçirmeleri ve düzeltme yapmaları için ilgili uzmanlarla yeniden görüşmeler yapılmıştır. Uzmanlardan gelen düzeltilmiş değerler yeniden analiz edilmiştir. Bireysel değerlendirme sonuçları, ana ve alt kriterlerin tutarlılık oranları bu bölümde verilmiştir. Bölüm 2.2 Denklem (2.7)'de verilen CR tutarlılık oranının sıfıra yakın olması, uzmanlardan alınan kararların daha tutarlı ve kabul edilebilir olduğu anlamına gelmektedir. Aşağıda Tablo 4.1 ile Tablo 4.6 arasında uzmanların bireysel değerlendirmeleri ve aritmetik ortalaması alınmasıyla lokal ağırlıkları verilmektedir.

Tablo 4.1. Ana kriter ağırlıkları

Bireyler	Proje yönetimi	Gereksinim yönetimi	Geliştirme	Konfigürasyon yönetimi	Risk yönetimi	Kalite güvence	Doğrulama ve geçerleme	Tutarlılık Oranı (CR)
1	0,068	0,114	0,131	0,158	0,160	0,107	0,262	0,075
2	0,177	0,332	0,260	0,066	0,034	0,033	0,097	0,073
3	0,076	0,227	0,227	0,130	0,076	0,076	0,190	0,077
4	0,171	0,152	0,203	0,130	0,138	0,116	0,090	0,088
Min	0,068	0,114	0,131	0,066	0,034	0,033	0,090	
Max	0,177	0,332	0,260	0,158	0,160	0,116	0,262	
Std Sapma	0,051	0,083	0,047	0,034	0,050	0,033	0,071	

Tablo 4.1.(Devam) Ana kriter ağırlıkları

Bireyler	Proje yönetimi	Gereksinim yönetimi	Geliştirme	Konfigürasyon yönetimi	Risk yönetimi	Kalite güvence	Doğrulama ve geçerleme	Tutarlılık Oranı (CR)
Lokal ağırlık	0,123	0,206	0,205	0,121	0,102	0,083	0,160	
Lokal ağır. %	12,28	20,60	20,50	12,12	10,20	8,32	15,99	

Tablo 4.2. Proje yönetimi süreci alt kriter ağırlıkları

Bireyler	PY 1.1	PY 1.2	PY 1.3	Tutarlılık Oranı (CR)
1	0,143	0,429	0,429	0,000
2	0,231	0,665	0,104	0,074
3	0,106	0,261	0,634	0,030
4	0,608	0,120	0,272	0,063
Min	0,106	0,120	0,104	
Max	0,608	0,665	0,634	
Std Sapma	0,199	0,203	0,195	
Lokal ağırlıklar	0,272	0,369	0,360	
Lokal ağırlıklar %	27,20	36,85	35,95	

Tablo 4.3. Geliştirme süreci alt kriter ağırlıkları

Bireyler	GEL 1.1	GEL 1.2	GEL 1.3	GEL 1.4	GEL 1.5	GEL 1.6	GEL 1.7	Tutarlılık Oranı (CR)
1	0,058	0,073	0,102	0,145	0,070	0,215	0,336	0,087
2	0,157	0,117	0,042	0,125	0,101	0,074	0,384	0,097
3	0,211	0,058	0,173	0,173	0,058	0,173	0,154	0,016
4	0,265	0,159	0,217	0,144	0,062	0,104	0,048	0,093
Min	0,058	0,058	0,042	0,125	0,058	0,074	0,048	
Max	0,265	0,159	0,217	0,173	0,101	0,215	0,384	
Std Sapma	0,076	0,040	0,067	0,017	0,017	0,055	0,136	
Lokal ağırlık	0,173	0,102	0,134	0,147	0,073	0,142	0,231	
Lokal ağır. %	17,28	10,18	13,35	14,69	7,28	14,16	23,05	

Tablo 4.4. Doğrulama ve geçerleme süreci alt kriter ağırlıkları

Bireyler	DG 1.1	DG 1.2	DG 1.3	DG 1.4	DG 1.5	Tutarlılık Oranı (CR)
1	0,070	0,382	0,154	0,177	0,218	0,040

Tablo 4.4.(Devam) Doğrulama ve geçerleme süreci alt kriter ağırlıkları

Bireyler	DG 1.1	DG 1.2	DG 1.3	DG 1.4	DG 1.5	Tutarlılık Oranı (CR)
2	0,039	0,189	0,062	0,281	0,430	0,064
3	0,072	0,262	0,087	0,289	0,289	0,008
4	0,049	0,082	0,146	0,297	0,426	0,080
Min	0,039	0,082	0,062	0,177	0,218	
Max	0,072	0,382	0,154	0,297	0,430	
Std Sapma	0,014	0,109	0,039	0,049	0,091	
Lokal ağırlık	0,058	0,229	0,112	0,261	0,341	
Lokal ağırlık. %	5,75	22,87	11,24	26,08	34,05	

Tablo 4.5. Kalite güvence süreci alt kriter ağırlıkları

Bireyler	KG 1.1	KG 1.2
1	0,750	0,250
2	0,750	0,250
3	0,750	0,250
4	0,750	0,250
Min	0,750	0,250
Max	0,750	0,250
Std Sapma	0,000	0,000
Lokal ağırlıklar	0,750	0,250
Lokal ağırlıklar %	75,00	25,00

Tablo 4.6. Konfigürasyon yönetimi süreci alt kriterler

Bireyler	KY 1.1	KY 1.2	KY 1.3	KY 1.4	Tutarlılık Oranı (CR)
1	0,231	0,070	0,499	0,200	0,084
2	0,079	0,519	0,201	0,201	0,016
3	0,100	0,300	0,300	0,300	0,000
4	0,515	0,282	0,145	0,058	0,052
Min	0,079	0,070	0,145	0,058	
Max	0,515	0,519	0,499	0,300	
Std Sapma	0,174	0,159	0,135	0,086	
Lokal ağırlıklar	0,231	0,293	0,286	0,190	
Lokal ağırlıklar %	23,12	29,27	28,64	18,97	

Nihai grup kararına ulaşmak için bireysel değerlendirme değerlerinin geometrik ortalaması alınarak elde edilen veriler sırasıyla aşağıda Tablo 4.7 ile Tablo 4.12 arasında verilmiştir.

Tablo 4.7. Ana kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu

Ana kriter	PY	GY	GEL	KY	RY	KG	DG
PY	1,000	0,485	0,577	0,931	1,278	1,565	0,841
GY	2,060	1,000	1,316	1,655	2,060	2,060	1,456
GEL	1,732	0,760	1,000	1,655	2,178	2,817	1,968
KY	1,075	0,604	0,604	1,000	1,107	1,316	0,760
RY	0,783	0,485	0,459	0,904	1,000	1,000	0,577
KG	0,639	0,485	0,355	0,760	1,000	1,000	0,439
DG	1,189	0,687	0,508	1,316	1,732	2,280	1,000

Tablo 4.8. Proje yönetimi süreci alt kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu

Proje Yönetimi Süreci	PY 1.1	PY 1.2	PY 1.3
PY 1.1	1,000	0,576	0,880
PY 1.2	1,732	1,000	0,863
PY 1.3	1,136	1,158	1,000

Tablo 4.9. Geliştirme süreci alt kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu

Geliştirme Süreci	GEL 1.1	GEL 1.2	GEL 1.3	GEL 1.4	GEL 1.5	GEL 1.6	GEL 1.7
GEL 1.1	1,000	1,456	1,732	0,841	1,456	1,495	0,997
GEL 1.2	0,687	1,000	0,576	0,577	1,565	0,931	0,508
GEL 1.3	0,577	1,728	1,000	0,760	1,561	0,760	0,676
GEL 1.4	1,189	1,732	1,316	1,000	2,280	1,000	0,707
GEL 1.5	0,687	0,639	0,637	0,439	1,000	0,439	0,408
GEL 1.6	0,669	1,072	1,316	1,000	2,280	1,000	0,615
GEL 1.7	0,997	1,968	1,480	1,414	2,449	1,622	1,000

Tablo 4.10. Doğrulama ve geçerleme süreci alt kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu

Doğrulama ve Geçerleme Süreci	DG 1.1	DG 1.2	DG 1.3	DG 1.4	DG 1.5
DG 1.1	1,000	0,258	0,439	0,221	0,202
DG 1.2	3,873	1,000	2,060	0,669	0,603
DG 1.3	2,280	0,484	1,000	0,298	0,355
DG 1.4	4,527	1,495	3,350	1,000	0,537
DG 1.5	4,949	1,655	2,817	1,861	1,000

Tablo 4.11. Kalite güvence süreci alt kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu

Kalite Güvence Süreci	KG 1.1	KG 1.2
KG 1.1	1,000	3,000
KG 1.2	0,333	1,000

Tablo 4.12. Konfigürasyon yönetimi süreci alt kriterler - geometrik ortalaması alınmış ikili karşılaştırma sonucu

Konfigürasyon Yönetimi Süreci	KY 1.1	KY 1.2	KY 1.3	KY 1.4
KY 1.1	1,000	0,880	0,577	1,117
KY 1.2	1,136	1,000	1,158	1,316
KY 1.3	1,732	0,863	1,000	1,732
KY 1.4	0,895	0,760	0,577	1,000

Ana kriterlerin ağırlık değerleri ve tutarlılık oranı Tablo 4.13'te verilmiştir. Tablo 4.13'te görülebileceği gibi, gereksinim yönetimi süreci % 21,7 ile proje başarısı üzerinde en önemli etkiye sahip kriterdir. Bunu sırasıyla % 21,5 ile geliştirme süreci, % 11,9 ile proje yönetimi ve konfigürasyon yönetimi süreçleri, % 15,1 ile doğrulama ve geçerleme süreci takip etmektedir. % 9,5 ile risk yönetimi süreci ve% 8,4 ile kalite güvence süreci son sıralarda yer almaktadır.

Tablo 4.13. Ana kriterleri ağırlık değerleri

Süreçler	Ağırlıklar
Project management	0,119
Requirements management	0,217
Development	0,215
Verification and Validation	0,151

Tablo 4.13.(Devam) Ana kriterleri ağırlık değerleri

Süreçler	Ağırlıklar
Quality assurance	0,084
Configuration management	0,119
Risk management	0,095
CI = 0,00963	CR = 0,00730

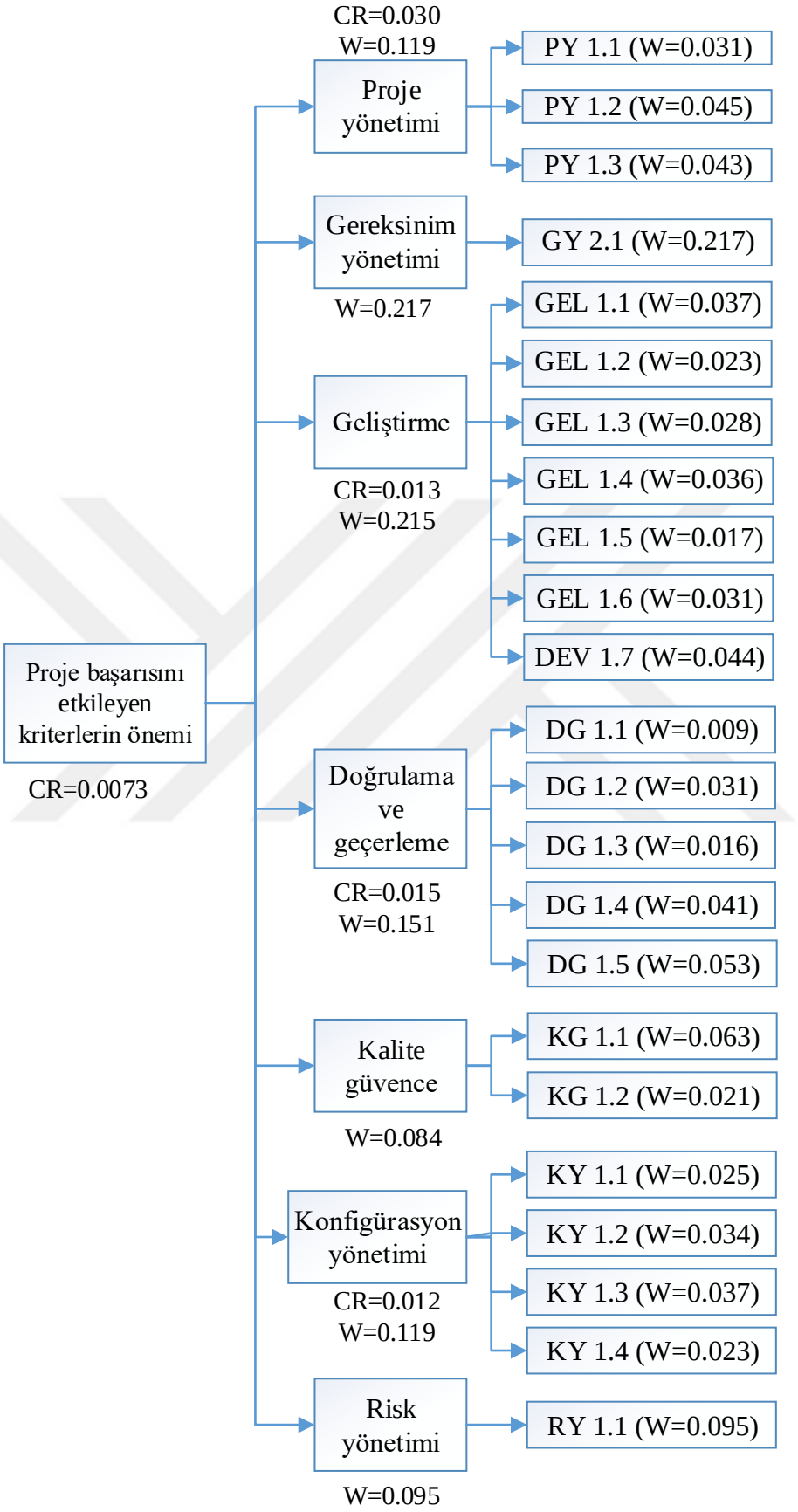
Seviye 3'te bulunan alt kriterlerin ikili karşılaştırma sonuçları için tutarlılık kontrolü yapılmıştır. Alt kriter ağırlıkları Şekil 4.2'de verilmiştir. Şekil 4.2'ye göre, tüm CR değerlerinin sıfıra yakın olması uzmanların değerlendirmelerinin gayet tutarlı olduğunu göstermektedir.

Şekil 4.2'de proje başarısına etki eden ana kriterlerin ve alt kriterlerin ağırlık değerleri verilmiştir. Metrik ağırlıklarını büyükten küçüğe doğru sıralayarak bir tablo hazırlanmıştır. Tablo 4.14'te verilen metrik ağırlıkları incelendiğinde projeden gereksinim yönetiminin proje başarısında en çok etkiye sahip olduğu görülmektedir. Proje gereksinimlerinin projenin analiz aşamasında belirlenmesi ve yazılım ürününden beklenen ürün özelliklerini tam ve doğru olarak yansıtması gerekmektedir.

Dokümante edilen gereksinimler müşteri ile beraber gözden geçirilmeli ve müşterinin onayı alınmalıdır. Analiz aşamasında iyi tanımlanmayan, eksik, yanlış ya da muğlak ifadelere sahip gereksinimler projenin kabul test aşamasında yapılan testlerden geçemeyeceği için ürünün kabulüne engel olacaktır.

Projenin ilerleyen aşamalarında ortaya çıkacak gereksinim değişiklikleri projedeki birçok elemanın değişmesine neden olabilir. Değişiklik meydana gelen gereksinimler aynı seviyede ya da alt/üst seviyedeki başka isterler ile bağımlı olabilir. Bu bağımlılık nedeniyle bir gereksinim değişikliği bir den fazla gereksinimin değişmesine yol açabilir. Gereksinimleri değişmesi bu gereksinimlere bağlı tasarım elemanlarında değişikliklere yol açabilir. Tasarım elemanlarına bağlı yazılım ve donanım birimlerinin de değişikliğe uğraması muhtemeldir.

Özetle; gereksinim değişiklikleri mümkün olduğunda projenin başında toplanmalıdır. Proje ilerlerken muhtemel gereksinim değişiklikleri mümkün olduğunca azaltılmalıdır.



Şekil 4.2. Süreç metrik kriter ağırlıkları

Süreç metriği alt kriterlerinin önemlerine göre sıralanması Tablo 4.14'de verilmiştir. Tablo 4.14'e göre proje başarısını etkileyen en önemli kriter proje başına düşen ortalama gereksinim değişikliği sayısıdır. Bu sonuç, proje devam ederken yüksek sayıda gereksinim değişikliğinin projenin başarısını engellediğini vurgulamaktadır.

Tablo 4.14. Süreç metrik kriterleri sıralama tablosu

Sıra	Kriter	Ağırlık
1	Proje başına ortalama gereksinim değişikliği sayısı	0,2166
2	Risk gözden geçirme oranı	0,0953
3	Proje uygunsuzluklarının ortalama açık kalma süresi	0,0633
4	Hata tespit yüzdesi	0,0531
5	Efor sapması	0,0447
6	Hata yoğunluğu	0,0435
7	Gelir/gider endeksi	0,0428
8	Hata yakalama yüzdesi	0,0414
9	Konfigürasyon uygunsuzluklarının açık kalma süresi	0,0371
10	Müşteri tarafından raporlanan hata açık kalma süresi	0,0370
11	Müşteri tarafından raporlanan hata açık kalma oranı	0.0362
12	Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı	0,0335
13	Bulgu yoğunluğu	0,0314
14	Kilometre taşı sapması	0,0313
15	Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı	0,0307
16	Müşteri tarafından raporlanan gereksinim değişikliği açık kalma süresi	0,0275
17	Konfigürasyon denetim planlamasından sapma	0,0253
18	Konfigürasyon uygunsuzluklarının açık kalma oranı	0,0230
19	Müşteri tarafından raporlanan iyileştirme açık kalma süresi	0,0229
20	Süreç denetim planlamasından sapma	0,0211
21	Müşteri tarafından raporlanan iyileştirme açık kalma oranı	0,0172
22	Hazırlık verimliliği	0,0165
23	Gözden geçirme hazırlık hızı	0,0087

Risk gözden geçirme oranı, proje başarısı üzerindeki etkisi açısından ikinci en önemli kriterdir. Bu kriter, projenin başından sonuna kadar projedeki risklerin yönetim

derecesini gösterir. Risk gözden geçirme oranı ne kadar yüksek olursa, proje başarı şansı o kadar yüksek olur.

Süreç denetim uygunsuzluklarının ortalama açık kalma süresi üçüncü en önemli kriterdir. Uzmanlar, kalite denetimleri sonucunda ürün ve süreçlerde bulunan uygunsuzlukların uzun süren kapanış sürelerinin proje başarısı üzerinde olumsuz bir etkiye sahip olduğunu vurgulamışlardır.

Hata tespit yüzdesi dördüncü en önemli kriterdir. Müşteriler tarafından tespit edilen hata sayısı ne kadar az olursa, proje o kadar başarılı olacaktır.

Efor sapması beşinci öneme sahiptir. Planlanan efor aşılırsa, toplam maliyet artacağından proje başarısı üzerinde olumsuz bir etkisi olacaktır. Hata yoğunluğu, proje başarısı üzerindeki etki bakımından altıncı sıradadır. Hata yoğunluğu yüksek olduğundan, hataların düzeltilmesi ek işçilik ve zaman gerektirecektir, bu nedenle proje bütçesi ve takvimi üzerinde olumsuz bir etkisi olacaktır.

Yedinci sırada yer alan gelir gideri endeksinin 1'den büyük olması, projenin başarısı üzerinde olumlu bir etkiye sahip olduğunu göstermektedir. Sekiz sıradaki hata yakalama oranı, yazılım ürünü müşteriye teslim edilmeden önce yakalanan hata sayısının toplam hata sayısına oranını verir. Daha yüksek oran, ürünün müşteriye daha az kusurla teslim edildiği anlamına gelir. Bu durumun proje başarısı üzerinde olumlu bir etkisi vardır. Konfigürasyon uygunsuzluklarının ortalama açık kalma süre kriteri dokuzuncu sırada önemlidir. Bu sürenin uzaması projenin başarısını azaltır.

Onuncu ve on birinci sıra metrikleri birbiriyle ilişkilidir. Müşteri tarafından rapor edilen hataların ortalama açık kalma süre ve oranının yüksek olması projenin başarısı üzerinde olumsuz bir etkiye sahiptir. Benzer şekilde uzmanlar, on iki ile yirmi bir arasındaki tüm metriklerin yüksek değerlerinin proje başarısı üzerinde olumsuz bir etkiye sahip olduğunu vurgulamışladır.

Yirmi ikinci ve yirmi üçüncü metrikler, yazılım ve dokümantasyon gözden geçirmeleri ile ilgili metriklerdir. Uzmanlar bu metriklerin en az öneme sahip olduğunu düşünmüşlerdir.

4.3. Kontrol Grafiklerinin Oluşturulması

Şekil 4.1'de belirtilen tüm süreç metrikleri için bireysel değerleri için kontrol grafikleri oluşturularak süreçlerin üst ve alt kontrol limitleri oluşturulmuştur. Müşteri tarafından raporlanan hata açık kalma süresi, müşteri tarafından raporlanan iyileştirme açık kalma süresi, müşteri tarafından raporlanan gereksinimlerin açık kalma süresi, bulgu yoğunluğu ve gözden geçirme hazırlık hızı metrikleri için kontrol kartı oluşturma adımları detaylı bir şekilde verilmiştir.

Genellikle yazılım dünyasında kontrol grafiği tipi olarak U ya da XMR kontrol grafikleri kullanılmaktadır [95]. Eğer verilerin dağılımı poisson ya da binominal dağılıma uygun ise U kontrol grafiği kullanılabilir. Verilerin dağılımı konusunda kesin fikir yok ise bireysel değerler için kontrol grafiği (XMR) kullanılır [95]. Bu tez çalışmasında süreç metrikleri kontrol grafiklerini oluşturmak için Minitab aracında XMR kontrol grafiği kullanılmıştır.

4.3.1. Müşteri tarafından raporlanan hata açık kalma süresi metriği kontrol grafiği

Müşteri tarafından raporlanan hata açık kalma süresi metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.15'te verilmiştir. Bu metriği ölçmenin amacı; projede müşteri tarafından raporlanan hataların ortalama olarak ne kadar süre açık kaldığını tespit etmek ve kurumda yürütülen projelerde müşteri tarafından iletilen hataların açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.15'te verilmiştir. Veriler projelerin kullandığı hata yönetim aracından toplanmıştır.

Tablo 4.15. Müşteri tarafından raporlanan hata açık kalma süresi metriği ölçüm verileri

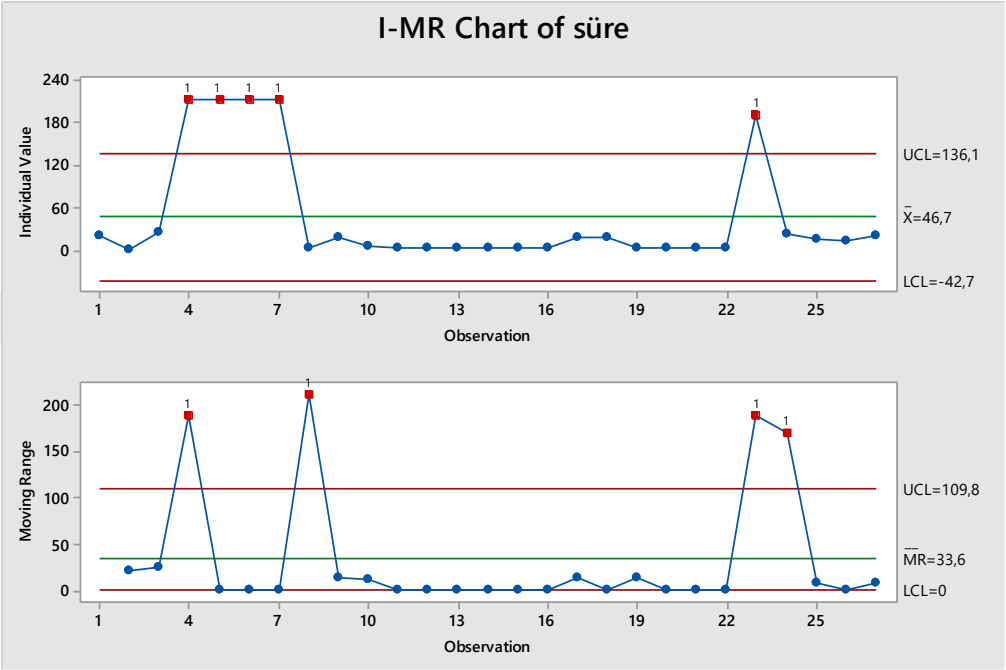
Kapanma Süresi	Id	Durum	Bildirim Tarihi	Kapanma Tarihi	Hata Derecesi	Hata Kaynağı	Test Tipi	Oluşma Sıklığı
20	20432	Kapandı	12.04.2016	02.05.2016	Küçük	Müşteri Hata	İşlevsel Test	Her Zaman
3	19995	Kapandı	27.03.2016	24.03.2016	Büyük	Müşteri Hata	İşlevsel Test	Her Zaman
25	17952	Kapandı	18.01.2016	12.02.2016	Küçük	Müşteri Hata	İşlevsel Test	Her Zaman
214	16152	Kapandı	27.11.2015	28.06.2016	Küçük	Müşteri Hata	Performans Testi	Her Zaman

Tablo 4.15.(Devam) Müşteri tarafından raporlanan hata açık kalma süresi metriği ölçüm verileri

Kapanma Süresi	Id	Durum	Bildirim Tarihi	Kapanma Tarihi	Hata Derecesi	Hata Kaynağı	Test Tipi	Oluşma Sıklığı
214	16151	Kapandı	27.11.2015	28.06.2016	Küçük	Müşteri Hata	Performans Testi	Her Zaman
214	16150	Kapandı	27.11.2015	28.06.2016	Küçük	Müşteri Hata	Performans Testi	Her Zaman
214	16149	Kapandı	27.11.2015	28.06.2016	Küçük	Müşteri Hata	Performans Testi	Her Zaman
3	16147	Kapandı	27.11.2015	30.11.2015	Küçük	Müşteri Hata	Performans Testi	Bazen
17	16146	Kapandı	27.11.2015	14.12.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
5	16145	Kapandı	27.11.2015	02.12.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
4	16143	Kapandı	27.11.2015	01.12.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
4	16142	Kapandı	27.11.2015	01.12.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
4	16141	Kapandı	27.11.2015	01.12.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
4	16140	Kapandı	27.11.2015	01.12.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
3	16139	Kapandı	27.11.2015	30.11.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
4	16136	Kapandı	27.11.2015	01.12.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
17	16134	Kapandı	27.11.2015	14.12.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
17	16133	Kapandı	27.11.2015	14.12.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
3	16131	Kapandı	27.11.2015	30.11.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
3	16129	Kapandı	27.11.2015	30.11.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
3	16128	Kapandı	27.11.2015	30.11.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
3	16127	Kapandı	27.11.2015	30.11.2015	Küçük	Müşteri Hata	Performans Testi	Her Zaman
191	16060	Kapandı	24.11.2015	02.06.2016	Küçük	Müşteri Hata	İşlevsel Test	Her Zaman
22	15813	Kapandı	09.11.2015	01.12.2015	Katastrofik	Müşteri Hata	Performans Testi	Her Zaman
15	15812	Kapandı	09.11.2015	24.11.2015	Katastrofik	Müşteri Hata	Performans Testi	Her Zaman
14	15811	Kapandı	09.11.2015	23.11.2015	Katastrofik	Müşteri Hata	Performans Testi	Her Zaman
22	15808	Kapandı	09.11.2015	01.12.2015	Katastrofik	Müşteri Hata	Performans Testi	Her Zaman

Tablo 4.15 incelendiğinde hatanın derecesi, oluşma sıklığı ve hangi test tipinde ortaya çıktığı gibi bilgiler de müşteri tarafından bulunan hatalar ile ilgili diğer özellikler olarak görülmektedir. Hatalar ile ilgili bu özellikler Şekil 2.18. Hata bildirim formunda da görülmektedir. Hatanın kapanma süresi bildirim tarihi ile kapanma tarihi arasındaki geçen sürenin gün cinsinden değeridir. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış hata bildirim formlarından alınmıştır. Tablo 4.15'te kapanma süresi

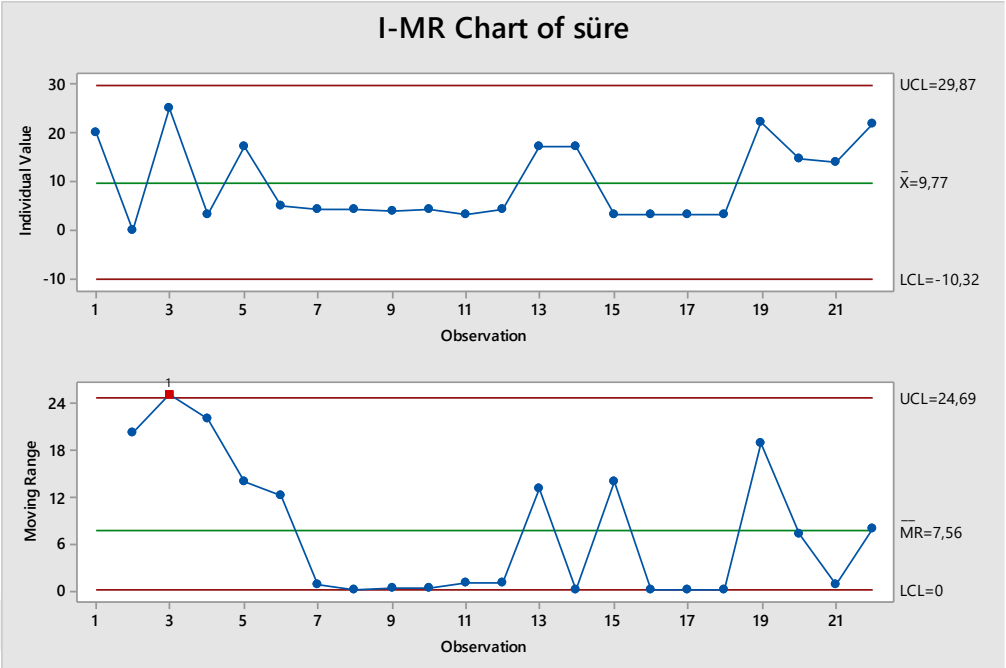
sütunundaki verilerden bireysel değerler için kontrol grafiğinin oluşturulması Şekil 4.3 ve Şekil 4.4'te gösterilmiştir. Kontrol grafiğinde kontrol limitleri dışındaki değerler çıkarılarak kalan veriler ile tekrar kontrol grafiği oluşturulmuştur. Süreç kontrol altında olana kadar bu işlemler tekrar edilmiştir. Limit dışındaki değerlerin kök neden analizi Bölüm 4.4'te verilmiştir.



Şekil 4.3. MTR hataların açık kalma süresi – ham veri

Şekil 4.3'teki kontrol grafiğinde 4. ve 7. arasındaki tüm noktalar ile 23. noktanın üst limit dışında olduğu görülmektedir. Bu noktaların çıkarılmasının ardından kalan verilerle oluşturulan kontrol grafiği Şekil 4.4'te verilmiştir.

Şekil 4.4'teki kontrol grafiğinde sürecin kontrol altında olduğu görülmektedir. Hata kapanma süresi metriği için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır [94]. Kontrol grafiğinde üst kontrol limit değeri 29,87 olarak hesaplanmıştır. Projelerin hata kapatma süreç performansı değerlendirmesinde bu kontrol grafiği referans alınacaktır. Eğer bir projenin hata kapatma süresi 0 ile 28,87 gün arasında ise sürecin kontrol altında olduğu, aksi takdirde sürecin kontrol dışında olduğu kabul edilecektir. Bu durumda kök nedeni bulmak için balık kılçığı yöntemi neden-sonuç analizi yapılacaktır.



Şekil 4.4. MTR hataların açık kalma süresi – birinci iterasyon

Yazılım yaşam döngüsündeki tüm süreç metrikleri için müşteri tarafından raporlanan hata açık kalma süresi metriğinde detaylı olarak anlatılan adımlar kullanılarak kontrol grafikleri oluşturulmuştur. Projelerde ölçülen süreç metrik değerleri kendi kontrol grafiğine ait limitler içerisinde ise AHS'den gelen metrik ağırlık katsayısı kadar proje başarısına katkı sağlayacaktır. Eğer ölçülen metrik değeri kontrol limitleri dışında ise metriğin ağırlık katsayısı sıfırlanacaktır. Bu durum proje başarı puanı hesaplaması yapılan projede kontrol dışındaki metrikler başarı puanına katkı yapmayacaktır.

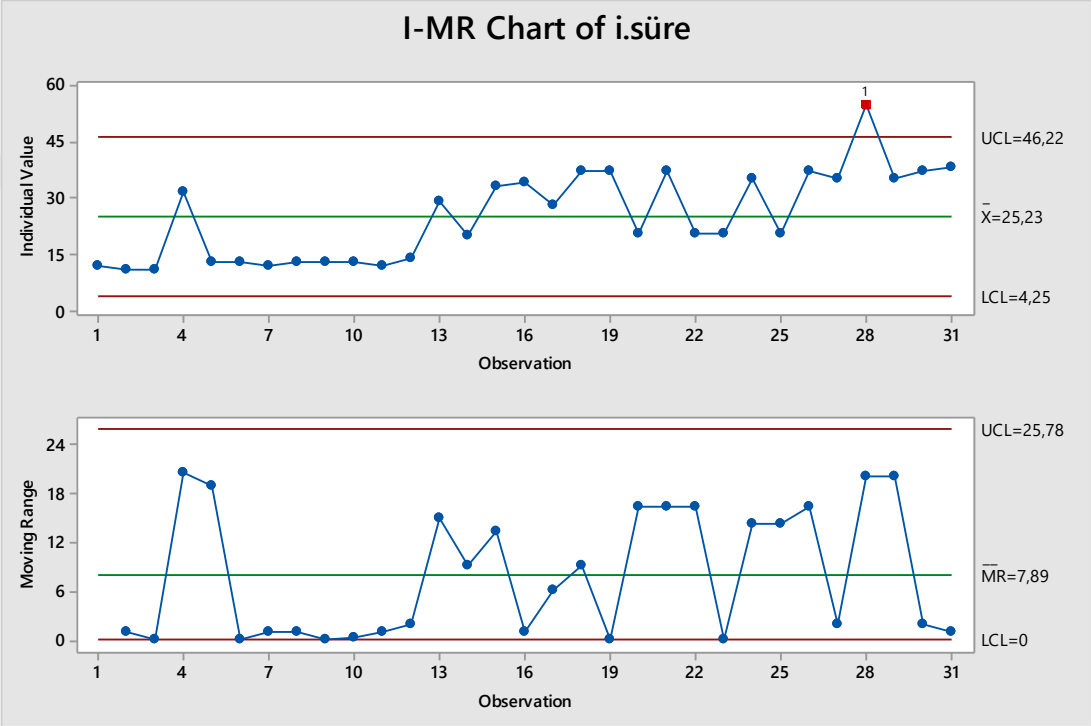
4.3.2. Müşteri tarafından raporlanan iyileştirme açık kalma süresi metriği kontrol grafiği

Müşteri tarafından raporlanan hata açık kalma süresi metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.16'da verilmiştir. Bu metriği ölçmenin amacı; projede müşteri tarafından raporlanan iyileştirmelerin ortalama olarak ne kadar süre açık kaldığını tespit etmek ve kurumda yürütülen projelerde müşteri tarafından iletilen iyileştirme isteklerinin açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.16'da verilmiştir. Veriler projelerin kullandığı hata yönetim aracından toplanmıştır.

Tablo 4.16. Müşteri tarafından raporlanan iyileştirme açık kalma süresi metriği ölçüm verileri

Kapanma Süresi	Bildirim Tarihi	Kapanma Tarihi
12	24.10.2019 10:36	05.11.2019 15:56
11	24.10.2019 10:28	04.11.2019 15:43
11	24.10.2019 10:26	04.11.2019 15:48
32	06.09.2019 16:42	08.10.2019 08:17
13	06.09.2019 16:14	19.09.2019 13:53
13	06.09.2019 15:53	19.09.2019 14:48
12	06.09.2019 15:24	18.09.2019 16:15
13	06.09.2019 14:26	19.09.2019 13:54
13	06.09.2019 14:24	19.09.2019 13:49
13	06.09.2019 11:26	19.09.2019 17:06
12	06.09.2019 09:57	18.09.2019 16:59
14	06.09.2019 09:34	20.09.2019 15:30
29	22.08.2019 14:08	20.09.2019 14:21
20	08.08.2019 09:17	28.08.2019 09:05
33	08.08.2019 09:13	10.09.2019 16:07
34	08.08.2019 08:45	11.09.2019 15:21
28	01.08.2019 15:26	29.08.2019 18:42
37	01.07.2019 13:32	07.08.2019 17:03
37	01.07.2019 13:32	07.08.2019 17:01
21	01.07.2019 13:31	22.07.2019 10:59
37	01.07.2019 13:29	07.08.2019 17:02
21	01.07.2019 13:27	22.07.2019 09:49
21	01.07.2019 13:26	22.07.2019 09:49
35	01.07.2019 13:20	05.08.2019 16:00
21	01.07.2019 13:17	22.07.2019 09:49
37	01.07.2019 13:07	07.08.2019 14:57
35	01.07.2019 13:06	05.08.2019 16:31
55	01.07.2019 13:06	25.08.2019 16:31
35	01.07.2019 13:06	05.08.2019 16:02
37	01.07.2019 13:04	07.08.2019 16:13
38	01.07.2019 13:04	08.08.2019 16:17

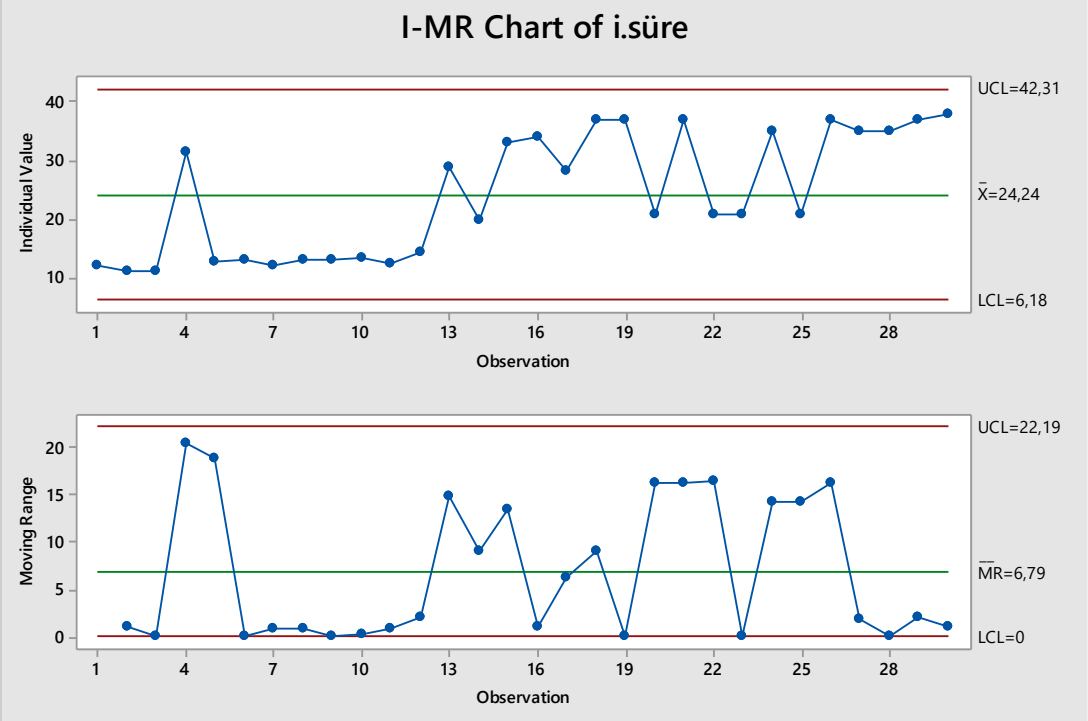
İyileştirme isteğinin kapanma süresi bildirim tarihi ile kapanma tarihi arasındaki geçen sürenin gün cinsinden değeridir. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış iyileştirme tipindeki hata bildirim formlarından alınmıştır. Tablo 4.16'da kapanma süresi sütunundaki verilerden bireysel değerler için kontrol grafiğinin oluşturulması Şekil 4.5 ve Şekil 4.6'da gösterilmiştir. Kontrol grafiğinde kontrol limitleri dışındaki değerler çıkarılarak kalan veriler ile tekrar kontrol grafiği oluşturulmuştur. Süreç kontrol altında olana kadar bu işlemler tekrar edilmiştir.



Şekil 4.5. MTR iyileştirme açık kalma süresi – ham veri

Şekil 4.5'teki kontrol grafiğinde 28. noktanın limit dışında olduğu görülmektedir. Bu noktanın çıkarılmasının ardından kalan verilerle oluşturulan kontrol grafiği Şekil 4.6'da verilmiştir.

Şekil 4.6'daki kontrol grafiğinde sürecin kontrol altında olduğu görülmektedir. Kontrol grafiğinde üst kontrol limit değeri 42,31 alt limit değeri 6,18 olarak hesaplanmıştır. Projelerin iyileştirme kapatma süreç performansı değerlendirmesinde bu kontrol grafiği referans alınacaktır. Eğer bir projenin hata kapatma süresi 6,18 ile 42,31 gün arasında ise sürecin kontrol altında olduğu, aksi takdirde sürecin kontrol dışında olduğu kabul edilecektir. Bu durumda kök nedeni bulmak için balık kılçığı yöntemi neden-sonuç analizi yapılacaktır.



Şekil 4.6. MTR iyileştirme açık kalma süresi kontrol grafiği – birinci iterasyon

Diğer süreç metrikleri için kontrol dışı verilerden arındırılmış veri seti ve kontrol grafikleri sırasıyla aşağıda verilmiştir.

4.3.3. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma süresi metriği kontrol grafiği

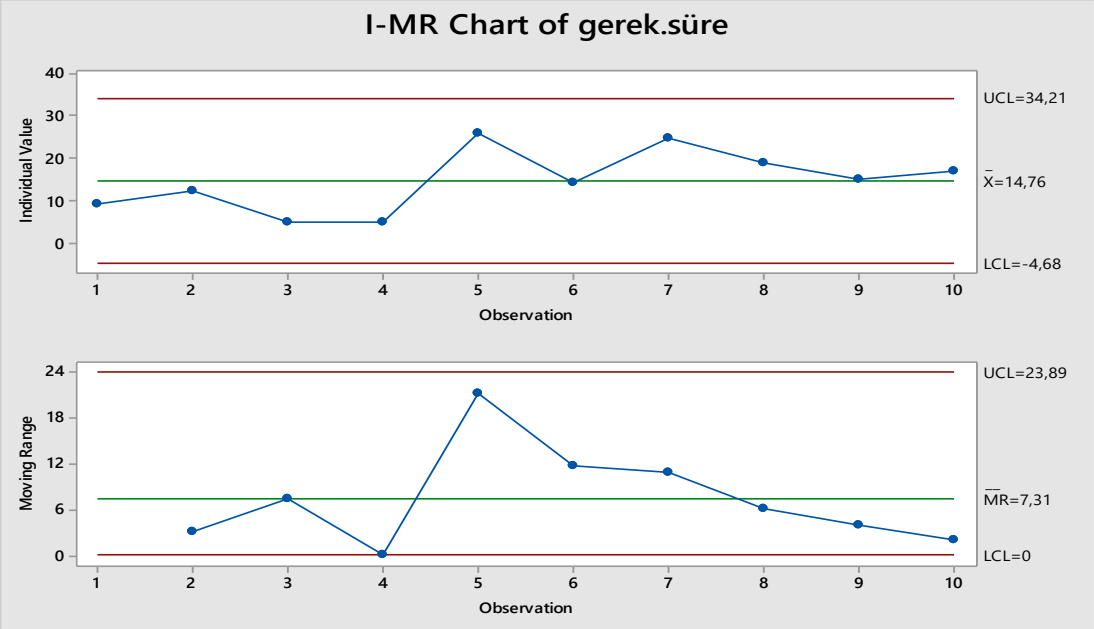
Müşteri tarafından raporlanan gereksinim değişikliği açık kalma süresi metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.17'de verilmiştir. Bu metriği ölçmenin amacı; projede müşteri tarafından raporlanan gereksinim değişiklik taleplerinin ortalama olarak ne kadar süre açık kaldığını tespit etmek ve kurumda yürütülen projelerde müşteri tarafından iletilen gereksinim değişikliklerinin açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirlemektir.

Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.17'de verilmiştir. Veriler projelerin kullandığı değişiklik yönetimi aracından toplanmıştır. Müşteri gereksinim değişikliği talepleri projenin bütçe ve takvimini etkileme durumu açısından analiz edilir. Proje bütçe ve takvimini etkileyen durumlarda müşteri ile proje sözleşme değişikliği işlemi gerçekleştirilir.

Tablo 4.17. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma süresi metriği ölçüm verileri

Gereksinim No	Kapanma Süresi	Kapanma Tarihi	Bildirim Tarihi
1	9	24.10.2019 14:47	15.10.2019 11:45
2	12	24.10.2019 14:52	12.10.2019 09:05
3	5	16.05.2019 13:22	11.05.2019 13:58
4	5	16.05.2019 13:22	11.05.2019 13:57
5	26	29.05.2019 11:18	03.05.2019 11:44
6	14	24.01.2019 14:06	10.01.2019 09:31
7	25	14.01.2019 20:10	20.12.2018 12:55
8	19	08.01.2019 11:46	20.12.2018 11:37
9	15	20.12.2018 14:22	05.12.2018 16:33
10	17	22.12.2018 14:24	05.12.2018 16:26

Gereksinim değişiklik isteğinin kapanma süresi bildirim tarihi ile kapanma tarihi arasındaki geçen sürenin gün cinsinden değeridir. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış değişiklik istek formlarından alınmıştır. Tablo 4.17'de kapanma süresi sütunundaki verilerden bireysel değerler için kontrol grafiğinin oluşturulması Şekil 4.7'de gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.7. MTR gereksinim değişikliği açık kalma süresi kontrol grafiği

4.3.4. Müşteri tarafından raporlanan hata açık kalma oranı metriği kontrol grafiği

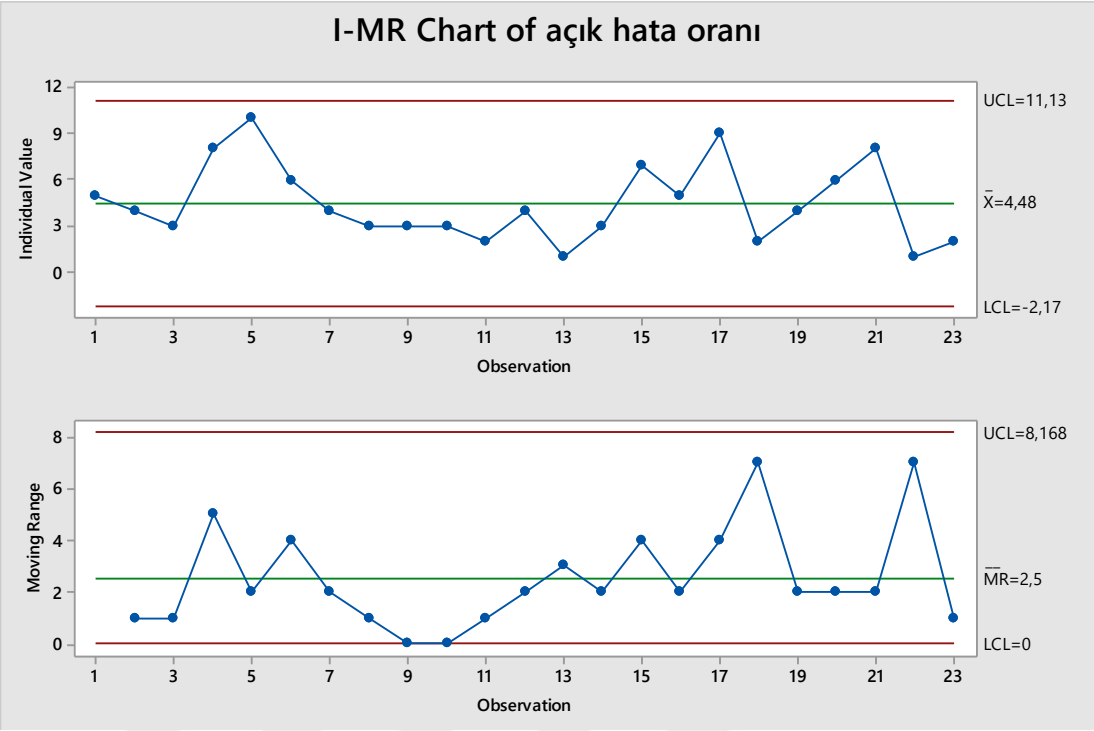
Müşteri tarafından raporlanan hata açık kalma oranı metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.18'de verilmiştir. Bu metriği ölçmenin amacı; projede müşteri tarafından raporlanan hataların oran olarak ne kadarının kapanmadığını tespit etmek ve kurumda yürütülen projelerde müşteri tarafından iletilen hataların açık kalma oranlarının merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.18'de verilmiştir. Veriler projelerin kullandığı hata yönetim aracından toplanmıştır.

Tablo 4.18. Müşteri tarafından raporlanan hata açık kalma oranı metriği ölçüm verileri

Proje No	Hata açık kalma oranı %	Proje No	Hata açık kalma oranı %
1	5	13	1
2	4	14	3
3	3	15	7
4	8	16	5
5	10	17	18
6	6	18	21
7	4	19	9
8	3	20	2
9	3	21	4
10	3	22	6
11	2	23	8
12	4	24	1
		25	2

Hata açık kalma oranının nasıl hesaplandığı Tablo 2.18'de belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış hata bildirim formlarından alınmıştır.

Tablo 4.18'de kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.8'de gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.8. MTR hata açık kalma oranı kontrol grafiği

4.3.5. Müşteri tarafından raporlanan iyileştirme açık kalma oranı metriği kontrol grafiği

Müşteri tarafından raporlanan iyileştirme açık kalma oranı metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.19'da verilmiştir.

Bu metriği ölçmenin amacı; projede müşteri tarafından raporlanan iyileştirmelerin oran olarak ne kadarının kapanmadığını tespit etmek ve kurumda yürütülen projelerde müşteri tarafından iletilen iyileştirmelerin açık kalma oranlarının merkezî eğilimini ve süreç değişkenliğini belirlemektir.

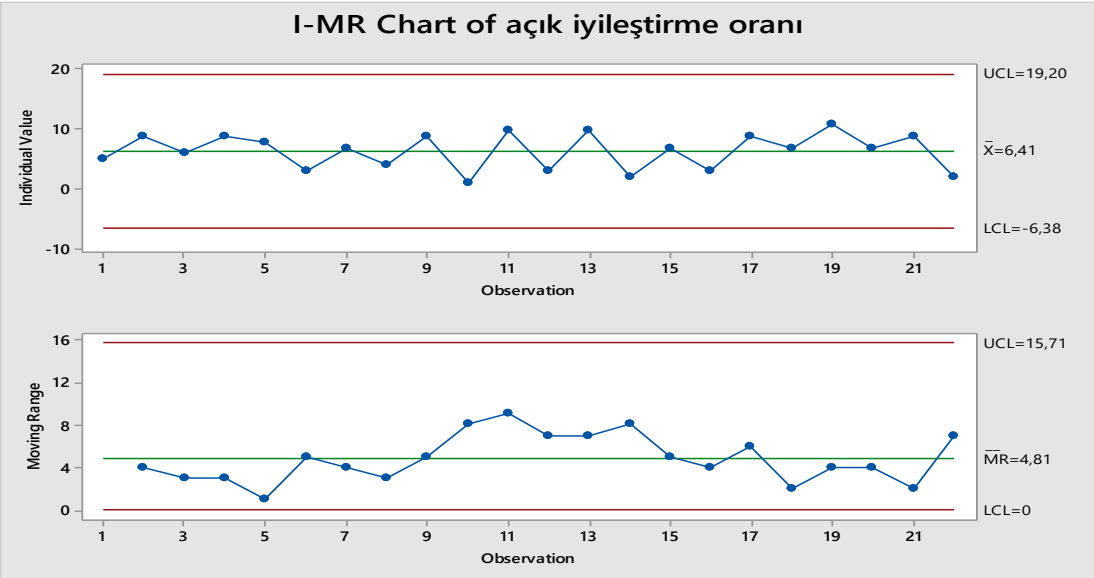
Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.19'da verilmiştir. Veriler projelerin kullandığı hata yönetim aracından toplanmıştır.

Müşterinin talep ettiği iyileştirme taleplerinin proje bütçesini ve takvimini etkileyecek nitelikte olup olmadığına karar vermek için gelen talepler analiz edilir. Analiz sonucu proje bütçesini ve takvimini etkileyecek talepler proje yönetimi tarafından kabul edilmez. Bu tipteki talepler için proje sözleşme değişikliği süreci işletilir. Bu tipteki değişiklik talepleri bu çalışmada değerlendirmeye alınmamıştır.

Tablo 4.19. Müşteri tarafından raporlanan iyileştirme açık kalma oranı metriği ölçüm verileri

Proje No	İyileştirme açık kalma oranı %	Proje No	İyileştirme açık kalma oranı %
1	5	13	3
2	9	14	10
3	6	15	2
4	9	16	7
5	35	17	3
6	8	18	9
7	3	19	40
8	7	20	7
9	4	21	11
10	9	22	7
11	1	23	9
12	10	24	43
		25	2

İyileştirme açık kalma oranının nasıl hesaplandığı Tablo 2.19'da belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış iyileştirme tipindeki hata bildirim formlarından alınmıştır. Tablo 4.19'da kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.9'da gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.9. MTR iyileştirme açık kalma oranı kontrol grafiği

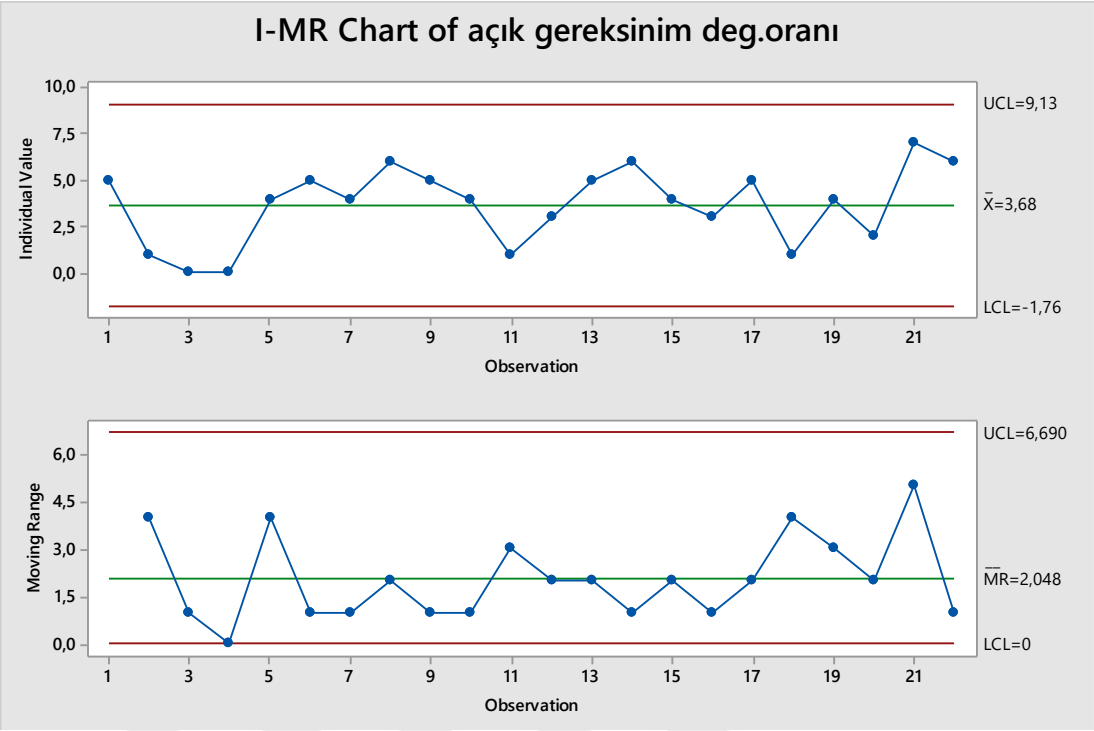
4.3.6. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı metriği kontrol grafiği

Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.20'de verilmiştir. Bu metriği ölçmenin amacı; projede müşteri tarafından raporlanan gereksinim değişiklik taleplerinin ortalama olarak ne kadar açık kaldığını tespit etmek ve kurumda yürütülen projelerde müşteri tarafından iletilen gereksinim değişikliklerinin açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.20'de verilmiştir. Veriler projelerin kullandığı değişiklik yönetim aracından toplanmıştır.

Tablo 4.20. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı metriği ölçüm verileri

Proje No	Gereksinim değişikliği açık kalma oranı %	Proje No	Gereksinim değişikliği açık kalma oranı %
1	5	13	1
2	1	14	3
3	0	15	5
4	0	16	18
5	4	17	6
6	20	18	4
7	5	19	3
8	4	20	5
9	6	21	1
10	24	22	4
11	5	23	2
12	4	24	7
		25	6

Gereksinim değişikliği açık kalma oranının nasıl hesaplandığı Tablo 2.20'de belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış değişiklik istek formlarından alınmıştır. Tablo 4.20'de kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.10'da gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.10. MTR gereksinim değişikliği açık kalma oranı kontrol grafiği

4.3.7. Hata yoğunluğu metriği kontrol grafiği

Hata yoğunluğu metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.21'de verilmiştir. Bu metriği ölçmenin amacı; projede üretilen yazılımların kalitesini ölçmektir. Bu değerin düşmesi, yazılımın kalitesinin artması anlamına gelir. Bu metrik kurumda yürütülen projelerde üretilen yazılımların kalitesinin merkezî eğilimini tespit eder ve süreç değişkenliğini belirler. Bu ölçüm müşteriye teslim edilen yazılım ürünlerinde bulunan hataları kapsar. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.21'de verilmiştir. Veriler projelerin kullandığı değişiklik yönetim aracından toplanmıştır.

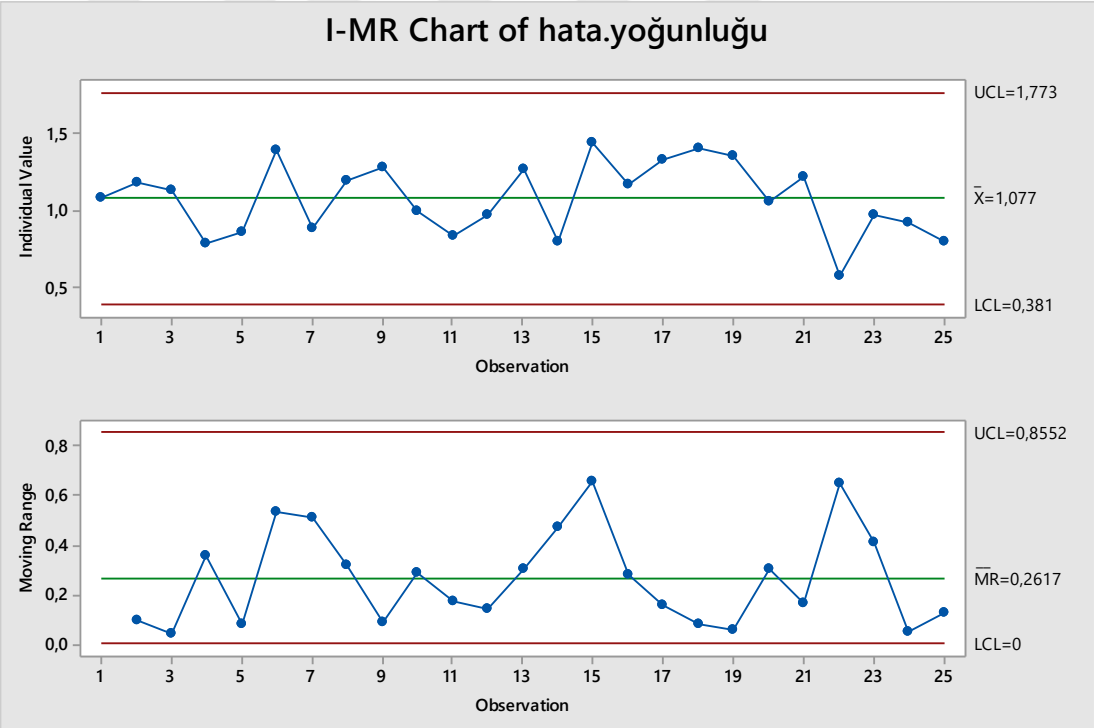
Tablo 4.21. Hata yoğunluğu metriği ölçüm verileri

Proje No	Hata yoğunluğu	Proje No	Hata yoğunluğu
1	1,084	13	1,271
2	1,177	14	0,801
3	1,136	15	1,451
4	0,783	16	1,177
5	0,860	17	1,331
6	1,392	18	1,411
7	0,882	19	1,354

Tablo 4.21.(Devam) Hata yoğunluğu metriği ölçüm verileri

Proje No	Hata yoğunluğu	Proje No	Hata yoğunluğu
8	1,200	20	1,053
9	1,283	21	1,219
10	0,996	22	0,569
11	0,829	23	0,975
12	0,968	24	0,926
		25	0,802

Yazılım hata yoğunluğu değerinin nasıl hesaplandığı Tablo 2.21'de belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış hata bildirim formlarından alınmıştır. Tablo 4.21'de verilerden oluşturulan bireysel değerler için kontrol grafiği Şekil 4.11'de gösterilmiştir.



Şekil 4.11. Hata yoğunluğu kontrol grafiği

4.3.8. Kilometre taşı sapması metriği kontrol grafiği

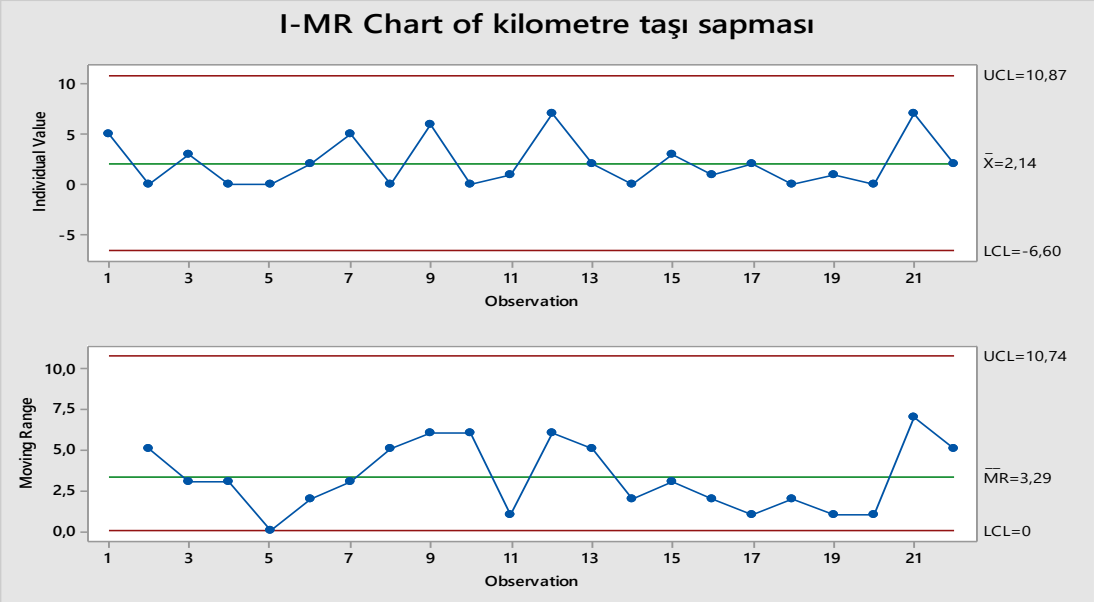
Kilometre taşı sapması metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.11'de verilmiştir. Bu metriği ölçmenin amacı; projenin zaman planına ne kadar uyumlu devam ettiğini ölçmektir. Projede tanımlanan kilometre taşlarının, proje takvimine uygun olup olmadığını tespit etmek ve böylece proje planından sapmayı

ölçer. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.22'de verilmiştir. Veriler projelerin kullandığı kaynak yönetimi aracı olan SAP aracından toplanmıştır.

Tablo 4.22. Kilometre taşı sapması metriği ölçüm verileri

Proje No	Kilometre taşı sapması %	Proje No	Kilometre taşı sapması %
1	5	13	7
2	0	14	2
3	3	15	0
4	0	16	3
5	0	17	1
6	15	18	2
7	2	19	19
8	5	20	0
9	0	21	1
10	6	22	0
11	0	23	7
12	1	24	34
		25	2

Kilometre taşı sapma yüzdesinin nasıl hesaplandığı Tablo 2.11'de belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Tablo 4.22'de kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.12'de gösterilmiştir.



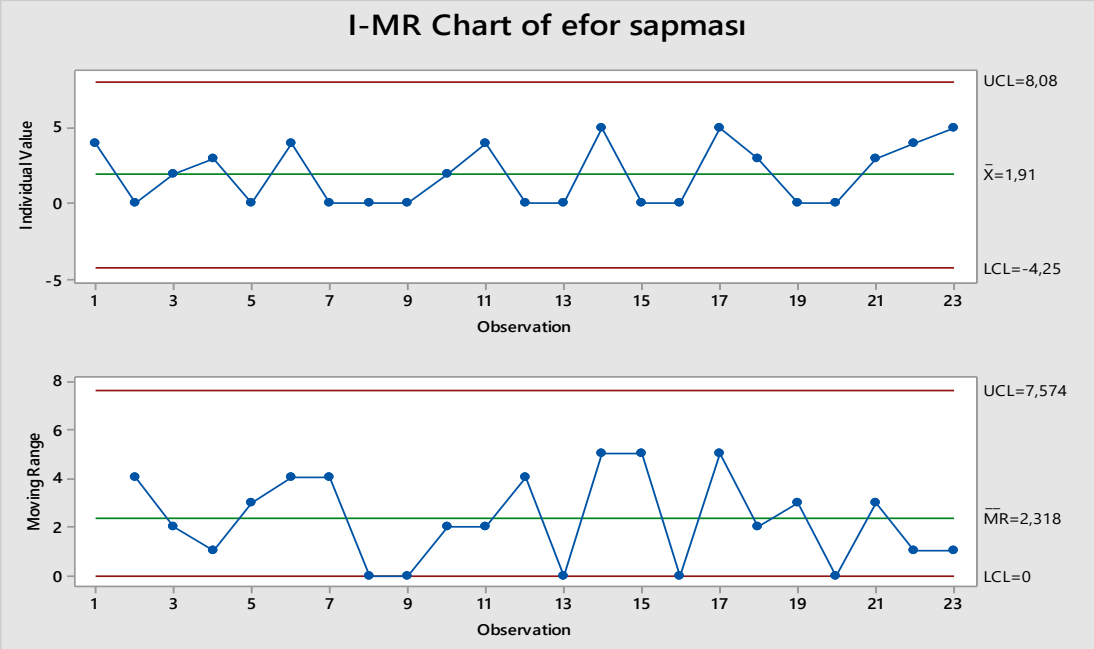
Şekil 4.12. Kilometre taşı sapması kontrol grafiği

4.3.9. Efor sapması metriği kontrol grafiği

Efor sapması metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.12'de verilmiştir. Bu metriği ölçmenin amacı; projenin işgücü planına ne kadar uyumlu devam ettiğini ölçmek ve proje işgücü kullanımının planlanan işgücü kullanımına uygun olup olmadığını tespit ederek proje planından sapmayı ölçmektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.23'te verilmiştir. Veriler projelerin kullandığı kaynak yönetimi aracı olan SAP aracından toplanmıştır.

Tablo 4.23. Efor sapması metriği ölçüm verileri

Proje No	Efor sapması %	Proje No	Efor sapması %
1	4	13	4
2	0	14	0
3	2	15	0
4	3	16	5
5	0	17	0
6	4	18	0
7	0	19	5
8	0	20	3
9	18	21	0
10	0	22	0
11	2	23	3
12	13	24	4
		25	5



Şekil 4.13. Efor sapması kontrol grafiği

Efor sapma yüzdesinin nasıl hesaplandığı Tablo 2.12'de belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Tablo 4.23'te kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.13'de gösterilmiştir.

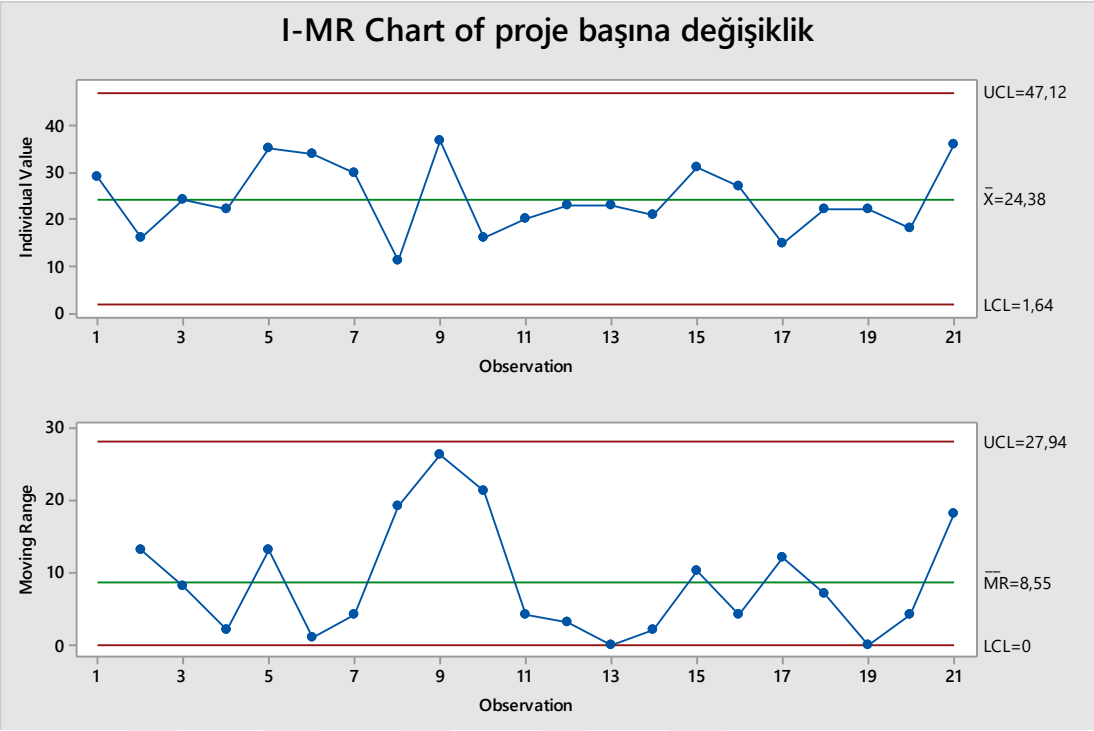
4.3.10. Proje başına ortalama gereksinim değişikliği sayısı metriği kontrol grafiği

Proje başına ortalama gereksinim değişikliği sayısı metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.14'te verilmiştir. Bu metriği ölçmenin amacı; yürütülen projelerde yaşanan gereksinim değişikliklerinin merkezî eğilimini ve süreç değişkenliğini tespit etmek ve projede ilgili dönem sonuna kadar gerçekleşen gereksinim değişikliklerinin kurumun süreç davranışına uygun olup olmadığını denetlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.24'te verilmiştir. Veriler projelerin kullandığı değişiklik yönetimi aracından toplanmıştır.

Tablo 4.24. Proje başı ortalama gereksinim değişiklik sayısı metrik ölçüm verileri

Proje No	Proje başına ortalama gereksinim değişikliği sayısı	Proje No	Proje başına ortalama gereksinim değişikliği sayısı
1	29	13	20
2	16	14	23
3	214	15	23
4	24	16	21
5	22	17	79
6	35	18	31
7	34	19	27
8	146	20	15
9	30	21	22
10	11	22	124
11	37	23	22
12	16	24	18
		25	36

Proje başına ortalama gereksinim değişikliği sayısının nasıl hesaplandığı Tablo 2.14'te belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış değişiklik istek formlarından alınmıştır. Tablo 4.24'te kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.14'te gösterilmiştir.



Şekil 4.14. Proje başına ortalama gereksinim değişikliği sayısı kontrol grafiği

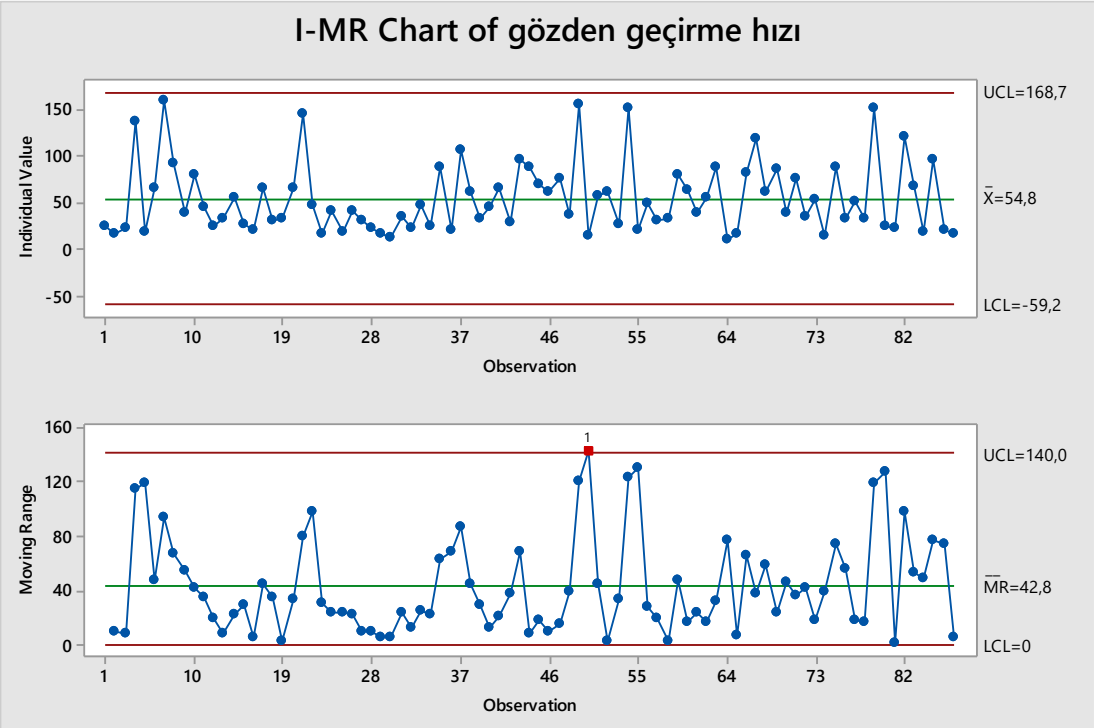
4.3.11. Gözden geçirme hazırlık hızı metriği kontrol grafiği

Gözden geçirme hazırlık hızı metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.22'de verilmiştir. Bu metriği ölçmenin amacı; yürütülen projelerde yaşanan gereksinim değişikliklerinin merkezî eğilimini ve süreç değişkenliğini tespit etmek ve projede ilgili dönem sonuna kadar gerçekleşen gereksinim değişikliklerinin kurumun süreç davranışına uygun olup olmadığını denetlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.25'te verilmiştir. Veriler projelerin kullandığı gözden geçirme aracından toplanmıştır.

Tablo 4.25. Gözden geçirme hazırlık hızı metriği ölçüm verileri

25	45	67	13	45	14	64	40	25
16	26	146	36	66	58	40	76	24
24	34	48	23	29	61	56	35	121
138	182	18	48	97	28	88	53	68
20	56	42	26	89	151	11	14	19
67	27	19	89	71	22	18	88	96
160	21	41	21	61	50	83	33	22
93	66	32	107	76	31	120	51	16
39	31	23	62	37	34	62	34	
80	34	18	33	156	81	86	152	

Tablodaki her hücre projelerde yapılan gözden geçirmelerden elde edilen gözden geçirme hazırlık hızı değerine karşılık gelmektedir. Gözden geçirme hazırlık hızının nasıl hesaplandığı Tablo 2.22'de belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Tablo 4.25'te kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.15'te gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.15. Gözden geçirme hazırlık hızı kontrol grafiği

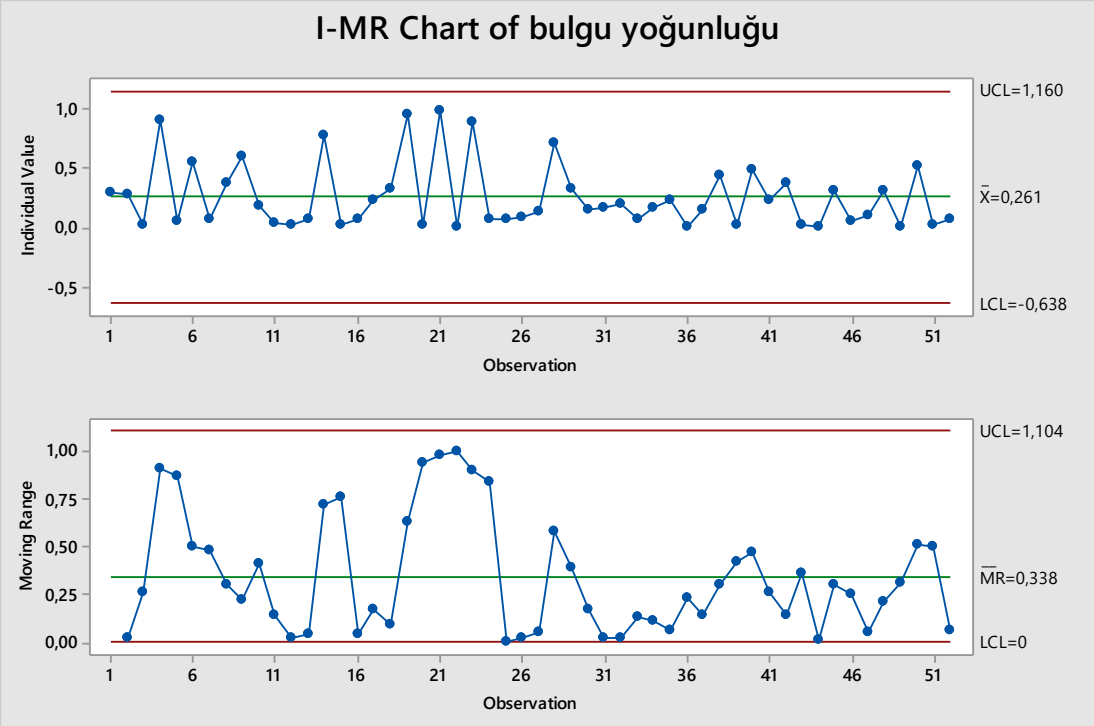
4.3.12. Bulgu yoğunluğu metriği kontrol grafiği

Bulgu yoğunluğu metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.23'te verilmiştir. Bu metriği ölçmenin amacı; projede gerçekleştirilen gözden geçirme bulgu yoğunluğunu ölçmek, kurumda yürütülen projelerde gözden geçirme bulgu yoğunluğunun merkezî eğilimini tespit etmek ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.26'da verilmiştir. Veriler projelerin kullandığı gözden geçirme aracından toplanmıştır.

Tablo 4.26. Bulgu yoğunluğu metriği ölçüm verileri

0,30	0,38	0,79	1,00	0,14	0,18	0,50	0,14	0,11
0,28	0,60	0,03	0,01	0,72	0,24	0,24	0,02	0,32
0,02	0,19	0,07	0,90	0,33	0,01	0,38	1,18	0,01
0,92	0,05	0,24	2,5	0,16	0,15	0,02	0,02	0,52
0,06	0,03	0,33	0,07	0,18	0,45	0,01	0,04	0,02
0,56	0,07	0,96	0,07	0,20	0,03	0,31	0,28	0,08
0,08	2,31	0,03	0,09	0,07	3,34	0,06		

Tablodaki her hücre projelerde yapılan doküman gözden geçirmelerden elde edilen bulgu yoğunluğu değerine karşılık gelmektedir. Bulgu yoğunluğunun nasıl hesaplandığı Tablo 2.23'te belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Tablo 4.26'da kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.16'da gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.16. Bulgu yoğunluğu kontrol grafiği

4.3.13. Gözden geçirme hazırlık verimliliği metriği kontrol grafiği

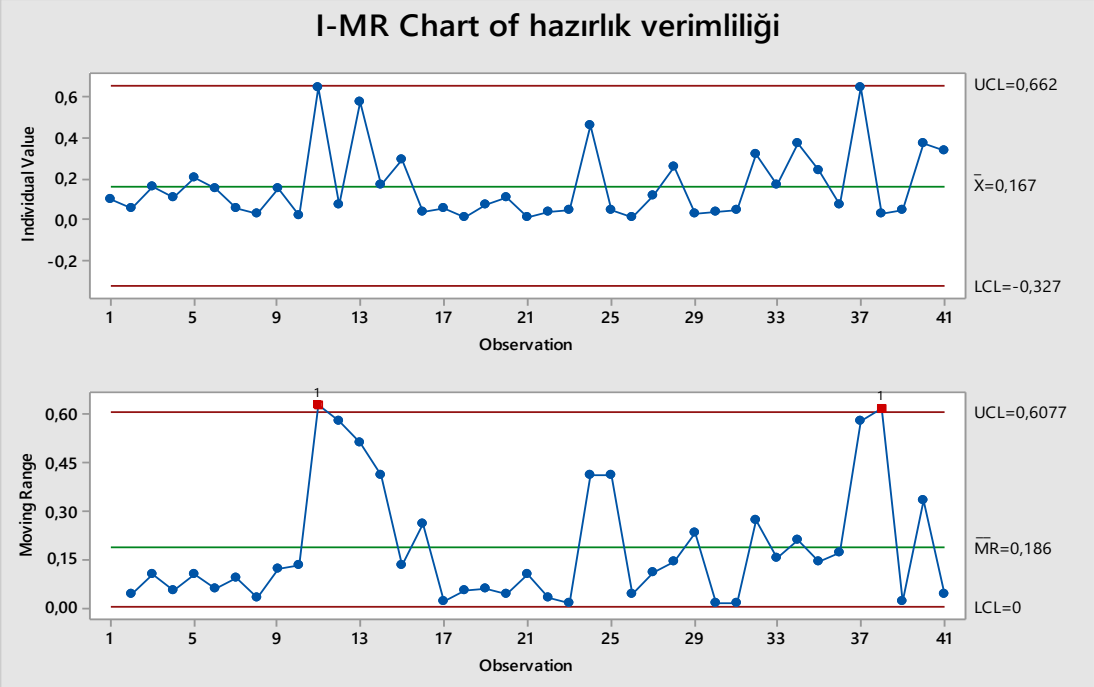
Gözden geçirme hazırlık verimliliği metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.24'te verilmiştir. Bu metriği ölçmenin amacı; projede gerçekleştirilen

gözden geçirme faaliyetlerinin verimliliğini ölçmek ve kurumda yürütülen projelerde gözden geçirme verimliliğinin merkezî eğilimini tespit etmek ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.27'de verilmiştir. Veriler projelerin kullandığı gözden geçirme aracından toplanmıştır.

Tablo 4.27. Gözden geçirme hazırlık verimliliği metriği ölçüm verileri

0,10	0,15	2,54	1,08	0,07	0,46	0,03	0,24	0,05
0,06	0,06	0,65	0,30	0,11	0,05	0,04	0,07	0,38
0,16	1,26	0,07	0,04	2,40	0,01	0,05	2,14	0,34
1,50	0,03	0,58	0,06	0,01	4,95	0,32	0,65	1,97
0,11	0,15	1,79	0,01	0,04	0,12	0,17	0,03	4,47
0,21	0,02	0,17	0,88	0,05	0,26	0,38		

Tablodaki her hücre projelerde yapılan doküman gözden geçirmelerden elde edilen hazırlık verimliliği değerine karşılık gelmektedir. Gözden geçirme hazırlık verimliliğinin nasıl hesaplandığı Tablo 2.24'te belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Tablo 4.27'de kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.17'de gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.17. Gözden geçirme hazırlık verimliliği kontrol grafiği

4.3.14. Hata yakalama yüzdesi metriği kontrol grafiği

Hata yakalama yüzdesi metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.25'te verilmiştir. Bu metriği ölçmenin amacı; yazılım ve dokümantasyondan oluşan iş ürünlerinin müşteri kabulüne sunulmadan önce hatanın ne kadarının geliştirme ve test faaliyetlerinde proje personeli tarafından tespit edilebildiğini ölçmek ve bu durumun merkezî eğilimini tespit etmek ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.28'de verilmiştir. Veriler projelerin kullandığı hata yönetim aracından toplanmıştır.

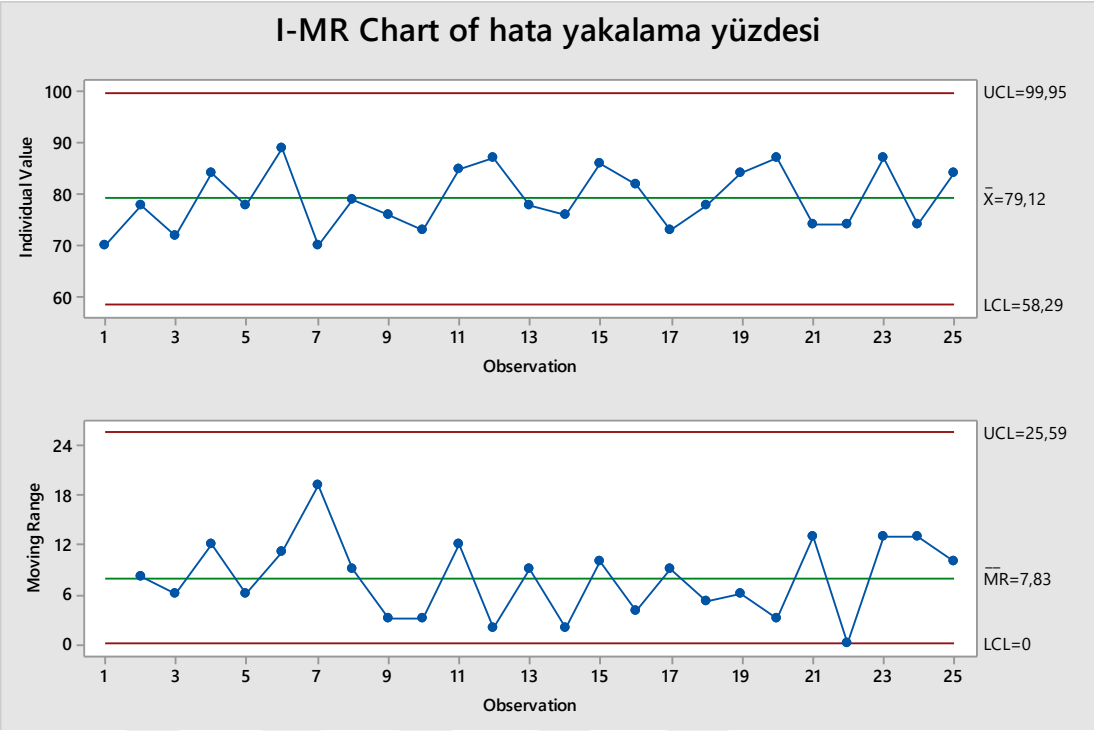
Tablo 4.28. Hata yakalama yüzdesi metriği ölçüm verileri

Proje No	Hata yakalama yüzdesi	Proje No	Hata yakalama yüzdesi
1	70	13	78
2	78	14	76
3	72	15	86
4	84	16	82
5	78	17	73
6	89	18	78
7	70	19	84
8	79	20	87
9	76	21	74
10	73	22	74
11	85	23	87
12	87	24	74
		25	84

Hata yakalama yüzdesinin nasıl hesaplandığı Tablo 2.25'te belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış hata bildirim formlarından alınmıştır.

Tablo 4.28'de kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.18'de gösterilmiştir.

Projelerde hata yakalama yüzdesinin %100'e yakın olması beklendiği için kontrol grafiğindeki üst kontrol limiti dikkate alınmayıp proje başarı hesaplamasında sadece alt kontrol limit değerine göre işlem yapılacaktır.



Şekil 4.18. Hata yakalama yüzdesi kontrol grafiği

4.3.15. Hata tespit yüzdesi metriği kontrol grafiği

Hata yakalama yüzdesi metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.26'da verilmiştir. Bu metriği ölçmenin amacı; projenin çeşitli sürümlerinde (bir ana dış sürüm, o sürüme ait yama sürümleri ve geliştirilmesindeki test sürümleri) tespit edilen hataların ne kadarının proje personeli tarafından tespit edilebildiğini ölmek ve bu durumun merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.29'da verilmiştir. Veriler projelerin kullandığı hata yönetim aracından toplanmıştır.

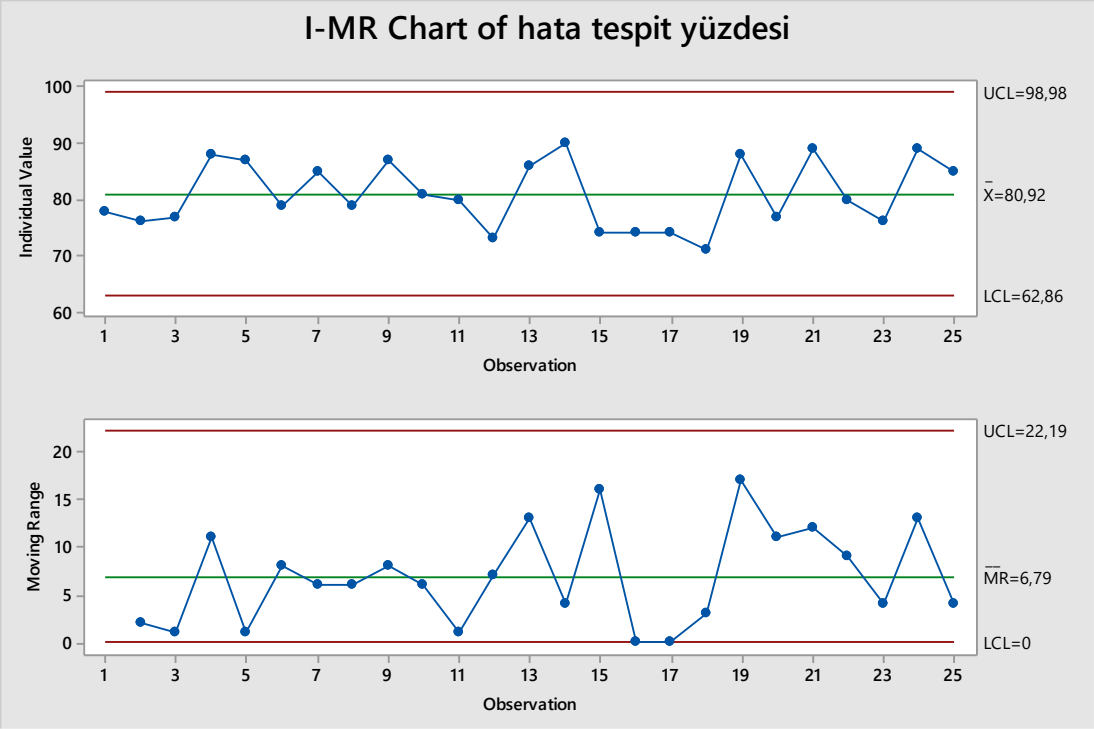
Tablo 4.29. Hata tespit yüzdesi metriği ölçüm verileri

Proje No	Hata yakalama yüzdesi	Proje No	Hata yakalama yüzdesi
1	70	13	78
2	78	14	76
3	72	15	86
4	84	16	82
5	78	17	73
6	89	18	78
7	70	19	84
8	79	20	87
9	76	21	74

Tablo 4.29.(Devam) Hata tespit yüzdesi metriği ölçüm verileri

Proje No	Hata yakalama yüzdesi	Proje No	Hata yakalama yüzdesi
10	73	22	74
11	85	23	87
12	87	24	74
		25	84

Hata tespit yüzdesinin nasıl hesaplandığı Tablo 2.26'da belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış hata bildirim formlarından alınmıştır. Tablo 4.29'da kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.19'da gösterilmiştir. Projelerde hata tespit yüzdesinin %100'e yakın olması beklendiği için kontrol grafiğindeki üst kontrol limiti dikkate alınmayıp proje başarı hesaplamasında sadece alt kontrol limit değerine göre işlem yapılacaktır.



Şekil 4.19. Hata tespit yüzdesi kontrol grafiği

4.3.16. Süreç denetim planlamasından sapma metriği kontrol grafiği

Süreç denetim planlamasından sapma metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.27'de verilmiştir. Bu metriği ölçmenin amacı; projenin süreç denetim

planlamasına ne kadar uyumlu devam ettiğini ölçmek ve kurumda kalite güvence yönetimi süreci kapsamında süreç denetimi yapılacak projeler için planlanan süreç denetimlerinden sapma miktarının merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.30'da verilmiştir. Veriler projelerin kullandığı iş yönetim aracından toplanmıştır.

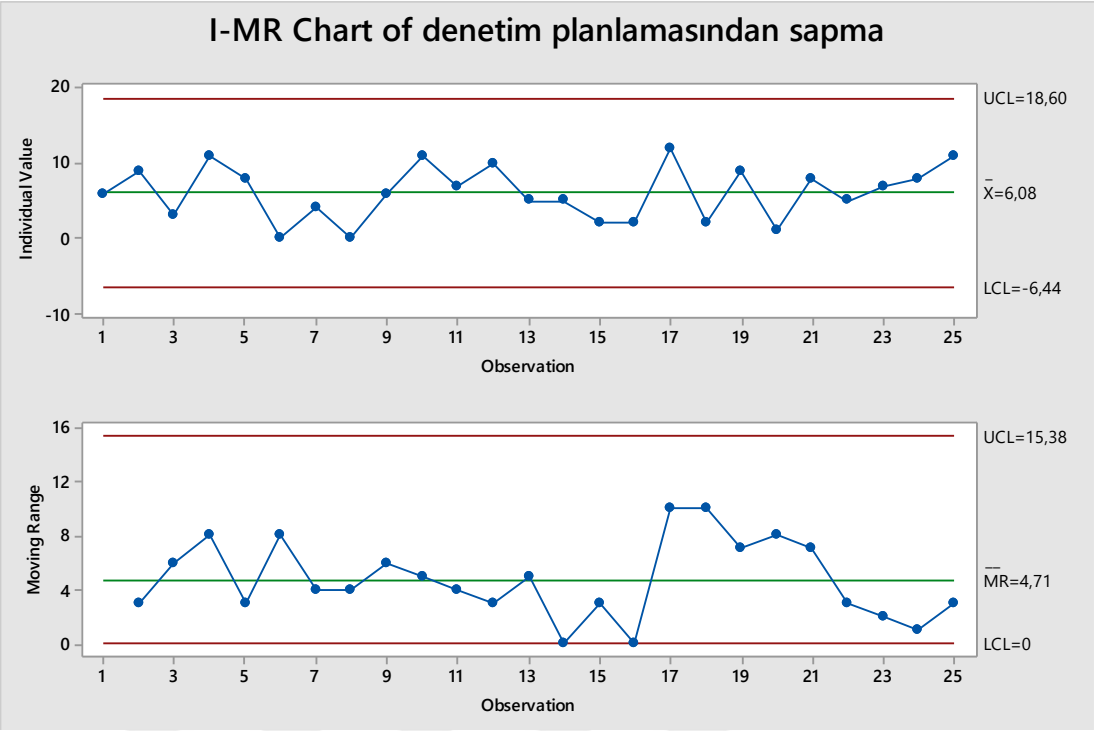
Tablo 4.30. Süreç denetim planlamasından sapma metriği ölçüm verileri

Proje No	Süreç denetiminden sapma	Proje No	Süreç denetiminden sapma
1	6	13	5
2	9	14	5
3	3	15	2
4	11	16	2
5	8	17	12
6	0	18	2
7	4	19	9
8	0	20	1
9	6	21	8
10	11	22	5
11	7	23	7
12	10	24	8
		25	11

Süreç denetim planlamasından sapma değerinin nasıl hesaplandığı Tablo 2.27'de belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış süreç denetim formlarından alınmıştır.

Tablo 4.30'da kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.20'de gösterilmiştir. 18,60 üst limit değerinden daha fazla sapma gösteren metrik ölçümlerinin kök neden analizi sürecine girerek sapmaya neden olan sebeplerinin araştırılması gerekmektedir. Projenin işlettiği süreç adımlarının kurumsal süreçlere uygunluğu planlanan zamanda gerçekleştirilmelidir. Zamanında gerçekleştirilmeyen denetimler proje ürünlerinde uygunsuzluğa neden olabilmektedir.

Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.20. Süreç denetim planlamasından sapma kontrol grafiği

4.3.17. Proje uygunsuzluklarının ortalama açık kalma süresi metriği kontrol grafiği

Proje uygunsuzluklarının ortalama açık kalma süresi metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.28'de verilmiştir. Bu metriği ölçmenin amacı; projede tespit edilen uygunsuzlukların ne kadar sürede çözümlendiğini ölçmek ve kurumda yürütülen projelerde kalite güvence süreci kapsamında projeler için açılan uygunsuzlukların ilgili dönem sonuna kadar ortalama açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.31'de verilmiştir. Veriler projelerin kullandığı iş yönetim aracından toplanmıştır.

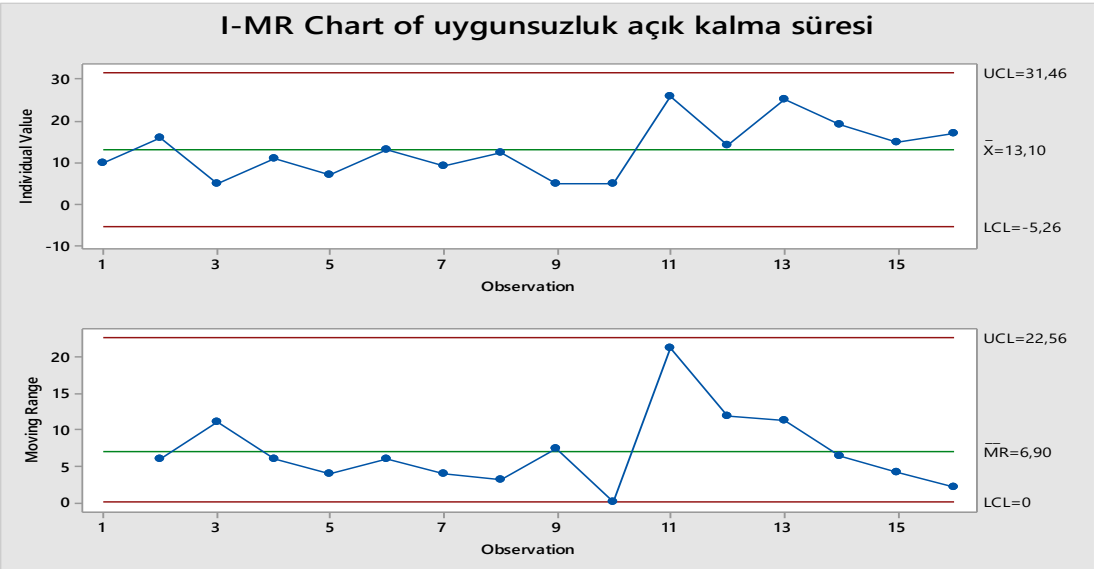
Tablo 4.31. Proje uygunsuzluklarının ortalama açık kalma süresi metriği ölçüm verileri

Kapanma Süresi	Kapanma Tarihi	Bildirim Tarihi
10	24.11.2019 12:56	14.11.2019 10:10
16	28.11.2019 17:03	12.11.2019 16:34
5	17.11.2019 12:34	12.11.2019 13:46
11	22.11.2019 13:56	11.11.2019 15:25

Tablo 4.31.(Devam) Proje uygunsuzluklarının ortalama açık kalma süresi metriği ölçüm verileri

Kapanma Süresi	Kapanma Tarihi	Bildirim Tarihi
7	11.11.2019 16:12	04.11.2019 09:12
13	17.11.2019 15:43	04.11.2019 10:33
9	24.10.2019 14:47	15.10.2019 11:45
12	24.10.2019 14:52	12.10.2019 09:05
5	16.05.2019 13:22	11.05.2019 13:58
5	16.05.2019 13:22	11.05.2019 13:57
26	29.05.2019 11:18	03.05.2019 11:44
14	24.01.2019 14:06	10.01.2019 09:31
25	14.01.2019 20:10	20.12.2018 12:55
19	08.01.2019 11:46	20.12.2018 11:37
15	20.12.2018 14:22	05.12.2018 16:33
17	22.12.2018 14:24	05.12.2018 16:26

Proje uygunsuzluklarının ortalama açık kalma süresi bildirim tarihi ile kapanma tarihi arasındaki geçen sürenin gün cinsinden değeridir. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış düzeltici faaliyet formlarından alınmıştır. Tablo 4.31'de kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.21'de gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.21. Proje uygunsuzluklarının ortalama açık kalma süresi kontrol grafiği

4.3.18. Konfigürasyon denetim planlamasından sapma metriği kontrol grafiği

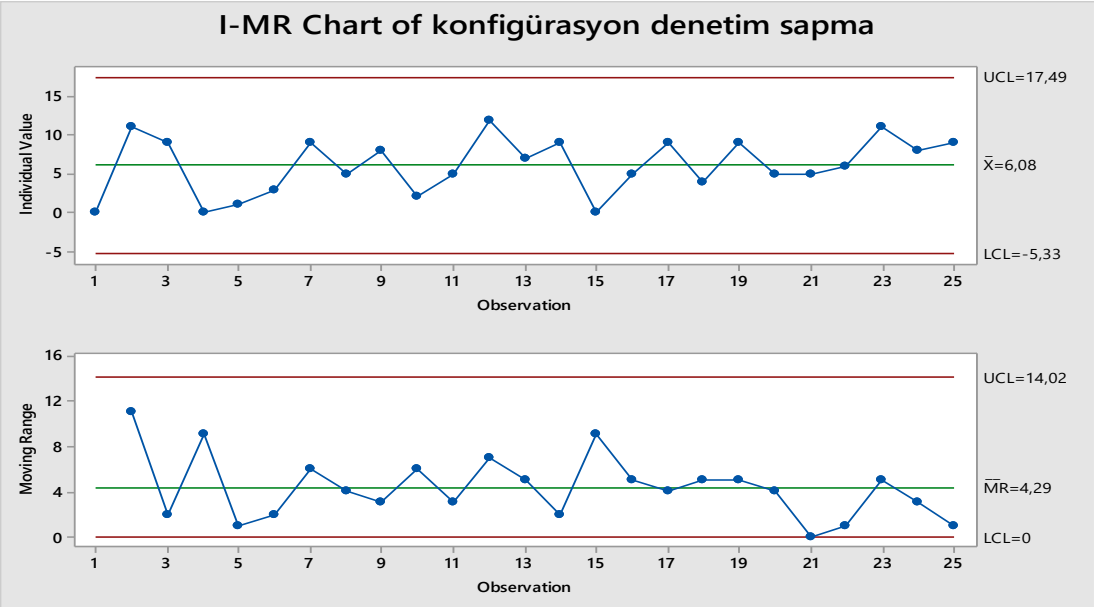
Konfigürasyon denetim planlamasından sapma metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.29'da verilmiştir. Bu metriği ölçmenin amacı; projenin konfigürasyon denetim planlamasına ne kadar uyumlu devam ettiğini ölçmek ve kurumda konfigürasyon yönetimi süreci kapsamında konfigürasyon denetimi yapılacak projeler için planlanan süreç denetimlerinden sapma miktarının merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.32'de verilmiştir. Veriler projelerin kullandığı iş yönetim aracından toplanmıştır.

Tablo 4.32. Konfigürasyon denetim planlamasından sapma metriği ölçüm verileri

Proje No	Konfigürasyon denetiminden sapma	Proje No	Konfigürasyon denetiminden sapma
1	0	13	7
2	11	14	9
3	9	15	0
4	0	16	5
5	1	17	9
6	3	18	4
7	9	19	9
8	5	20	5
9	8	21	5
10	2	22	6
11	5	23	11
12	12	24	8
		25	9

Konfigürasyon denetim planlamasından sapma değerinin nasıl hesaplandığı Tablo 2.29'de belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış konfigürasyon denetim formlarından alınmıştır.

Tablo 4.32'de kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.22'de gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.22. Konfigürasyon denetim planlamasından sapma kontrol grafiği

4.3.19. Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı metriği kontrol grafiği

Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.30'da verilmiştir. Bu metriği ölçmenin amacı; kurumda konfigürasyon yönetimi süreci kapsamında denetim yapılacak projeler için planlanan denetimlerde bulunan proje başına düşen uygunsuzluk miktarını tespit etmektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.33'te verilmiştir. Veriler projelerin kullandığı değişiklik yönetimi aracından toplanmıştır.

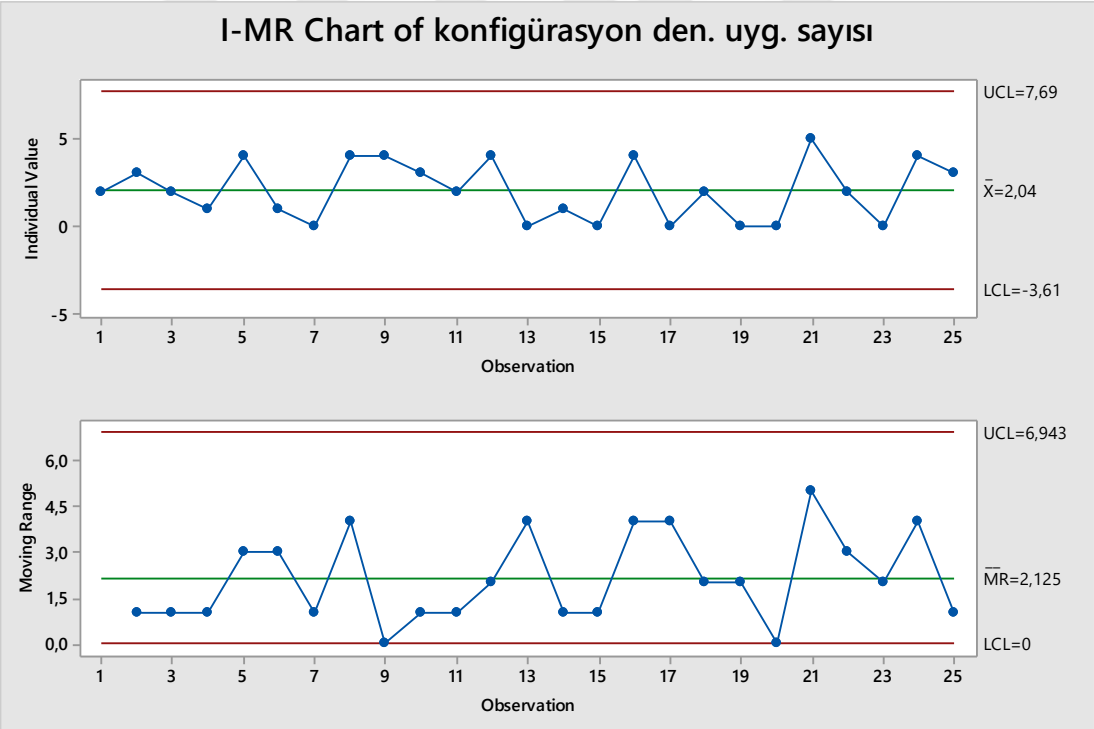
Tablo 4.33. Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı metriği ölçüm verileri

Proje No	Konfigürasyon denetimi başına uygunsuzluk sayısı	Proje No	Konfigürasyon denetimi başına uygunsuzluk sayısı
1	2	13	0
2	3	14	1
3	2	15	0
4	1	16	4
5	4	17	0
6	1	18	2
7	0	19	0
8	4	20	0
9	4	21	5

Tablo 4.33.(Devam) Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı metriği ölçüm verileri

Proje No	Konfigürasyon denetimi başına uygunsuzluk sayısı	Proje No	Konfigürasyon denetimi başına uygunsuzluk sayısı
10	3	22	2
11	2	23	0
12	4	24	4
		25	3

Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısının nasıl hesaplandığı Tablo 2.30'da belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler düzeltilmiş ve kapatılmış düzeltici faaliyet formlarından alınmıştır. Tablo 4.33'te kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.23'te gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.23. Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı kontrol grafiği

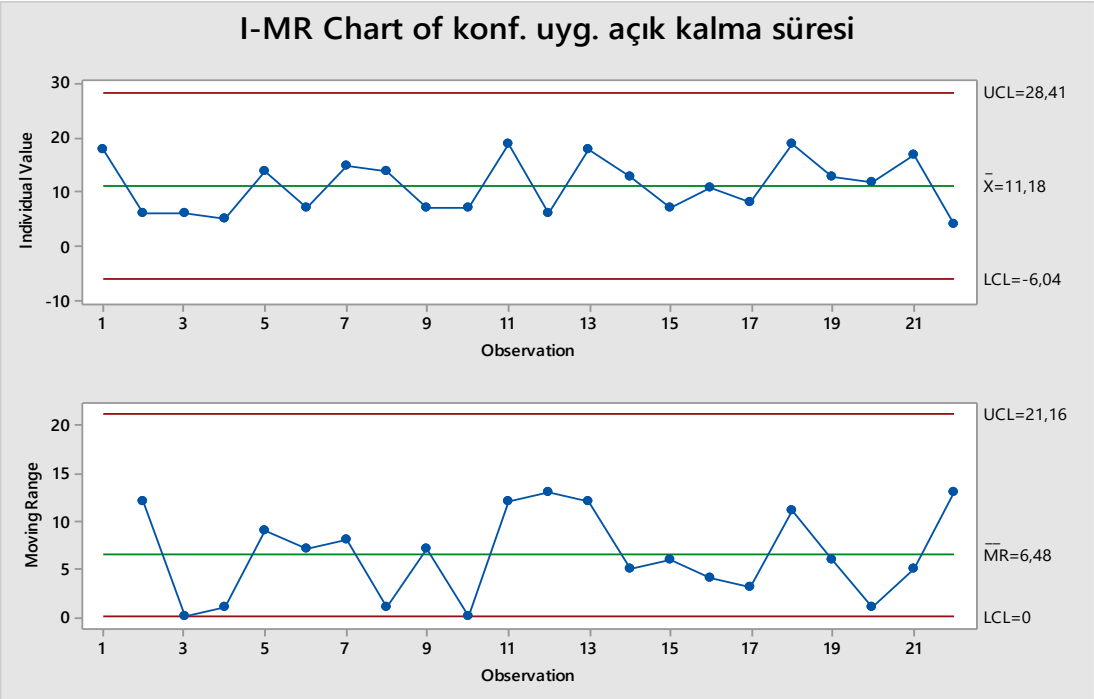
4.3.20. Konfigürasyon uygunsuzluklarının ortalama açık kalma süresi metriği kontrol grafiği

Konfigürasyon uygunsuzluklarının ortalama açık kalma süresi metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.31'de verilmiştir. Bu metriği ölçmenin amacı; projede konfigürasyon denetimleri kapsamında açılan konfigürasyon uygunsuzluklarının ortalama açık kalma sürelerini tespit etmek ve kurumda yürütülen projelerde konfigürasyon uygunsuzluklarının ortalama açık kalma sürelerinin merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.34'de verilmiştir. Veriler projelerin kullandığı iş yönetim aracından toplanmıştır.

Tablo 4.34. Konfigürasyon uygunsuzluklarının ortalama açık kalma süresi metriği ölçüm verileri

Proje No	Konfigürasyon uygunsuzluklarının açık kalma süresi	Proje No	Konfigürasyon uygunsuzluklarının açık kalma süresi
1	18	13	19
2	6	14	55
3	6	15	6
4	5	16	18
5	14	17	13
6	7	18	7
7	34	19	11
8	15	20	8
9	14	21	19
10	41	22	13
11	7	23	12
12	7	24	17
		25	4

Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısının nasıl hesaplandığı Tablo 2.31'de belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler kapatılmış düzeltici faaliyet formlarından alınmıştır. Tablo 4.34'de kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.24'de gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.24. Konfigürasyon uygunsuzluklarının ortalama açık kalma süresi kontrol grafiği

4.3.21. Konfigürasyon uygunsuzluklarının açık kalma oranı metriği kontrol grafiği

Konfigürasyon uygunsuzluklarının açık kalma oranı metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.32'de verilmiştir. Bu metriği ölçmenin amacı; ilgili dönemde açık durumda bulunan konfigürasyon uygunsuzluk sayısının toplam konfigürasyon uygunsuzluk sayısı içindeki payını tespit etmek ve kurumda yürütülen projelerde konfigürasyon uygunsuzluklarının açık kalma oranının merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.35'te verilmiştir. Veriler projelerin kullandığı değişiklik yönetim aracından toplanmıştır.

Tablo 4.35. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı metriği ölçüm verileri

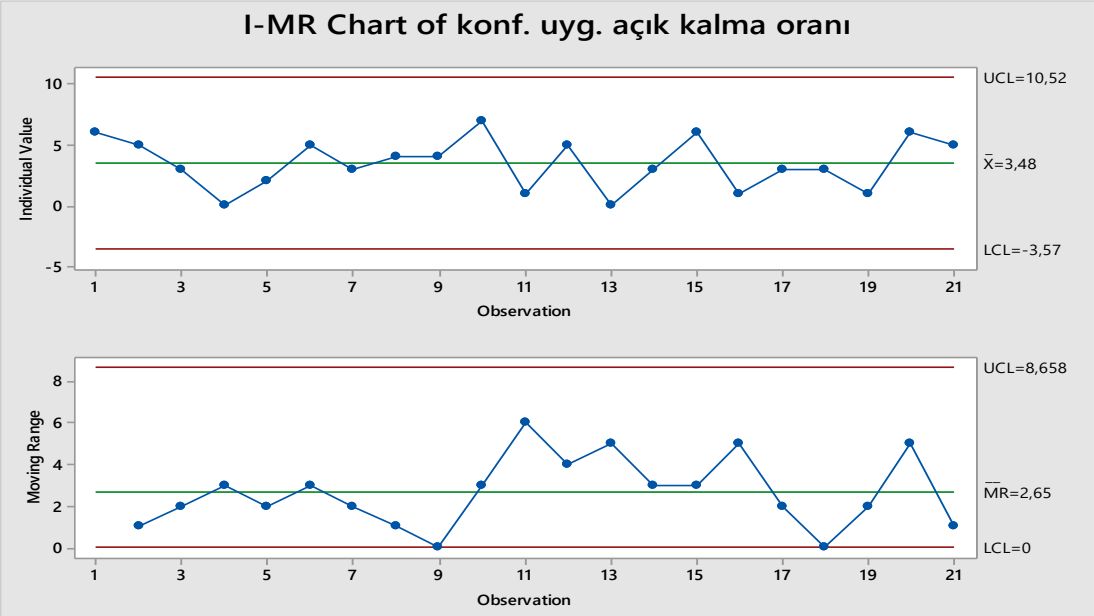
Proje No	Konfigürasyon uygunsuzluklarının açık kalma oranı	Proje No	Konfigürasyon uygunsuzluklarının açık kalma oranı
1	6	13	1
2	5	14	5
3	3	15	19
4	0	16	0
5	2	17	3

Tablo 4.35.(Devam) Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı metriği ölçüm verileri

Proje No	Konfigürasyon uygunsuzluklarının açık kalma oranı	Proje No	Konfigürasyon uygunsuzluklarının açık kalma oranı
6	22	18	6
7	5	19	1
8	3	20	3
9	4	21	3
10	4	22	1
11	26	23	6
12	7	24	14
		25	5

Konfigürasyon uygunsuzluklarının açık kalma oranı nasıl hesaplandığı Tablo 2.32'de belirtilmiştir. Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler kapatılmış düzeltici faaliyet formlarından alınmıştır.

Tablo 4.35'te kontrol limitleri dışındaki veriler çıkarılıp kalan verilerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.25'te gösterilmiştir. Bu metrik için negatif bir değere sahip olmak mantık olarak mümkün olmadığından negatif değer olarak hesaplanan alt kontrol sınırının değeri sıfıra kaydırılır.



Şekil 4.25. Konfigürasyon uygunsuzluklarının açık kalma oranı kontrol grafiği

4.3.22. Risk gözden geçirme oranı metriği kontrol grafiği

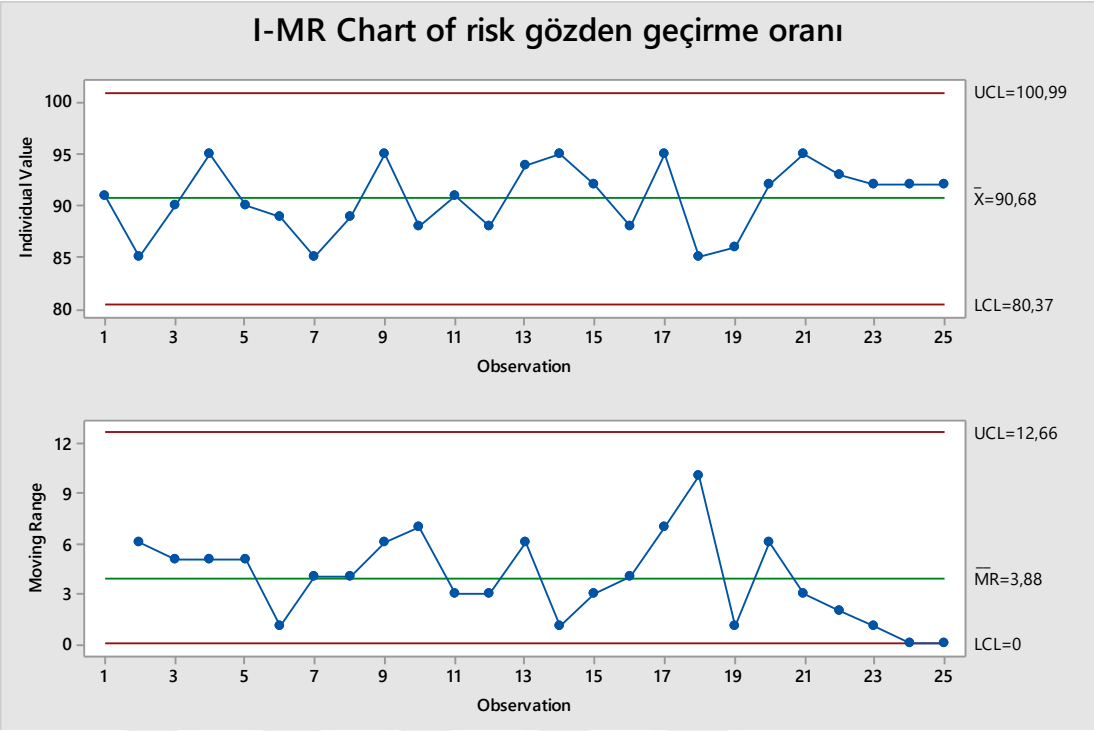
Risk gözden geçirme oranı metriği ile ilgili detaylı bilgi ölçüm bilgi formu şeklinde Tablo 2.33'te verilmiştir. Bu metriği ölçmenin amacı; proje başından ilgili dönem sonuna kadar planlanan risk gözden geçirme faaliyetlerinin ne kadarının gerçekleştirildiğini tespit etmek ve kurumda yürütülen projelerde risk gözden geçirme oranının merkezî eğilimini ve süreç değişkenliğini belirlemektir. Çalışmanın yapıldığı Ar-Ge kurumundaki projelerden toplanan veriler Tablo 4.36'da verilmiştir. Veriler projelerin kullandığı değişiklik yönetim aracından toplanmıştır.

Tablo 4.36. Risk gözden geçirme oranı metriği ölçüm verileri

Proje No	Risk gözden geçirme oranı	Proje No	Risk gözden geçirme oranı
1	91	13	94
2	85	14	95
3	90	15	92
4	95	16	88
5	90	17	95
6	89	18	85
7	85	19	86
8	89	20	92
9	95	21	95
10	88	22	93
11	91	23	92
12	88	24	92
		25	92

Risk gözden geçirme oranı nasıl hesaplandığı Tablo 2.33'te belirtilmiştir.

Veriler, çalışmanın yapıldığı kurumdaki 25 projeden alınmıştır. Veri analizine giren bilgiler projenin kullandığı iş takip aracında bulunan risk formlarından toplanmıştır. Bir risk formunda bildirme, değerlendirme ve kapandı durumları vardır. Bir riskin gözden geçirilmesi ilgili riskin bildirme durumunda olmaması anlamına gelmektedir. Proje çalışanları tarafından iş takip aracına kaydedilen bir risk önce bildirme durumunda bulunur. Proje çalışanları tarafından gözden geçirilen bir risk değerlendirme durumuna geçer. Değerlendirme ve kapandı durumunda olan risklerden elde edilen ölçümlerden oluşturulan kontrol altındaki sürece ait bireysel değerler için kontrol grafiği Şekil 4.26'da gösterilmiştir.



Şekil 4.26. Risk gözden geçirme oranı kontrol grafiği

4.3.23. Üst ve alt kontrol limitleri özet tablosu

Bölüm 4.3'te süreç metrikleri için oluşturulan kontrol grafiklerindeki üst ve alt kontrol limit değerleri Tablo 4.37'de verilmiştir.

Tablo 4.37. Süreç metrikleri üst ve alt kontrol limit değerleri

Metrik Adı	Alt limit değeri	Üst limit değeri
PY 1.1. Kilometre taşı sapması (%)	0	10,87
PY 1.2. Efor sapması (%)	0	8,08
PY 1.3. Gelir/gider endeksi	>1	
GY 1.1. Proje başına ortalama gereksinim değişikliği sayısı (adet)	1,64	47,12
GEL 1.1. Müşteri tarafından raporlanan hata açık kalma süresi (gün)	0	29,87
GEL 1.2. Müşteri tarafından raporlanan iyileştirme açık kalma süresi (gün)	4,25	46,22
GEL 1.3. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma süresi (gün)	0	34,21
GEL 1.4. Müşteri tarafından raporlanan hata açık kalma oranı (%)	0	11,13

Tablo 4.37.(Devam) Süreç metrikleri üst ve alt kontrol limit değerleri

Metrik Adı	Alt limit değeri	Üst limit değeri
GEL 1.5. Müşteri tarafından raporlanan iyileştirme açık kalma oranı (%)	0	19,20
GEL 1.6. Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı (%)	0	9,13
GEL 1.7. Hata yoğunluğu (adet/1000 satır kod)	0,38	1,77
DG 1.1. Gözden geçirme hazırlık hızı (büyüklük/saat)	0	168,70
DG 1.2. Bulgu yoğunluğu (adet/büyüklük)	0	1,16
DG 1.3. Gözden geçirme hazırlık verimliliği (adet/saat)	0	0,66
DG 1.4. Hata yakalama yüzdesi (%)	58,29	99,95
DG 1.5. Hata tespit yüzdesi (%)	62,86	98,98
KG 1.1. Süreç denetim planlamasından sapma (gün)	0	18,60
KG 1.2. Proje uygunsuzluklarının ortalama açık kalma süresi (gün)	0	31,46
KY 1.1. Konfigürasyon denetim planlamasından sapma (gün)	0	17,49
KY 1.2. Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı (adet)	0	7,69
KY 1.3. Konfigürasyon Uygunsuzluklarının Açık Kalma Süresi (gün)	0	28,41
KY 1.4. Konfigürasyon Uygunsuzluklarının Açık Kalma Oranı (%)	0	10,52
RY 1.1. Risk gözden geçirme oranı (%)	80,37	100

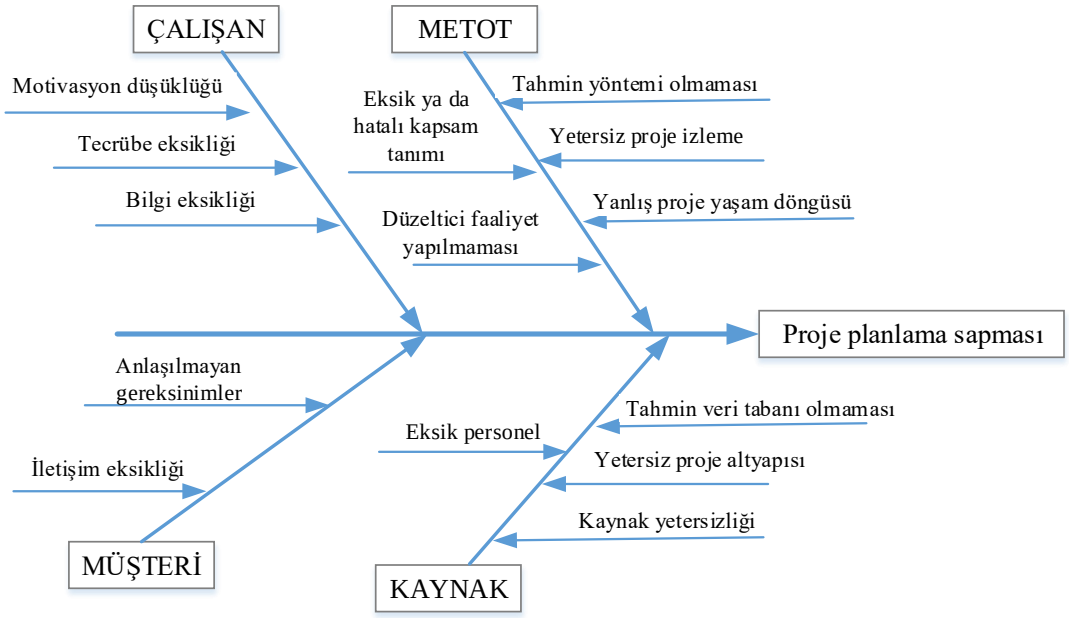
4.4. Neden-Sonuç Analizleri

Bu bölümde; Bölüm 4.3'te kontrol grafiklerinin alt ve üst kontrol limitleri dışındaki noktaların kök nedenlerini tespit etmek için Bölüm 2.3.7'de belirtilen neden-sonuç diyagramı kullanılarak analizler gerçekleştirilmiştir. Kök neden analizi ana kriterler için gerçekleştirilmiştir.

4.4.1. Proje yönetimi süreci neden-sonuç diyagramı

Proje yönetimi sürecine ait alt kriterler kilometre taşı sapması, efor sapması ve gelir/gider endeksidir. Alt kriterlerin ortak özelliği proje planlama ile ilgili olmasıdır. Proje planlama sapmasının neden sonuç diyagramı Şekil 4.27'de verilmiştir. Buna göre metot, çalışan, kaynak ve müşteri ana nedenler olarak gözlemlenmiştir. Bu ana nedenler kendi içerisinde alt nedenlere ayrıştırılarak gerekli önlemlerin alınması yönünde nereden başlanılacağı konusunda doğru karar vermek amaçlanmıştır.

Neden-sonuç diyagramı uygulanarak proje planlamasından sapmaya sebep olabilecek ana nedenler ve hataya sebep olabilecek alt nedenler önem sırasına göre ayrılmıştır. Buna göre; çalışanların motivasyon düşüklüğü, tecrübe eksikliği, proje alanında bilgi eksikliği, müşteri ile iletişim eksikliği, müşteri ile ortak anlaşmaya varılamayan gereksinimlerin varlığı, proje planlamasına esas teşkil eden tahminleme ile ilgili bir yöntemin olmaması, projeye yeterli çalışan personel ataması yapılmaması, proje geliştirme ile ilgili yeterli altyapı araçlarının olmaması, projeye malzeme, teçhizat gibi yeterli kaynak ayrılmaması, projenin başında proje kapsamının tam ve doğru olarak belirlenememesi, proje izleme ve kontrol alt süreçlerinin işletilmemesi, projedeki problemler için düzeltici ve önleyici faaliyetlerin uygulanmaması ve projeye uygun olmayan yaşam döngüsü modeli seçimi proje planlama sapmasının sebepleri olarak sayılabilmektedir.



Şekil 4.27. Proje yönetimine ait metriklerin neden sonuç diyagramı

Planlama sapmasına yol açan tüm bu nedenler, Ar-Ge kurumunda çalışan uzmanlardan oluşan bir ekip ile beyin fırtınası tekniği kullanılarak belirlenmiştir. Buna göre; hata nedenleri, kutucuklar içerisinde gösterilirken bu nedenlere sebebiyet alt nedenler oklarla belirtilmiştir.

Şekil 4.27'de verilen neden-sonuç diyagramı incelendiğinde; proje planlama sapmasına genelde metot ve kaynak kökenli sebeplerin yol açtığı gözlenmiştir. Yine müşteri ve çalışan kaynaklı nedenlerin de aslında metot ve kaynak merkezli olduğu belirlenmiştir. Proje planlaması kapsamında tahminleme ile ilgili metot eksikliği, tahminleme için benzer eski proje bilgileri kaydının olmaması ya da kolay erişilememesi, proje karakteristiğine uygun yaşam döngüsü seçilmemesi ve projeye yeterli insan, malzeme ve teçhizat kaynağı aktarılmaması sapmanın temel hata nedenleri olduğu belirlenmiştir. Bu sapmanın düzeltilmesi veya yeniden meydana gelmesinin önlenebilmesi için bu nedenlerin ortadan kaldırılması gerektiği belirlenmiştir.

4.4.2. Gereksinim yönetimi süreci neden-sonuç diyagramı

Gereksinim yönetimi sürecine ait alt kriterler proje başına düşen ortalama gereksinim değişikliği sayısıdır. Projelerde gereksinimlerin çok değişmemesi beklenir. Çünkü değişen gereksinimler projede birçok değişikliğe neden oldukları için projenin bütçesine aşmasına ve teslim zamanının uzamasına neden olmaktadır. Bu nedenle gereksinim değişikliği projeler için bir problem teşkil etmektedir. Gereksinim değişikliği neden sonuç diyagramı Şekil 4.28'de verilmiştir. Buna göre metot, çalışan, kaynak ve müşteri ana nedenler olarak gözlemlenmiştir. Bu ana nedenler kendi içerisinde alt nedenlere ayrıştırılarak gerekli önlemlerin alınması yönünde nereden başlanılacağı konusunda doğru karar vermek amaçlanmıştır. Neden-sonuç diyagramı uygulanarak gereksinim değişikliği sayısının artmasına neden olabilecek ana nedenler ve hataya sebep olabilecek alt nedenler önem sırasına göre ayrılmıştır. Buna göre; çalışanların motivasyon düşüklüğü, tecrübe eksikliği, proje alanında uzmanlık olmaması, gereksinimlerin detaylandırılmaması, müşteri ile iletişim eksikliği, müşteri ile ortak anlaşmaya varılamayan gereksinimlerin varlığı, gereksinimlerin müşteri ile beraber gözden geçirilmemesi, gereksinimler için müşteriden taahhüt/onay alınmaması, gereksinimlerin tanımlanma ve temin etme yöntemlerinin tanımlı

olmaması, projenin kavramsal tasarımının yapılmamış ve geliştirilecek sistemin/yazılım operasyonel senaryolarının tanımlı olmaması, gereksinim yönetimi ile ilgili araç kullanılmaması ve dolayısıyla gereksinimlerin proje boyunca izlenememesi, projeye malzeme ve teçhizat gibi yeterli kaynak ayrılmaması ve projeye uygun olmayan yaşam döngüsü modeli seçimi gereksinim değişikliği probleminin sebepleri olarak sayılabilmektedir.



Şekil 4.28. Gereksinim yönetimine ait metriklerin neden sonuç diyagramı

Gereksinim değişikliğine yol açan tüm bu nedenler, Ar-Ge kurumunda çalışan uzmanlardan oluşan bir ekip ile beyin fırtınası tekniği kullanılarak belirlenmiştir. Buna göre; hata nedenleri, kutucuklar içerisinde gösterilirken bu nedenlere sebebiyet alt nedenler oklarla belirtilmiştir.

Şekil 4.28'de verilen neden-sonuç diyagramı incelendiğinde; gereksinim değişikliği problemine genelde metot ve müşteri kökenli sebeplerin yol açtığı gözlenmiştir. Projede geliştirilecek ürün ile ilgili gereksinimlerin müşteri ile detaylı bir şekilde belirlenmesi, gözde geçirilmesi ve taahhüt alınması noktasındaki sorunlar, gereksinimlerin nasıl temin edileceği ve nasıl iyi derecede tanımlanacağı konusundaki metot eksikliği ve gereksinimlerin değişiklik kaydı ve izlenebilirlik bilgilerinin yönetildiği araç kullanımındaki eksiklikler gereksinim değişikliğindeki temel hata nedenleri olduğu belirlenmiştir. Gereksinim değişikliği sayısını azaltmak için bu nedenlerin ortadan kaldırılması gerektiği belirlenmiştir.

4.4.3. Geliştirme süreci neden-sonuç diyagramı

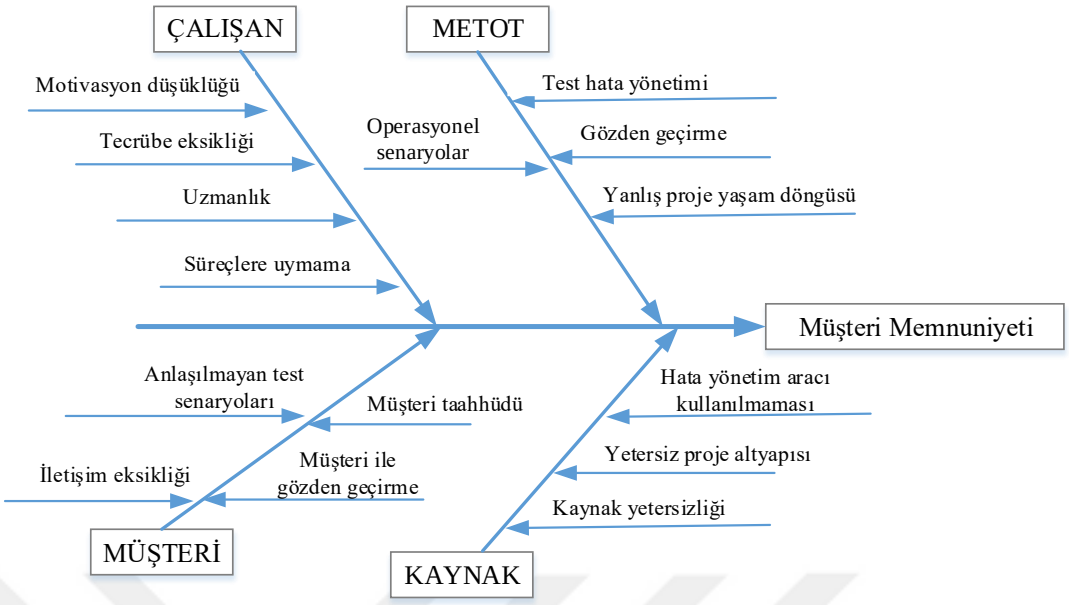
Geliştirme sürecine ait alt kriterler müşteri tarafından raporlanan hata, iyileştirme ve gereksinim değişikliklerinin ortalama kapatılma süresi ve bunların ortalama açık kalma oranıdır. Bu metrikleri ölçmenin amacı müşteri memnuniyetinin artırılmasıdır. Bu metriklerin ortalama açık kalma süresi ve oranının mümkün olduğunca az olması beklenir ve zaman içerisinde azalan bir trende sahip olması istenir.

Diğer taraftan hata yoğunluğu metriği ise ürün kalitesi ile ilgilidir. Hata yoğunluğunun az olması ürünün kalitesinin yüksek olması anlamına gelmektedir. Bu nedenle geliştirme sürecine ait iki ana problem olduğu söylenebilir. Birincisi müşteri memnuniyeti ikincisi ürün kalitesi problemidir. Bu iki probleme ait neden sonuç diyagramları Şekil 4.29 ve Şekil 4.30'da verilmiştir.

Her iki diyagrama göre metot, çalışan, kaynak ve müşteri ana nedenler olarak gözlemlenmiştir. Bu ana nedenler kendi içerisinde alt nedenlere ayrıştırılarak gerekli önlemlerin alınması yönünde nereden başlanılacağı konusunda doğru karar vermek amaçlanmıştır.

Neden-sonuç diyagramı uygulanarak müşteri memnuniyetinden ve ürün kalitesinden sapmaya sebep olabilecek ana nedenler ve hataya sebep olabilecek alt nedenler önem sırasına göre ayrılmıştır.

Müşteri memnuniyetine yönelik diyagrama göre; çalışanların motivasyon düşüklüğü, tecrübe eksikliği, proje alanında uzmanlık olmaması, çalışanın süreç adımlarına uymaması, müşteri ile iletişim eksikliği, müşteri ile ortak anlaşmaya varılamayan test senaryoları, test dokümanlarının müşteri ile beraber gözden geçirilmemesi, test dokümanları ve test senaryoları için müşteriden taahhüt/onay alınmaması, test hata yönetimi ve gözden geçirme yöntemlerinin tanımlı olmaması, geliştirilecek sistemin/yazılım operasyonel senaryolarının tanımlı olmaması, hata yönetimi ile ilgili araç kullanılmaması, projenin geliştirme ve test altyapısının yetersiz olması, projeye malzeme ve teçhizat gibi yeterli kaynak ayrılmaması ve projeye uygun olmayan yaşam döngüsü modeli seçimi geliştirme sürecine ait müşteri memnuniyeti probleminin sebepleri olarak sayılabilmektedir.

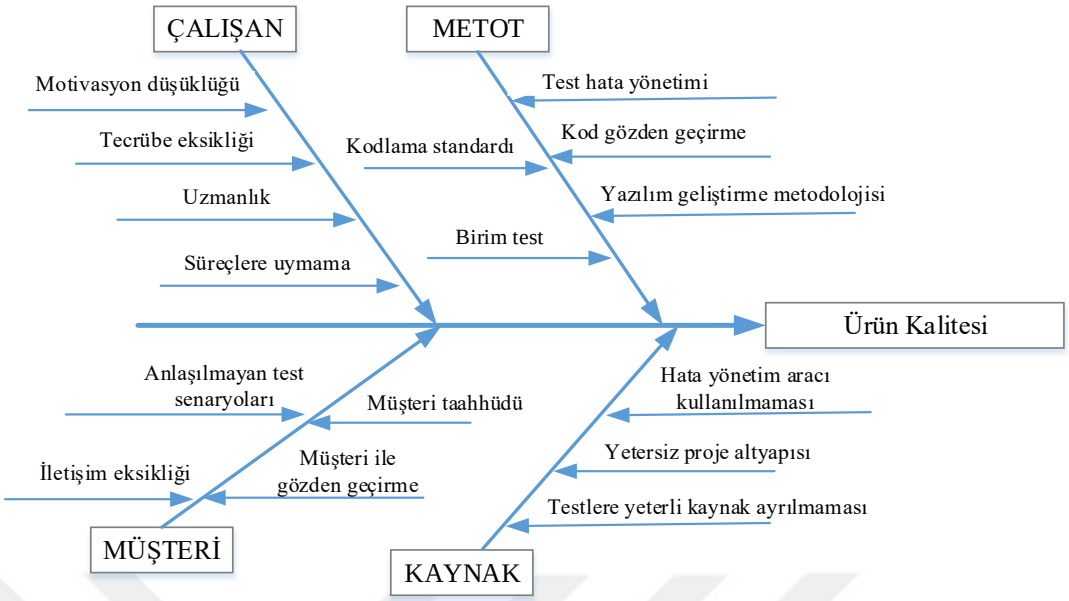


Şekil 4.29. Geliştirme sürecine ait metriklerin neden sonuç diyagramı

Diğer taraftan ürün kalitesine yönelik neden-sonuç diyagramına göre; çalışanların motivasyon düşüklüğü, tecrübe eksikliği, proje alanında uzmanlık olmaması, çalışanın süreç adımlarına uymaması, müşteri ile iletişim eksikliği, müşteri ile ortak anlaşmaya varılamayan test senaryoları, test dokümanlarının müşteri ile beraber gözden geçirilmemesi, test dokümanları ve test senaryoları için müşteriden taahhüt/onay alınmaması, test hata yönetimi ve kod gözden geçirme yöntemlerinin tanımlı olmaması, yazılım kodlama standardının tanımlı olmaması, yazılım birim testlerinin gerçekleştirilmemesi, hata yönetimi ile ilgili araç kullanılmaması, projenin geliştirme ve test altyapısının yetersiz olması, projede testlere yeterli zaman ve kaynak ayrılmaması ve projeye uygun olmayan yazılım geliştirme metodolojisi seçimi geliştirme sürecine ait ürün kalitesi probleminin sebepleri olarak sayılabilmektedir.

Şekil 4.30'daki neden sonuç diyagramı incelendiğinde ürün kalitesini en çok etkileyen alt nedenlerin başında testlere yeterli kaynak ayrılmaması gelmektedir. Projeye test mühendisi atanmaması ve testlerin geliştiriciler tarafından gerçekleştirilmesi test sürecinin verimsiz işletilmesine neden olmaktadır. Bu durum geliştirilen yazılım ürününde daha fazla hatanın gözden kaçması ile sonuçlanmaktadır.

Ayrıca projeye atanan test mühendisinin tecrübesiz olması ürün kalitesini en çok etkileyen bir diğer neden olarak öne çıkmaktadır.



Şekil 4.30. Geliştirme süreci hata yoğunluğu metriği neden sonuç diyagramı

Müşteri memnuniyetsizliğine ve ürün kalite eksikliğine yol açan tüm bu nedenler, Ar-Ge kurumunda çalışan uzmanlardan oluşan bir ekip ile beyin fırtınası tekniği kullanılarak belirlenmiştir. Buna göre; hata nedenleri, kutucuklar içerisinde gösterilirken bu nedenlere sebebiyet alt nedenler oklarla belirtilmiştir.

Şekil 4.29'da verilen neden-sonuç diyagramı incelendiğinde; müşteri memnuniyeti eksikliği problemine genelde metot ve kaynak kökenli sebeplerin yol açtığı gözlenmiştir. Yanlış proje yaşam döngüsü seçimi, geliştirilecek sistemin/yazılım operasyonel senaryolarının tanımlı olmaması, projenin geliştirme ve test altyapısının yetersiz olması, test senaryolarının nasıl iyi derecede tanımlanacağı ve gözden geçirileceği konusundaki yöntem eksikliği ve test hatalarının yönetildiği araç kullanımındaki eksiklikler ile projeye malzeme ve teçhizat gibi yeterli kaynak ayrılmaması müşteri memnuniyetsizliğindeki temel hata nedenleri olduğu belirlenmiştir. Müşteri memnuniyetini artırmak için bu nedenlerin ortadan kaldırılması gerektiği belirlenmiştir.

Şekil 4.30'da verilen neden-sonuç diyagramı incelendiğinde; ürün kalite eksikliği problemine genelde metot ve kaynak kökenli sebeplerin yol açtığı gözlenmiştir. Test senaryolarının nasıl iyi derecede tanımlanacağı ve gözden geçirileceği konusundaki yöntem eksikliği ve test hatalarının yönetildiği araç kullanımındaki eksiklikler, kodlama standardının olmaması, yazılım birim testlerinin gerçekleştirilmemesi, test

faaliyetlerine yeterli kaynak ve süre ayrılmaması ile projeye uygun olmayan yazılım geliştirme metodolojisi seçimi ürün kalite sorunundaki temel hata nedenleri olduğu belirlenmiştir. Müşteri memnuniyetini artırmak için bu nedenlerin ortadan kaldırılması gerektiği belirlenmiştir.

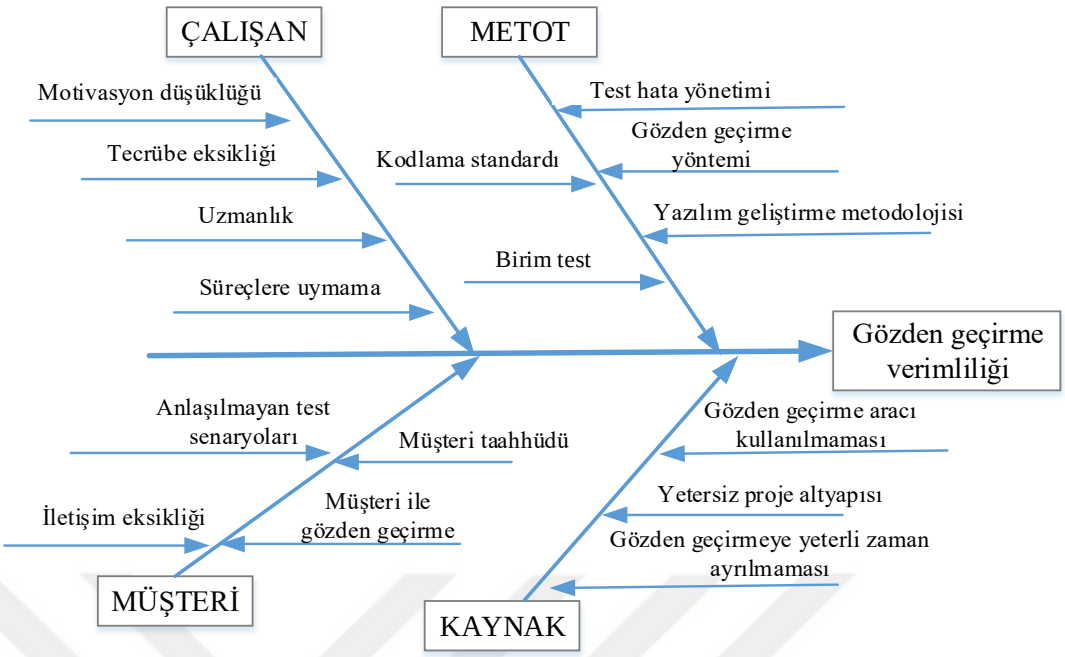
4.4.4. Doğrulama ve geçerleme süreci neden-sonuç diyagramı

Doğrulama ve geçerleme sürecine ait alt kriterler gözden geçirme verimliliği ile ilgili metrikler ve ürün kalitesine etki eden hata yakalama ve tespit yüzdeleridir. Hata yakalama ve tespit yüzdeleri müşteriye giden ürünün mümkün olduğunca az hata ile gitmesini hedefler. Ürün kalitesi ile ilgili neden sonuç diyagramı Şekil 4.30'da verilmiştir.

Doğrulama ve geçerleme sürecindeki diğer problem gözden geçirme verimliliği ile ilgilidir. Gözden geçirme ile ilgili hazırlık verimliliği, gözden geçirme hızı ve bulgu yoğunluğu gözden geçirmeye yeterli zamanın ayrılmasını sağlamaya çalışır. Gözden geçirme verimliliği ile ilgili neden sonuç diyagramı Şekil 4.31'de verilmiştir. Buna göre metot, çalışan, kaynak ve müşteri ana nedenler olarak gözlemlenmiştir. Bu ana nedenler kendi içerisinde alt nedenlere ayrıştırılarak gerekli önlemlerin alınması yönünde nereden başlanılacağı konusunda doğru karar vermek amaçlanmıştır.

Neden-sonuç diyagramı uygulanarak gözden geçirme verimliliğinden sapmaya sebep olabilecek ana nedenler ve hataya sebep olabilecek alt nedenler önem sırasına göre ayrılmıştır.

Buna göre; çalışanların motivasyon düşüklüğü, tecrübe eksikliği, proje alanında uzmanlık olmaması, çalışanların süreçlere uymaması, müşteri ile iletişim eksikliği, müşteri ile ortak anlaşmaya varılamayan test senaryoları, test dokümanlarının müşteri ile beraber gözden geçirilmemesi, test dokümanları ve test senaryoları için müşteriden taahhüt/onay alınmaması, test hata yönetimi ve gözden geçirme yöntemlerinin tanımlı olmaması, kodlama standardının olmaması, birim testlerin gerçekleştirilmemesi, gözden geçirme ile ilgili araç kullanılmaması, projenin geliştirme ve test altyapısının yetersiz olması, projede gözden geçirme faaliyetlerine yeterli zaman ayrılmaması ve projeye uygun olmayan yazılım geliştirme metodolojisi seçimi gözden geçirme verimliliğini düşüren sebepler olarak sayılabilmektedir.



Şekil 4.31. Doğrulama ve geçerleme süreci metrikleri neden sonuç diyagramı

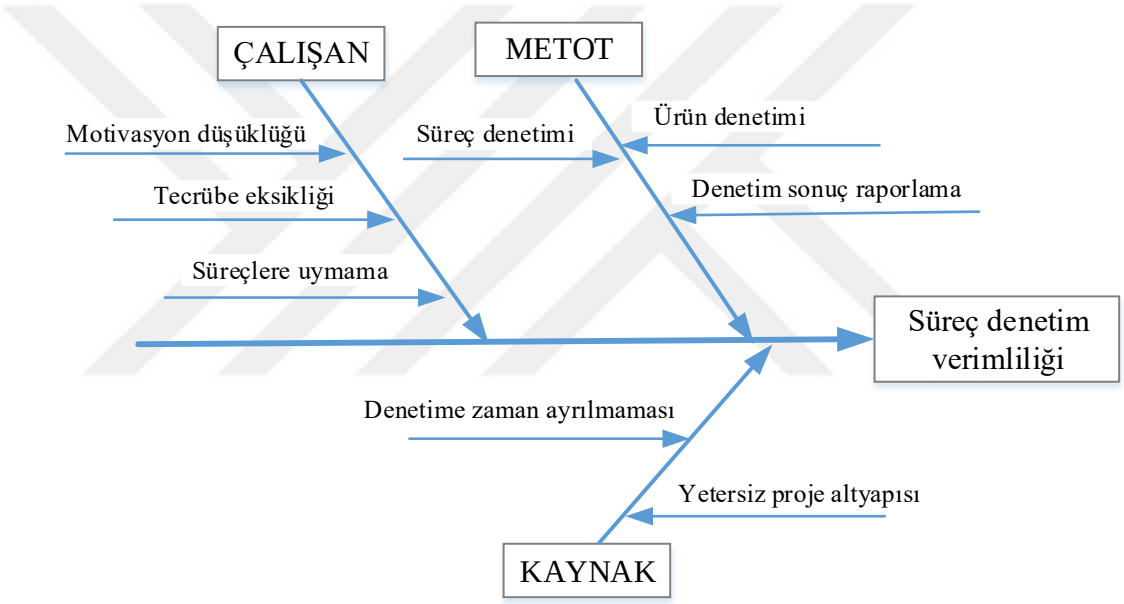
Gözden geçirme verimliliğini düşüren tüm bu nedenler, Ar-Ge kurumunda çalışan uzmanlardan oluşan bir ekip ile beyin fırtınası tekniği kullanılarak belirlenmiştir. Buna göre; hata nedenleri, kutucuklar içerisinde gösterilirken bu nedenlere sebebiyet alt nedenler oklarla belirtilmiştir.

Şekil 4.31'de verilen neden-sonuç diyagramı incelendiğinde; gözden geçirme verimliliği problemine genelde metot ve kaynak kökenli sebeplerin yol açtığı gözlenmiştir. Test hata yönetimi ve gözden geçirme ile ilgili yöntem eksikliği ve gözden geçirmelerin yönetildiği araç kullanımındaki eksiklikler, kodlama standardının olmaması, yazılım birim testlerinin gerçekleştirilmemesi, gözden geçirme faaliyetlerine yeterli kaynak ve süre ayrılmaması ile projeye uygun olmayan yazılım geliştirme metodolojisi seçiminin gözden geçirme verimliliğindeki temel hata nedenleri olduğu belirlenmiştir. Gözden geçirme verimliliğini artırmak için bu nedenlerin ortadan kaldırılması gerektiği belirlenmiştir.

4.4.5. Kalite güvence süreci neden-sonuç diyagramı

Kalite güvence sürecine ait alt kriterler süreç denetiminde ortaya çıkan uygunsuzluklarının ortalama kapanma süresi ve denetim planından sapma değeridir. Bu kriterler süreç denetim verimliliğini artırmaya yöneliktir. Süreç denetim verimliliği ile ilgili neden sonuç diyagramı Şekil 4.32'de verilmiştir. Buna göre metot, çalışan, ve

kaynak ana nedenler olarak gözlemlenmiştir. Bu ana nedenler kendi içerisinde alt nedenlere ayrıştırılarak gerekli önlemlerin alınması yönünde nereden başlanılacağı konusunda doğru karar vermek amaçlanmıştır. Neden-sonuç diyagramı uygulanarak süreç denetim verimliliğinden sapmaya sebep olabilecek ana nedenler ve hataya sebep olabilecek alt nedenler önem sırasına göre ayrılmıştır. Buna göre; çalışanların motivasyon düşüklüğü, tecrübe eksikliği, çalışanların süreçlere uymaması, süreç ve ürün denetimi ile ilgili yöntemlerin olmaması, denetim sonuçlarının raporlanması ve düzeltici faaliyetler konusundaki sorunlar, denetimlere yeterli zaman ayrılmaması ve yetersiz proje altyapısı süreç denetim verimliliğini düşüren sebepler olarak sayılabilmektedir.



Şekil 4.32. Kalite güvence süreci metrikleri neden sonuç diyagramı

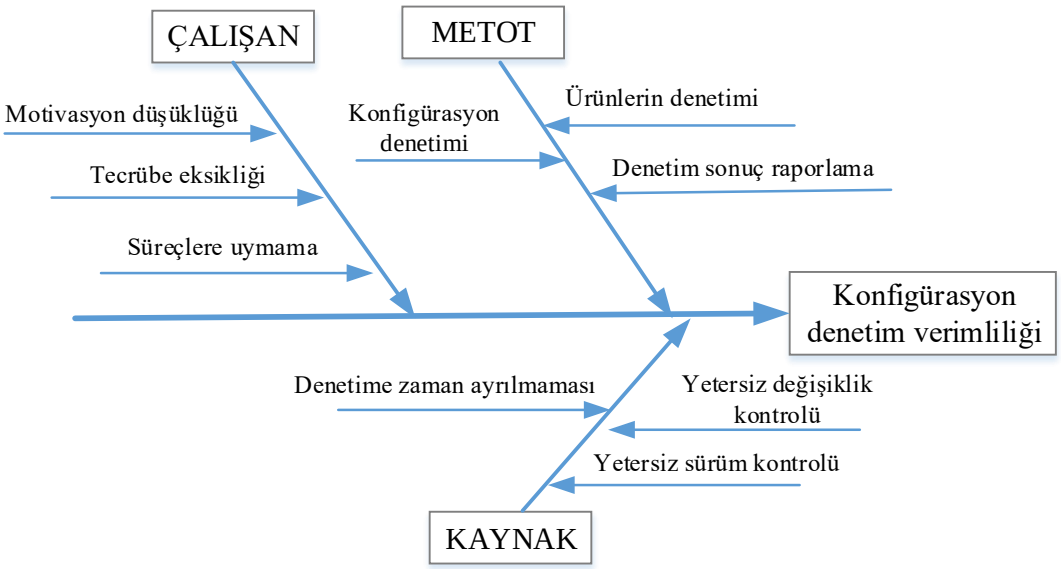
Süreç denetim verimliliğini düşüren tüm bu nedenler, Ar-Ge kurumunda çalışan uzmanlardan oluşan bir ekip ile beyin fırtınası tekniği kullanılarak belirlenmiştir. Buna göre; hata nedenleri, kutucuklar içerisinde gösterilirken bu nedenlere sebebiyet alt nedenler oklarla belirtilmiştir.

Şekil 4.32'de verilen neden-sonuç diyagramı incelendiğinde; Süreç denetim verimliliğini düşmesine genelde çalışan ve kaynak kökenli sebeplerin yol açtığı gözlenmiştir. Denetimlere yeterli zaman ayrılmaması ve çalışanların süreçlere uymaması sapmanın temel hata nedenleri olduğu belirlenmiştir. Bu sapmanın

düzeltilmesi veya yeniden meydana gelmesinin önlenebilmesi için bu nedenlerin ortadan kaldırılması gerektiği belirlenmiştir.

4.4.6. Konfigürasyon yönetimi süreci neden-sonuç diyagramı

Konfigürasyon yönetimi sürecine ait alt kriterler konfigürasyon denetiminde ortaya çıkan uygunsuzluklarının ortalama kapanma süresi ile oranı, denetim planından sapma değeri ve konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısıdır. Bu kriterler konfigürasyon denetim verimliliğini artırmaya yöneliktir. konfigürasyon denetim verimliliği ile ilgili neden sonuç diyagramı Şekil 4.33'te verilmiştir. Buna göre metot, çalışan, ve kaynak ana nedenler olarak gözlemlenmiştir. Bu ana nedenler kendi içerisinde alt nedenlere ayrıştırılarak gerekli önlemlerin alınması yönünde nereden başlanılacağı konusunda doğru karar vermek amaçlanmıştır. Neden-sonuç diyagramı uygulanarak konfigürasyon denetim verimliliğinden sapmaya sebep olabilecek ana nedenler ve hataya sebep olabilecek alt nedenler önem sırasına göre ayrılmıştır. Buna göre; çalışanların motivasyon düşüklüğü, tecrübe eksikliği, çalışanların süreçlere uymaması, konfigürasyon ve ürün denetimi ile ilgili yöntemlerin olmaması, denetim sonuçlarının raporlanması ve düzeltici faaliyetler konusundaki sorunlar, denetimlere yeterli zaman ayrılmaması, yetersiz sürüm kontrolü ve yetersiz proje altyapısı konfigürasyon denetim verimliliğini düşüren sebepler olarak sayılabilmektedir.



Şekil 4.33. Konfigürasyon yönetimi süreci metrikleri neden sonuç diyagramı

Konfigürasyon denetim verimliliğini düşüren tüm bu nedenler, Ar-Ge kurumunda çalışan uzmanlardan oluşan bir ekip ile beyin fırtınası tekniği kullanılarak belirlenmiştir. Buna göre; hata nedenleri, kutucuklar içerisinde gösterilirken bu nedenlere sebebiyet alt nedenler oklarla belirtilmiştir.

Şekil 4.33'te verilen neden-sonuç diyagramı incelendiğinde; konfigürasyon denetim verimliliğini düşmesine genelde çalışan ve kaynak kökenli sebeplerin yol açtığı gözlenmiştir. Denetimlere yeterli zaman ayrılmaması ve çalışanların süreçlere uymaması sapmanın temel hata nedenleri olduğu belirlenmiştir. Bu sapmanın düzeltilmesi veya yeniden meydana gelmesinin önlenebilmesi için bu nedenlerin ortadan kaldırılması gerektiği belirlenmiştir.

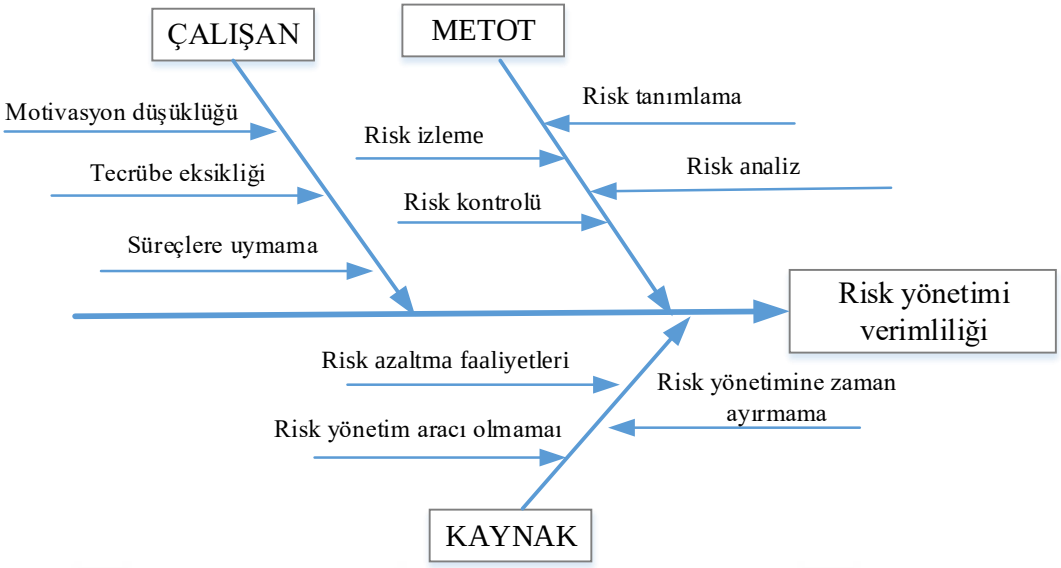
4.4.7. Risk yönetimi süreci neden-sonuç diyagramı

Konfigürasyon yönetimi sürecine ait alt kriterler konfigürasyon denetiminde ortaya çıkan uygunsuzluklarının ortalama kapanma süresi ile oranı, denetim planından sapma değeri ve konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısıdır.

Bu kriterler konfigürasyon denetim verimliliğini artırmaya yöneliktir. konfigürasyon denetim verimliliği ile ilgili neden sonuç diyagramı Şekil 4.34'te verilmiştir. Buna göre metot, çalışan, ve kaynak ana nedenler olarak gözlemlenmiştir. Bu ana nedenler kendi içerisinde alt nedenlere ayrıştırılarak gerekli önlemlerin alınması yönünde nereden başlanılacağı konusunda doğru karar vermek amaçlanmıştır.

Neden-sonuç diyagramı uygulanarak konfigürasyon denetim verimliliğinden sapmaya sebep olabilecek ana nedenler ve hataya sebep olabilecek alt nedenler önem sırasına göre ayrılmıştır. Buna göre; çalışanların motivasyon düşüklüğü, tecrübe eksikliği, çalışanların süreçlere uymaması, konfigürasyon ve ürün denetimi ile ilgili yöntemlerin olmaması, denetim sonuçlarının raporlanması ve düzeltici faaliyetler konusundaki sorunlar, denetimlere yeterli zaman ayrılmaması, yetersiz sürüm kontrolü ve yetersiz proje altyapısı konfigürasyon denetim verimliliğini düşüren sebepler olarak sayılabilmektedir.

Şekil 4.34 incelendiğinde risk yönetimi faaliyetine yeterli zaman ayrılmaması risk yönetim verimliliğini olumsuz etkileyen başlıca neden olarak öne çıkmaktadır.



Şekil 4.34. Risk yönetimi süreci metrikleri neden sonuç diyagramı

Risk yönetimi verimliliğini düşüren tüm bu nedenler, Ar-Ge kurumunda çalışan uzmanlardan oluşan bir ekip ile beyin fırtınası tekniği kullanılarak belirlenmiştir. Buna göre; hata nedenleri, kutucuklar içerisinde gösterilirken bu nedenlere sebebiyet alt nedenler oklarla belirtilmiştir.

Şekil 4.34'te verilen neden-sonuç diyagramı incelendiğinde; risk yönetimi verimliliğini düşmesine genelde kaynak ve metot kökenli sebeplerin yol açtığı gözlenmiştir. Denetimlere yeterli zaman ayrılmaması ve risk yönetimi faaliyetleri ile ilgili yöntemlerin olmaması sapmanın temel hata nedenleri olduğu belirlenmiştir. Bu sapmanın düzeltilmesi veya yeniden meydana gelmesinin önlenebilmesi için bu nedenlerin ortadan kaldırılması gerektiği belirlenmiştir.

4.5. Örnek Bir Projenin Proje Başarı Puanının Hesaplanması

Model bir Ar-Ge kurumunda kabul test aşamasında bulunan gerçek bir proje verisi ile işletilmiştir. Yazılım yaşam döngüsü aşamaları olan proje planlama, analiz, uygulama ve test aşamalarında proje performans değerlendirmesi yapılmıştır.

Her değerlendirmenin bir sonucu olarak, sayısal bir proje başarı puanı oluşturulmuştur. Projeden kabul testi aşamasında toplanan süreç metrik ölçüm sonuçları Tablo 4.38'de verilmiştir.

Tablo 4.38. Kabul aşamasındaki bir projenin ölçülen metrik değerleri

Metrik adı	Ölçülen değer
Kilometre taşı sapması (%)	14,5
Efor sapması (%)	13,2
Gelir/gider endeksi	0.83
Proje uygunsuzluklarının ortalama açık kalma süresi (gün)	11
Süreç denetim planlamasından sapma (gün)	0
Müşteri tarafından raporlanan hata açık kalma süresi (gün)	24
Müşteri tarafından raporlanan iyileştirme açık kalma süresi (gün)	22
Müşteri tarafından raporlanan gereksinim değişikliği açık kalma süresi (gün)	18
Müşteri tarafından raporlanan hata açık kalma oranı (%)	8,3
Müşteri tarafından raporlanan iyileştirme açık kalma oranı (%)	7,2
Müşteri tarafından raporlanan gereksinim değişikliği açık kalma oranı (%)	8.9
Hata yoğunluğu (#/KLOC)	1.1
Konfigürasyon denetim planlamasından sapma (gün)	7
Konfigürasyon denetimi başına düşen ortalama uygunsuzluk sayısı (adet)	2
Konfigürasyon Uygunsuzluklarının Açık Kalma Süresi (gün)	15
Konfigürasyon Uygunsuzluklarının Açık Kalma Oranı (%)	6
Risk gözden geçirme oranı (%)	100
Proje Başına Ortalama Gereksinim Değişikliği Sayısı (adet)	14
Gözden geçirme hazırlık hızı (büyüklük/saat)	80
Bulgu yoğunluğu (adet/büyüklük)	0,76
Gözden geçirme hazırlık verimliliği (adet/saat)	0,23
Hata yakalama yüzdesi (%)	60
Hata tespit yüzdesi (%)	70

Planlama aşaması sırasında modele girilecek metrikler Tablo 3.1'de verilmiştir. Proje planlama aşamasında ölçülen ilgili süreç metrikleri üst ve alt kontrol sınırları içindeyse, metriklerin toplam ağırlığı 0.355'tir. Kontrol dışı olan süreç metriklerinin ağırlık değerinin sıfır olduğu varsayılır. Metriklerin analiz aşaması toplam ağırlığı 0.630'dur. Projenin uygulanması ve test aşamasında performans değerlendirmesi yapılırken, yazılım yaşam döngüsü süreç ölçümlerinin tamamı modele girdi niteliğindedir. Yani her iki aşama için de toplam metrik ağırlığı 1'e eşittir.

Gerçek proje ölçüm sonuçlarında, projenin kilometre taşı sapması, efor sapması ve gelir / gider endeksi değerlerinin ilgili kontrol grafiklerinin sınırları dışında olduğu bulunmuştur. Diğer süreç metrik değerleri kontrol sınırları içinde bulunmuştur. Bu

durumda, kontrol sınırları dâhilindeki metriklerin toplam ağırlık değerleri 0,236 olarak hesaplanmıştır. Projenin beklenen toplam metrik ağırlığı 0,355'dir. Bu iki değerin birbirine oranı ve 100 ile çarpımı, projenin planlama aşaması için başarı puanı verir.

Planlama aşaması başarı puanı = (0,236/0,355)*100 = 66

Planlama aşaması sonu itibarıyla proje beklenen başarının % 66'sını karşılamaktadır.

Süreç metrik değerlerinin bu projede hedeflenen sınırlardan sapmasının nedenleri neden-sonuç diyagramı ile analiz edilmiş ve kök nedenleri bulunmuştur. Analiz sonucunda, projedeki teknik zorluklar nedeniyle, projenin kilometre taşı bitiş süresinin uzadığı ve bu durumda efor değerinin arttığı sonucuna varılmıştır. Analiz sonucunda düzeltici faaliyet olarak, deneyimli bir çalışanı projeye atanmıştır. Uygulama yapılan projenin diğer aşamalarındaki değerlendirme sonuçları Tablo 4.39'da verilmiştir.

Tablo 4.39. Başarı puanı hesaplama sonuçları

Proje aşaması	Modele girecek metrikler	Hesaplanan proje başarı puanı
Planlama	PY 1.1,PY 1.2, PY1.3,KG 1.1, KG 1.2, DG 1.1, DG 1.2, DG 1.3, GY 1.1	66
Analiz	PY 1.1,PY 1.2, PY 1.3,GY 1.1,GEL 1.3, GEL 1.6, KG 1.1, KG 1.2, DG 1.1, DG 1.2, DG 1.3	81
Gerçekleştirme	Tüm süreç metrik kriterleri	88
Test	Tüm süreç metrik kriterleri	88

4.6. Hata Kapatma Süresi Tahmini

Bu bölümde bir Ar-Ge kurumunda gerçekleştirilen yazılım projelerindeki testlerde bulunan hatalar toplanmış ve hataların çözüm süresinin tahminlenmesi ile ilgili bir analiz gerçekleştirilmiştir. Hata çözüm süresi tahmini için çoklu regresyon analizi kullanılmıştır. Öncelikle çoklu regresyon hakkında kısa bilgi verilip ardından gerçekleştirilen çalışma ve yorumlar verilecektir.

Regresyon analizi, değişkenler arasındaki ilişkiyi araştırmak ve modellemek için kullanılan istatistiksel bir tekniktir. Regresyon uygulamaları çoktur ve mühendislik, fizik ve kimya bilimleri, ekonomi, yönetim, yaşam ve biyolojik bilimler ve sosyal bilimler dâhil olmak üzere hemen hemen her alanda kullanılır [96].

4.6.1. Basit doğrusal regresyon ve korelasyon

İstatistikte, modeller değişkenler arasındaki ilişkileri ölçer ve tahmin yapmayı sağlar. Basit bir doğrusal regresyon en temel modeldir. Modelde sadece iki değişken vardır ve bir hata terimiyle doğrusal bir ilişki olarak modellenmiştir:

$$Y_t = \alpha + \beta X_t + u_t \quad (4.1)$$

Denklem (4.1)'de kesişme α ve eğim β bilinmeyen sabitlerdir ve u rastgele bir hata bileşenidir. Hataların ortalamasının sıfır ve varyansının σ^2 olduğu varsayılmaktadır. Ayrıca, genellikle hataların ilişkisiz olduğunu varsayılır. Bu, bir hatanın değerinin başka bir hatanın değerine bağlı olmadığı anlamına gelir. Regresyonun amacı Y'yi X temelinde tahmin etmek veya Y'nin X'e nasıl bağlı olduğunu tanımlamaktır.

Bu modelin α ve β parametrelerini (regresyon katsayıları) bulmak için "X" serbest değişkeni ve "Y" bağlı değişkeniyle ilgili gözlemlere gerek duyulur. Bu gözlemlerde değişkenler "eşit aralıklı veya oranlı ölçme düzeyi"nde ölçülür [97].

"X" ve "Y" değişkenlerinin anakütlelerini oluşturan ve bu değişkenler için bütün değerlere sahip olunması uygulamada olanaksız olduğundan, söz konusu değişkenler için örneklemeye başvurulur. Böylece "α ve β parametrelerinin" tahminleri olan "a ve b istatistikleri" bulunabilir. Anakütle için birer "α ve β parametresi" varken, bu anakütleden çekilen her bir örneklem için ayrı birer "a ve b istatistiği" elde edilmektedir. Bu "a ve b istatistikleri" normal dağılıma sahip olup, beklenen değerleri sırasıyla "α ve β"dır. Uygulamada tek bir örneklem alınmakta ve bu örneklem yardımıyla anakütle parametreleri tahmin edilmektedir.

Y bağlı değişkeni ile X serbest değişkeni arasındaki gerçek ilişki Denklem (4.2) de verilmiştir. Bu denklemden yola çıkarak gerçek regresyon doğrusu (anakütle regresyon doğrusu):

$$E(Y_t) = \alpha + \beta X_t \quad (4.2)$$

Y ile X arasındaki tahmin edilen ilişki:

$$Y_t = a + bX_t + e_t \quad (4.3)$$

Şeklinde gösterilmektedir. Tahmin edilen regresyon doğrusu (örneklem regresyon doğrusu):

$$\widehat{Y}_t = a + bX_t \quad (4.4)$$

Yukarıdaki eşitliklerde;

Y_t: Y değişkeninin gözlemlenen değerini,

$\widehat{Y}_t$: X değişkeninin belli bir değeri var iken Y değişkeninin tahmin edilen değerini,

a: α parametresinin tahminini,

b: β parametresinin tahminini,

e_t: u_t hata teriminin tahminini

simgelemektedir.

"a" regresyon doğrusunun "Y" eksenini kestiği noktadır. Öte yandan "b" regresyon doğrusunun eğimini, yani "X"in "1 birim" değişmesi durumunda "Y"nin hangi yönde ve ne kadar değişeceğini ifade eder.

"α ve β parametreleri"nin tahminleri "a ve b istatistikleri"nin hesaplanmasında en küçük kareler tekniğinden yararlanılabilir. En küçük kareler tekniğinin esası "a ve b"yi hata terimleri tahminlerinin kareleri toplamını minimum olacak şekilde seçmektir.

$$\sum e_t^2 = \sum (Y_t - \widehat{Y}_t)^2 = \min \quad (4.5)$$

Denklem (4.5)'te belirtilen durum yapılabildiğinde verilere en iyi uyan bir doğru elde edilmiş olacaktır. Basit doğrusal regresyon modelindeki "α ve β parametreleri"nin tahminleri olan "a ve b istatistikleri"nin bulunmasında yararlanılacak denklemler şunlardır:

$$\sum Y_t = na + b \sum X_t \quad (4.6)$$

$$\sum X_t Y_t = a \sum X_t + b \sum X_t^2 \quad (4.7)$$

Denklem (4.6) ve Denklem (4.7)'deki "X_t ve Y_t" değerleri sıfır orijinine göre ifade edilmiştir.

İki değişkenin doğrusal birlikte değişmesinin derecesi "r" simgesi ile gösterilen "korelasyon katsayısı" ile ölçülür. Korelasyon katsayısı, iki değişkenin birlikte değişimlerinde ne dereceye kadar uygun olduğunu belirler, fakat hiçbir şekilde neden-sonuç ilişkisi kurmaz. Değişkenler arasında neden-sonuç ilişkisi bulunduğunda ise korelasyon katsayısı yüksek çıkmaktadır. "Değişkenlerin ölçü birimine bağlı olmayan" ve ilişkinin yönünü de belirten korelasyon katsayısı "-1'den küçük" olamayacağı gibi "+1!den büyük" de çıkamaz. Bu durum aşağıdaki formül ile ifade edilir;

$$-1 \leq r \leq +1 \tag{4.8}$$

Pozitif işaretli korelasyon katsayısı, değişkenlerden birinin değeri artarken (azalırken) diğerinin değerinin de arttığını (azaldığını), negatif işaretli korelasyon katsayısı değişkenlerden birinin değeri artarken (azalırken) diğerinin değerinin de azaldığını (arttığını) gösterir. "r=0" olduğunda değişkenler arasında doğrusal ilişki bulunmadığı söylenir. "r=+1" olması pozitif tam doğrusal ilişki, "r=-1" olması negatif tam doğrusal ilişki varlığını ortaya koyar. Uygulamada korelasyon katsayısının "+1" veya "-1"e eşit çıkması beklenmez. Değişkenler arasındaki ilişki kuvvetlendikçe "± 1"e, aksine zayıfladıkça "0"a yaklaşan korelasyon katsayısı elde edilmektedir. Korelasyon katsayısı aşağıdaki formülle hesaplanır:

$$r = \frac{\sum x_t y_t}{\sqrt{\sum x_t^2 . \sum y_t^2}} \tag{4.9}$$

Korelasyon katsayısının karesi "belirlilik katsayısı"dır. Bu katsayı bağlı değişkendeki değişmelerin yüzde kaçının serbest değişkendeki değişmelerle açıklanabileceğini gösterir. "0" ile "± 1" arasındaki hangi "r" değerlerinin istatistiksel açıdan anlam taşıdığını belirlemek için "T ve F testleri" gerçekleştirilir.

"T" testine başvurulduğunda "n" örneklem hacmi olmak üzere;

$$T = \left| \frac{r.\sqrt{n-2}}{\sqrt{1-r^2}} \right| \tag{4.10}$$

formülüne göre hesaplanan test istatistiğine başvurulur. Anakütlenin korelasyon katsayısını (“n” çok büyük olduğunda “r”nin yaklaşacağı değeri) “ρ” simgesi ile gösterdiğimizde; örneklem korelasyon katsayılarının dağılımı “$\rho = 0$” sıfır etrafında simetrik olup “T dağılımına” uyar. “T testi”nde test istatistiği belli bir anlamlılık düzeyine ve “v=n-k” serbestlik derecesine göre Ek-B’deki “kritik T değerleri tablosu”nda bulunan kritik değer (T_K) ile kıyaslanır. Hesaplanan değer kritik “T” değerinden büyük çıktığında söz konusu korelasyon katsayısının göz ardı edilemeyecek kadar önemli bir bağlantıyı ifade ettiği yorumu yapılır.

“T ve F dağılımları” arasındaki ilişkiden yararlanarak, tahmin edilen parametre sayısı “2” ve dolayısıyla “$v_1 = k - 1$”, yanı “$v_1 = 1$” durumunda “$F = T^2$” olduğundan “T testi” yerine “F testi”ne başvurmak olanaklıdır. Test istatistiği formülü aşağıdaki gibidir:

$$F = \frac{r^2/(k-1)}{(1-r^2)/(n-k)} \tag{4.11}$$

“n” örneklem hacmini, “k” tahmin edilen parametre sayısını ifade etmektedir. Denklem (4.8)’e göre hesaplanan test istatistiği belli bir anlamlılık düzeyine, ayrıca “$v_1 = k - 1$” ve “$v_2 = n - k$”serbestlik derecelerine göre Ek-C’deki “kritik F değerleri tablosu”nda bulunan kritik değer (F_K)ile kıyaslanır. Hesaplanan değer kritik “F” değerinden büyük çıktığında söz konusu korelasyon katsayısının göz ardı edilemeyecek kadar önemli bir bağlantıyı ifade ettiği yorumu yapılır.

4.6.2. Çoklu doğrusal regresyon ve korelasyon

Basit doğrusal regresyonda bir serbest değişken ve bir de bağlı değişken modele dahil edilmektedir. Bağlı değişkenin birden fazla serbest değişken tarafından etkilendiği durumlarda çoklu doğrusal regresyon modellemesi yapılır. Çoklu doğrusal regresyon modelinin temel varsayımları, hata teriminin sıfır ortalama ve sabit varyans ile normal dağılım göstermesi, hata terimleri arasında otokorelasyonun olmaması, hata terimi ile bağımsız değişkenler arasında bir ilişkinin olmaması, bağımsız değişkenler arasında çoklu doğrusal bağlantı durumunun olmaması şeklinde ifade edilebilir [98].

Serbest değişken sayısının iki olduğu farz edildiğinde, “Y bağlı değişkeni ile X ve Z serbest değişkenleri arasındaki gerçek ilişki”:

$$Y_t = \alpha + \beta X_t + \gamma Z_t + u_t \quad (4.12)$$

iken, "gerçek regresyon doğrusu (anakütle regresyon doğrusu)":

$$E\,(Y_t) = \alpha + \beta X_t + \gamma Z_t \quad (4.13)$$

şeklindedir. Öte yandan "Y ile X ve Z arasındaki tahmin edilen ilişki":

$$Y_t = a + bX_t + cZ_t + e_t \quad (4.14)$$

şeklinde gösterilmektedir. Son olarak "tahmin edilen regresyon doğrusu (örneklem regresyon doğrusu)" aşağıdaki gibidir:

$$\widehat{Y}_t = a + bX_t + cZ_t \quad (4.15)$$

Yukarıdaki eşitliklerde;

Y_t: Y değişkeninin gözlemlenen değerini,

$\widehat{Y}_t$: X ve Z değişkenlerinin belli bir değeri var iken Y değişkeninin tahmin edilen değerini,

a: α parametresinin tahminini,

b: β parametresinin tahminini,

c: γ parametresinin tahminini,

e_t: u_t hata teriminin tahminini

simgelemektedir.

Çoklu doğrusal regresyon modelindeki "α, β ve γ parametreleri"nin tahminleri olan "a, b ve istatistikleri" en küçük kareler tekniği ile bulunabilir. Bu tahminde yararlanılabilecek denklemler aşağıdaki gibidir:

$$\sum Y_t = na + b\sum X_t + c\sum Z_t \quad (4.16)$$

$$\sum X_t Y_t = a\sum X_t + b\sum X_t^2 + c\sum X_t Z_t \quad (4.17)$$

$$\sum Z_t Y_t = a \sum Z_t + b \sum X_t Z_t + c \sum Z_t^2 \quad (4.17)$$

Denklem (4.16), Denklem (4.17) ve Denklem (4.18)'deki "X_t, Y_t ve Z_t" değerleri sıfır orijinine göre ifade edilmiştir.

Birden fazla serbest değişkenin varlığı durumunda, bağlı değişken ile serbest değişkenler arasında ne derece sıkı bir ilişki olduğunu açıklamak üzere çoklu korelasyon katsayısı (R) hesaplanabilir. Bunun karesi olan çoklu belirlilik katsayısı yorum yapma açısından daha elverişli olduğundan tercih edilir. Bağlı değişkendeki değişmelerin yüzde kaçının serbest değişkenlerdeki değişmelerle açıklanabileceğini gösteren çoklu belirlilik katsayısı aşağıdaki genel formül ile hesaplanır:

$$R^2 = \frac{b \sum x_t y_t + c \sum z_t y_t + d \sum v_t y_t + \cdots}{\sum y_t^2} \quad (4.18)$$

Modele yeni bir serbest değişken eklendiğinde çoklu belirlilik katsayısı genellikle artar, fakat hiçbir zaman azalmaz. Çoklu belirlilik katsayısının yaklaşık 0,80 olması yeterli kabul edilmektedir [97].

Modelde yer alan serbest değişkenlerin sayısı artarken serbestlik derecesi azalır. Dolayısıyla, serbestlik derecesini de hesaba katmak amacıyla belirlilik katsayısını düzeltmek gereklidir. Bu formül aşağıdaki gibidir:

$$R_d^2 = 1 - (1 - R^2).\frac{n-1}{n-k} \quad (4.19)$$

Serbest değişken sayısının artması düzeltilmiş çoklu belirlilik katsayısını azaltıcı bir etkiye sahiptir. Bu nedenle, yeni bir serbest değişkenin eklenmesiyle regresyonun açıklayıcı gücünün artıp artmadığını ortaya koymada "R_d^2" tercih edilir. Eklenen yeni değişken R_d^2'nin artmasını sağlıyorsa modele dâhil edilir, aksi halde model dışı bırakılır.

Çoklu belirlilik katsayısının anlamlı olup olmadığı "F testi" ile incelenebilir. Örneklem hacmi "n" ve tahmin edilen parametre sayısı "k" olmak üzere;

$$F = \frac{R^2/(k-1)}{(1-R^2)/(n-k)} \quad (4.20)$$

Denklem (4.20)'deki formül ile hesaplanan test istatistiği, belli bir anlamlılık düzeyine, ayrıca "$v_1 = k - 1$" ve "$v_2 = n - k$" serbestlik derecelerine göre Ek-C'deki "kritik F değerleri tablosu"nda bulunan kritik değer (F_K) ile kıyaslanır. Hesaplanan değer kritik "F" değerinden büyük çıktığında söz konusu korelasyon katsayısının göz ardı edilemeyecek kadar önemli bir bağlantıyı ifade ettiği yorumu yapılır.

4.6.3. Grafik analizler, korelasyon ve regresyon analizleri

Bu bölümde bir Ar-Ge kurumundaki yazılım geliştirme projelerinden alınan test hata verileri ile çoklu regresyon analizi gerçekleştirilmiştir. Analiz sonucunda regresyon doğrusu bulunmuş ve sonuçlar yorumlanmıştır.

Analize girdi olan hata bilgileri Şekil 2.18. Hata bildirim formundan toplanmıştır. Analizde toplam 120 adet hata verisi kullanılmıştır. Kullanılan veriler Ek-D'de verilmiştir. Regresyon modelindeki bağılı ve bağımsız değişkenler Tablo 4.40'da verilmiştir.

Tablo 4.40. Regresyon modeli bağımlı ve bağımsız değişkenler

Bağımlı değişken	Bağımsız değişkenler							
$Y=f(x_1 \dots x_n)$ Hata kapanma süresi	X1 Öncelik	X2 İş paketi	X3 Hata derecesi	X4 Hatayı tespit eden	X5 Test tipi	X6 Test Seviyesi	X7 Test Yeri	X8 Oluşma sıklığı

"Hata kapanma süresi" Ek-D'de görünen "Kapanma tarihi" ile "Bildirim tarihi" arsındaki farkın gün cinsinden değeridir.

"Öncelik" bağımsız değişkeninin "Yüksek" ve "Orta" olmak üzere iki seviyesi bulunmaktadır.

"İş paketi" bağımsız değişkeni "İP1, İP2, İP3" olmak üzere üç farklı değer alabilmektedir.

"Hata derecesi" bağımsız değişkeni "Katastrofik, Kritik, Büyük, Küçük, Kozmetik" olmak üzere 5 farklı değer alabilmektedir.

“Hatayı tespit eden” bağımsız değişkeninin “Müşteri, Çalışan” olmak üzere iki farklı seviyesi vardır.

“Test tipi” bağımsız değişkeni “İşlevsel, Performans” olarak iki farklı değer alabilmektedir.

“Test seviyesi” bağımsız değişkeni “Sistem, Yazılım” olarak iki farklı değer alabilmektedir.

“Test yeri” bağımsız değişkeni “Müşteri ortamı, Proje test ortamı” olarak iki farklı değer alabilmektedir.

“Oluşma sıklığı” bağımsız değişkeni “Her zaman, Genellikle, Bazen” olarak üç farklı değer alabilmektedir.

Bağımsız değişkenlerin alabileceği değerlerin tablo şeklinde görünümü Tablo 4.41’de verilmiştir.

Tablo 4.41. Bağımsız değişkenler ve alabileceği değerler

Bağımsız değişken adı	Alabileceği değerler
Öncelik	{Yüksek, Orta}
İş paketi	{İP1, İP2, İP3}
Hata derecesi	{Katastrofik, Kritik, Büyük, Küçük, Kozmetik}
Hatayı tespit eden	{Müşteri, Çalışan}
Test tipi	{İşlevsel, Performans}
Test seviyesi	{Sistem, Yazılım}
Test yeri	{Müşteri ortamı, Proje test ortamı}
Oluşma sıklığı	{Her zaman, Genellikle, Bazen}

Analiz için IBM SPSS aracı kullanılmıştır. Kantitatif bir değişken kalitatif değişkenler ile tahmin edilmeye çalışıldığından dolayı SPSS aracında geçici değişkenler (dummy variables) yaratılmak zorundadır. Örneğin “hata derecesi” değişkeni için “katastrofik” adıyla bir geçici değişken oluşturulmuştur. Hata derecesi katastrofik olan satırlarda bu değer “1”, olmayanlarda “0” olarak gözükecektir. Aynı şekilde “kritik”, “büyük” ve “küçük” adıyla geçici değişkenler oluşturulmuştur. Bir hata verisinde hata derecesi hangi değere sahipse o değere ait oluşturulan geçici değişken “1”, diğer değişkenler

“0” olarak gözükür. Eğer hata derecesi için oluşturulan tüm geçici değişkenler “0” ise bu durumdaki satırlar “kozmetik” değişkenini temsil eder. SPSS aracı bu satırın “kozmetik” geçici değişkenini temsil ettiğini anlayacaktır.

Benzer şekilde örneğin öncelik değişkeni için “Orta” adıyla geçici değişken oluşturulmuştur. “Orta” geçici değişkeninin “0” değerini aldığı satırlar hatanın önceliğinin yüksek olduğu anlamına gelmektedir. SPSS aracında oluşturulan geçici değişkenler Tablo 4.42’de verilmiştir.

Tablo 4.42. SPSS aracında oluşturulan geçici değişkenler

Bağımsız değişken adı	SPSS aracında oluşturulan geçici değişkenler
Öncelik	{ Orta}
İş paketi	{İP1, İP2}
Hata derecesi	{Katastrofik, Kritik, Büyük, Küçük, }
Hatayı tespit eden	{ Çalışan}
Test tipi	{İşlevsel, }
Test seviyesi	{Sistem, }
Test yeri	{ Proje test ortamı}
Oluşma sıklığı	{Her zaman, Bazen}

Ek-D’deki hata verilerine ait betimleyici istatistikler Tablo 4.43’de verilmiştir. Tablodan da görüleceği üzere yapılan çalışmada eksik veri olmadığı tespit edilmiştir.

Tablo 4.43. Test hatalarına ait betimleyici istatistikler

	Mean	Std. Deviation	N
Süre	11,2174	5,81256	120
Orta	,53	,501	120
İP1	,52	,502	120
İP2	,30	,460	120
Küçük	,33	,470	120
Büyük	,22	,414	120
Katastrofik	,17	,374	120
Kritik	,18	,382	120
Çalısan	,81	,395	120
İşlevsel	,34	,476	120
Sistem	,33	,473	120
Bazen	,13	,332	120
HerZaman	,71	,456	120

Tablo 4.43.(Devam) Test hatalarına ait betimleyici istatistikler

	Mean	Std. Deviation	N
ProjeTestOrtamı	,59	,494	120

Bağımlı değişkenin bağımsız değişkenler tarafından ne kadarının açıklanabildiğini ifade eden Tablo 4.44'de gösterilmiştir. Tablo 4.44'de görüleceği üzere bağımsız değişkenlerin bağımlı değişken üzerindeki önemlerini belirlemek için yapılan regresyon analizi sonucunda, R^2 = 0,791 ve R_d^2 = 0,765 olarak gerçekleşmiştir. Bağımlı değişkenin %76,5'i oluşturulan bağımsız değişkenler tarafından açıklanabilmektedir.

Tablo 4.44. Regresyon analizi belirlilik katsayısı

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	,889[a]	,791	,765	2,81671	,791	30,827	13	106	,000

a. Predictors: (Constant), ProjeTestOrtamı, Orta, HerZaman, Büyük, İşlevsel, İP1, Çalısan, Sistem, Bazen, Kritik, İP2, Küçük, Katastrofik

b. Dependent Variable: Süre

Tablo 4.45'te verilen regresyon analizinde "Analysis of Variance (ANOVA)" başlığı altında, F=30,827 değeri için p değerinin 0,000 düzeyinde gerçekleşmesi de modelin anlamlı olduğunu göstermektedir.

ANOVA, regresyon modelinin geçerli olup olmadığını aşağıdaki hipotez testleri ile %95 anlamlılık düzeyinde test eder. P değerinin "0,000" olduğu için H_0 hipotezi reddedilir.

$H_0: p \geq 0{,}05$ $\beta_1 = \beta_2 = \cdots = \beta_{13} = 0$ modeldeki tüm katsayılar "0" dır ve model geçersizdir.

$H_1: p < 0{,}05$ regresyon modeli katsayılardan en az biri "0" dan farklıdır ve regresyon geçerlidir.

Tablo 4.45. Regresyon analizi ANOVA tablosu

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3179,525	13	244,579	30,827	,000[b]
	Residual	840,990	106	7,934		
	Total	4020,514	119			

a. Dependent Variable: Süre

b. Predictors: (Constant), ProjeTestOrtamı, Orta, HerZaman, Büyük, İşlevsel, İP1, Çalısan, Sistem, Bazen, Kritik, İP2, Küçük, Katastrofik

Hata kapatma süresi regresyon modeli detayları Tablo 4.46'te verilmiştir.

Tablo 4.46. Hata kapatma süresi regresyon analizi

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	Collinearity Statistics	
		B	Std. Error	Beta			Tolerance	VIF
1	(Constant)	17,960	1,926		9,325	,000		
	Orta	-,045	1,149	-,004	-,040	,969	,201	4,970
	İP1	,109	,719	,009	,152	,880	,511	1,955
	İP2	-,819	,828	-,065	-,989	,325	,460	2,176
	Küçük	-2,960	1,027	-,240	-2,882	,005	,286	3,501
	Büyük	-8,369	1,212	-,596	-6,906	,000	,265	3,771
	Katastrofik	-14,325	1,627	-,922	-8,806	,000	,180	5,559
	Kritik	-11,398	1,515	-,748	-7,522	,000	,199	5,014
	Çalısan	1,032	,886	,070	1,165	,247	,543	1,840
	İşlevsel	-,017	,677	-,001	-,026	,980	,641	1,559
	Sistem	,469	,717	,038	,653	,515	,578	1,730
	Bazen	,094	1,079	,005	,087	,931	,519	1,927
	HerZaman	,386	,774	,030	,498	,619	,534	1,872
	ProjeTestOrtamı	-1,083	,671	-,092	-1,613	,110	,607	1,647

Yorum: Tablo 4.46'teki hata derecesi ile ilgili tüm bağımsız değişkenlerin p değerlerinin 0,05'ten küçük olması bağımsız değişkenlerin anlamlı olduğunu ifade etmektedir. Katastrofik, kritik, büyük ve küçük bağımsız değişkenleri anlamlıdır ve

regresyon modeline girecektir. Bu durumda regresyon doğrusu denklemi Denklem (4.21)'deki gibi bulunmuştur.

$$\text{Kapanma süresi} = 17{,}960 * \beta_0 - 2{,}960 * \beta_1 - 8{,}369 * \beta_2 - 14{,}325 * \beta_3 - 11{,}398 * \beta_4 \quad (4.21)$$

β_0: kozmetik

β_1: küçük

β_2: büyük

β_3: katastrofik

β_4: kritik

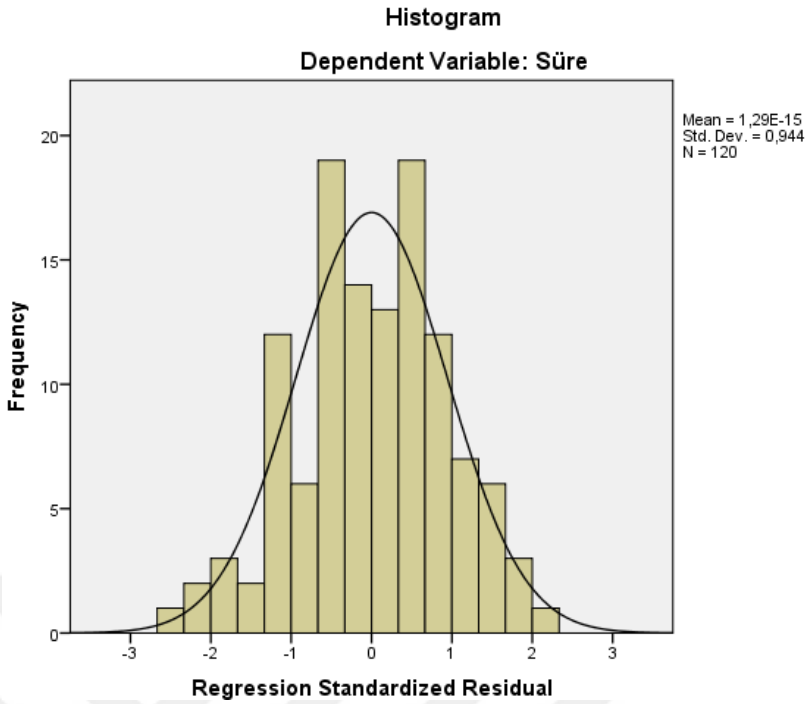
bağımsız değişkenlerini ifade etmektedir.

Model çoklu doğrusal bağlantı durumu için de test edilmiştir; tüm bağımsız değişkenlerin varyans artış faktörü (variance inflation factor - VIF) değerleri "10"un altındadır. Bu değerin 10'un altında olması çoklu doğrusal bağlantı sorunu bulunmadığını göstermektedir. VIF değeri "1"e yakınsa o değişken ile ilgili sorun bulunmadığı, "10" veya daha fazla olması durumunda önemli bir sorun olduğu sonucuna varılır [99].

"t"nin mutlak değeri bağımsız değişkenlerin, bağımlı değişken üzerinde etki büyüklüğünü göstermektedir. Tablo 4.46 incelendiğinde "katastrofik" bağımsız değişkenin hata kapatma süresi bağımlı değişkeni üzerinde en büyük etkiye sahip olduğu görülmektedir. Bunu sırasıyla "kritik", "büyük" ve son olarak "küçük" değişkenleri izlemektedir.

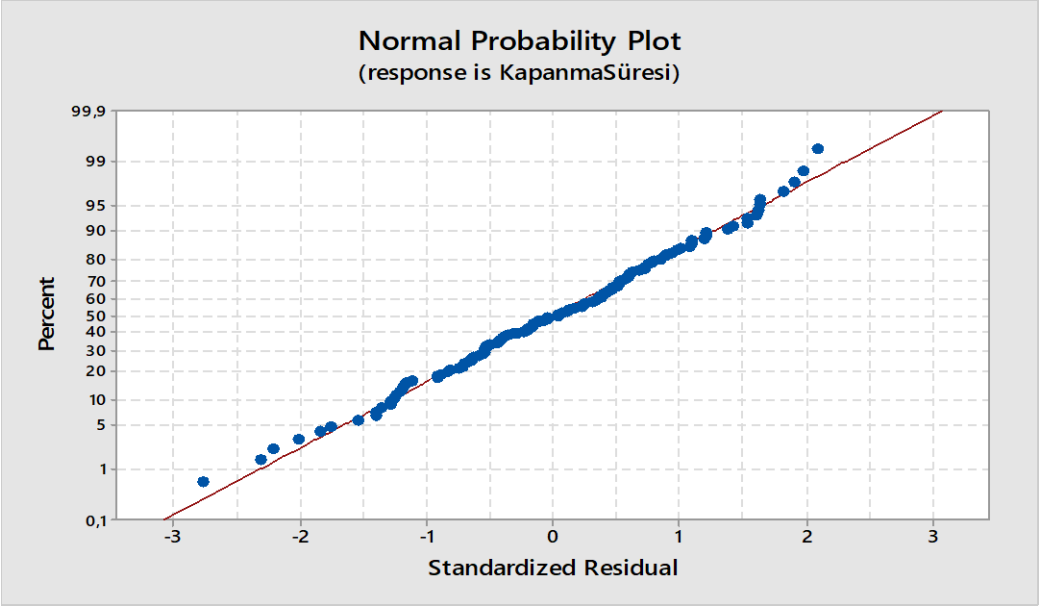
Hata terimlerinin ortalamasının sıfır ve normal dağılım göstermektedir. Bu durum modelle gözlenen veriler arasındaki farkların sıfır ya da sıfıra yakın olduğu şeklinde yorumlanır. Hata terimleri grafiği Şekil 4.35'te gösterilmiştir.

Tablo 4.46 incelendiğinde hatanın bulunduğu iş paketinin hata kapatma süresine etkisinin olmadığı görülmektedir. Benzer şekilde hatayı tespit eden tarafın müşteri ya da çalışan olmasının hata kapatma süresine etkisi bulunmamaktadır. Testin yeri, test seviyesi, test tipi ve hatanın oluşma sıklığı değişkenlerinin hata kapatma süresi üzerinde anlamlı etkisi bulunmamaktadır. Hatanın öncelik bilgisinin de hata kapatma süresine anlamlı bir şekilde etki etmediği gözlenmiştir.



Şekil 4.35. Hata kapatma süresi regresyon analizi – hata terimleri grafiği

Aşağıda Şekil 4.36'da verilen olasılık grafiğinde hata kapatma süresine ait regresyon modelinin hata terimlerinin normal dağılım özelliği gösterdiği daha açık olarak görülmektedir.



Şekil 4.36. Hata kapatma süresi regresyon analizi – hata terimleri olasılık grafiği

Şekil 4.35 ve Şekil 4.36, regresyon varsayımlarını karşıladığı için Tablo 4.46'de verilen hata yoğunluğu regresyon denklemi tahminleme amaçlı kullanılabilecektir.

5. MODEL İÇİN GELİŞTİRİLEN YAZILIM

Yazılım geliştirme süreç metrikleri ile proje performanslarının ölçülmesi ve proje başarı puanı hesaplanması amacıyla geliştirilen yazılım hakkında bu bölümde bilgi verilecektir. Geliştirilen yazılımda bu tez çalışmasında anlatılan yazılım yaşam döngüsü süreçleri esas alınmıştır. Yazılım sunucu ve istemci tarafında javascript dili kullanılarak geliştirilmiştir. Platform bağımsız tüm işletim sistemlerin çalışabilecek şekilde tasarlanmış olup web tabanlı kullanılmaktadır. Her bir yazılım süreci çıktısı ayrı birer bölüm olarak verilecektir. Öncelikle yazılımdan beklenen işlevlerinin anlatıldığı yazılım gereksinimleri dokümantasyonu hakkındaki bilgi aşağıdaki bölümde verilmiştir.

5.1. Yazılım Gereksinimleri Belirtimi

Gereksinim yönetimi standartlarına göre gereksinimlerin birbirine karışmasını önlemek için her bir yazılım gereksiniminin tekil bir betimleyicisi olmalıdır [100]. Bu kurala için yazılım isteri betimleyicisi olarak "YİS 1, YİS 2…" şeklinde bir gösterim kullanılmıştır.

YİS-1…Yazılım, konfigürasyon araçlarından veri çekecektir.

YİS-2… Yazılım, konfigürasyon araçlarından veri çekmek için sunucu kullanacaktır

YİS-3… Yazılım, bu sunucuda modül adı verilen kodu içerecektir.

YİS-4… Modüller, adapter, converter ve calculator tipinde olacaktır. Bu tiplerin işlevleri aşağıdaki gibi olacaktır.

Adapter: konfigürasyon aracına bağlantı sağlayan modül,
Converter: veri mapping işlemi yapmayı sağlayan modül,
Calculator: hesaplama işlemi yapmayı sağlayan modül.

YİS-5… Yazılım, modüller arasındaki ilişkiyi yönetecektir.

YİS-6… Yazılım, kullanıcıya bir arayüz sağlayacaktır. Bu arayüz üzerinden kullanıcı metrik seti, metrik oluşturacağı, çalıştıracağı ve sonuçları görüntüleyebileceği bir ekrana sahip olacaktır.

YİS-7… Yazılımın bu ekranları oluşturma ve düzenleme ekranları bir olacaktır ancak oluşturma ve çalıştırma işlevlerine göre farklı görünümler gösterecektir.

YİS-8… Metrik seti oluşturma ekranında metrik seti adı, açıklaması alanı ve metrik ekleme ile metrik oluşturma butonu bulunacaktır.

YİS-9… Metrik set oluşturma ekranında, metrik listesi bulunacaktır. Bu liste üzerinde metrik düzenlenebilecektir.

YİS-10… Metrik düzenleme alanında metriklerin içerisine 3 adet modül tiplerinden herhangi biri eklenebilecektir.

YİS-11… Metrik düzenleme bölümünde, metrik adı girilen bir alan bulunacaktır bu alan metrik düzenleme bölümünde bulunana butona göre metrik analiz detayı alanına dönüşerek açıklama girilmesini sağlayacaktır.

YİS-12… Metrik düzenleme bölümünde, metrik detaylarının girilmesini sağlayan alanı açacak olan buton, metriğe bağlı modülleri gösteren bölümü açacak olan buton, değişlikleri kayıt etmeye yarayan buton, metriği metrik setinden çıkarmaya yarayan buton ve metriği tamamen silmeye yarayan buton bulunacaktır.

YİS-13… Metrik set ekranın altında “Hazırla” butonu bulunacaktır. Buton ile metrik listesindeki metriklerin içerisine eklenen modüllerden ilk modüllerin giriş (input) bilgisinin doldurulması için kullanıcıya açılır pencere (popup) çıkarılacaktır.

YİS-14… Metrik set ekranında, “Hesapla” butonu bulanacak ve buton metrik set içerisindeki metrik listesinde bulunan metrikleri hesaplayacaktır.

YİS-15… Yazılım, kullanıcı tarafından post isteği ile metrik çalıştırma isteğini, metrik girdileri ile alacaktır

YİS-16... Alınan metrik set çalıştırma isteğinin ardından veri tabanından ilgili metriğin modül listesi alınacaktır.

YİS-17... Modül yönetim sistemi ilgili modüllerin kodlarını çalışır hale getirecektir ve sırasıyla çalıştıracaktır.

YİS-18... Modül Yönetim Sistemi, kullanıcı tarafından gelen girdi bilgisini ilk modüle verecektir.

YİS-19... İlk Modül, gerekli işlemleri yaptıktan sonra çıktı (output) verecektir. Bu çıktı bir modül yönetim sistemi aracılıyla bir sonraki modüle girdi olarak gelecektir.

YİS-20... Modül yönetim sistemi, metrik içerisindeki son modülü çalıştırdıktan sonra son modülün çıktı kullanıcı tarafında bulunan metrik hazırlama (metric builder) sisteminde geçici olarak saklanır.

YİS-21… Metric Builder sisteminde geçici olarak saklanan metrik sonuçları metrik setteki tüm metrikler hesaplandıktan sonra metrik set sonucu olarak veri tabanındaki sonuç bölümünde saklanır.

YIS-22… Metriklerin alınacağı projelerin belirlenmesi için “Projeler” menüsü bulunacaktır.

YIS-23… Proje bilgileri dinamik alanlarda tutulur. Bu dinamik alanlar “anahtar” ve “değer” ikilisinden oluşacaktır.

YIS-24… Projeler alanındaki dinamik alanlar belirlenmiş ön tanımlı anahtarlar bulunacaktır.

Bunlar;

ID
projectName
projectKey
projectJIRAKey
projectJAZZKey
projectCOLLABORATORKey
projectSVNKey

YIS-25… Ön tanımlı dinamik alanlara ek yeni alan eklenebilecektir. Yeni alan ekleyebilmek için dinamik alan listesinin hemen sonunda "Yeni Alan Ekle" butonu bulunacaktır.

YIS-26… Yazılım document-based veritabanı kullanacaktır.

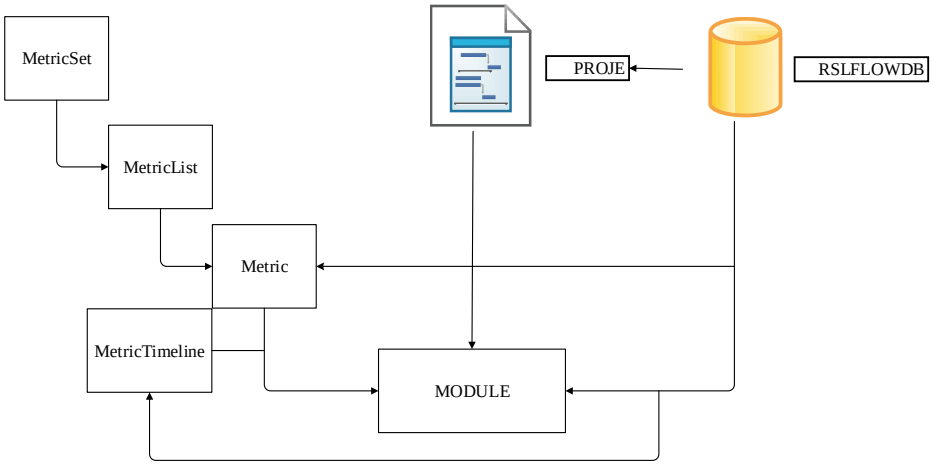
YIS-27… Yazılım gösterge (dashboard) ekranına sahip olacaktır.

YIS-28… Dashboard ekranında sisteme kayıt edilen eklentiler (widget) dinamik olarak eklenebilecektir.

YIS-29… Ön tanımlı olarak "Metrik Genel Bilgileri, Proje Detayları, Metrik Kartı" eklentileri bulunacaktır.

5.2. Yazılım Mimari Tasarım

Yazılım mimari tasarımı sürecinde, yazılım gereksinimleri ve sistem mimarisi, üst seviye tasarım kavramlarına dönüştürülür. Bu süreçte yazılım sistemi, belirli işlevleri ve alt işlevleri yerine getiren bileşenlerle mantıksal bölümlere ayrılarak tasarımın modüler olması sağlanır. Şekil 5.1'de geliştirilen yazılımın mimari tasarımı verilmiştir.



Şekil 5.1. Yazılım mimari tasarım şeması

Yazılım mimari tasarımında görülen RslFlowDB veri tabanı için NoSQL veri tabanı kullanılmıştır. Veri tabanında metrikler ilgili veriler bir dizin içerisinde dosya olarak depolanır ve istenilen anahtar-değer eklenebilen bir sistem olma özelliğini taşır.

“MetricSet” bileşeninde metrik listesi bulunmaktadır. Tek seferde birden fazla metrik çalıştırmak için mantıksal bir gruplama yapmaktadır. “Metriclist” bileşeni içerisinde metrikler bulunmaktadır. Metrik bileşeninin içerisinde de “modüle” bileşeni bulunur. Bu modüller bir “timeline” bileşeni ile dinamik olarak eklenebilmektedir. Modül bileşeni ise süreç iş akışlarının çalıştığı ilgili araçlardan (IBM JAZZ, JIRA vb.) veri çekmenin yanı sıra hesaplama işlemi de yapmaktadır.

5.3. Yazılım Detaylı Tasarım

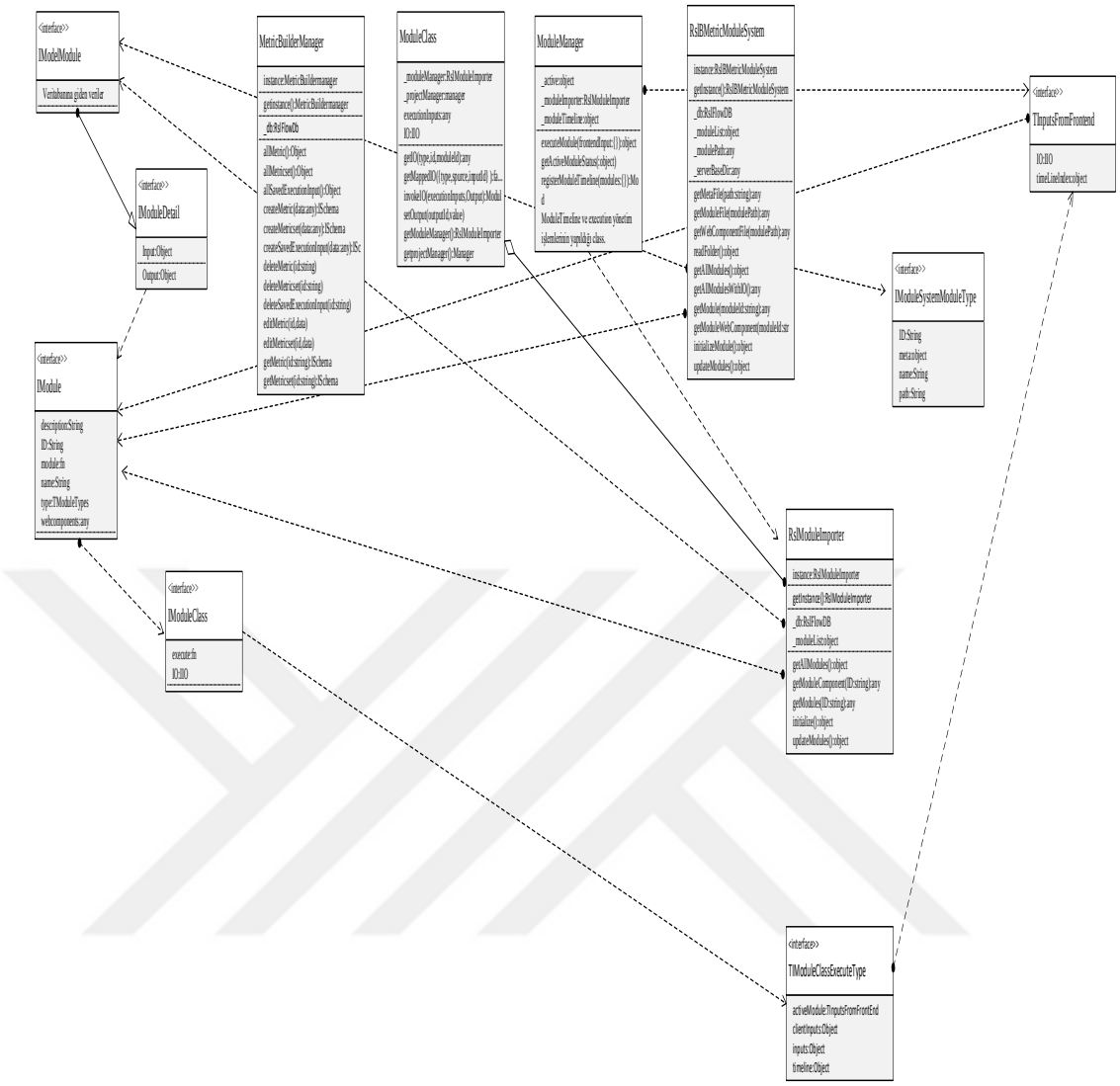
Yazılım detaylı tasarım süresinde, mimari tasarım detaylandırılarak kodlanıp, derlenip test edilebilecek birim seviyesine kadar tasarlanır. Yazılım birimleri arasındaki ara yüzler de tasarlanarak dokümante edilir. Her bir yazılım bileşeni için veri yapıları, giriş ve çıkış parametreleri, algoritmalar belirlenir. Belirlenen yazılım yapısına ve yazılım gereksinimlerine uygun olan etkinlik diyagramı (activity diagram), sınıf diyagramı (class diagram), ardışım diyagramı (sequence diagram), durum diyagramı (state diagram) vb. çizilir. Şekil 5.2’de geliştirilen yazılımın sınıf diyagramı verilmiştir. Diyagramda gösterilen sınıfların açıklamaları aşağıdaki gibidir:

MetricBuilderManager: Sunucu tarafında metrik set ve metrik ile ilgili oluşturma, silme, düzenleme gibi işlemler buradan yapılır.

ModuleManager: Metrik çalıştırılırken MetricTimeline içerisindeki modülleri çalıştıran ve yöneten sınıftır. Belirli bir sırayla çalıştırır. Bu sıra istemci tarafında belirlenir ve veri tabanına kayıt edilir ve buradaki sırayla çalıştırır. Bir modüle girdi (input) verilir ve modül çıktı (output) oluşturur. Bu oluşturulan çıktının bir sonraki modüle girdi olarak verilmesini sağlar.

RslBMetricModuleSystem: Sisteme kayıt edilen modüllerin bilgilerini alan, yöneten ve düzenleyen sınıftır.

RslModuleImporter: Sistemde kod olarak bulunan modülleri sisteme kayıt eden sınıftır. Modüller kod olarak güncellendiyse yine bu sınıf kullanılır.



Şekil 5.2. Yazılım sınıf diyagramı

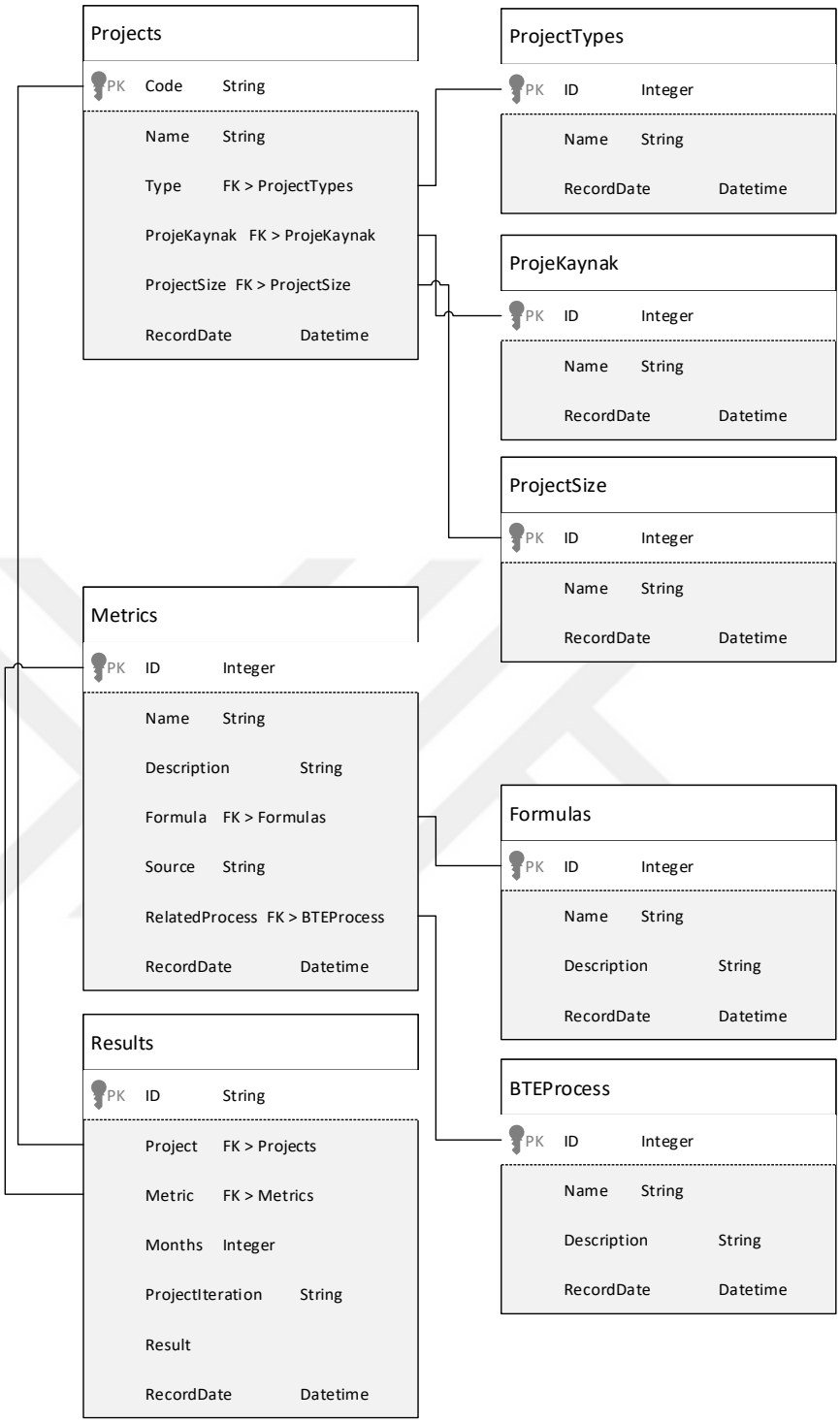
Veri tabanında projelerin isim, tip, kaynak, büyüklük gibi bilgilerinin saklandığı bir tablo oluşturulmuştur.

Metrikler tablosunda ise süreç metriklerinin isimleri, metrik açıklamaları, metriklerin hesaplanma formülleri, kaynağı ve metriğin ilgili olduğu süreç adı bulunmaktadır.

Sonuçlar tablosunda ölçüm yapılan proje ve metrik bilgileri, ölçümün hangi tarihte yapıldığı ve ölçüm sonuçları yer almaktadır.

5.4. Yazılımda Kullanılan Veri Tabanı Tasarımı

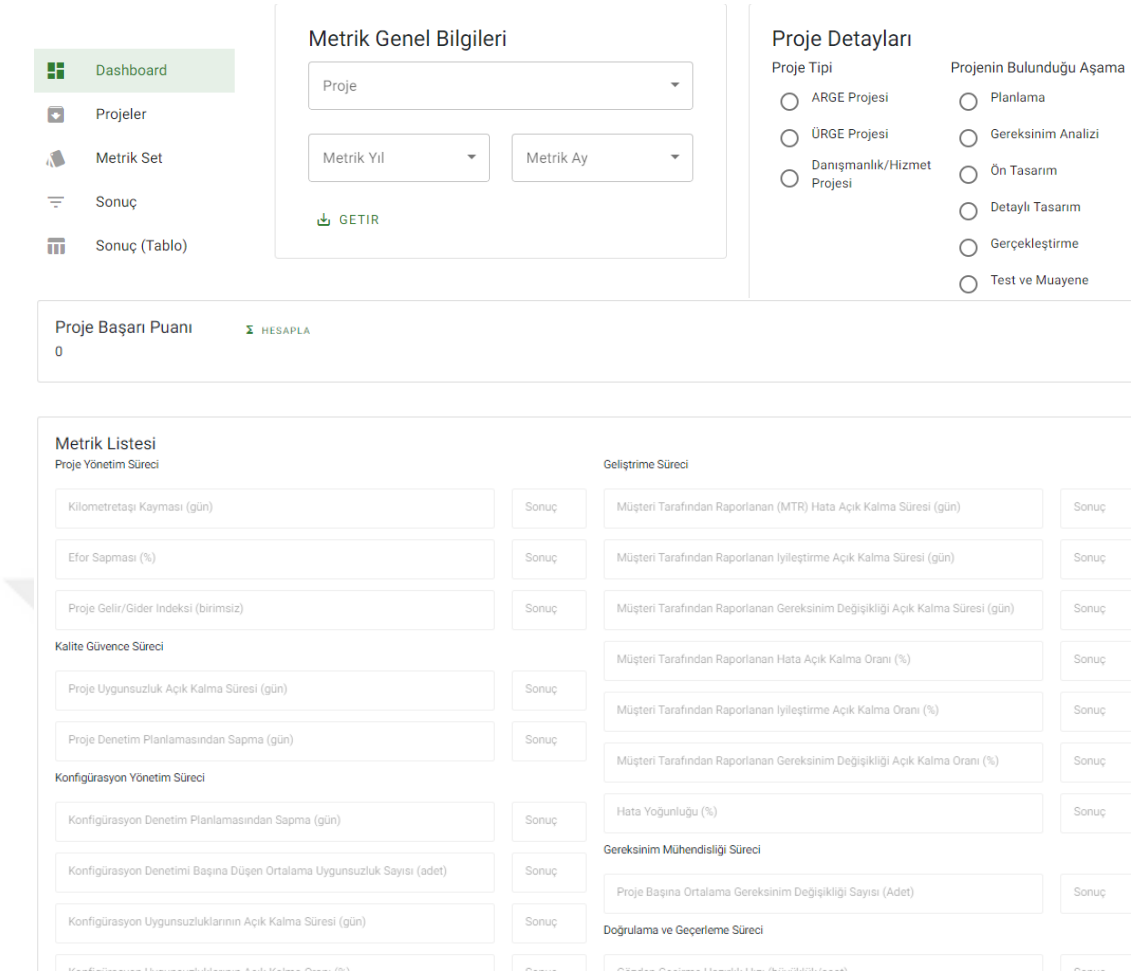
Yazılımda kullanılan veri tabanı tasarımı Şekil 5.3'te gösterilmiştir.



Şekil 5.3. Yazılımda kullanılan veri tabanı tasarım gösterimi

5.5. Yazılımda Ara Yüz Ekranlar

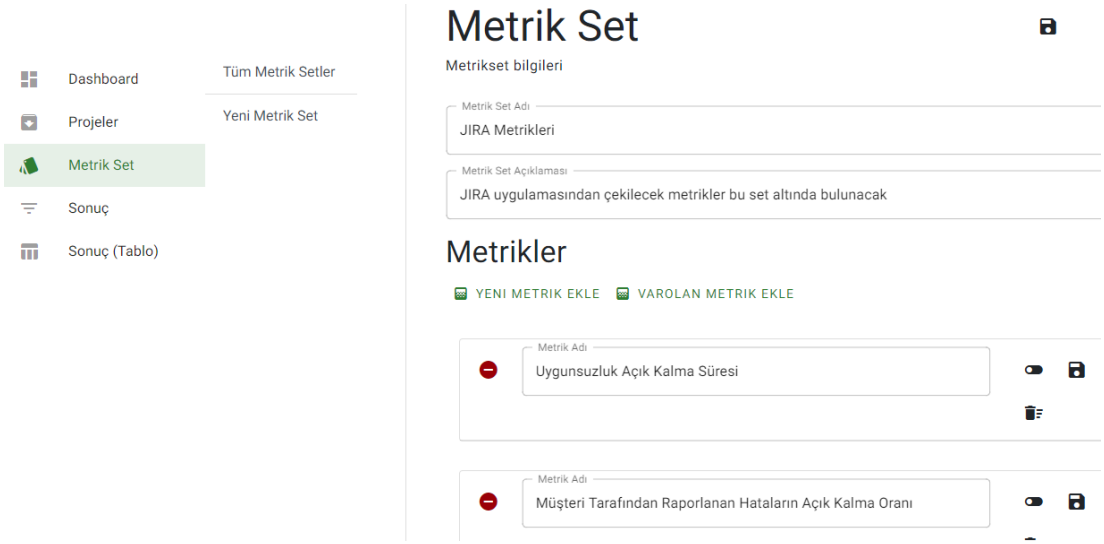
Bu bölümde geliştirilen yazılımdan ara yüzler resim olarak verilmiştir.



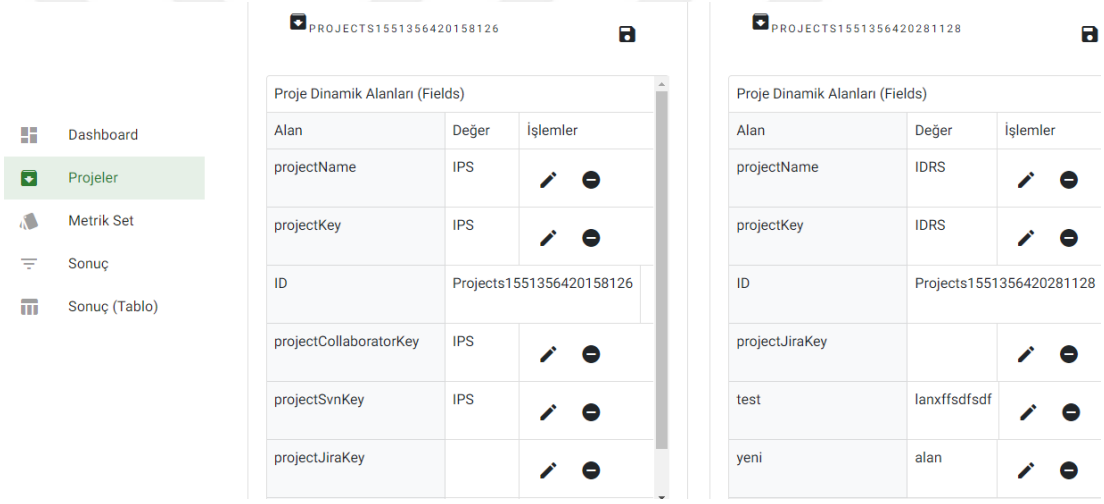
Şekil 5.4. Yazılım ana sayfa ekranı



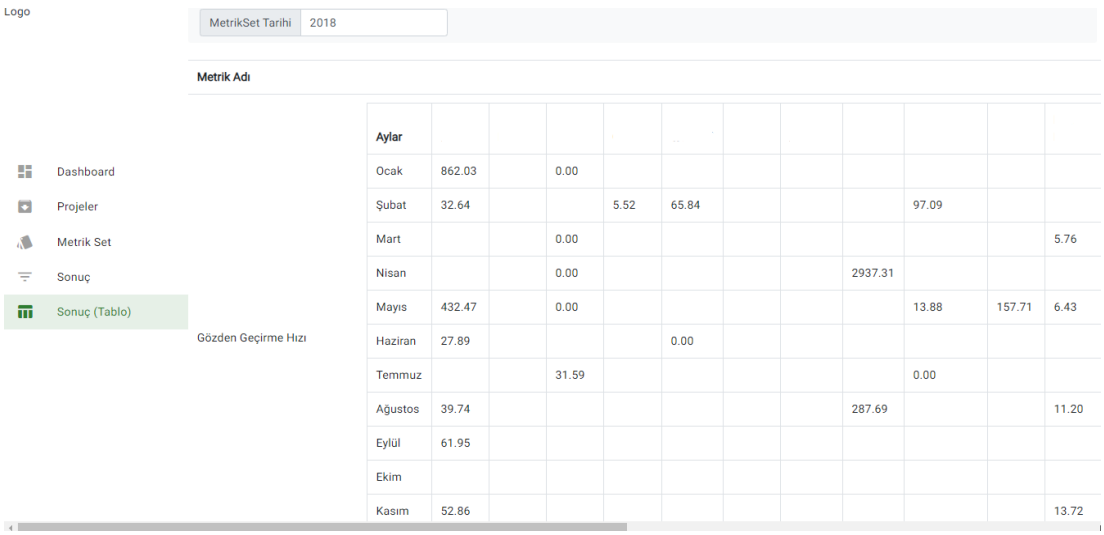
Şekil 5.5. Yazılım metrik tanımlama ekranı



Şekil 5.6. Yazılım metrik tanımlama detay ekranı



Şekil 5.7. Yazılım proje tanımlama ekranı



Şekil 5.8. Yazılım ölçüm sonuç ekranı

5.6. Yazılım Kodları

Geliştirilen yazılımın süreç metrik toplama ve hesaplama ile ilgili javascript kodları Ek-E'de verilmiştir.

6. SONUÇLAR VE ÖNERİLER

Yapılan çalışmanın sonuçları bu bölümde değerlendirilmekte, çalışma hakkında özet bilgiler verilmekte, kısıtlar belirtilmekte ve bundan sonra bu alanda yapılacak çalışmalar için öneriler sunulmaktadır.

Proje başarısına etki eden faktörlerin proje başlangıcında belirlenmesi ve bunun bilicinde olarak planlama yapılması ve önlem alınması proje başarısı açısından önem arz etmektedir. Projelerin başarıyla sonuçlanabilmesi için projeleri etkileyen faktörlerin kontrol edilmesi, izlenmesi ve yönetilmesi gerekmektedir. Bu başarıyı sağlayabilmenin yollarından bir tanesi de kurumsal verilerin toplanması, analiz edilmesi ve sonuçlara göre önlemler alınmasıdır.

Proje yönetimi felsefesi uluslararası standartlarla desteklenmiş ve bir metodolojiye dönüşmüştür. Bu standardizasyonun en önemli sonuçlarından biri olan ortak anlayış ilkesi ile proje yönetimi faaliyet adımları için asgari gereklilikler belirlenmiştir. Araştırmalara paralel olarak, bu faaliyetlerin ölçme, analiz, değerlendirme, iyileştirme hakkında ortak bir anlayış oluşturması amaçlandığı, ancak bu konudaki bazı boşlukların gözlemlendiği ve tartışıldığı görülmektedir. Literatür araştırması sonucunda proje başarı faktörlerinin genel anlamda belirlendiği sonucuna varılmıştır, ancak bunlar sektör bazında tartışıldığında bu genel hatların daha da özelleştirilmesi gerekmektedir. Başka bir deyişle, literatür incelendiğinde, ağırlıklı olarak üretim ve hizmet sektörlerinde ölçüm, analiz ve değerlendirme modelleri oluşturulmuş, ancak yazılım Ar-Ge sektörü için model ve uygulama örnekleri olmadığı görülmüştür.

Bu çalışmanın amacı, yazılım yaşam döngüsü süreç ölçümleriyle proje performansını değerlendirmek ve proje başarı puanını hesaplamak için bir model geliştirmektir. Literatürde eksiklik olarak görülen proje başarı faktörleri, Ar-Ge organizasyonunda ileri düzeyde deneyime ve yetkinliğe sahip uzmanlarla tartışılmıştır, eksik olduğu düşünülen proje başarı kriterleri belirlenip ve bu tez çalışmasında öneri olarak ileri sürülmüştür.

Bu kriterlerden bir proje performans değerlendirme modeli oluşturmak için, bir Ar-Ge organizasyonundaki deneyimli araştırmacılardan ve proje yöneticilerinden değerlendirmeler alınmıştır. Süreç metriklerinin göreceli önemi AHS yöntemi ile belirlenmiştir. Uzmanların kararları ile yapılan hesaplamalar sonucunda her bir kriterin ağırlığı tespit edilmiş ve model üzerindeki etkisi ortaya konulmuştur. Ana kriterlerin ağırlıklandırılması sonucunda, gereksinim yönetimi sürecinin proje başarısı üzerinde% 21,7 etkiye sahip olacağı gözlenmiştir. Bunu % 21,5 ile geliştirme süreci, % 11,9 ile proje yönetimi süreci ve konfigürasyon yönetimi süreci, % 15,1 ile doğrulama ve geçerleme süreci, % 9,5 ile risk yönetimi süreci ve % 8,4 ile kalite güvence süreci izlemektedir. Alt kriterlerin ağırlıklandırılması sonucunda, her bir süreç metriğinin proje başarısı üzerindeki etkisi ortaya konulmuştur.

Çalışmada analitik hiyerarşi sürecinin kullanılması tercih edilmiştir. Bu kararı alma sebepleri 1) sorunun çözümü için hiyerarşik bir karar yapısının oluşturulması, 2) çoklu uzman görüşlerinin değerlendirilmesini mümkün kılması, 3) ikili karşılaştırma avantajı, 4) karar verme gruplarının uzun tartışmalara katılımına gerek olmaması, 5) uzman kararlarındaki tutarsızlığı ele alma becerisi olarak sıralanabilir. Öte yandan, yeni kriterler eklendiğinde karar verme sürecinin çok zaman aldığı gözlemlenmiştir.

İstatistiksel süreç kontrolü tekniği, süreçlerin performansının izlenmesi, kontrol edilmesi ve yönetilmesi için istatistiki yöntemleri uygulayan bir araçtır. İstatistiksel süreç kontrolü için farklı yöntemler vardır. Bunlar serpilme diyagramları, çalışma grafikleri, neden ve sonuç diyagramları, histogramlar, çubuk grafikler, pareto grafikleri ve kontrol grafikleri olarak sayılabilir. Kontrol grafikleri imalat ve yazılım sektöründe süreci izlemek ve kontrol etmek için yaygın ve başarılı bir araç olarak kullanılmaktadır. Bu tez çalışmasında, süreç ölçümlerinin alt ve üst kontrol limitlerini belirlemek için kontrol grafikleri kullanılmıştır. İstatistiksel süreç kontrolünde farklı kontrol grafik türleri vardır. Yazılım süreçlerinde, veri noktaları genellikle imalat süreçlerinde olduğu kadar sık değildir. Her veri noktası ayrı ayrı çizilir ve değerlendirilir. Bu nedenle, yazılım mühendisliği alanında en çok kullanılan kontrol grafiği Bireysel (X) ve Hareketli Aralık (MR) kontrol grafikleridir. Bu tez çalışmasında süreç performans dayanaklarını belirlemek için bireysel gözlem değerleri için kontrol grafiği kullanılmıştır.

Önerilen modelin uygulanabilirliğini ve sonuçların elde edilmesinin etkilerini gözlemlemek için bir Ar-Ge kuruluşundan bir vaka çalışması yapılmıştır. Bu uygulama örneği sonucunda süreç kontrol grafikleri oluşturulmuş ve süreç performansları değerlendirilmiştir. Süreç metrik ölçümlerinin kontrol grafiklerindeki üst ve alt kontrol sınırlarını aşması durumunda, proje yöneticileri bu durumun nedenini kök neden analizi ile araştırabilir ve gerekirse düzeltici eylem başlatarak proje performansını artırmayı hedefleyebilirler.

Bu çalışmanın proje yöneticileri ve üst düzey yöneticiler için önemli etkileri vardır. İlk olarak, bu çalışma proje performans değerlendirmesinde kullanılacak proje başarı faktörlerinin bir listesini oluşturmuştur. Bu liste, proje yöneticilerinin proje başarısı için önemli olan temel süreç başarı kriterlerini anlamalarına yardımcı olacaktır. İkinci olarak, bu çalışma proje yöneticilerine projelerinin başarısını ölçmek için bir model sunmaktadır. Model, proje yöneticilerinin projelerin süreç performansını iyileştirmesi için bir araç görevi görebilir.

Yazılım geliştirme yaşam döngüsü süreç metrikleri ile proje performanslarının ölçülmesi ve proje başarı puanı hesaplanması amacıyla bir yazılım geliştirilmiştir. Geliştirilen yazılımda bu tez çalışmasında anlatılan yazılım yaşam döngüsü süreçleri esas alınmıştır. Yazılım sunucu ve istemci tarafında javascript dili kullanılarak geliştirilmiştir. Yazılım platform bağımsız tüm işletim sistemlerinde çalışabilecek şekilde tasarlanmış olup web tabanlı kullanılmaktadır.

Bu çalışma verilerin toplandığı ve uzmanların değerlendirmelerinin alındığı Ar-Ge kurumu bağlamıyla sınırlıdır ve bu nedenle sonuçlar sadece bu bağlamda geçerli kabul edilebilir. Diğer kuruluşlar kendi yazılım yaşam döngüsü süreç metriklerini tanımlar ve projelerden süreç metrik ölçümlerini toplayabilir ise bu tez kapsamında geliştirilen yazılımı kullanarak kendi kurumlarının proje performans değerlendirme ve proje başarı puanı hesaplama faaliyetlerini gerçekleştirebilirler. Bu tez çalışmasında AHS çalışması ile metrik ağırlıklarını belirleme işlemi ayrı bir çalışma olarak yapılmış ve sonuçları yazılımda kullanılmıştır. Metrik ağırlık belirleme işlemi yazılım ile yapıldığında önerilen proje performans değerlendirme sistemi etkinliğinin daha da artacağı değerlendirilmektedir.

KAYNAKLAR

[1] Presidency of the Republic of Turkey Strategy and Budget Department (in 681 article).

[2] Presidency of the Republic of Turkey Strategy and Budget Department Medium Term Program 2016-2018 (in 180 article).

[3] Republic of Turkey Ministry of Industry and Technology, Turkey Software Sector Strategy and Action Plan 2017-2019.

[4] Radujković M., Sjekavica M., Project Management Success Factors, *Creative Construction Conference*, Primošten, Croatia, 19-22 June 2017.

[5] Merrow E, Independent Project Analysis Report, *IPA*, 2011.

[6] ISO/IEC 12207, International Standard Systems and Software Engineering - Software Life Cycle Processes, ISO/IEC/IEEE, 2017 DOI:10.1109/IEEESTD.2017.8100771.

[7] Santhanam R., Schniederjans M. J., A Model Formulation System for Information System Project Selection, *Computers & operations research*, 1993, **20**(7), 755-767.

[8] Akyol M. M, Yazılım Süreçlerini Etkileyen Faktörlerin Belirlenmesine İlişkin Bir Ölçüm ve İyileştirme Modeli ve Uygulaması, Doktora Tezi, İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2013, 337117.

[9] Ayyıldız M., Kalıpsız O., Yavuz S., YEEM: Yazılım Projeleri Maliyet Tahminleme Ölçev Seti ve Modeli, *ELECO*, Bursa, Türkiye, 6-10 Aralık 2006.

[10] Sanchez O. P., Terlizzi M. A., Moraes H., Cost and Time Project Management Success Factors for Information Systems Development Projects, *International Journal of Project Management*, 2017, **35**(8), 1608–1626.

[11] Mir F. A., Pinnington A. H., Exploring the Value of Project Management: Linking Project Management Performance and Project Success, *International Journal of Project Management*, 2014, **32** (2), 202–217.

[12] Todorović M. J., Petrović D. Č, Mihić M. M., Obradović V. L., Bushuyev S. D., Project Success Analysis Framework: A Knowledge-Based Approach in Project Management, *International Journal of Project Management*, 2015, **33**(4), 772–783.

[13] Verner J., Kitchenham B., Cerpa N., Estimating Project Outcomes, *ICSSEA Conference*, Paris, France, 4-6 December 2007.

[14] Garousi V., Tarhan A., Pfahl D., Coşkunçay A., Demirörs O., Correlation of Critical Success Factors with Success of Software Projects: An Empirical Investigation, *Software Quality Journal*, 2019, **27**(1), 429–493.

[15] Wadugodapitiya R. R. M. M. K., Sandanayake Y. G., Thurairajah N., Building Project Performance Evaluation Model, *Proc. CIB World Congress,* Salford, UK, 10-13 May 2010.

[16] Ilbeigi M., Heravi G., Developing a Model to Evaluate Project Performance: Contractor Company's Viewpoint, *Joint Int. Symp. CIB*, Dubrovnik, Croatia, 27-30 September 2009.

[17] Koelmans R.G., Project Success and Performance Evaluation, *Int. Platinum Conf. The South African Institute of Mining and Metallurgy*, Sun City, South Africa, 3-7 October 2004.

[18] Leu S. S., Lin Y. C., Project Performance Evaluation Based on Statistical Process Control Techniques, *Journal of Construction Engineering and Management*, 2008, **134**(10), 813-819.

[19] Bauch G. T., Chung C. A., A Statistical Project Control Tool for Engineering Managers, *Project Management Journal*, 2001, **32**(2), 37-44.

[20] Bower D. C., Finegan A. D., New Approaches in Project Performance Evaluation Techniques, *International Journal of Managing Projects in Business*, 2009, **2**(3), 435-444.

[21] Shahzad B., Said A. M., Identification and Quantitative Analysis of Project Success Factors for Large Scale Projects, *International Journal of Knowledge Society Research (IJKSR)*, 2014, **5**(1), 83-95.

[22] Viglioni T. G., Cunha J. A. O., Moura H. P., A Performance Evaluation Model for Project Management Office Based on a Multicriteria Approach, *Procedia Computer Science*, 2016, **100**, 955-962.

[23] Doskočil R., Škapa S., Olšová P., Success Evaluation Model for Project Management, *Economics and Management*, DOI: 10.15240/tul/001/2016-4-012.

[24] Shashi K. N., Nair T. G., Suma V., SLI, A New metric to Determine Success of a Software Project, 2014 *International Conference on Electronics and Communication Systems (ICECS)*, DOI: 10.1109/ECS.2014.6892814.

[25] Badewi A., The Impact of Project Management (PM) and Benefits Management (BM) Practices on Project Success: Towards Developing a Project Benefits Governance Framework, *International Journal of Project Management*, 2016, **34**(4), 761-778.

[26] Mossalam A., Arafa M., The Role of Project Manager in Benefits Realization Management as a Project Constraint/Driver, *HBRC Journal*, 2016, **12**(3), 305-315.

[27] Gomes J., Romão M., Improving Project Success: A Case Study Using Benefits and Project Management, *Procedia Computer Science*, 2016, **100**, 489-497.

[28] Bryde D. J., Robinson L., Client versus Contractor Perspectives on Project Success Criteria, *International Journal of Project Management*, 2005, **23**(8), 622-629.

[29] Jugdev K., Müller R., A Retrospective Look at Our Evolving Understanding of Project Success, *Project Management Journal*, 2005, **36**(4), 19-31.

[30] Baccarini D., The Logical Framework Method for Defining Project Success, *Project Management Journal*, 1999, **30**(4), 25-32.

[31] Besner C., Hobbs B., The Project Management Tools and Techniques: The Portrait of Current Professional Practice, *Project Management Journal*, 2006, **37**(3), 37-48.

[32] Lim C. S., Mohamed M. Z., Criteria of Project Success: An Exploratory Re-Examination, *International Journal of Project Management*, 1999, **17**(4), 243-248.

[33] Radujković M., The Role of Project Management in Construction Industry Modernization, *13th China International Construction Project Management Summit*, Hangzhou, Zhejiang, 11-13 October 2014.

[34] Cooke-Davies T., The Real Success Factors on Projects, *International Journal of Project Management*, 2002, **20**(3), 337-342.

[35] Müller R., Jugdev K., Critical Success Factors in Projects: Pinto, Slevin, and Prescott - The Elucidation of Project Success, *International Journal of Managing Projects in Business*, 2012, **5**(4), 757-775.

[36] Bryde D. J., Project Management Concepts, Methods and Application, *International Journal of Operations & Production Management*, 2003, **23**(7), 775 – 793.

[37] Chou J. S., Irawan N., Pham A. D., Project Management Knowledge of Construction Professionals: Cross-Country Study of Effects on Project Success, *Journal of Construction Engineering and Management*, 2013, **139**(11), 1-15.

[38] Ellatar S. M. S., Towards Developing an Improved Methodology for Evaluating Performance and Achieving Success in Construction Projects, *Scientific Research and Essay*, 2009, **4**(6), 549-554.

[39] Linde K. V., PricewaterCuppers (PWC) Insights and Trends: Current Portfolio, Programme and Project Management Practices, *3rd Global Survey on the Current State of Project Management*, PWC, 2012.

[40] Feger A. L. R., Thomas G. A., A Framework for Exploring the Relationship Between Project Manager Leadership Style and Project Success, *International Journal of Management*, 2012, **1**(1), 1-19.

[41] Keil M., Rai A., Liu S., How User Risk and Requirements Risk Moderate the Effects of Formal and Informal Control on the Process Performance of IT Projects, *European Journal of Information Systems*, 2013, **22**(6), 650-672.

[42] Taylor H., Critical Risks in Outsourced IT Projects: The Intractable and the Unforeseen, *Communications of the ACM*, 2006, **49**(11), 74-79.

[43] Al-Tmeemy S. M. H. M., Abdul-Rahman H., Harun Z., Future Criteria for Success of Building Projects in Malaysia, *International Journal of Project Management*, 2011, **29**(3), 337-348.

[44] Blindenbach-Driessen F., Van Den Ende J., Innovation in Project-Based Firms: The Context Dependency of Success Factors, *Research Policy*, 2006, **35**(4), 545-561.

[45] Bryde D. J., Perceptions of the Impact of Project Sponsorship Practices on Project Success, *International Journal of Project Management*, 2008, **26**(8), 800–809.

[46] Nahod M. M., Radujković M., The Impact of ICB 3.0 Competences on Project Management Success, *Procedia-Social and Behavioral Sciences*, 2013, **74**(0), 244-254.

[47] Ramazani J., Jergeas G., Project Managers and the Journey From Good to Great: The Benefits of Investment in Project Management Training and Education, *International Journal of Project Management*, 2015, **33**(1), 41-52.

[48] Turner J. R., Müller R., Dulewicz V., Comparing the Leadership Styles of Functional and Project Managers, *International Journal of Managing Projects in Business*, 2009, **2**(2), 198-216.

[49] Ika L.A., Project Success as a Topic in Project Management Journals, *Project Management Journal*, 2009, **40**(4), 6-19.

[50] Radujković M., Voditelj Projekta, *Građevinar*, 2000, **52**(03), 143-151.

[51] Westerveld E., The Project Excellence Model®: Linking Success Criteria and Critical Success Factors, *International Journal of Project Management*, 2003, **21**(6), 411-418.

[52] Skulmoski G., Project Maturity and Competence Interface, *Cost Engineering - Ann Arbor Then Morgantown*, 2001, **43**(6), 11-24.

[53] Yang L. R., Huang C. F., Wu K. S., The Association among Project Manager's Leadership Style, Teamwork and Project Success, *International Journal of Project Management*, 2011, **29**(3), 258–267.

[54] De Lange W. J., Multi-Criteria Decision-Making for Water Resource Management in the Berg Water Management Area, PhD Thesis, University of Stellenbosch, Stellenbosch, South Africa, 2006.

[55] Saaty T. L., *The Analytic Hierarchy Process*, McGraw-Hill, New York, 1980.

[56] Saaty T. L., Vargas L. G., *Models, Methods, Concepts & Applications of the Analytic Hierarchy Process*, Springer, Boston, MA, 1-25, 2001.

[57] Forman E. H., Selly M. A., *Decision by Objectives (How to Convince Others That You Are Right)*, World Scientific Pub. Co., USA, 2001.

[58] Saaty T. L., Decision Making with the Analytic Hierarchy Process, *International Journal of Services Sciences*, 2008, **1**(1), 83-98.

[59] Lee D., Hwang J., Decision Support for Selecting Exportable Nuclear Technology Using the Analytic Hierarchy Process: A Korean Case, *Energy Policy*, 2010, **38**(1), 161-167.

[60] Saaty T. L., Ozdemir M. S., Why the Magic Number Seven Plus or Minus Two, *Mathematical and Computer Modelling*, 2003, **38**(3–4), 233–244.

[61] Saaty T. L., How to Make a Decision: The Analytic Hierarchy Process, *Interfaces*, 1994, **24**(6), 19-43.

[62] Saaty R. W., The Analytic Hierarchy Process—What It is and How It is Used, *Mathematical Modelling*, 1987, **9**(3-5), 161-176.

[63] Saaty T. L., *Multicriteria Decision Making the Analytic Hierarchy Process*, Eta Services Ltd, USA, 1988.

[64] Chu A. T. W., Kalaba R. E., Spingarn K, A Comparison of Two Methods for Determining the Weights of Belonging to Fuzzy Sets, *Journal of Optimization Theory and Applications*, 1979, **27**(4), 531-538.

[65] Crawford G., Williams C., A Note on the Analysis of Subjective Judgment Matrices, *Journal of Mathematical Psychology*, 1985, **29**(4), 387-405.

[66] Barzilai J., Deriving Weights From Pairwise Comparison Matrices, *Journal of the Operational Research Society*, 1997, **48**(12), 1226-1232.

[67] Bryson N., A Goal Programming Method for Generating Priority Vectors, *Journal of the Operational Research Society*, 1995, **46**(5), 641-648.

[68] Erginel N. M., Tasarım Hata Türü ve Etkileri Analizinin Etkinliği için Bir Model ve Uygulaması, *Endüstri Mühendisliği Dergisi*, 2004, **15**(3), 25-44.

[69] Saaty T. L., Some Mathematical Concepts of the Analytic Hierarchy Process, *Behaviormetrika*, 1991, **18**(29), 1-9.

[70] Dalkey N., Helmer O., An Experimental Application of the Delphi Method to the Use of Experts, *Management Science*, 1963, **9**(3), 458-467.

[71] Delbecq A. L., Van de Ven A. H., Gustafson D. H., *Group Techniques for Program Planning : A Guide to Nominal Group and Delphi Processes*, Scott Foresman Company, Glenview, Illinois, 1975.

[72] Mergen Y., Sistem Tercihinde Analitik Hiyerarşi Modelinin Tapu ve Kadastro Genel Müdürlüğü'nde Uygulaması, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2006, 180361.

[73] Çam H., Toraman A., Hazar Petrollerinin Pazar Stratejisi ve AHS Esaslı Alternatif Güzergâh Değerlendirme Modeli, *İTÜ Mühendislik Dergisi*, 2003, **2**(6), 41-46.

[74] Akyıldız E., Analitik Hiyerarşi Süreci ve Bankacılık Sektöründe Bir Uygulama, Yüksek Lisans Tezi, Gazi Üniversitesi, Sosyal Bilimler Enstitüsü, Ankara, 2006, 190861.

[75] Kuruüzüm A., Atsan N., Analitik Hiyerarşi Yöntemi ve İşletmecilik Alanındaki Uygulamaları, *Akdeniz İ.İ.B.F. Dergisi*, 2001, **1**, 83-105.

[76] Özkan Ö., Personel Seçiminde Karar Verme Yöntemlerinin İncelenmesi: AHS, Electre ve Topsis Örneği, Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi, Sosyal Bilimler Enstitüsü, İzmir, 2007, 207839.

[77] Dağdeviren M., Eraslan E., Kurt M., Dizdar E. N., Tedarikçi Seçimi Problemine Analitik Ağ Süreci ile Alternatif Bir Yaklaşım, *Teknoloji Z.K.Ü. Karabük Teknik Eğitim Fakültesi Dergisi*, 2005, **8**(2), 115-122.

[78] Aydın G., Analitik Hiyerarşi Prosesi (AHS) ve Bir Sanayi İşletmesinde Uygulanması, Yüksek Lisans Tezi, Kocaeli Üniversitesi, Sosyal Bilimler Enstitüsü, Kocaeli, 2008, 229507.

[79] Florac W.A., Carleton A.D., *Measuring the Software Process - Statistical Process Control for Software Process Improvement*, Addison Wesley, SEI Series on Software Engineering, Boston, USA, 1999.

[80] Jacob A. L., Pillai S. K., Statistical Process Control to Improve Coding and Code Review, *IEEE software*, 2003, **20**(3), 50-55.

[81] Florac W. A., Carleton A. D., Barnard J. R., Statistical Process Control: Analyzing Space Shuttle Onboard Software Process, *IEEE Software*, 2000, **17**(4), 97-106.

[82] Jalote P., Saxena A., Optimum Control Limits for Employing Statistical Process Control in Software Process, *IEEE Transactions on Software Engineering*, 2002, **28**(12), 1126-1134.

[83] Wang Q., Jiang N., Gou L., Liu X., Li M., Wang Y., BSR: A Statistic-Based Approach for Establishing and Refining Software Process Performance

Baseline, *Proceedings of the 28th International Conference on Software Engineering*, Shanghai, China, May, 2006.

[84] Card D., Statistical Process Control For Software?, *IEEE Software*, 1994, 11(3), 95–97.

[85] Montgomery D. C., *Introduction to Statistical Process Control*, 6th ed., Wiley, USA, 2009.

[86] Small B. B., *Statistical Quality Control Handbook*, 1st ed., Western Electric Company, Indiana, 1956.

[87] Juran J. M., Godfrey A. B., (1999). *Juran's Quality Handbook*, 5th ed., McGraw-Hill, New York, 1999.

[88] Wadsworth H. M., Stephens K. S., Godfrey A. B., *Modern Methods for Quality Control and Improvement*, 2nd ed., Wiley, New York, 2002.

[89] SEI CMMI® for Development Version 1.3, Capability Maturity Model Integration, *Software Engineering Institute*, Pittsburgh, 2010.

[90] ISO/IEC JTC 1/SC 7, ISO/IEC TR 15504, Information Technology-Software Process Assessment, *International Organization for Standardization*, Switzerland, 1998.

[91] IEEE Std 15939, Systems and Software Engineering-Measurement Process, *IEEE Computer Society*, New York, 2008.

[92] Tarhan A., An Assessment Model for the Applicability of Statistical Process Control for Software Processes, PhD Thesis, Middle East Technical University, Graduate School Of Informatics, Ankara, 2006, 199161.

[93] ISO/IEC/IEEE 12207, Systems and Software Engineering -Software Life Cycle Processes, *International Organization for Standardization*, Switzerland, 2017.

[94] Günel B. D., Erdoğan O., Dikici A., Learner Control Charts: A Structured Way for Finding Common Causes with Decision Trees, *Euro SPI*, Dundalk, Ireland, 2013.

[95] Florac W. A., Park R. E., Carleton, A. D., Practical Software Measurement: Measuring for Process Management and Improvement, *Software Engineering Institute*, CMU/SEI-97-HB-003, 97-100, 1997.

[96] Montgomery D. C., Runger, G. C., *Applied Statistics and Probability for Engineers*, 6th ed., Wiley, USA, 2014.

[97] Serper Ö., *Uygulamalı İstatistik*, 7. Baskı, Ezgi Kitabevi, İstanbul, 2014.

[98] Bahçecitapar M., Aktaş S., Çoklu Doğrusal Bağlantı Durumunda Doğrusal Karma Modelin Kullanımı ve Bir Uygulama, *Sakarya University Journal of Science*, 2017, **21**(6), 1349-1359.

[99] Ott R. L., Longnecker M., *An Introduction to Statistical Methods and Data Analysis*, 5th ed., Duxbury, Pacific Grove,2001.

[100] IEEE Std 830, Recommended Practice for Software Requirements Specifications, *IEEE Computer Society*, New York, 1998.

EKLER

Ek-A

Tablo A.1. Kontrol limitleri hesaplamaları için katsayılar tablosu

Observations in Sample, n	Chart for Averages: Factors for Control Limits A	A_2	A_3	Chart for Averages: Factors for Center Line c_4	$1/c_4$	Chart for Standard Deviations: Factors for Control Limits B_3	B_4	B_5	B_6	Factors for Center Line d_2	$1/d_2$	Chart for Ranges: Factors for Control Limits d_3	D_1	D_2	D_3	D_4
2	2.121	1.880	2.659	0.7979	1.2533	0	3.267	0	2.606	1.128	0.8865	0.853	0	3.686	0	3.267
3	1.732	1.023	1.954	0.8862	1.1284	0	2.568	0	2.276	1.693	0.5907	0.888	0	4.358	0	2.574
4	1.500	0.729	1.628	0.9213	1.0854	0	2.266	0	2.088	2.059	0.4857	0.880	0	4.698	0	2.282
5	1.342	0.577	1.427	0.9400	1.0638	0	2.089	0	1.964	2.326	0.4299	0.864	0	4.918	0	2.114
6	1.225	0.483	1.287	0.9515	1.0510	0.030	1.970	0.029	1.874	2.534	0.3946	0.848	0	5.078	0	2.004
7	1.134	0.419	1.182	0.9594	1.0423	0.118	1.882	0.113	1.806	2.704	0.3698	0.833	0.204	5.204	0.076	1.924
8	1.061	0.373	1.099	0.9650	1.0363	0.185	1.815	0.179	1.751	2.847	0.3512	0.820	0.388	5.306	0.136	1.864
9	1.000	0.337	1.032	0.9693	1.0317	0.239	1.761	0.232	1.707	2.970	0.3367	0.808	0.547	5.393	0.184	1.816
10	0.949	0.308	0.975	0.9727	1.0281	0.284	1.716	0.276	1.669	3.078	0.3249	0.797	0.687	5.469	0.223	1.777
11	0.905	0.285	0.927	0.9754	1.0252	0.321	1.679	0.313	1.637	3.173	0.3152	0.787	0.811	5.535	0.256	1.744
12	0.866	0.266	0.886	0.9776	1.0229	0.354	1.646	0.346	1.610	3.258	0.3069	0.778	0.922	5.594	0.283	1.717
13	0.832	0.249	0.850	0.9794	1.0210	0.382	1.618	0.374	1.585	3.336	0.2998	0.770	1.025	5.647	0.307	1.693
14	0.802	0.235	0.817	0.9810	1.0194	0.406	1.594	0.399	1.563	3.407	0.2935	0.763	1.118	5.696	0.328	1.672
15	0.775	0.223	0.789	0.9823	1.0180	0.428	1.572	0.421	1.544	3.472	0.2880	0.756	1.203	5.741	0.347	1.653
16	0.750	0.212	0.763	0.9835	1.0168	0.448	1.552	0.440	1.526	3.532	0.2831	0.750	1.282	5.782	0.363	1.637
17	0.728	0.203	0.739	0.9845	1.0157	0.466	1.534	0.458	1.511	3.588	0.2787	0.744	1.356	5.820	0.378	1.622
18	0.707	0.194	0.718	0.9854	1.0148	0.482	1.518	0.475	1.496	3.640	0.2747	0.739	1.424	5.856	0.391	1.608
19	0.688	0.187	0.698	0.9862	1.0140	0.497	1.503	0.490	1.483	3.689	0.2711	0.734	1.487	5.891	0.403	1.597
20	0.671	0.180	0.680	0.9869	1.0133	0.510	1.490	0.504	1.470	3.735	0.2677	0.729	1.549	5.921	0.415	1.585
21	0.655	0.173	0.663	0.9876	1.0126	0.523	1.477	0.516	1.459	3.778	0.2647	0.724	1.605	5.951	0.425	1.575
22	0.640	0.167	0.647	0.9882	1.0119	0.534	1.466	0.528	1.448	3.819	0.2618	0.720	1.659	5.979	0.434	1.566
23	0.626	0.162	0.633	0.9887	1.0114	0.545	1.455	0.539	1.438	3.858	0.2592	0.716	1.710	6.006	0.443	1.557
24	0.612	0.157	0.619	0.9892	1.0109	0.555	1.445	0.549	1.429	3.895	0.2567	0.712	1.759	6.031	0.451	1.548
25	0.600	0.153	0.606	0.9896	1.0105	0.565	1.435	0.559	1.420	3.931	0.2544	0.708	1.806	6.056	0.459	1.541

Ek-B

Tablo B.1. Kritik T değerleri tablosu

V (serbestlik derecesi)	α (Anlamlılık Düzeyi)			
	Tek Taraflı Test			
	0,05	0,025	0,01	0,005
	Çift Taraflı Test			
	0,10	0,05	0,02	0,01
1	6,314	12,706	31,821	63,657
2	2,920	4,303	6,965	9,925
3	2,353	3,182	4,541	5,841
4	2,132	2,776	3,747	4,604
5	2,015	2,571	3,365	4,032
6	1,943	2,447	3,143	3,707
7	1,895	2,365	2,998	3,499
8	1,860	2,306	2,896	3,355
9	1,833	2,262	2,821	3,250
10	1,812	2,228	2,764	3,169
11	1,796	2,201	2,718	3,106
12	1,782	2,179	2,681	3,055
13	1,771	2,160	2,650	3,012
14	1,761	2,145	2,624	2,977
15	1,753	2,131	2,602	2,947
16	1,746	2,120	2,583	2,921
17	1,740	2,110	2,567	2,898
18	1,734	2,101	2,552	2,878
19	1,729	2,093	2,539	2,861
20	1,725	2,086	2,528	2,845
21	1,721	2,080	2,518	2,831
22	1,717	2,074	2,508	2,819
23	1,714	2,069	2,500	2,807
24	1,711	2,064	2,492	2,797
25	1,708	2,060	2,485	2,787
26	1,706	2,056	2,479	2,779
27	1,703	2,052	2,473	2,771
28	1,701	2,048	2,467	2,763
29	1,699	2,045	2,462	2,756
30	1,697	2,042	2,457	2,750
40	1,684	2,021	2,423	2,704
60	1,671	2,000	2,390	2,660
120	1,658	1,980	2,358	2,617
∞	1,645	1,96	2,326	2,576
V	0,90	0,95	0,98	0,99

Ek-C

Tablo C.1. Kritik F değerleri tablosu (α = 0,05)

V1 V2	1	2	3	4	5	6	8	12	24	∞
1	161,45	199,50	215,71	224,58	230,16	233,99	238,88	243,91	249,05	254
2	18,51	19,00	19,16	19,25	19,30	19,33	19,37	19,41	19,45	19,50
3	10,13	9,55	9,28	9,12	9,01	8,94	8,85	8,74	8,64	8,53
4	7,71	6,94	6,59	6,39	6,26	6,16	6,04	5,91	5,77	5,63
5	6,61	5,79	5,41	5,19	5,05	4,95	4,82	4,68	4,53	4,36
6	5,99	5,14	4,76	4,53	4,39	4,28	4,15	4,00	3,84	3,67
7	5,59	4,74	4,35	4,12	3,97	3,87	3,73	3,57	3,41	3,23
8	5,32	4,46	4,07	3,84	3,69	3,58	3,44	3,28	3,12	2,93
9	5,12	4,26	3,86	3,63	3,48	3,37	3,23	3,07	2,90	2,71
10	4,96	4,10	3,71	3,48	3,33	3,22	3,07	2,91	2,74	2,54
11	4,84	3,98	3,59	3,36	3,20	3,09	2,95	2,79	2,61	2,40
12	4,75	3,89	3,49	3,26	3,11	3,00	2,85	2,69	2,51	2,30
13	4,67	3,81	3,41	3,18	3,03	2,92	2,77	2,60	2,42	2,21
14	4,60	3,74	3,34	3,11	2,96	2,85	2,70	2,53	2,35	2,13
15	4,54	3,68	3,29	3,06	2,90	2,79	2,64	2,48	2,29	2,07
16	4,49	3,63	3,24	3,01	2,85	2,74	2,59	2,42	2,24	2,01
17	4,45	3,59	3,20	2,96	2,81	2,70	2,55	2,38	2,19	1,96
18	4,41	3,55	3,16	2,93	2,77	2,66	2,51	2,34	2,15	1,92
19	4,38	3,52	3,13	2,90	2,74	2,63	2,48	2,31	2,11	1,88
20	4,35	3,49	3,10	2,87	2,71	2,60	2,45	2,28	2,08	1,84
21	4,32	3,47	3,07	2,84	2,68	2,57	2,42	2,25	2,05	1,81
22	4,30	3,44	3,05	2,82	2,66	2,55	2,40	2,23	2,03	1,78
23	4,28	3,42	3,03	2,80	2,64	2,53	2,37	2,20	2,01	1,76
24	4,26	3,40	3,01	2,78	2,62	2,51	2,36	2,18	1,98	1,73
25	4,24	3,39	2,99	2,76	2,60	2,49	2,34	2,16	1,96	1,71
26	4,23	3,37	2,98	2,74	2,59	2,47	2,32	2,15	1,95	1,69
27	4,21	3,35	2,96	2,73	2,57	2,46	2,31	2,13	1,93	1,67
28	4,20	3,34	2,95	2,71	2,56	2,45	2,29	2,12	1,91	1,65
29	4,18	3,33	2,93	2,70	2,55	2,43	2,28	2,10	1,90	1,64
30	4,17	3,32	2,92	2,69	2,53	2,42	2,27	2,09	1,89	1,62
40	4,08	3,23	2,84	2,61	2,45	2,34	2,18	2,00	1,79	1,51
60	4,00	3,15	2,76	2,53	2,37	2,25	2,10	1,92	1,70	1,39
120	3,92	3,07	2,68	2,45	2,29	2,18	2,02	1,83	1,61	1,25
∞	3,84	2,99	2,60	2,35	2,21	2,09	1,94	1,75	1,52	1,00

Ek-D

Tablo D.1. Hata kapatma süresi tahminleme regresyon analizi çalışmasında kullanılan veriler

Kapanma Süresi	Önc.	Bildirim Tarihi	Kapanma Tarihi	İş P.	Hata derece	Hata tespit	Test Tipi	Test Seviyesi	Test Yeri	Oluşma Sıklığı
12	Orta	16.06.2016 11:27	28.06.2016 15:50	İP 1	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
12	Orta	16.06.2016 10:32	28.06.2016 11:52	İP 1	Büyük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
12	Orta	16.06.2016 10:26	28.06.2016 11:46	İP 1	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
9	Yüksek	16.06.2016 10:24	25.06.2016 12:01	İP 1	Büyük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
12	Orta	16.06.2016 10:17	28.06.2016 11:59	İP 3	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Genellikle
12	Orta	16.06.2016 09:43	28.06.2016 11:58	İP 1	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Genellikle
17	Orta	16.05.2016 11:57	02.06.2016 09:08	İP 1	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
17	Orta	16.05.2016 10:48	02.06.2016 09:23	İP 2	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Bazen
17	Orta	16.05.2016 10:47	02.06.2016 09:22	İP 1	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
17	Orta	16.05.2016 10:40	02.06.2016 09:20	İP 3	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Genellikle
16	Yüksek	13.05.2016 15:28	29.05.2016 09:26	İP 1	Büyük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
11	Orta	13.05.2016 15:27	24.05.2016 15:27	İP 1	Büyük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Genellikle
12	Yüksek	13.05.2016 15:00	25.05.2016 15:27	İP 1	Büyük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
9	Yüksek	13.05.2016 14:44	22.05.2016 15:27	İP 2	Büyük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
8	Orta	13.05.2016 14:14	21.05.2016 15:27	İP 3	Büyük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
11	Orta	13.05.2016 13:41	24.05.2016 15:27	İP 1	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
20	Orta	13.05.2016 13:36	02.06.2016 09:16	İP 1	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Genellikle
20	Orta	13.05.2016 13:11	02.06.2016 09:16	İP 1	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Bazen
20	Orta	13.05.2016 12:08	02.06.2016 09:15	İP 3	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
10	Orta	08.04.2016 09:41	18.04.2016 14:09	İP 1	Küçük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
10	Orta	08.04.2016 09:40	18.04.2016 13:36	İP 2	Küçük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
3	Yüksek	12.02.2016 18:19	15.02.2016 10:03	İP 2	Katastrofik	Müşteri	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
16	Orta	12.02.2016 18:03	28.02.2016 14:56	İP 3	Küçük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Bazen
4	Yüksek	12.02.2016 17:59	16.02.2016 14:56	İP 3	Katastrofik	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
16	Orta	12.02.2016 17:51	28.02.2016 14:56	İP 3	Küçük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
16	Orta	12.02.2016 17:41	28.02.2016 14:56	İP 2	Küçük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
16	Orta	12.02.2016 17:13	28.02.2016 14:56	İP 2	Küçük	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
6	Yüksek	12.02.2016 15:47	18.02.2016 14:59	İP 3	Kritik	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
7	Yüksek	05.02.2016 12:16	12.02.2016 17:15	İP 3	Kritik	Müşteri	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
7	Yüksek	05.02.2016 11:44	12.02.2016 17:02	İP 3	Kritik	Müşteri	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
13	Orta	05.02.2016 09:43	18.02.2016 15:17	İP 3	Küçük	Çalışan	Performans	Yazılım	Proje Test Ortamı	Genellikle

Tablo D.1.(Devam) Hata kapatma süresi tahminleme regresyon analizi çalışmasında kullanılan veriler

Kapanma Süresi	Önc.	Bildirim Tarihi	Kapanma Tarihi	İş P.	Hata derece	Hata tespit	Test Tipi	Test Seviyesi	Test Yeri	Oluşma Sıklığı
8	Yüksek	12.01.2016 16:41	20.01.2016 12:05	İP 1	Kritik	Çalışan	Performans	Yazılım	Müşteri Ortamı	Genellikle
11	Yüksek	12.01.2016 16:33	23.01.2016 16:33	İP 1	Büyük	Çalışan	Performans	Sistem	Müşteri Ortamı	Bazen
20	Orta	12.01.2016 16:32	01.02.2016 09:41	İP 2	Küçük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
12	Yüksek	12.01.2016 16:31	24.01.2016 16:33	İP 3	Büyük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
13	Orta	12.01.2016 16:26	25.01.2016 16:33	İP 1	Büyük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Bazen
8	Yüksek	12.01.2016 16:25	20.01.2016 16:33	İP 2	Büyük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
20	Orta	12.01.2016 16:24	01.02.2016 09:44	İP 1	Küçük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
7	Yüksek	12.01.2016 16:12	19.01.2016 16:33	İP 2	Büyük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
20	Orta	12.01.2016 16:11	01.02.2016 09:50	İP 3	Küçük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Genellikle
3	Yüksek	12.01.2016 16:10	15.01.2016 16:10	İP 1	Katastrofik	Çalışan	İşlevsel	Sistem	Müşteri Ortamı	Bazen
4	Yüksek	12.01.2016 16:09	16.01.2016 16:10	İP 1	Katastrofik	Çalışan	Performans	Sistem	Müşteri Ortamı	Her Zaman
14	Yüksek	12.01.2016 16:07	26.01.2016 16:33	İP 1	Büyük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
7	Orta	12.01.2016 16:05	19.01.2016 16:05	İP 1	Büyük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
11	Yüksek	12.01.2016 16:03	23.01.2016 16:05	İP 2	Büyük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
10	Orta	12.01.2016 16:00	22.01.2016 16:05	İP 1	Büyük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
20	Orta	12.01.2016 15:56	01.02.2016 09:59	İP 1	Küçük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
20	Orta	12.01.2016 15:41	01.02.2016 10:00	İP 1	Küçük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
21	Orta	11.01.2016 17:01	01.02.2016 10:02	İP 2	Küçük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
21	Orta	11.01.2016 14:36	01.02.2016 10:35	İP 1	Kozmetik	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
3	Yüksek	11.01.2016 12:39	14.01.2016 12:39	İP 1	Katastrofik	Çalışan	Performans	Yazılım	Müşteri Ortamı	Bazen
21	Orta	11.01.2016 12:29	01.02.2016 10:36	İP 1	Küçük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Genellikle
3	Yüksek	11.01.2016 12:28	14.01.2016 12:39	İP 1	Katastrofik	Müşteri	Performans	Yazılım	Müşteri Ortamı	Her Zaman
4	Yüksek	11.01.2016 12:26	15.01.2016 12:39	İP 1	Katastrofik	Müşteri	Performans	Yazılım	Müşteri Ortamı	Her Zaman
6	Yüksek	11.01.2016 12:22	17.01.2016 12:39	İP 1	Katastrofik	Müşteri	Performans	Yazılım	Müşteri Ortamı	Genellikle
8	Yüksek	11.01.2016 12:21	19.01.2016 16:05	İP 1	Büyük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
9	Orta	11.01.2016 12:20	20.01.2016 16:05	İP 2	Büyük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
18	Orta	11.01.2016 12:19	29.01.2016 12:19	İP 1	Kozmetik	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
14	Orta	11.01.2016 12:17	25.01.2016 12:19	İP 1	Küçük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
13	Orta	11.01.2016 12:15	24.01.2016 12:19	İP 1	Küçük	Çalışan	İşlevsel	Yazılım	Müşteri Ortamı	Genellikle
13	Orta	11.01.2016 12:15	24.01.2016 12:19	İP 1	Küçük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
18	Orta	11.01.2016 12:13	29.01.2016 12:19	İP 1	Kozmetik	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
21	Orta	11.01.2016 12:08	01.02.2016 10:52	İP 1	Küçük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman

Tablo D.1.(Devam) Hata kapatma süresi tahminleme regresyon analizi çalışmasında kullanılan veriler

Kapanma Süresi	Önc.	Bildirim Tarihi	Kapanma Tarihi	İş P.	Hata derece	Hata tespit	Test Tipi	Test Seviyesi	Test Yeri	Oluşma Sıklığı
21	Orta	11.01.2016 12:06	01.02.2016 10:53	İP 2	Kozmetik	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
21	Orta	11.01.2016 12:05	01.02.2016 10:53	İP 1	Küçük	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
3	Yüksek	11.01.2016 12:02	14.01.2016 12:02	İP 1	Katastrofik	Müşteri	İşlevsel	Yazılım	Müşteri Ortamı	Her Zaman
4	Yüksek	11.01.2016 11:24	15.01.2016 11:24	İP 1	Katastrofik	Müşteri	Performans	Yazılım	Müşteri Ortamı	Her Zaman
11	Orta	11.01.2016 11:17	22.01.2016 16:05	İP 1	Büyük	Çalışan	Performans	Sistem	Müşteri Ortamı	Her Zaman
14	Orta	05.01.2016 16:04	19.01.2016 12:55	İP 1	Küçük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Genellikle
12	Yüksek	05.01.2016 14:26	17.01.2016 16:05	İP 2	Büyük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Bazen
14	Orta	05.01.2016 14:23	19.01.2016 16:05	İP 1	Büyük	Çalışan	İşlevsel	Sistem	Proje Test Ortamı	Her Zaman
21	Orta	15.12.2015 12:08	05.01.2016 12:35	İP 1	Kozmetik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
11	Orta	07.12.2015 17:25	18.12.2015 15:39	İP 1	Kozmetik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
7	Yüksek	07.12.2015 14:44	14.12.2015 15:38	İP 1	Kritik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Bazen
8	Yüksek	07.12.2015 12:18	15.12.2015 12:18	İP 2	Kritik	Çalışan	Performans	Sistem	Proje Test Ortamı	Her Zaman
7	Yüksek	07.12.2015 11:24	14.12.2015 15:04	İP 2	Kritik	Müşteri	Performans	Sistem	Proje Test Ortamı	Her Zaman
7	Yüksek	07.12.2015 11:19	14.12.2015 16:14	İP 3	Kritik	Müşteri	Performans	Yazılım	Proje Test Ortamı	Her Zaman
16	Orta	07.12.2015 11:19	23.12.2015 15:36	İP 1	Küçük	Çalışan	Performans	Yazılım	Proje Test Ortamı	Genellikle
7	Yüksek	07.12.2015 11:17	14.12.2015 15:50	İP 3	Kritik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Bazen
16	Orta	07.12.2015 11:14	23.12.2015 15:36	İP 1	Küçük	Çalışan	Performans	Sistem	Proje Test Ortamı	Genellikle
15	Orta	02.12.2015 10:20	17.12.2015 12:53	İP 3	Kozmetik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
15	Orta	02.12.2015 09:30	17.12.2015 12:53	İP 3	Kozmetik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
3	Yüksek	27.11.2015 12:27	30.11.2015 11:28	İP 2	Katastrofik	Müşteri	Performans	Yazılım	Proje Test Ortamı	Genellikle
7	Yüksek	27.11.2015 12:24	04.12.2015 16:39	İP 2	Büyük	Müşteri	Performans	Sistem	Proje Test Ortamı	Her Zaman
5	Yüksek	27.11.2015 12:13	02.12.2015 12:46	İP 2	Katastrofik	Müşteri	Performans	Yazılım	Proje Test Ortamı	Her Zaman
4	Yüksek	27.11.2015 12:04	01.12.2015 16:45	İP 2	Kritik	Müşteri	Performans	Yazılım	Proje Test Ortamı	Her Zaman
8	Orta	27.11.2015 12:03	05.12.2015 16:45	İP 2	Küçük	Çalışan	Performans	Yazılım	Proje Test Ortamı	Genellikle
3	Yüksek	27.11.2015 12:03	30.11.2015 11:28	İP 2	Katastrofik	Müşteri	Performans	Yazılım	Proje Test Ortamı	Genellikle
11	Orta	27.11.2015 10:52	08.12.2015 15:36	İP 2	Küçük	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
10	Orta	27.11.2015 10:49	07.12.2015 15:36	İP 2	Küçük	Müşteri	Performans	Sistem	Proje Test Ortamı	Genellikle
9	Orta	27.11.2015 10:47	06.12.2015 14:56	İP 2	Küçük	Müşteri	Performans	Sistem	Proje Test Ortamı	Her Zaman
8	Yüksek	27.11.2015 10:42	05.12.2015 16:32	İP 2	Büyük	Çalışan	Performans	Yazılım	Proje Test Ortamı	Bazen
9	Orta	27.11.2015 10:25	06.12.2015 16:33	İP 2	Büyük	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
6	Yüksek	27.11.2015 10:13	03.12.2015 16:13	İP 2	Büyük	Müşteri	Performans	Yazılım	Proje Test Ortamı	Her Zaman
4	Yüksek	27.11.2015 09:57	01.12.2015 16:13	İP 2	Katastrofik	Müşteri	Performans	Yazılım	Proje Test Ortamı	Her Zaman

Tablo D.1.(Devam) Hata kapatma süresi tahminleme regresyon analizi çalışmasında kullanılan veriler

Kapanma Süresi	Önc.	Bildirim Tarihi	Kapanma Tarihi	İş P.	Hata derece	Hata tespit	Test Tipi	Test Seviyesi	Test Yeri	Oluşma Sıklığı
4	Yüksek	27.11.2015 09:54	01.12.2015 16:13	İP 2	Katastrofik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
4	Yüksek	27.11.2015 09:45	01.12.2015 16:13	İP 2	Kritik	Müşteri	Performans	Yazılım	Proje Test Ortamı	Her Zaman
4	Yüksek	24.11.2015 10:04	28.11.2015 12:54	İP 3	Katastrofik	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
1	Yüksek	09.11.2015 15:52	10.11.2015 17:11	İP 2	Katastrofik	Müşteri	Performans	Yazılım	Müşteri Ortamı	Genellikle
4	Yüksek	09.11.2015 15:52	13.11.2015 10:08	İP 3	Katastrofik	Çalışan	Performans	Yazılım	Müşteri Ortamı	Her Zaman
5	Yüksek	09.11.2015 15:51	14.11.2015 15:39	İP 2	Katastrofik	Müşteri	Performans	Sistem	Müşteri Ortamı	Her Zaman
6	Yüksek	09.11.2015 15:49	15.11.2015 15:24	İP 1	Katastrofik	Müşteri	Performans	Sistem	Müşteri Ortamı	Genellikle
10	Orta	03.11.2015 13:31	13.11.2015 13:31	İP 3	Büyük	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
20	Orta	03.11.2015 11:51	23.11.2015 16:13	İP 3	Kozmetik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
21	Orta	03.11.2015 11:49	24.11.2015 10:01	İP 3	Kozmetik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
3	Yüksek	27.10.2015 13:59	30.10.2015 13:59	İP 1	Kritik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Bazen
17	Orta	27.10.2015 13:50	13.11.2015 13:31	İP 1	Küçük	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
8	Yüksek	27.10.2015 13:46	04.11.2015 13:59	İP 2	Kritik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
5	Yüksek	27.10.2015 13:43	01.11.2015 13:59	İP 1	Kritik	Çalışan	Performans	Sistem	Proje Test Ortamı	Bazen
6	Yüksek	27.10.2015 13:37	02.11.2015 13:59	İP 1	Kritik	Çalışan	Performans	Sistem	Proje Test Ortamı	Her Zaman
5	Yüksek	27.10.2015 13:35	01.11.2015 13:59	İP 1	Kritik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Bazen
7	Yüksek	27.10.2015 13:29	03.11.2015 13:59	İP 1	Kritik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Bazen
19	Orta	27.10.2015 10:10	15.11.2015 10:10	İP 1	Kozmetik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
24	Orta	27.10.2015 10:07	20.11.2015 10:10	İP 1	Kozmetik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
18	Orta	26.10.2015 15:11	13.11.2015 13:31	İP 1	Kozmetik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
18	Orta	26.10.2015 14:46	13.11.2015 13:31	İP 1	Kozmetik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
5	Yüksek	26.10.2015 14:13	31.10.2015 14:13	İP 1	Kritik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
7	Yüksek	26.10.2015 14:08	02.11.2015 14:08	İP 2	Kritik	Çalışan	Performans	Sistem	Proje Test Ortamı	Her Zaman
8	Yüksek	26.10.2015 11:41	03.11.2015 14:08	İP 2	Kritik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman
9	Yüksek	26.10.2015 11:34	04.11.2015 14:08	İP 1	Kritik	Çalışan	Performans	Yazılım	Proje Test Ortamı	Her Zaman

Ek-E Yazılım Kodları

```
return new Promise<any>(resolve => {
        let issueArray: number[] = [];
        let result;
        for (const module in inputs) {
          result = inputs[module].Inputs.find(inpt => inpt.id === "results")
            .value;
          for (const issue of result) {
            if (
              issue.fields.resolutiondate !== undefined &&
              issue.fields.resolutiondate !== null
            ) {
              issueArray.push(
                Time.DateDiff(issue.fields.resolutiondate, issue.fields.created)
              );
            } else {
              try {
                issueArray.push(
                  Time.DateDiff(
                    inputs[module].Inputs.find(inpt => inpt.id === "EmptyDate")
                      .value,
                    issue.fields.created
                  )
                );
              } catch (error) {
              }
            }
          }
        }
```

Şekil E.1. Hataların ortalama açık kalma süresi hesaplama program kodu

```
new Promise<any>(resolve => {
      let issueArray: number[] = [];
      let result;
      for (const module in inputs) {
        result = inputs[module].Inputs.find(inpt => inpt.id === "results")
          .value;
        for (const issue of result) {
          if (
            issue.fields.resolutiondate !== undefined &&
            issue.fields.resolutiondate !== null
          ) {
            issueArray.push(
              Time.DateDiff(issue.fields.resolutiondate, issue.fields.created)
            );
          } else {
            try {
              issueArray.push(
                Time.DateDiff(
                  inputs[module].Inputs.find(inpt => inpt.id === "EmptyDate")
                    .value,
                  issue.fields.created
                )
              );
            } catch (error) {
            }}}}
      try {
        resolve({
          issues: result,
          "Müşteri Tarafından Raporlanan Gereksinim Değişikliği Açık Kalma Süresi": TwoDigit(
            median(issueArray)
          )
        });
      } catch (error) {
        resolve({
          "Müşteri Tarafından Raporlanan Gereksinim Değişikliği Açık Kalma Süresi": 0
        });
      }
    });
  }
```

Şekil E.2. Gereksinim değişikliği açık kalma süresi hesaplama program kodu

```
new Promise<any>(resolve => {
      let issueArray: number[] = [];
      let result;
      for (const module in inputs) {
        result = inputs[module].Inputs.find(inpt => inpt.id === "results")
          .value;
        for (const issue of result) {
          if (
            issue.fields.resolutiondate !== undefined &&
            issue.fields.resolutiondate !== null
          ) {
            issueArray.push(
              Time.DateDiff(issue.fields.resolutiondate, issue.fields.created)
            );
          } else {
            try {
              issueArray.push(
                Time.DateDiff(
                  inputs[module].Inputs.find(inpt => inpt.id === "EmptyDate")
                    .value,
                  issue.fields.created
                )
              );
            } catch (error) {
            }
          }
        }
      }
      try {
        resolve({
          issues: result,
          "Konfigürasyon Uygunsuzluklarının Açık Kalma Süresi": TwoDigit(
            median(issueArray)
          )
        });
      } catch (error) {
        resolve({
          "Konfigürasyon Uygunsuzluklarının Açık Kalma Süresi": 0
        });
      }
    });
  }
```

Şekil E.3. Konfigürasyon uygunsuzlukları açık kalma süresi hesaplama program kodu

```
new Promise<any>(resolve => {
      let openIssues: number = 0;
      let result;
      for (const module in inputs) {
        result = inputs[module].Inputs.find(inpt => inpt.id === "results")
          .value;
        for (const issue of result) {
          if (issue.fields.status.name.indexOf("Kapandı") === -1) {
            openIssues++;
          }
        }
      }
      try {
        resolve({
          issues: result,
          "Konfigürasyon Uygunsuzluklarının Ortalama Açık Kalma Oranı": TwoDigit(
            (openIssues / (result.length))
          )
        });
      } catch (error) {
        resolve({
          "Konfigürasyon Uygunsuzluklarının Ortalama Açık Kalma Oranı": 0
        });
      }
    });
```

Şekil E.4. Konfigürasyon uygunsuzlukları açık kalma oranı hesaplama program kodu

KİŞİSEL YAYIN VE ESERLER

Gün Ö., Kumru P., Aladağ Z., Developing a Model for Measuring Project Performance with Software Life Cycle Process Metrics and Calculating Project Success Score, *Sakarya University Journal of Science*, 2020, **24**(3), 536-554.

Gün Ö., Kumru P. Y., Yazılım Projelerinde Gözden Geçirme Bulgu Yoğunluğunu Etkileyen Faktörlerin Belirlenmesi: Bir Ar-Ge Kurumunda Uygulama Örneği, *Kocaeli Üniversitesi Fen Bilimleri Dergisi*, 2019, **2**(1) , 36-42.

Gün Ö., Kumru P. Y., Yazılım Proje Yaşam Döngüsü Süreç Metrikleri ile Proje Performanslarının Ölçülmesi ve Proje Başarı Puanının Hesaplanmasına Yönelik Bir Model Geliştirilmesi, *13. Ulusal Yazılım Mühendisliği Sempozyumu*, İzmir, Türkiye, 23-25 Eylül 2019.

Şahin Y., Yıldız Kumru P., **Gün Ö.**, Solution Of Constraints Through Thinking Process In Software Code Reviews, *International Conference on Technology, Engineering and Science*, Antalya, Türkiye, 26 - 29 Ekim 2017.

Gün Ö., Akman G., Kumru P. Y., Yazılım Projelerinin Süreç Performanslarının Ölçülmesinde Düşünce Süreçlerinin Kullanılması, *TMMOB Makina Mühendisleri Odası X. Endüstri-İşletme Mühendisliği Kurultayı*, İstanbul, Türkiye, 25-26 Aralık 2015.

Özvural Ö. G., **Gün Ö.**, & Ak E., Etkin Bir Yazılım Süreç Yönetimi İçin Süreç Yönetim Aracı Seçimi, *8. Ulusal Yazılım Mühendisliği Sempozyumu*, Güzelyurt, KKTC, 08-10 Eylül 2014.

Kumru P. Y., **Gün Ö.**, Ar-Ge Faaliyeti Yürüten Bir KOBİ'de Ar-Ge Bilinci ve Müşteri Memnuniyetinin Ölçülmesi, YAEM 34. Ulusal Yöneylem Araştırması ve Endüstri Mühendisliği Kongresi, Bursa, Türkiye, 25-27 Haziran 2014.

Gün Ö., An Intelligent Support for Requirements Management, *International XII. Turkish Symposium on Artificial Intelligence and Neural Networks TAINN*, Çanakkale, Turkey, 02-04 July 2003.

ÖZGEÇMİŞ

06 Nisan 1977 tarihinde Adapazarı'nda doğan Özgür GÜN evli ve iki çocuk babasıdır. 2000 yılında Sakarya Üniversitesi Bilgisayar Mühendisliği bölümünden mezun olmuş, yüksek lisansını aynı bölümde 2003 yılında tamamlamıştır.

2000-2001 yılları arasında Sakarya Üniversitesi Enformatik Bölümünde araştırma görevliliği yapan Özgür GÜN 2001 yılından bu yana TÜBİTAK BİLGEM Gebze Yerleşkesinde çalışmaktadır.

2008 yılında kamu sektöründe bir ilk olarak sistem, yazılım ve donanım alanında CMMI Seviye 3 kalite belgesini alan TÜBİTAK Bilişim Teknolojileri Enstitüsü'nün süreç iyileştirme projesinin yöneticiliğini yapmıştır. En son 2014 yılında yenilenen CMMI Seviye 3 sertifikasyonundan sonra CMMI Seviye 5 süreç iyileştirme ve yaygınlaştırma çalışmalarını yürütmektedir.

20 yılı aşkın bir süredir yazılım mühendisliği alanında kalite yöneticisi ve eğitmen olarak çalışmaktadır. Yazılım kalite değerlendirme, yazılımda ölçme, süreç iyileştirme ve kalite yönetim sistemleri konularında uzmanlığa sahiptir.

Halen Kocaeli Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliğinde doktora çalışmasına devam etmektedir.