



**FARKLI YAZILIM SÜREÇ REFERANS MODELLERİ  
İLE DEĞERLENDİRMELERİ DESTEKLEMELİK İÇİN  
ONTOLOJİ TABANLI BİR ARAÇ**

**Rukiye BAKIRHAN**

**Danışman: Dr. Öğr. Üyesi Ahmet COŞKUNÇAY**

**Yüksek Lisans Tezi**

**Bilgisayar Mühendisliği Ana Bilim Dalı**

**2022**

(Her hakkı saklıdır.)

T.C.  
ATATÜRK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI

**FARKLI YAZILIM SÜREÇ REFERANS MODELLERİ İLE  
DEĞERLENDİRMELERİ DESTEKLEMELİK İÇİN ONTOLOJİ TABANLI BİR ARAÇ**

(An Ontology Based Tool for Supporting Assessments with Different Software Process  
Reference Models)

YÜKSEK LİSANS TEZİ

Rukiye BAKIRHAN

Danışman: Dr. Öğr. Üyesi Ahmet COŞKUNÇAY

Erzurum  
Ocak, 2022

## KABUL VE ONAY TUTANAĐI

Rukiye BAKIRHAN tarafından hazırlanan “FARKLI YAZILIM SÜREÇ REFERANS MODELLERİ İLE DEĐERLENDİRMELERİ DESTEKLEMELİK İÇİN ONTOLOJİ TABANLI BİR ARAÇ” başlıklı çalışması 03 / 01 / 2022 tarihinde yapılan tez savunma sınavı sonucunda başarılı bulunarak jürimiz tarafından Bilgisayar MühendisliĐi Ana Bilim Dalı, Bilgisayar MühendisliĐi Bilim Dalında yüksek lisans tezi olarak kabul edilmiştir.

Jüri Başkanı:	Doç. Dr. Ebru GÖKALP Hacettepe Üniversitesi	Aslı Islak İmzalıdır
Danışman:	Dr. Öğr. Üyesi Ahmet COŞKUNÇAY Atatürk Üniversitesi	Aslı Islak İmzalıdır
Jüri Üyesi:	Dr. Öğr. Üyesi Barış ÖZYER Atatürk Üniversitesi	Aslı Islak İmzalıdır

Enstitü Yönetim Kurulunun  
.../.../.... tarih ve ..... sayılı  
kararı.

Bu tezin Atatürk Üniversitesi Lisansüstü Eğitim ve Öğretim YönetmeliĐi'nin ilgili maddelerinde belirtilen şartları yerine getirdiĐini onaylarım.

**Enstitü Müdürü**

Aslı Islak İmzalıdır

**Not:** Bu tezde kullanılan özgün ve başka kaynaklardan yapılan bildiriş, çizelge, şekil ve fotoğrafların kaynak olarak kullanımı, 5846 sayılı Fikir ve Sanat Eserleri Kanunundaki hükümlere tabidir.

## ETİK BİLDİRİM VE İNTİHAL BEYAN FORMU

Yüksek Lisans Tezi olarak Dr. Öğr. Üyesi Ahmet COŞKUNÇAY danışmanlığında sunulan “Farklı yazılım süreç referans modelleri ile değerlendirmeleri desteklemek için ontoloji tabanlı bir araç” başlıklı çalışmanın tarafımızdan bilimsel etik ilkelere uyularak yazıldığını, yararlanılan eserlerin kaynakçada gösterildiğini, Fen Bilimleri Enstitüsü tarafından belirlenmiş olan Turnitin Programı benzerlik oranlarının aşılmadığını ve aşağıdaki oranlarda olduğunu beyan ederiz.

Tez Bölümleri	Tezin Benzerlik Oranı (%)	Maksimum Oran (%)
Giriş	3	30
Kuramsal Temeller	7	30
Materyal ve Yöntem	3	35
Bulgular	0	20
Tartışma	0	20
Tezin Geneli	4	25

*Not: Yedi kelimeye kadar benzerlikler ile Başlık, Kaynakça, İçindekiler, Teşekkür, Dizin ve Ekler kısımları tarama dışı bırakılabilir. Yukarıdaki azami benzerlik oranları yanında tek bir kaynaktan olan benzerlik oranlarının %5'den büyük olmaması gerekir.*

Beyan edilen bilgilerin doğru olduğunu, aksi halde doğacak hukuki sorumlulukları kabul ve beyan ederiz.

Tez Yazarı (Öğrenci)	Tez Danışmanı
Rukiye BAKIRHAN	Dr. Öğr. Üyesi Ahmet COŞKUNÇAY
3.1.2022	3.1.2022
İmza:	İmza:

\* Tez ile ilgili YÖKTEZ’de yayınlamasına ilişkin bir engelleme var ise aşağıdaki alanı doldurunuz.

Tezle ilgili patent başvurusu yapılması / patent alma sürecinin devam etmesi sebebiyle Enstitü Yönetim Kurulunun .../.../... tarih ve ..... sayılı kararı ile teze erişim 2 (iki) yıl süreyle engellenmiştir.

Enstitü Yönetim Kurulunun .../.../... tarih ve ..... sayılı kararı ile teze erişim 6 (altı) ay süreyle engellenmiştir.

## TEŐEKKÜR

Tezimin konusunun belirlenmesinde, arařtırma ařamasında ve tamamlanma sürecindeki destek ve katkılarından dolayı danıřmanım Sayın Dr. Öğr. Üyesi Ahmet COŐKUNÇAY'a teőekkür ederim. Tezimin bařlangıcından bitimine kadar bana inanan, her zaman yanımda olan, desteklerini esirgemeyen ve motivasyonumu her daim canlı tutan bařta annem olmak üzere tüm aileme teőekkür ederim.

Rukiye BAKIRHAN



## ÖZET

### YÜKSEK LİSANS TEZİ

#### FARKLI YAZILIM SÜREÇ REFERANS MODELLERİ İLE DEĞERLENDİRMELERİ DESTEKLEMELİK İÇİN ONTOLOJİ TABANLI BİR ARAÇ

**Rukiye BAKIRHAN**

**Danışman: Dr. Öğr. Üyesi Ahmet COŞKUNÇAY**

**Amaç:** Bu tezin amacı, organizasyonlar tarafından yaygın olarak kullanılan CMMI, SPICE ve TMMI süreç referans modellerini ortak bir ontoloji üzerinde buluşturarak organizasyonların yazılım süreç değerlendirmesini destekleyen bir araç geliştirmektir.

**Yöntem:** Bu çalışmada ontoloji tabanlı bir süreç değerlendirme aracı geliştirmek için ontolojileri oluşturma programı olarak Protégé programını, geliştirme ortamı olarak Eclipse' i, kavram tabanı olarak OWL ontolojisini, sorgulama dili olarak SPARQL' i, programlama dili olarak Java'yı, ontolojileri yüklemek için ise JENA kütüphanesini kullanmıştır. Geliştirilen araç OSPAT (Ontology-Based Software Process Assessment Tool) ile bir organizasyonda üç farklı modelin süreç değerlendirmesi yapılarak doğrulaması yapılmıştır.

**Bulgular:** Bu araç ile kabul görmüş standart modellerde değerlendirme yapabilme, farklı modellerde değerlendirme yapabilme, basit arayüze sahip olma, ontoloji tabanlı olma ve bu sayede kolay güncellenebilme gibi özellikleri ile organizasyonlara süreç değerlendirmesi yaparken destek sağlamıştır. Bu araç üç modelin eşleştirilerek bir ontolojide buluşabileceğini gösterdi.

**Sonuç:** Bu çalışmada OSPAT adındaki üç farklı modelde değerlendirme yapmaya olanak sağlayan hem manuel hem de otomatik olarak kullanılabilen ontoloji tabanlı yazılım süreç değerlendirme aracı geliştirilmiştir. Bu aracın giriş kısmında hedeflenen özelliklere yeterli ölçüde sahip olduğu kanıtlanmıştır. Ayrıca bu araç organizasyonlara düzenli yapacakları süreç değerlendirmelerinin sonuçlarını karşılaştırma yapma imkânı sağlamıştır.

**Anahtar Kelimeler:** Yazılım Süreç Değerlendirme Aracı, Yazılım Süreç Referans Modelleri, Yazılım Süreç İyileştirme, Anlamsal Ağ, Ontoloji, CMMI, SPICE, TMMI

**Ocak 2022, 81 sayfa**

## ABSTRACT

### MASTER'S THESIS

#### AN ONTOLOGY BASED TOOL FOR SUPPORTING ASSESSMENTS WITH DIFFERENT SOFTWARE PROCESS REFERENCE MODELS

**Rukiye BAKIRHAN**

**Supervisor: Dr. Öğr. Üyesi Ahmet COŞKUNÇAY**

**Purpose:** The aim of this thesis is to develop a tool that supports the overall process evaluation of organizations by bringing together the CMMI, SPICE and TMMI models, which are popularly used by organizations, on a common ontology, by enabling computers to understand the concepts.

**Method:** In this study, to develop an ontology-based process evaluation tool, we used the Protégé program as the ontology creation program, Eclipse as the development environment, OWL ontology as the concept base, SPARQL as the query language, Java as the programming language, and to load the ontologies, it used the JENA library. With the developed tool OSPAT (Ontology-Based Software Process Assessment Tool), in three different models in an organization were verified by making process evaluations.

**Findings:** With this tool, it has provided support to organizations when evaluating processes with its features such as making evaluations in accepted standard models, evaluating in different models, having a simple interface, being ontology-based and thus easy to update. This tool showed that three models can be matched to meet in an ontology.

**Conclusion:** In this study, an ontology-based software process evaluation tool that can be used both manually and automatically, which allows evaluation in three different models called OSPAT, has been developed. This tool has adequately been proven to have targeted features in the introduction. In addition, this tool provided organizations with the opportunity to compare the results of their regular process evaluations.

**Keywords:** Software Process Assessment Tool, Software Process Reference Models, Semantic Web, Ontology, CMMI, SPICE, TMMI

**January 2022, 81 pages**

## İÇİNDEKİLER

KABUL VE ONAY TUTANAĞI.....	i
ETİK BİLDİRİM VE İNTİHAL BEYAN FORMU .....	ii
TEŞEKKÜR .....	iii
ÖZET .....	iv
ABSTRACT .....	v
İÇİNDEKİLER.....	vi
TABLolar DİZİNİ.....	ix
ŞEKİLLER DİZİNİ .....	x
KISALTMALAR VE SİMGELER DİZİNİ.....	xi
GİRİŞ.....	1
KURAMSAL TEMELLER.....	4
Yazılım Süreç Değerlendirmesi .....	4
Yazılım Süreç Değerlendirme Referans Modelleri.....	5
ISO 15504 (SPICE) modeli.....	6
CMMI modeli .....	6
TMMI modeli.....	7
Ontoloji .....	7
Yazılım Süreç Değerlendirme Araçları.....	8
SPiCE-Lite .....	9
CMM-Quest .....	9
SPiCE 1-2-1 .....	10
Appraisal assistant.....	10
SEAL QQ.....	11
CMMiPal v1.0.....	11
Appraisal Wizard .....	11
Genel Literatür Değerlendirmesi.....	11
MATERYAL VE METOT .....	15
Modellerin Seçilmesi .....	15
Ontolojilerin Oluşturulması .....	16
Araçın Geliştirilmesi .....	22
OSPAT(Ontology-Based Software Process Assessment Tool).....	26
Geliştirilen Araç ve Özellikleri .....	26
Araçın Doğrulanması .....	29

ARAŞTIRMA BULGULARI VE TARTIŞMA.....	35
SONUÇLAR VE ÖNERİLER .....	37
KAYNAKLAR.....	39
EKLER .....	42
EK-1. SPICE Süreçleri.....	42
EK-2. CMMI Süreç Alanları.....	44
EK-3. TMMI Süreç Alanları .....	45
EK-4. SPICE-Proje Yönetimi Süreç Değerlendirme Soruları .....	46
EK-5. CMMI-Proje Planlama Süreç Değerlendirme Soruları .....	47
EK-6. CMMI-Gereksinim Yönetimi Süreç Değerlendirme Soruları.....	48
EK-7. TMMI-Test Planlama Süreç Değerlendirme Soruları.....	49
EK-8. CMMI Kavramsal Şeması .....	50
EK-9. TMMI Kavramsal Şeması .....	51
EK-10. Yazılım Gereksinim Analizi Sürecine Ait Notların, İş Ürünlerinin ve Başarı Yüzdelerinin Girildiği Sayfa.....	51
EK-11. OWL Dosyasına Değerlendirmenin Eklenebildiği ve Daha Önce Eklenen Değerlendirmenin Silinebildiği Sayfa.....	52
EK-12. Gereksinim Yönetimi Süreci Arayüz Görüntüsü .....	52
EK-13. Gereksinim Yönetimi Sürecinin Arayüz İle Eklenen OWL Dosyasındaki İş Ürünlerini ve Başarı Yüzdesinin Protégé İle Gösterilmesi.....	53
EK-14. Proje Planlama Sürecinin Arayüz Görüntüsü.....	53
EK-15. Proje Planlama Sürecinin Arayüz İle Eklenen OWL Dosyasındaki İş Ürünlerini ve Başarı Yüzdesinin Protégé İle Gösterilmesi.....	54
EK-16. Proje Yönetimi Süreci Arayüz Görüntüsü.....	55
EK-17. Proje Yönetimi Sürecinin Arayüz İle Eklenen OWL Dosyasındaki İş Ürünlerini ve Başarı Yüzdesinin Protégé İle Gösterilmesi.....	55
EK-18. Test Planlama Sürecinin Arayüz Görüntüsü .....	56
EK-19. Test Planlama Sürecinin Arayüz İle Eklenen OWL Dosyasındaki İş Ürünlerini ve Başarı Yüzdesinin Protégé İle Gösterilmesi.....	56
EK-20. Arayüz İle Eklenen Üç Modelin Birleştirildiği YGO OWL Dosyasındaki İş Ürünlerini ve Başarı Yüzdesinin Protégé İle Gösterilmesi .....	57
EK 21 CMMI Ontolojisinin OWLGrEd İle Görselleştirilmesi.....	58
EK-22. SPICE Ontolojisinin OWLGrEd ile Görselleştirilmesi.....	59
EK-23. TMMI Ontolojisinin OWLGrEd ile Görselleştirilmesi.....	60
EK-24. YGO Ontolojisinin OWLGrEd ile Görselleştirilmesi .....	61

EK-25. CMMI Ontolojisi Oluřturmak İin Yazılan Java Kodu .....	62
ÖZGEMİŐ.....	68



## TABLolar DİZİNİ

<b>Tablo 1.</b> Niteliklerin Notlandırılması.....	6
<b>Tablo 2.</b> Araçların Özellik Tablosu .....	12
<b>Tablo 3.</b> Ontolojilerin Kapsamını Belirlemek İçin Cevapları Aranan Sorular .....	16
Tablo 4. Kavramsal Şemalardaki İlişkilerin Tanımı .....	17
<b>Tablo 5.</b> Modellerin Kavramlarının Eşleştirilmesi .....	21
<b>Tablo 6.</b> SPICE Modeli Yazılım Gereksinim Analizi Sürecinde Yapılan Değerlendirmeye Ait Verilerin Girilmesi Kullanıcı Senaryosu .....	27
<b>Tablo 7.</b> SPICE Modeli Yazılım Gereksinim Analizi Sürecinde Yapılan Değerlendirmeyi OWL Dosyasına Ekleme Kullanıcı Senaryosu .....	28
<b>Tablo 8.</b> Yapılan Değerlendirmeyi Silme Kullanıcı Senaryosu .....	28
<b>Tablo 9.</b> Doğrulama Çalışmasında Kullanılan Modellerin Süreçlerinin Eşleştirilmesi.....	29
<b>Tablo 10.</b> Demografik Sorular .....	30
<b>Tablo 11.</b> SPICE Modeli Yazılım Gereksinim Analizi Soruları.....	30
<b>Tablo 12.</b> Kullanıcı Arayüzü Anket Soruları ve Cevapları.....	34

## ŞEKİLLER DİZİNİ

Şekil 1. İlk çizilen kavramsal SPICE şeması .....	18
Şekil 2. Karşılaştırılan SPICE modeli kavramsal şeması.....	19
Şekil 3. En son çizilen olan SPICE kavramsal şeması .....	20
Şekil 4. YGOM kavramsal şeması .....	22
Şekil 5. YGOM kavramsal şemasına ait bağlı veri grafiği .....	22
Şekil 6. Anasayfa: yeni assessment oluşturma ve model seçim sayfası.....	23
Şekil 7. Assessment bireyi ekleme ve bireye ait bilgileri girme sayfası .....	23
Şekil 8. Seçilen sürece ait verileri girme sayfası.....	24
Şekil 9. SPARQL ile OWL dosyasından çekilen süreçlerin Sorgu kodu.....	25
Şekil 10. SPARQL ile OWL dosyasından çekilen Süreç listesinin Arayüz görüntüsü.....	25
Şekil 11. Yazılım gereksinim analizi sürecinin arayüzü .....	31
Şekil 12. Değerlendirme sonrası Yazılım Gereksinim Analizi sürecinin Protégé editöründeki OWL dosyasına eklenen bireylerin ve özelliklerin gösterimi .....	32

## KISALTMALAR VE SİMGELER DİZİNİ

- CMMI** : The Capability Maturity Model Integration (Yetenek Olgunluk Model Entegrasyonu)
- TMMI** : The Test Maturity Model Integration (Test Olgunluk Model Entegrasyonu)
- SPICE** : Software Process Improvement and Capability Determination (Yazılım Süreç geliştirme yetenek düzeyi belirlenmesi)
- OSPAT** : Ontology-Based Software Process Assessment Tool (Ontoloji Tabanlı Yazılım Süreç Değerlendirme Aracı)
- YGOM** : Yeni Genel Ontoloji Model



## GİRİŞ

Yaklaşık 30 yıldır kurumlar, mevcut süreçlerinin değerlendirmelerini yapmak için CMMI ve SPICE gibi yazılım süreç referans modellerini yaygın olarak kullanmaktadır. Süreç değerlendirme, süreç iyileştirme faaliyetleri için temel adımdır. Süreç değerlendirme işlemi kurumlardaki yazılım süreçlerinin uygulanmasında güçlü, zayıf ve / veya eksik noktaları detaylı olarak araştırır (ISO 2004). Bu değerlendirme sayesinde kurumlar yürütülen sürecin mevcut durumu hakkında bilgi edinirler ve kurumlara süreçlerinin yetkinliği ve olgunluğu hakkında derecelendirme vermek için süreç referans modellerini kullanırlar. Ontolojiler ise bir alana ait çeşitli kavramları ve bu kavramlar arasındaki ilişkileri formel ve açık bir şekilde tanımlamaktadır. Ayrıca bilginin paylaşılması, tekrar kullanımı ve zamandan kazanç gibi faktörler açısından uzmanlara kolaylık sağlamaktadır. Ontolojilerin kullanımı, yazılım süreç değerlendirmesinde de bilgisayar destekli yapıyı sağlamak için uygun bir çözüm olabilir (Feloni and Braga 2015). Ontolojiler, süreç referans modelleri ve kuruluşun süreçlerini temsil etmenin yanı sıra, makine tarafından anlaşılabilir bir dilde ifade edildikleri için aralarındaki eşlemeleri belirlemede ve sürdürmede bir araç sağlamaktadır (Mora *et al.* 2010).

Kurumlar süreç referans modelleri ile süreç değerlendirmelerini yaparak süreçlerini iyileştirmezlerse yazılım projelerini müşterilere planlanan teslim tarihinde ve beklenen kalitede sunmakta zorlanırlar. Bu sebeple projelerindeki adımların verimliliğini ölçmek isterler. Organizasyonların yazılım projeleri süreçlerini uluslararası düzeyde kabul görmüş çerçeve modeller ile değerlendirmesi, süreçlerinin sahip olduğu eksiklikleri görmelerini sağlar. Süreç değerlendirmelerini yaptıktan sonra gözlemledikleri sonuçlara göre süreç iyileştirilmelerini yapabilirler. Bunların yanı sıra, organizasyonlara projelerin süreç değerlendirmelerini yapabilmeleri için zaman, çaba ve kalifiye personel gerekir. Bir araç desteği olmadan bir kurumun süreçlerini birden fazla süreç referans modeline uygun olarak değerlendirmesi, kuruluşun süreçlerinin her bir modele uygunluğunu izlemek ve aralarındaki sapmayı anlamayı daha zor hale getirmektedir (Gazel *et al.* 2009). Organizasyonların farklı modeller ile değerlendirme yaparken harcadıkları çabayı azaltmak için kullanımı kolay ve genel değerlendirme yapabilen bir araç desteğine ihtiyacı vardır. Bu araç ile daha kısa sürede ve herhangi bir harici danışmana ihtiyaç duyulmadan değerlendirmelerin yapılması sağlanmış olur.

Web’ de bilgiye erişim sunan arama motorları günümüzde kullanılan temel araçlardır. Web’ de yazılışları aynı olan fakat anlamları farklı olan konuyla ilgisi olmayan birçok terim bulunduğundan dolayı bu araçlar bilgi yönetiminde sıkıntı yaşamaktadır. Belgelere ulaşmak, hem zaman hem de enerji kaybına neden olur. Buna karşın Semantik Web’ in gerçekleştirilmesi için kullanılan Ontolojiler ile, aynı anlama gelip farklı terimlerle ifade edilen bilgilere erişimi engeller. Ontolojilerde anahtar kelimelerin girilmesi ile bilgi arama sürecinde kişi ve kurumlara konuyla doğrudan alakalı bilgilere ulaşmasında kolaylık sağlanır. Ayrıca semantik web sayesinde verilere makinelerin mevcut iyi tanımlanmış ontolojilerin desteği ile verilerin anlamına erişmesi ve böylece ilgili problemleri çözebilme yeteneği verilir(Türkyılmaz and Yaşar 2008). Bunun yanında araştırmacıların ontolojileri sıfırdan oluşturmak yerine literatüre önceden eklenen ontolojileri kullanabilmeleri sayesinde zamandan kazanç sağlanır. Semantik web teknolojileri sahip olduğu bu özellikler sayesinde yukarıda bahsedilen olumsuzluklara ve zayıflıklara çözüm olarak düşünülmektedir.

Literatüre bakıldığında tekrar kullanılmak veya kaynak olarak faydalanılmak üzere CMMI, SPICE ve TMMI modellerinde ontolojiler bulunmaktadır. Tek bir modelde süreç değerlendirmesini destekleyen araçlar da bulunmaktadır. Ancak literatürde hem ontoloji tabanlı hem de CMMI, SPICE ve TMMI modellerinde süreç değerlendirmesine destek olan araç bulunmamaktadır. Bu modellerle ilgili ön çalışmalar yapılmış fakat geliştirilmiş bir araç yoktur. Bu tezin amacı, özellikle CMMI, SPICE ve TMMI modellerini ortak bir ontoloji üzerinde buluşturarak organizasyonların genel süreç değerlendirmesini destekleyen bir araç geliştirmektir. Üç modelin eşleştirilerek ortak bir ontolojide buluşturulması sayesinde aynı organizasyona ait farklı modellerle yapılan değerlendirme sonuçlarının karşılaştırılması hedeflenmiştir. Oluşturulan ontolojiler ise yazılım süreç değerlendirme alanında çalışan insanlar için modellerin öğrenilmesi, kavramların aralarındaki ilişkilerin daha kolay anlaşılması ve modellerin kendi aralarındaki benzerlik ve farklılıkların belirlenmesi için referans olacaktır. Bunların dışında bu ontolojiyi ve aracı oluştururken referans model tabanlı süreç değerlendirmesi yapan tüm kurumlara uygulanabilir olması hedeflenmiştir.

Yukarıda bahsedilenlere göre bu çalışmada belirlenen araştırma soruları aşağıdadır.

1. Birden fazla süreç değerlendirme referans modellerini bir araya getirecek bir ontoloji neleri içermelidir?

Ontolojilerin biçimsel, makineler tarafından anlaşılabilir ve paylaşılabilir bir dil ile ifade edilmesi gerekir (Gazel 2009). Modelleri ve modellere ait kavramları sorgulayabilmesi için verileri sorgulama özelliğine sahip olması gerekir. Üç modele ait ontoloji bir OWL dosyasında tutulması gerekir. Üç modeli eşleştirmek için ontolojilerin eşleştirilebilir yapıda olması gerekir.

Ayrıca kullanıcıların modelleri daha iyi anlayabilmesi için ontolojilerin görselleştirilmesi gerekir.

2. Birden fazla süreç referans modeli ile süreç değerlendirmesi yapabilen aracı geliştirmenin avantajları nelerdir? Bu aracın özellikleri neler olmalıdır?

Kaliteli bir yazılım süreç değerlendirme aracı bütüncül bir meta model ile yeni bir model tanımlamaya olanak sağlamalıdır (Yürüm, Top *et al.* 2014). Süreç değerlendirme aracının birçok süreç referans modeli ile değerlendirme yapmaya uygun olması süreç değerlendirme aracının organizasyonlara uygun olan modeli kullanma açısından daha faydalı olacaktır. Yani yazılım süreç değerlendirme araçlarının farklı modelleri desteklemede esnek olup olmadığı önemlidir. Çünkü organizasyonlar genelde tek modeli destekleyen süreç değerlendirme araçlarındansa herhangi bir süreç referans çerçevesini destekleyebilecek bir araca ihtiyaç duyarlar. Birden fazla model ile değerlendirme yapabilen araçlar değerlendirme sonuçlarını karşılaştırma imkânı sağlar. Araç ile organizasyonlara farklı süreç referans modelleri ile süreçlerinin değerlendirmelerini yaparak her model için hangi seviyelerde olduğunu görme imkânı sağlanır. Özetle bu araç kabul görmüş standart modellerde değerlendirme yapabilme, farklı modellerde değerlendirme yapabilme, basit arayüze sahip olma, ontoloji tabanlı olma ve bu sayede kolay güncellenebilme gibi özelliklere sahip olmalıdır.

Bütün bu sorular ve ihtiyaçlar doğrultusunda organizasyonlar tarafından popüler olarak kullanılan üç farklı modeli ortak bir ontolojide buluşturarak ontoloji tabanlı genel bir süreç değerlendirme aracını geliştirmek amaçlanmıştır.

Bu tez çalışması beş bölümden oluşmaktadır. İlk bölümde tez konusuna benzer çalışmalar incelenmiştir. İkinci bölümde aracı geliştirmek için kullanılan yöntem ve uygulanan adımlardan bahsedilmiştir Üçüncü bölümde araç, aracın özellikleri ve aracın doğrulanması anlatılmıştır. Dördüncü bölümde geliştirilen araç sayesinde ortaya çıkan bulgulara yer verilmiştir. Son bölümde ise çalışma sonrası elde edilen sonuçlar ve araçtaki eksiklikler anlatılmıştır.

## KURAMSAL TEMELLER

Bu bölüm kurumların projelerinin yazılım süreçlerini neden değerlendirmeleri gerektiğini açıklamasının yanında değerlendirme yapmak için yazılım süreç referans modellerinin de gerekliliği ile alakalı bilgileri içerir. Bunların dışında, ontoloji kavramı ile ilgili bilgiler verilerek mevcut çalışmalar açıklanmıştır. Son olarak süreç değerlendirmesi yapan araçlar detaylı incelenerek sahip oldukları özelliklere göre karşılaştırılmıştır.

### Yazılım Süreç Değerlendirmesi

Yazılım süreçleri, yazılım projelerinde geliştiricilere rehberlik eden çeşitli faaliyetleri, yöntemleri ve uygulamaları kapsar (Kitson and Masters 1993). Yazılım geliştirme organizasyonları, yazılım süreçlerinin belirlenen modellere uygun olarak gerçekleşmesini sağlamak için süreç değerlendirmesi yapma ihtiyacı duyarlar. Bu sayede bu organizasyonlar değerlendirme sonuçlarına göre süreçlerinin kalite düzeylerini gözlemlemiş olur. Kalite düzeylerine göre süreçlerinde yaptıkları uygulamaları değiştirerek iyileştirmeyi sağlamak isteyebilirler. Mevcut durumun nesnel ölçütlerle ve olabildiğince nicel değerlendirilmesi doğru iyileştirmenin yapılması için önemlidir (Aytaç 2008). Ayrıca süreçlerin bu nicel standartlarla değerlendirilmesi projenin daha kolay yönetilmesi sağlar. Yani değerlendirme neticesinde, yazılım organizasyonlarının önemli sorunları ve iyileştirme öncelikleri belirlenerek gelişmeleri sağlanmış olur (Lepasaar and Makinen 2002). Ayrıca yazılım organizasyonlarının süreçlerini en iyi şekilde değerlendirmek için belirli özelliklere sahip olmaları gerekmektedir. Aşağıdaki listede (Rout 2003) verilen süreç değerlendirmesi için olması gereken standartlar, Rout tarafından yapılan anket çalışmalarından faydalanılarak oluşturulmuştur.

Standart süreç değerlendirmesi için gereksinimler aşağıda listelenmiştir:

- Değerlendirme güvenilir ve tutarlı bir değerlendirme performansına sahip olmalı.
- Herhangi bir yazılım geliştirme organizasyonunda kullanılabilir olmalı.
- Süreç iyileştirmesini sağlamak için değerlendirme sonuçlarını kullanılabilir olmalı.
- Satın alma ve geliştirmedeki risklerin değerlendirilmesi için süreç yeterliliğinin belirlenmesinde değerlendirme sonuçlarını kullanılabilmesi.
- Yazılım mühendisliğindeki mevcut en iyi uygulamaları yansıtmalı.
- Hem projeleri hem de organizasyonları değerlendirebilmeli.
- Kendisinden beklenen yeterlilik hakkında bilgi içermeli.

- Değerlendirmeler, tercihen bağımsız olarak doğrulanabilmeli ve kurum içinde gerçekleştirilmeli.

Buna göre, iyi bir yazılım süreç değerlendirmesinin yapılabilmesi için değerlendirmenin güvenilir ve tutarlı bir değerlendirme performansına sahip olması beklenmelidir. Bir süreç değerlendirmesi standardı, uygulama alanı, iş ihtiyaçları veya organizasyonun büyüklüğü fark etmeksizin, yazılım geliştirme ile ilgili herhangi bir organizasyonda kullanılabilir olması gerekir. Bu gereksinimler süreçlerin standartlaşması açısından önemlidir.

### **Yazılım Süreç Değerlendirme Referans Modelleri**

Kurumlar standart yazılım süreç referans modelleri ile süreçlerinin hangi seviyede olduğunu belirlemek için bazı metotlarla gözlem yaparlar. Süreç modelleri disiplinli ve kapsamlı uygulamalara sahip olduğundan kurumların bu modelleri kullanması ile projenin istenilen zamanda ortaya çıkması sağlanır. Ayrıca yazılım süreçlerinin üretkenliği ve verimliliği iyileştirildiğinde, organizasyon yüksek kaliteli ürünler geliştirebilir, sürdürebilir, teslim edebilir, iş hedeflerini karşılayabilir ve daha yüksek bir müşteri memnuniyeti elde edebilir (Mesquida *et al.* 2012).

Literatürdeki var olan CMMI, ISO 15504 ve ISO 12207 modelleri temel alınarak oluşturulmuş Brazilian Süreç Değerlendirme modeli, Brezilyadaki küçük ve orta ölçekli işletmelerin süreçlerinin değerlendirmesini ve iyileştirmesini sağlar (Rocha *et al.* 2005). Bu modelde yedi tane süreç olgunluk seviyesi bulunmaktadır. Medi SPICE modeli, medical endüstrisinde ve sağlık teknolojisinde özel gereksinimlerini karşılamak için geliştirilmekte olan bir yazılım süreci değerlendirme ve iyileştirme modelidir. Medi SPICE, SPICE modelini temel alarak tıbbi cihaz düzenlemeleri ve standartlarını kapsar (Casey and McCaffery 2012). Automotive SPICE Süreç Değerlendirme Modeli ise, ISO/IEC 15504 gerekliliklerine uygun olarak otomotiv sistemlerinin geliştirilmesinde yazılım süreç kapasitesinin uygun değerlendirmelerini yapmak için kullanılır (Messnarz *et al.* 2009).

Bu çalışmada kullanılan CMMI, SPICE ve TMMI modelleri süreç, süreç tipine ve yetenek düzeyine göre ayrılabilirler. Bu gruplamalar yazılım projelerinin ve süreçlerinin çalışanlar tarafından daha iyi anlaşılmasını sağlar. Denetim sırasında modellerdeki her süreç ayrı ayrı değerlendirilir. İlgili sürecin temel uygulamalarının başarı düzeylerini değerlendirmek için Tablo 1'deki dört puandan biri verilir.

**Tablo 1.** Niteliklerin Notlandırılması<sup>1</sup>

%0-15	Hiç başarılmamış (Not Achieved)	İş çıktıları yok, deliller yetersiz veya gösterilen içerik kabul edilmemiş
%16-50	Az kısmı başarılmış (Partially Achieved)	Bazı istenilen iş çıktıları var fakat amaçlandığı gibi değil.
%51-85	Büyük kısmı başarılmış (Largelly Achieved)	Sistematik bir yaklaşım olmasına rağmen istenilen çıktıları başarmada küçük eksiklikler olabiliyor.
%86-100	Tamamı başarılmış (Fully Achieved)	Tam ve sistematik bir yaklaşım var. İstenilen iş çıktılarına elde edememe gibi bir risk

Aşağıdaki kısımda bu çalışma için önemli olan yazılım süreç referans modelleri ayrı başlıklar altında detaylı olarak açıklanmıştır.

### **ISO 15504 (SPICE) modeli**

Yaklaşık 20 yıl önce Uluslararası bir standart olarak ortaya çıkan SPICE (Software-Process Improvement and Capability dEtermination- Yazılım Süreç İyileştirme ve Yetenek Belirleme) ISO ve IEC tarafından geliştirilmiştir. ISO 15504 modelinin avantajlarından biri küçük, orta ve büyük ölçekteki her türlü projeye, organizasyona ve endüstriye kolaylıkla uyarlanabilir. Süreçleri kategorize etmesi sayesinde değerlendirmeleri gerçekleştirmek için gereksinimlerin anlaşılması, izlenmesi ve değerlendirme sonuçların anlaşılması kolaydır (Rout 2003). Bu model sahip olduğu uygulamalar sayesinde süreç içerisindeki iyileşmeyi ve gelişmeyi ölçebilir. ISO 15504, herhangi bir belgelendirmeye yönelik sertifikasyon amacı taşımayan bir süreç standardıdır (Yücalar 2006). Ekler kısmında (15504 2003) SPICE modeline ait 48 standart sürece yer verilmiştir. Bu süreçler 9 farklı kategoride gruplandırılmıştır.

### **CMMI modeli**

CMMI, kurum yazılım süreçlerinin, ürünlerin ve hizmetlerin geliştirilmesi ve sürdürülmesi için kapsamlı bir çözüm sunar (Chrissis *et al.* 2011). Geliştirmeye yönelik tasarımdan bakıma kadar ürün yaşam döngüsünü kapsayan ve en iyi uygulamalardan oluşan CMMI organizasyonların süreçlerini iyileştirmelerine yardımcı olur (Team 2006). Çeşitli yazılım kuruluşları CMMI modelini yazılım geliştirme süreçlerinin olgunluğunu değerlendirmek için kullanarak değerlendirmenin sonuçlarını da sonrasında süreç iyileştirme için analiz eder (Soydan and Kokar 2006).

<sup>1</sup> <https://www.tse.org.tr/IcerikDetay?ID=944&ParentID=1212>

CMMI modeli süreç iyileştirmeye yönelik hem aşamalı hem de sürekli gösterime sahiptir. Her temsilin değerine göre avantajları olması organizasyonlar için kendilerine uygun gösterimi kullanma seçeneği sunar(Team 2006). Ekler kısmında CMMI modelinin süreç alanları kategorilerine ayrılmıştır(Mollaoğlu and Bolatan 2020).

### **TMMI modeli**

TMMI, organizasyonların test süreçlerini değerlendirmek için TMMI Foundation tarafından geliştirilen bir modeldir. Birçok süreç referans modelinin geliştirilmesinden sonra 2000 li yıllarda test odaklı bir test değerlendirme modeline ihtiyaç duyulmuştur (Garousi and van Veenendaal 2021). TMMI modeline ait olgunluk düzeyi, süreç alanı, hedef ve uygulama gibi kavramlar CMMI modelinden aktarılmıştır (Çiflikli and Coşkunçay 2018).

Ekler kısmındaki Tablo 4'te(van Veenendaal 2018) TMMI modelinin olgunluk düzeylerine göre gruplandırılmış süreç alanları gösterilmiştir.

### **Ontoloji**

Ontoloji, bir alandaki kavramlar dizisi ve bu kavramlar arasındaki ilişkiler ile bilginin soyut modelidir. Ontolojinin bilgisayar bilimleri açısından en çok kabul gören tanımı “Ontoloji, kavramsallaştırmanın açıkça belirtilmesidir.” (Gruber 1993). Günümüzde birçok farklı alandaki uzmanların birbirlerine daha kısa sürede bilgilerini paylaşması için ortak bir sözlük gibi kullanabilecekleri standartlaşmış ontolojiler geliştirilmektedir. Bu araştırmacılar için ortak bir kelime dağarcığı olan ontolojiler, alandaki temel kavramların makine tarafından yorumlanabilir tanımlarını ve bunlar arasındaki ilişkileri içerir (Noy and McGuinness 2001). Böylece bazı kaynaklar hızlı bir şekilde diğer uzmanlar tarafından farklı alanlar için de tekrar kullanılabilir. Ayrıca bu ontolojiler diğer uzmanlar tarafından kolaylıkla analiz edilebilir.

Ontolojiler, belirli bir alanın bilgisini ve Bilgisayar Bilimi alanında akıllı sistemler üretmek için kullanılırlar (Athiththan *et al.* 2018). Özellikle son zamanlarda zeki sistemler, e-ticaret, doğal dil işleme ve bilgi yönetimi gibi birçok çeşitli alanda kullanılmaktadır (Türkyılmaz and Yaşar 2008). Bununla birlikte kurumların ontolojileri kullanımı, yazılım süreç değerlendirmesinde de bilgisayar destekli yapıyı sağlamak için bir çözümdür. Ontolojileri oluşturmak için birkaç uygulanabilir alternatif model arasında, öngörülen görev için hangisinin daha iyi çalışacağını, daha sezgisel, daha genişletilebilir ve daha sürdürülebilir olacağını belirleyip ona göre seçim yapılması gerekir. Ontolojinin ilk sürümü tanımlandıktan sonra, uygulamalarda veya problem çözme yöntemlerinde kullanılarak veya alandaki uzmanlarla tartışılarak eksikliklere göre güncellenebilir. Sonuç olarak, çoğunlukla ilk ontolojiyi gözden

geçirmek gerekebilir. Bu yinelemeli tasarım sürecinin ontolojinin tüm yaşam döngüsü boyunca devam etmesi önerilir.

Tarhan ve arkadaşları, Yazılım süreç değerlendirmelerinde ontolojilerin kullanımı üzerinde sistematik bir literatür incelemesi olan bir makale oluşturmuşlardır. Ontolojilerin kullanımı ve faydalarını anlamak amacıyla yaptıkları bu çalışmada, en çok bilinen dijital kütüphaneleri araştırarak 14 ontoloji çalışmasını incelemişlerdir. Seçilen çalışmaları hedeflenen yazılım süreçleri, araştırma yönü, kullanılan süreç iyileştirme modeli, kullanılan süreç referans modeli ve ontoloji temsil dili gibi birçok konu açısından analiz etmişlerdir. Bu çalışmanın yazılım süreç değerlendirmedeki ontolojilerin kullanımı üzerine yapılan ilk literatür çalışması olduğunu söylemişlerdir (Tarhan and Giray 2017).

Liao ve arkadaşları, Bir yazılım süreç ontolojisi ve uygulaması adlı makalede yazılım süreç ontoloji(SPO) olarak adlandırılan yazılım süreçleri için OWL tabanlı bir ontoloji tasarlamışlardır. SPO'nun anlamsal açıdan zengin, yeniden kullanılabilir ve iyi bir genişletilebilirliğe sahip olmasını istemişlerdir. Web tabanlı süreç değerlendirmesinin bir prototipi olan bir aracı yaklaşımın avantajlarını göstermek için geliştirmişlerdir. Yazılım süreç ontolojilerinin ana bileşeni olarak birleşik bir atomik uygulamalar kümesi tasarlamışlardır. Ayrıca iki modelin süreçlerinin isimleri aynı olmasa da içeriklerinin büyük ölçüde benzer olduklarını sunmuşlardır. Böylece bu makale farklı referans modellerinin bir arada yer alabileceğini göstermiştir (Liao and Leung 2014).

Grambow ve arkadaşları, Yazılım Mühendisliğinde Otomatik Süreç Değerlendirmesine Doğru adlı makalede süreç değerlendirmesini otomatik olarak yapma yeteneği sağlamak için süreç odaklı pratik ve değişken bir yaklaşım sunmuşlardır. Yazılım mühendisliği süreç desteğini ve otomatik bir şekilde rehberliği geliştirmek için Bağlam Bilinçli Yazılım Mühendisliği Ortamı Olay Güdümlü Çerçeveyi(CoSEEEK) geliştirmişlerdir. Bu yaklaşımı özellikle CMMI, ISO/IEC 15504 ve ISO 9001 modelleri için uygun olarak oluşturmuşlardır. Bu 3 süreç modelinin ontolojide nasıl birleştiğini göstermişlerdir (Grambow *et al.* 2012).

### **Yazılım Süreç Değerlendirme Araçları**

Yazılım süreç referans modelleri ile yapılan değerlendirmelerden kaynaklanan çıktılar hacimli ve karmaşık olabilir. Bunları basitleştirmek, kolay anlamak ve analiz etmek için araç desteği gerekebilir (Hunter *et al.* 1997). Bu araçlar, yazılım geliştirme organizasyonlarına proje süreçlerini hızlı ve doğru bir şekilde değerlendirmek için avantaj sağlar. Aynı zamanda bu araçlar ile değerlendirmenin sonuçlarını ve elde edilen verileri otomatik olarak analiz edip raporlayabilirler. Kurumun değerlendirilen sürecinin seviyesini belirleme konusunda destek

olurlar. Böylece organizasyonların bu araçları kullanmaları değerlendirdikleri sürecin güçlü ve zayıf yönlerini hızlı bir şekilde anlamalarına yardımcı olur.

Bu bilgisayar destekli araçlar, kullanması daha ucuz, değişimi, güncellenmesi kolay ve kurumların harici danışmanları işe almaya ihtiyaç olmadan kullanılabilmesi için üretilir (Soydan and Kokar 2006). Bu araçların temel amacı, kurumların değerlendirme için kullandığı harcamaları en aza indirmek ve değerlendirme raporlarının güvenilirliğini artırmak için değerlendirmeye yardımcı olmaktır. Ayrıca değerlendirme yapılırken harcanan zamanı azaltmak ve farklı modellerden süreç değerlendirme yaparak seviye belirlemek amacıyla bu süreç değerlendirme araçları geliştirilmiştir.

Süreç değerlendirmesinin veri toplama aşamasını desteklemek ve kurumların yazılım süreçlerinin süreç referans modellerine uygunluğunu izlemek için yazılım süreç değerlendirme araçları kullanılmıştır (Gazel *et al.* 2012). Bu araçlar yapılan değerlendirmelerin sonucunda süreç iyileştirmeye destek verirler. Süreç olgunluk değerlendirmesi sırasında süreci denetleyen kişileri doğru yönlendirerek daha güvenilir bir değerlendirme yapılmasına sebep olurlar. Bunların yanında bu araçlara esneklik özelliği eklenmesiyle yeni geliştirilen ve güncellenen modeller için de kullanılmasına yol açılabilir. Ayrıca bu araçların çoğu ticari araçlardır (Garcia and Pacheco 2009).

Bu kısımda literatürde ve piyasada olan 7 tane yazılım süreç geliştirme aracından ve onların özelliklerinden bahsedeceğiz (Gazel *et al.* 2012).

### **SPiCE-Lite**

SPiCE-Lite<sup>2</sup> kurumsal süreçlerin ISO / IEC 15504 gerekliliklerine uygunluğunu değerlendirmek için geliştirilmiştir. Her büyüklükteki yazılım şirketleri ve çok küçük takımlar için kullanıma uygundur. Sonuç, çizelgeler şeklinde kapsamlı değerlendirmeler ve analizler içermektedir. İstenirse, bir Web Raporu oluşturulabilir. Derecelendirme öz değerlendirme şeklinde organizasyondan çalışan biri ya da bir grup tarafından yapılabilir veya bir danışman yardımıyla yapılabilir. Ayrıca dışarıdan eğitilmiş, deneyimli değerlendiriciler tarafından yapılması sağlanabilir.

### **CMM-Quest**

iCMM-Quest<sup>3</sup>, yazılım kuruluşları için hızlı ve verimli şekilde çalışan bir öz değerlendirme aracıdır. Sadece CMMI-Dev v1.3 tabanlı süreç değerlendirmelerini desteklemek

---

<sup>2</sup> <http://www.spicelite.com/cms/en/>

<sup>3</sup> <http://www.cmm-quest.com/cms/en/>

üzere geliştirilmiştir. Yani farklı bir süreç referans modeli tanımlanmasına izin verilmemektedir. Bu araç süreç değerlendirmesinde hazırlık olarak seçmek için işlevler, veri toplama için metin ekranlar, veri analizleri için grafikler ve MS Word ve HTML formatında raporlama sağlar. Bu araç sayesinde alınan gerekli tüm girdiler analiz edilir ve otomatik raporlanır.

### **SPiCE 1-2-1**

SPiCE 1-2-1<sup>4</sup>, kurumlardaki süreçlerin sadece SPiCE modeli özelinde değerlendirme yapmak için geliştirilmiştir. Yeni ve ya farklı bir süreç referans modeli tanımlanmasına izin vermemektedir. Değerlendirme verileri bir veri tabanında saklanır. Ayrıca kullanıcılar bu aracı kullanmaya çalıştıklarında orta derecede zorluk yaşarlar. Buna rağmen bu araç otomatik raporlama yapmayı ve denetçiyi yönlendirme özelliklerini barındırır.

### **Appraisal assistant**

APPRAİSAL ASSISTANT<sup>5</sup> Aracı Griffith Üniversitesi Software Quality Institute tarafından kurumsal olgunluk ve süreç derecelerini belirleyen bir süreç değerlendirme aracı olarak geliştirilmiştir. Araç, SPiCE ve CMMI süreç referans modelleri üzerinden değerlendirme yapabilmektedir. Ayrıca bu araç, CMMI V1.2, CMMI V1.3, ISO / IEC 12207 ve diğerleri dâhil olmak üzere çoklu referans modellerinde kullanılabilir (Tayamanon *et al.* 2011). Bunun yanında bu araç sayesinde, Değerlendirme kaydı, değerlendirme, değerlendirme bulguları, güçlü ve zayıf yanları, derecelendirme dosyaları ve iş yükü özetleri de dâhil olmak üzere çeşitli raporların otomatik olarak oluşturulması gerçekleştirilir. Bu araçta süreç referans modelleri ve eşleme verileri bir Microsoft Access veri tabanında depolanır. Modelde bulunan elemanların içerikleri değiştirilebilir. Fakat Modelde bulunan kavramların isimleri aynı kalmak zorundadır. Fakat Araç, Kullanıcılar yeni bir model tanımlanırken yetenek ve olgunluk seviyelerinin sayısının belirleyemezler.

Bu aracın avantajı bu araç sayesinde amaçların ve süreçlerin sonuç profili, değerlendirmenin güçlü ve iyileştirmeye açık yönleri, değerlendirme bulguları, kanıtlar, değerlendirme girdileri ayrı şekilde raporlanabilmektedir. Bu aracın diğer araçlardan farklı olmasının sebeplerinden biri de farklı projelerde değerlendirme yapmaya olanak sağlamasıdır.

---

<sup>4</sup> <http://www.spice121.com/cms/en/>

<sup>5</sup> <https://www.sqi.griffith.edu.au/AppraisalAssistant/indexFrameset.html>

## SEAL QQ

SEAL<sup>6</sup> tarafından geliştirilen ve bir yazılım değerlendirme aracı olan SEAL QQ, Microsoft Windows 3.1 \ 95 altında çalışan, bir veri tabanı aracılığıyla oluşturulan ayrıntılı raporların hızlı bir şekilde oluşturulmasını içeren grafiksel bir uygulamadır (Walker and Lok 1995). Bu araç, diğer araçlardan farklı olarak kullanıcının her kurum için birden fazla proje oluşturmasına izin verir. Bu araç kurumdaki süreçler, temel uygulamalar, yönetim uygulamaları, iş ürünleri ve süreç yönetimi gibi göstergeleri kullanılarak değerlendirmeyi yapar. Değerlendirme sonuçları yirmiden fazla görüntülenebilir ve yazdırılabilir. Ayrıca rapor görünümü mevcuttur; otomatik başarı derecesi toplama, işlem ayrıntı özeti grafikleri ve süreç veya kapasite derecelendirme profili rapor görüntülerinde sunulur.

## CMMiPal v1.0

CMMiPal v1.0<sup>7</sup> Chemuturi Consultancy tarafından geliştirilmiş bu araç ile sadece CMMI pratikleri ile kurum süreçleri eşleştirilerek süreç değerlendirmesi yapılır. Farklı modellerde kullanılması uygun değildir. Değerlendirme raporunda süreçler, amaçlar, pratikler ve iş ürünleri sadece yazı olarak bulunmaktadır ve değerlendirmeyi özetleyen bir bölüm yoktur (Yürüm *et al.* 2014). Değerlendirme girdileri ise raporda bulunmamaktadır. Ayrıca Model ve veri işlemesi Microsoft Access veri tabanında depolanır. Model ve eşleştirme verileri MS Access veri tabanında tutulmaktadır.

## Appraisal Wizard

Aracı geliştiren firma Integrated System Diagnostics Incorporated'tir. Appraisal Wizard<sup>8</sup> aracı sadece CMMI tabanlı süreç değerlendirmesi yapabilmektedir. Bu aracın eksikliklerinden biri farklı projelerin değerlendirilmeye izin verilmemesidir (Yürüm *et al.* 2014). Temelde SCAMPI yöntemini benimser. Bu araç raporlama dahil olmak üzere Planlama, Görüşme, Gözlem Yapma, Derecelendirme, Sunum Oluşturma gibi bir çok özelliğe sahiptir. Fakat Appraisal Wizard aracı içerisinde olan karmaşıklıktan dolayı bu aracı kullanmak zordur.

## Genel Literatür Değerlendirmesi

Yazılım Süreç Değerlendirmesi yapan araçların sahip olması gereken özellikler vardır. Bu araçlar farklı model ile uyum, CMMI ile uyum, SPICE ile uyum, otomatik raporlama, Öz-

---

<sup>6</sup> SPICE Assessments using the SEAL assessment tool

<sup>7</sup> <http://www.chemuturi.com/cmmipaldtls.html>

<sup>8</sup> <https://www.isd-inc.com/summary-of-services/tools/appraisal-wizard-and-wizard-lite/>

değerlendirmeyi destekleme gibi özellikler açısından değerlendirilmiştir. Tablo 2’de (Yürüm *et al.* 2014; Tarhan *et al.* 2015) araçların bazı özellikler açısından karşılaştırılması verilmiştir.

**Tablo 2.** Araçların Özellik Tablosu

ARAÇLARIN ÖZELLİK TABLOSU						
ARAÇ İSMİ	ARAÇ SAHİBİ	FARKLI MODEL İLE UYUM	CMMI	SPICE	OTOMATİK RAPORLAMA	ÖZ-DEĞERLENDİRMEYİ DESTEKLEME
Appraisal Wizard	Integrated System Diagnostics Incorporated	×	✓	×	✓	✓
CMMiPal v1.0	Chemuturi Consultancy	×	✓	×	-	×
SEAL QQ	SEAL	✓	×	✓	✓	-
Appraisal Assistant	Software Institute of Griffith University	Quality of	✓	✓	✓	✓
SPiCE 1-2-1	HM&S Consulting	IT-×	×	✓	✓	✓
SPiCE-Lite	HM&S Consulting	IT-×	×	✓	-	✓
CMM-Quest	HM&S Consulting	IT-×	✓	×	✓	✓

Bu özellikler incelenerek tabloda yer verilmelerinin sebepleri açıklanmıştır. Araç sahibine yer verilmesi araçların hangi şirket ya da kurum tarafından geliştirildiği gösterir. Bu sayede ürünü kullanmak ve araştırmak için doğru yönlendirme yapılmıştır. Yazılım süreç değerlendirme araçlarının farklı modelleri destekleme açısından esnek olup olmadığı önemlidir. Çünkü kurumlar ihtiyaçları çerçevesinde kullanacakları farklı referans modelleri destekleyecek bir araca gereksinim duyarlar. Farklı modeller ile değerlendirme yapabilen araçlar değerlendirme sonuçlarını karşılaştırma imkânı sağlar. Yani kurumlar farklı süreç referans modelleri ile süreçlerinin değerlendirmelerini yaparak her model için hangi seviyelerde olduğunu öğrenmek isteyebilirler. Bu yüzden çeşitli özelliklere sahip bir yazılım süreç değerlendirme aracı bütüncül bir meta model ile yeni bir model tanımlanmasına da imkân sağlamalıdır (Yürüm *et al.* 2014). Bu araçların CMMI modeli uygun olması firmaların belirli CMMI standartlarında süreçlerinin yeterliliklerini görmeleri için önemlidir. Bu standart modeller ile kurumlar değerlendirmeler yaparak süreçler için belirli olgunluk derecelendirmeleri kazanırlar. Bu tabloda araçların değerlendirme modeli olarak CMMI modeli için uygun olup olmadığına bakılmıştır. Bu araçların SPICE modeli uygun olması firmaların

belirli CMMI standartlarında süreçlerinin yeterliliklerini görmeleri için önemlidir. Bu standart modeller ile kurumlar değerlendirmeler yaparak süreçler için belirli olgunluk derecelendirmeleri kazanırlar. Bu tabloda araçların değerlendirme modeli olarak SPICE modeli için uygun olup olmadığına bakılmıştır. Yazılım süreç değerlendirme araçları, organizasyonlara yapılan değerlendirmelere ait raporlama gereksinimlerini (değerlendirme tarihi, değerlendirme girdileri, kanıtlar, değerlendirme bulguları, her bir süreç için değerlendirme sonuç profili) karşılamalı ve bu bulguları otomatik olarak raporlayabilmelidir (Yürüm *et al.* 2014). Ayrıca araçların değerlendirmeyi kurum dışından herhangi bir uzman çağırılmadan kurumun bir yazılım ekibi ile ya da bir danışmanı yardımıyla yapabilmesi kurumun değerlendirmelerde harcanan masrafin azalması açısından faydalı olur.

Literatürdeki çoğu aracın ontoloji tabanlı olmayan yazılım süreç değerlendirme araçları olması, bu alandaki eksikliği işaret etmektedir. Çünkü ontoloji tabanlı bir araç kullanarak süreç değerlendirmesini yapmak ile manuel olarak yapmak karşılaştırılırsa araç ile daha kısa sürede değerlendirmeler yapılabilir (Athiththan *et al.* 2018). Ayrıca ontolojilerin bilgisayarda çalıştırılabilir semantiği olduğundan, kurum tarafından sağlanan bazı verilere dayanarak şirketlerin modellere ait olgunluk düzeyleri hakkında çıkarımlar da sağlanabilir (Soydan and Kokar 2006).

Literatürdeki bu araçların dışındaki ontoloji tabanlı süreç değerlendirme araçlarına bakılırsa OCMQ-E isminde CMMI modeline uyumlu ontoloji-tabanlı bir süreç değerlendirme destek aracı oluşturulmuştur(Gazel 2009). Daha önceden var olan bir süreç yönetimi aracı olan EPF aracına Eclipse eklentileri (plug-in) ile yeni özellikler eklenerek oluşturulan bu aracın oluşturulmasındaki amaç, kurum süreçleri ile CMMI süreç referans modeline ait süreçler arasındaki uyumluluğunun izlenmesi, çeşitli ontoloji sorgulamalarının yapılması ve süreç değerlendirme faaliyetinin veri toplama aşamasının desteklenmesidir. Bu oluşturulan aracın, süreç ile CMMI arasında eşleme kurmak için ve kurulan eşlemeleri yönetmek için test edildiğinde Excel tablolarından daha hızlı ve kolay bir kullanım ortamı sağladığı gözlenmiştir. Süreç ile ilgili sorgulanmak istenen bilgilere dokümanlara bakma ihtiyacı duymadan zamandan kazanılmıştır. Ayrıca bu araç takip eden doğrulama, derecelendirme ve raporlama aşamalarının otomatik olarak yapılmasına olanak vermese de bu aşamaların manuel olarak gerçekleşmesini destekleyecek altyapıyı sağlamaktadır. SPONTO ismindeki diğer ontoloji tabanlı süreç değerlendirme aracına bakılırsa ontolojiler kullanılarak çok fazla zaman harcamadan yeniden kullanılabilirlik desteklenmiştir (Athiththan *et al.* 2018).

İlgili çalışmalar arasında yer alan Ontoloji tabanlı CMMI Eşleme ve Sorgulama Aracı (OCMQT), süreç değerlendirmesinin veri toplama aşamasını desteklemek ve CMMI modeline

uygunluğunu gözlemek için geliştirilmiş yazılım süreç değerlendirme aracı olup, açık kaynak kodlu bir EPF tabanlı süreç yönetim aracına bir eklentidir (Gazel *et al.* 2012). Bu araç, süreç tanımlarını ve CMMI ile ilgili bilgileri arama ihtiyacını desteklemektedir. Bununla birlikte CMMI bilgisini anlama, süreç varlıklarına erişme ve süreç değerlendirme ve iyileştirme için eşlemeleri yönetme konusunda OCMQT aracı kullanışlıdır. Ayrıca OCMQT aracının avantajlarından birisinin bir kurumda uzman olmayan kişilerin istenmeyen hatalar yapmasına engel olmasıdır. Ancak OCMQT aracının değerlendirme derecelendirmesi hakkında çıkarım yapma yeteneğinin yoktur.

Tüm tanımlanmış süreç referans modellerini desteklemek için genel bir yazılım süreç değerlendirme aracı (GSPA) geliştirilmiştir(Yürüm 2014). Bu araç, farklı süreç referans modellerini destekleyerek her farklı süreç referans modeli için farklı bir araç geliştirilmesinin gerekliliğini ortadan kaldırmak ve bu araçla desteklenen değerlendirmelerin etkinliğini ölçmek amacıyla kullanılmıştır. GSPA, en yaygın kullanılan süreç referans modelleri olan CMMI ve ISO 15504 modellerini tek bir modelde entegre etmek için sınıf diyagramları kullanılarak oluşturulmuş bir meta model süreç değerlendirme aracıdır. Bu araç her iki model ile süreç değerlendirmesini desteklese de, ontoloji tabanlı bir araç değildir.

Mevcut süreç değerlendirme aracı çalışmaları incelendiğinde, ontoloji tabanlı olup üç farklı model ile değerlendirme ve karşılaştırma imkânı sunan bir araç olmadığı gözlemlenmiştir. Bu çalışmada ise, üç farklı modeli ortak bir ontolojide buluşturarak ontoloji tabanlı bir yazılım süreç değerlendirme destek aracı önerilmiştir.

## MATERYAL VE METOT

Literatür arařtırmaları ile çeřitli süreç deęerlendirmeleri, ontolojiler, modeller, deęerlendirme araçları ve anlamsal aę tabanlı çalıřmalar detaylı olarak incelenmiřtir. İncelemeler sonucunda yazılım süreç deęerlendirmeleri yapılırken deęerlendirmeyi yapan uzmanların ve deęerlendirme sırasında kullanılan materyallerin eksiklerinin olduęu gözlemlenmiřtir. Kurumların bir araç desteęi olmadan deęerlendirmeleri manuel olarak yapması sürecin karmařık, çevre hatalarına açık ve zaman alıcı hale gelmesine sebep olduęu görülmüřtür (Gazel *et al.* 2009). Bu araç ile yukarıdaki olumsuzlukların giderileceęi hedeflenmiřtir. Geliřtirilecek araç deęerlendirme yaparken makine tarafından yorumlanabilir olduęundan ontoloji tabanlı olması hedeflenmiřtir. Bu aracın ontoloji tabanlı olması dięer uyumlu modellerle ortak bir dil üzerinden haberleřebilme ve birlikte çalıřabilirlik yeteneęi kazandırmıřtır (Demirdizen *et al.*2010). Bu bölüm üç bölümden oluřmaktadır. Birinci bölümde süreç referans modelleri seçilmiřtir. İkinci bölümde ontolojiler oluřturulmuřtur. Üçüncü bölümde yazılım süreç deęerlendirme aracı oluřturulmuřtur.

### Modellerin Seçilmesi

Yazılım kalitesinin güvencesinde önemli rol oynayan yazılım süreç referans modellerinden yaygın kullanılan CMMI, SPICE ve TMMI modelleri incelenerek süreç deęerlendirme aracında kullanılmalarının uygun olduklarına karar verilmiřtir. Bu modeller yapı ve içerik açısından farklılıklara sahip olsa da bunların ortak amacı, bir deęerlendirmenin yürütülmesini desteklemektir(Rout and Tuffley 2007). Bu modellerin süreç tanımlarındaki ve uygulamalarındaki benzerlikler üç modelin eřleřtirilmesinin uygun olduęunu gösterir. Bu modellerin çalıřma için uygun bulunmalarının sebeplerinden biri süreç deęerlendirmeleri yapılması için çalıřmalarda en çok kullanılan modeller olmasıdır. CMMI ve SPICE modellerinin önemli ortak özellięi, her süreci bir süreç yeterlilik seviyesi ile ayrı ayrı deęerlendirme ve derecelendirme seçeneęine olanak tanıyan sürekli gösterimin olmasıdır (Yolcular and Erdoğan 2009). Üç modelin ortak özelliklere sahip olmaları bu modellerin birbirleriyle eřleřen kavramlara sahip olduęunu göstermiřtir. Literatüre bakıldıęında tekrar kullanmak ve kaynak olarak faydalanmak üzere CMMI, SPICE ve TMMI modellerine iliřkin çeřitli ontolojilerin oluřturulduęu görülmüřtür. Buna raęmen farklı modellerin bir arada deęerlendirme yaptıęı ve ortak ontolojide eřleřtięi bir araç üzerinde çalıřılmadıęı saptanmıřtır.

Bu eksiklikten dolayı üç modeli ortak bir ontoloji üzerinde buluşturarak yazılım organizasyonlarının genel süreç değerlendirme aracının geliştirilmesine karar verilmiştir.

### **Ontolojilerin Oluşturulması**

Ontoloji geliştirmek için ilk olarak Ontoloji Oluşturma Kılavuzu incelenmiştir (Noy and McGuinness 2001). Bu kılavuzda ontoloji geliştirmek için gerekli olan ontolojideki sınıfların tanımlanması, sınıfların bir taksonomik (alt sınıf-üst sınıf) hiyerarşisinde düzenlenmesi, slotların tanımlanması ve bu slotlar için izin verilen değerlerin tanımlanması, örnekler için slotların değerlerinin doldurulması gibi işlemlerin nasıl yapılması gerektiği açıklanmıştır. İlk olarak, Ontolojileri oluştururken kapsam, ontolojilerin ne için kullanılacağını, hangi sorulara cevap vereceğini ve kimin kullanacağına dair sorularla belirlenmiştir. Ontolojilerin kapsayacağı alanla alakalı sorulara Tablo 3’de yer verilmiştir. Bu sorular her hangi bir zamanda modelin kapsamını belirlemeye ve sınırlamaya yardımcı olur. Bu sorulara ontolojiler oluşturulurken cevap aranmıştır.

**Tablo 3.** Ontolojilerin Kapsamını Belirlemek İçin Cevapları Aranılan Sorular

---

1	SPICE, TMMI ve CMMI modellerinin amaçları ne?
2	SPICE, TMMI ve CMMI modellerinin kapsam ve kısıtları nelerdir?
3	SPICE, TMMI ve CMMI hangi risk faktörlerini taşırlar?
4	SPICE, TMMI ve CMMI hangi uygulamaları kullanır?
5	SPICE, TMMI ve CMMI hangi iş ürünlerini kullanır?
6	SPICE, TMMI ve CMMI modellerinin kullandığı metotlar nelerdir?
7	SPICE, TMMI ve CMMI modellerinde roller nelerdir?
8	SPICE, TMMI ve CMMI hangi olgunluk seviyelerini içerir?
9	SPICE, TMMI ve CMMI hangi tip projelerde kullanılır?
10	SPICE, TMMI ve CMMI modellerinin süreç ve süreç alanları nelerdir? Bu süreçler, kavramlar ve ilişkiler hangi hedeflere aittir?
11	Bu organizasyon hangi projeleri barındırır?
12	Bu organizasyon hangi tip projeleri barındırır?
13	Bu organizasyon değerlendirmenin yapılacağı hangi projesinde hangi süreçleri kullanır?
14	Bu organizasyon değerlendirmenin yapılacağı hangi projesinde hangi uygulamaları kullanır?
15	Bu organizasyon değerlendirmenin yapılacağı hangi projesinde hangi iş ürünlerini kullanır?
16	Bu organizasyon değerlendirmenin yapılacağı hangi projesinde hangi metotları kullanır?
17	Bu organizasyon değerlendirmenin yapılacağı hangi projesinde hangi rolleri kullanır?
18	Bu değerlendirmede hangi olgunluk derecesi verilmiştir?
19	Yapılan değerlendirmede çıktı olarak neler olur?
20	Hangi değerlendirmede hangi süreç varlıkları ele alınır?

---

**Tablo 3.** (devamı)

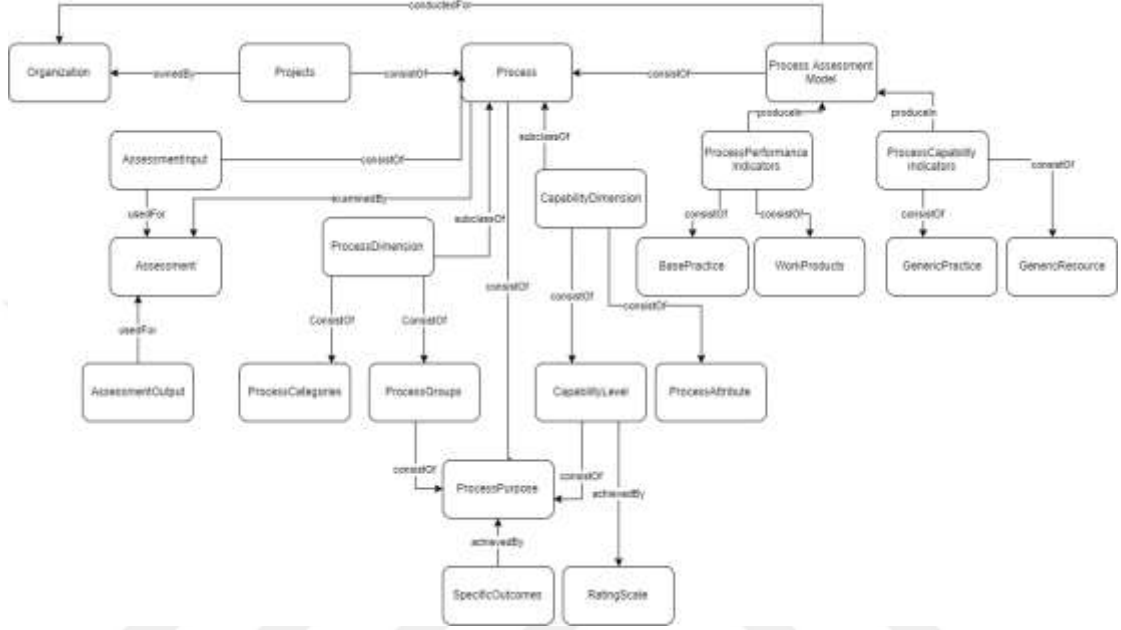
21	Hangi süreçler hangi riskleri barındırır?
22	Hangi olgunluk seviyeleri hangi süreçlerle ilişkilidir?
23	Hangi kurum hangi projede hangi değerlendirmede hangi süreç ve uygulamaları içerir?
24	Hangi süreçler hangi hedeflere ait?
25	Hangi uygulamalar hangi riskleri barındırır?
26	Organizasyon model ile ilişkisi nedir?
27	Hangi projeden hangi iş ürünleri elde edilmiştir?
28	Hangi iş ürünleri için hangi değerlendirme yapılmıştır?

Kılavuzdaki adımlar takip edilerek modellere ait sınıflar, alt sınıflar ve ilişkiler belirlenmiştir. Tablo 4’de çizilen üç model kavramsal şemasındaki kavramlar arasındaki ilişkiler şemalardaki anlamlarıyla tanımlanmıştır.

**Tablo 4.** Kavramsal Şemalardaki İlişkilerin Tanımı

Kavramların arasındaki ilişkiler	Kavramsal şemalardaki örnek ilişkiler
1 ownedBy (Sahibi olmak)	Organizasyonlar projelerin sahibidir.
2 ConsistOf (-den oluşmak)	Süreçler projelerden oluşur.
3 ConductedFor (İçin yürütülen)	Değerlendirmeler organizasyonlar için yapılır
4 isA (olmak)	Özel Uygulamalar bir uygulamadır. Üst sınıf-alt sınıf ilişkilerinde kullanılır.
5 achievedBy (erişilmiş)	Bir hedefe ait tüm uygulamalar başarı ile sonuçlandığında ait olduğu modelde belirtilir. Hedeflerden Uygulamalara erişilir
6 representedBy (Tarafından temsil edilmiş)	Süreç alanından sunumda temsil edilmiş
7 producedIn (-de üretilmiş)	Değerlendirme çıktısı Süreç Alanında üretilmiş
8 isLeveledBy(derecelendirilmiş)	Gösterimler olgunluk seviyelerine göre derecelendirilmiş
9 satisfiedBy(Yerine getirilmiş)	Hedefler süreç alanları tarafından yerine getirilmiş
10 outputOf (Çıktısı olmak)	İş ürünleri kurumların çıktısıdır
11 developedBy (geliştirilmek/geliştiren)	Aşamalı temsil Olgunluk Seviyesi tarafından geliştirilmiş
12 subclassOf (Altsınıfı olmak)	Süreç boyutu Süreç’ in alt sınıfıdır
13 fulfilledBy (Tarafından uygulanmış)	Değerlendirme çıktısı Temel uygulama tarafından gerçekleştirilmiş
14 associatedWith (İlişkilendirilmiş)	İş ürünleri Süreç çıktısı ile ilişkilendirilmiş
15 Uses (kullanılır)	Genel kaynaklar yine uygulamalar tarafından kullanılır
16 hasLevel (Seviyeye sahiptir)	Değerlendirmeler yetenek seviyelerine sahiptir
17 sameAs (-İle aynı olmak)	Eserler iş ürünleri ile aynıdır

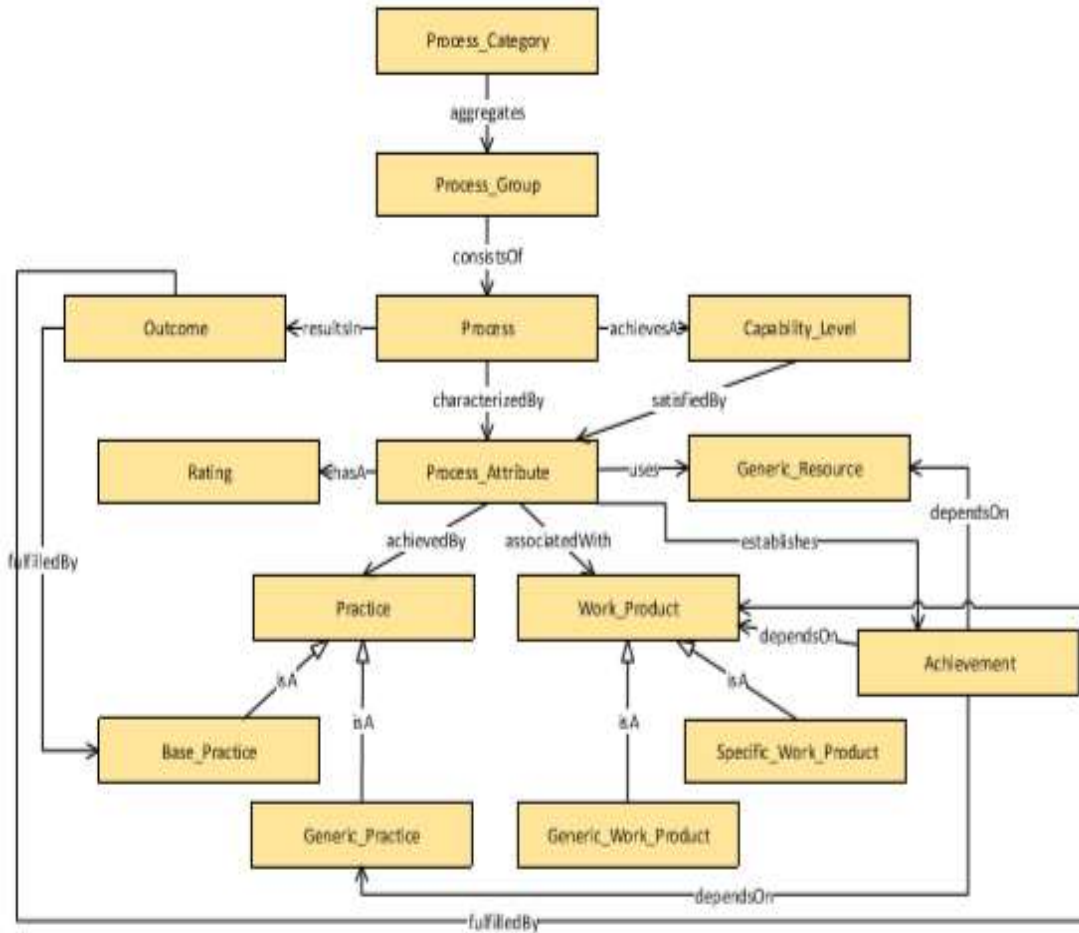
Draw.io<sup>9</sup> programında her bir modele ait sınıfların, alt sınıfların ve onların hiyerarşik ilişkilerinin oluşturulduğu kavramsal diyagramlar çizilmiştir. Ekler kısmında sırasıyla CMMI modeline ait ve TMMI modeline ait kavramsal diyagramlara yer verilmiştir. Oluşturulan ontoloji şemalarını literatürdeki ontoloji şemaları ile karşılaştırarak eksikler tespit edilerek güncellemeleri yapılmıştır. Şekil 1’de ilk çizilen SPICE modeline ait kavramsal şema gösterilmiştir.



Şekil 1. İlk çizilen kavramsal SPICE şeması

Şekil 2’deki (Proença and Borbinha 2018) kavramsal diyagramda ilk çizilen SPICE şeması ile literatürdeki karşılaştırılan SPICE modeline ait kavramsal şema gösterilmiştir.

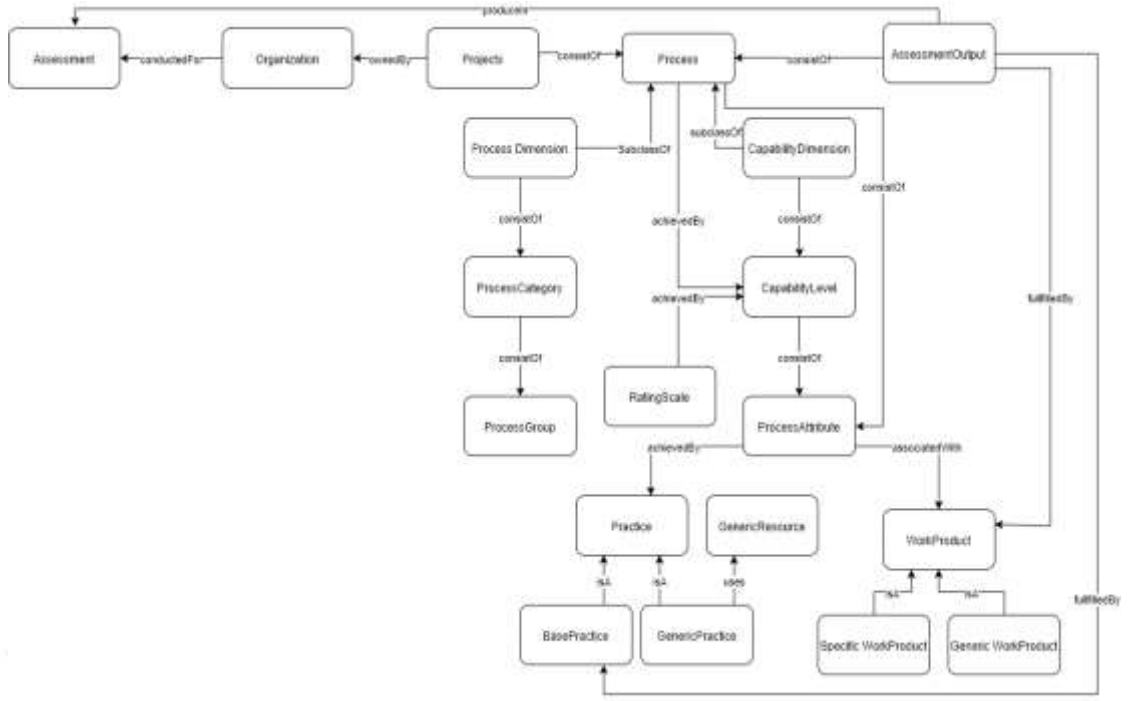
<sup>9</sup> Draw.io: <https://app.diagrams.net/>



**Şekil 2.** Karşılaştırılan SPICE modeli kavramsal şeması

İlk olarak Proença ve Barbinha'nın çalışmalarında yer verdikleri ISO/IEC 15504-5 Ontoloji şemasına bakıldığında "ProcessCategory" kavramı ile "Process" kavramının arasındaki ilişkinin kurulması için "ProcessCategory" kavramının "ProcessGroup" kavramı ile de ilişki kurdukları görülmüştür. "Process" ile "ProcessAttribute" doğrudan ilişkilendirilmiştir. "Outcome"ın "WorkProduct" ve "BasePractice" tarafından ortaya çıktığı ilişkisi gösterilmiştir. "ProcessAttribute" kavramı, "Practice" ve "WorkProduct" kavramları ile ayrı ayrı ilişkilendirilmiştir. Miras ilişkisi olarak "BasePractice" ve "GenericPractice" kavramları bir Practice kavramıdır ilişkisi kurulmuştur. Buna ek olarak "SpecificWorkProduct" ve "GenericWorkProduct" bir "WorkProduct" kavramıdır ilişkileri de kurulmuştur. Son olarak "ProcessAttribute" kavramının, "Rating" ve "GenericResource" kavramları ile doğrudan ilişkilendirildiği gözlemlenmiştir.

Şekil 3'teki kavramsal şemada, karşılaştırıldıktan sonra güncellenen SPICE modeline ait güncel kavramsal şema gösterilmiştir.



**Şekil 3.** En son çizilen olan SPICE kavramsal şeması

İlk oluşturulan SPICE kavramsal şemasında “ProcessCategory” kavramı “Process” ile “ProcessDimension” üzerinden ilişkilendirilmiştir. “ProcessAttribute” kavramı “Process” ile “capabilityDimension” üzerinden bağlantı sağlanmıştır. “SpecificOutcome”, “Process” ile “ProcessPurpose” üzerinden ilişkilendirilmiştir. Bu ilişkilerdeki eksiklikler karşılaştırılan makaledeki kavramsal şemaya bakılarak tamamlanmıştır. Bununla birlikte “BasePractice” ve “GenericPractice” ‘i “Practice”in alt sınıfları yapan ilişkilerin ve kavramların olması gerektiğinin farkına varılmıştır. Bu yapılan karşılaştırmalar sayesinde modellerin kavramsal diyagramlarındaki eksiklikler görülmüştür ve ontolojilerin doğrulanması yapılmıştır.

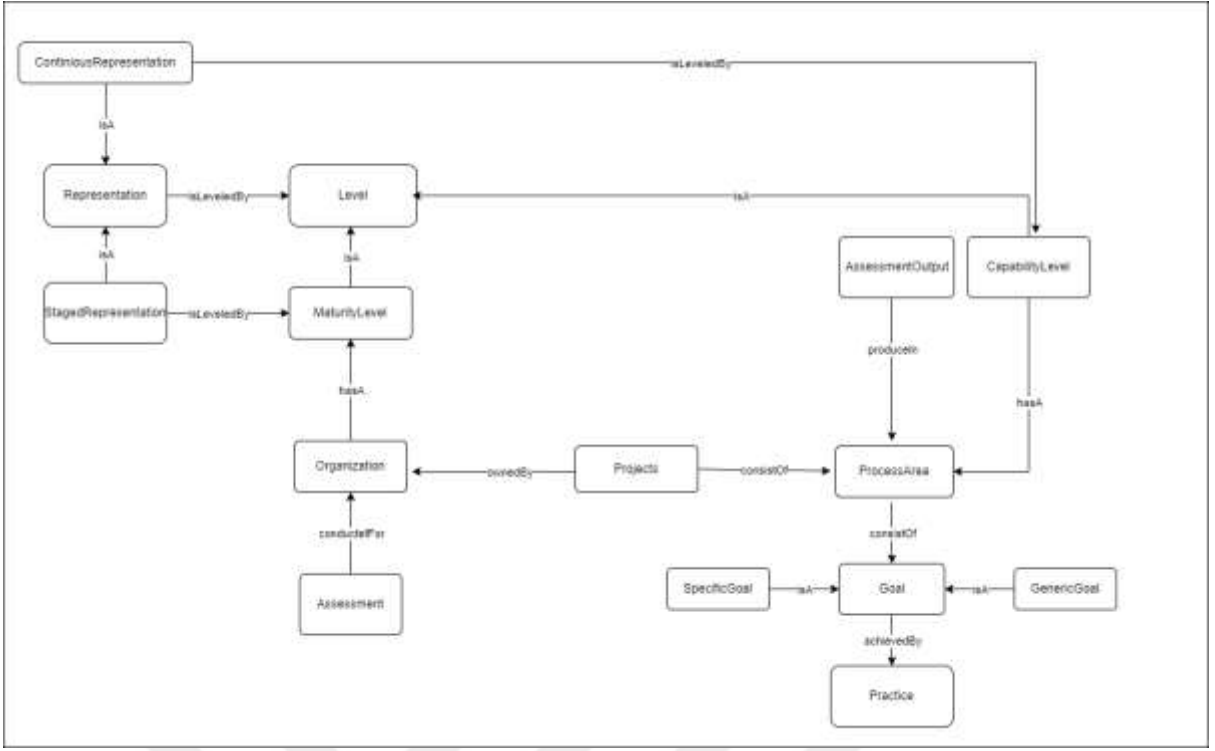
Bunların yanı sıra ilk başta çizilen modellerin kavramsal diyagramlarının çalışmanın ilerlemesi neticesinde güncellenebilir olduğu göz önüne alınmıştır. Ayrıca ontoloji modellemesinin tek bir yolu olmadığı için olabilecek alternatiflerden birinde karar verilmiştir. Ortak ontolojiyi oluştururken 3 modelin ortak sınıfları ve ilişkileri dikkate alınmıştır. Ontoloji eşleştirirken modellerin kavramlarının aynı işlevi taşımasına önem verilmiştir.

Tablo 5’de (Vanamali *et al.* 2008) Süreç referans modellerine ait benzer tanımdaki kavramların eşleştirilmesi gösterilmiştir. Bu tablodaki bazı kavramların isimleri farklı olsa da aynı anlama sahip oldukları gözlemlenmiştir. Bunun yanı sıra TMMI modeli CMMI modelinden üretildiği için aynı isimli kavramlara sahip olduğundan eşleştirmesi zor olmamıştır. Sadece CMMI modelinde olan “Continous Capability” kavramı TMMI modelinde yoktur.

**Tablo 5.** Modellerin Kavramlarının Eşleştirilmesi

CMMI	SPICE	TMMI
Process Area	Process	Process Area
Purpose	Process Purpose	Purpose
Specific Goal	Process Outcomes	Specific Goal
Specific Practice	Base Practices	Specific Practice
Subpractices	-	Subpractices
Typical Work Products	Output Work Products	Typical Work Products
-	Work Product Characteristics	-
Generic Goal	Process Attributes	Generic Goal
Generic Practice	Generic Practices	Generic Practice
Generic Practice Elaborations	-	Generic Practice Elaborations
-	Generic Resources	-
Examples	-	Examples
Amplifications	-	Amplifications
Continuous Capability	-	-
Staged Capability	-	Staged Capability
Capability Levels	Capability Levels	Capability Levels

Bu üç süreç referans modellerinin birleşimi olarak YGOM (Yeni Genel Ontoloji Modeli) ontoloji modeline ait kavramsal diyagram da oluşturulmuştur. YGOM genel ontolojiyi 3 modelin ontolojisindeki ortak kavramlar belirlenerek oluşturulmuştur. Şekil 4'te YGOM kavramsal şemasına yer verilmiştir. Buna göre CMMI ve TMMI'da yer alan ve uygulamaları süreç amacına göre ayırmayı sağlayan süreç alanlarına SPICE'da süreç kavramı karşılık gelmektedir. Bu sebeple Process ile ProcessArea aynı sınıfta gösterilmiştir. Ayrıca her üç model de gösterime (representation) sahiptir. CMMI ve SPICE modelleri sürekli ve basamaklı olmak üzere iki çeşit gösterime sahipken TMMI sadece basamaklı gösterime sahiptir. Bunun için Continuous/Staged bir Representation'dir ilişkisi eklenmiştir.



**Şekil 4.** YGOM kavramsal şeması

Çizimler tamamlandıktan sonra Protégé 5.5.0 programının kurulumu yapılarak her bir modele ait ontolojiler Protégé’de OWL dili ile geliştirilmiştir. Bu OWL dosyaları sayesinde bilgisayarların anlayabileceği bir kavramsallaştırma ve belirtim oluşturulmuş oldu. OWL dilinin kendine has kuralları (Örn:İki sınıf aynı isimli olamaz gibi) vardır.

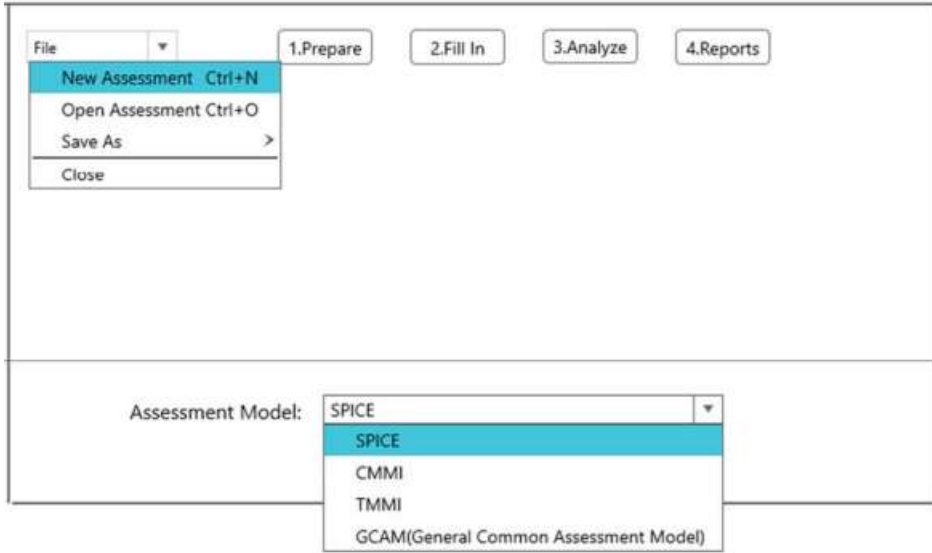
Şekil 5’te YGOM ontolojisinin Protégé programındaki OntoGraph kısmındaki bağlı veri grafiği aşağıda gösterilmiştir.



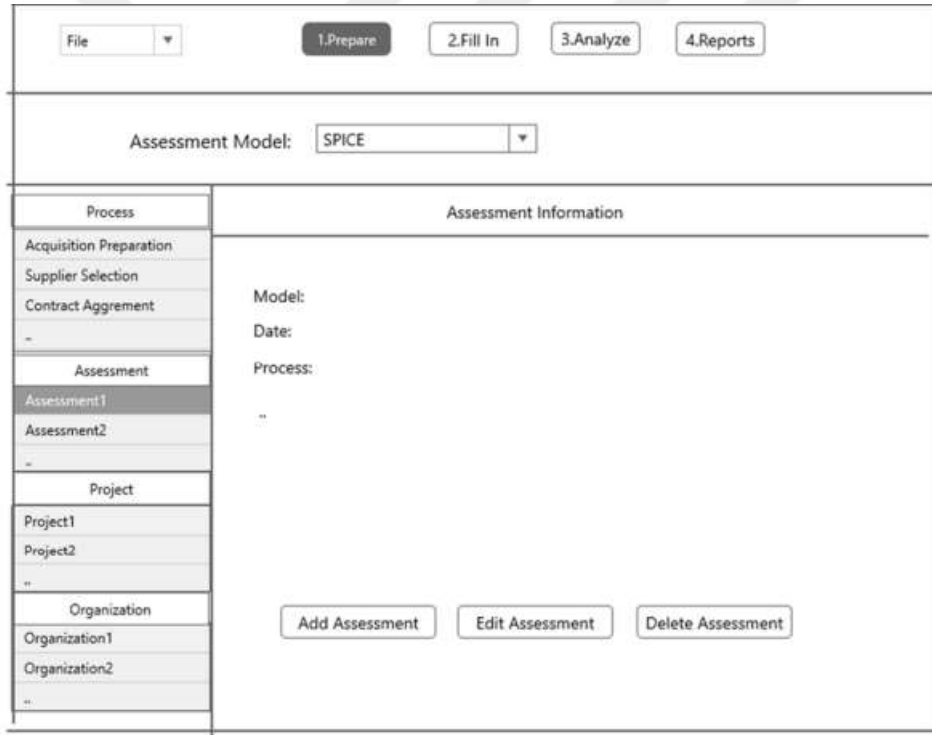
**Şekil 5.** YGOM kavramsal şemasına ait bağlı veri grafiği

### Aracın Geliştirilmesi

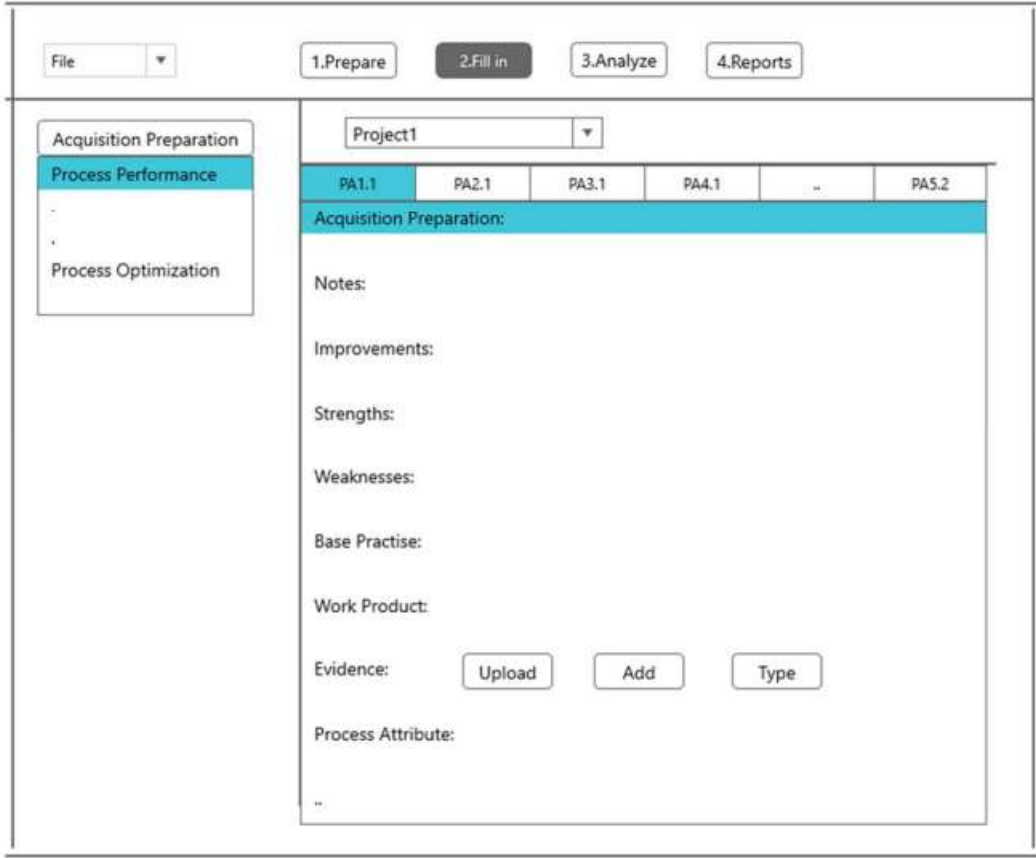
Aracı geliştirmek için ilk olarak literatürdeki mevcut süreç değerlendirme araçları incelenmiştir. Bu araçlara ait özellikler incelenmiştir. Aracın olabilecek sayfalarının arayüzü draw.io programında oluşturulmuştur. Anasayfa, Model seçimi, Assessment Ekleme, Süreç seçimi gibi sayfaların ekran görüntüleri aşağıdadır.



Şekil 6. Anasayfa: yeni assessment oluşturma ve model seçim sayfası



Şekil 7. Assessment bireyi ekleme ve bireye ait bilgileri girme sayfası



**Şekil 8.** Seçilen sürece ait verileri girme sayfası

Arayüz çiziminden sonra aracın hangi programlama diliyle yazılması gerektiği araştırılarak ontoloji ile entegre çalışabilen Java<sup>10</sup>'nın uygun olduğu görülmüştür. Java'nın semantik web uygulamaları geliştirmeyi sağlayan açık kaynaklı ve ücretsiz kütüphanelere sahip olan bir dil olması seçilmesinde etkili olmuştur. Java AWT<sup>11</sup> paketinin üzerine kurulan olan ve Java ile Grafiksel Arayüzler (GUI) hazırlamak için gerekli araçlara sahip Java Swing<sup>12</sup> ile uygulamanın arayüzü oluşturulmuştur. Geliştirme platformu olarak Java tabanlı uygulama ortamı sunan açık kaynak kodlu Eclipse<sup>13</sup> ya da Netbeans uygundur. Bu projede Netbeans'in AWT ve Swing grafiksel kullanıcı arayüzü desteği daha iyi olmasına rağmen Eclipse'in daha önce oluşturulan frame kodlarını test etme imkânı daha uygun olduğu için Eclipse ortamı tercih edilmiştir. Bu görsel arayüz hazırlarken, her sayfa için en alta bir çerçeve (JFrame) ve bu çerçevelerin içlerine ise birer panel konmuştur. Panel üzerine düğme, text alanı, progressbar, menü gibi görsel araçlar yerleştirilmiştir. Aracı ontoloji tabanlı bir araca dönüştürmek için Java çerçevesi olan Jena Kütüphanesi kullanılmıştır. Jena, Semantik Web uygulamaları oluşturmak için uygun bir kütüphane olduğu için seçilmiştir. Bu sayede ontolojide tutulan bilgiler Jena

<sup>10</sup> Java: <https://www.java.com/tr/>

<sup>11</sup> Java AWT: <https://docs.oracle.com/javase/7/docs/api/java/awt/package-summary.html>

<sup>12</sup> Java SWING: <https://docs.oracle.com/javase/tutorial/uiswing/>

<sup>13</sup> Eclipse: <https://www.eclipse.org/>

Kütüphanesi ile Eclipse ortamına aktarılmıştır. Ontolojiler çekildikten sonra RDF dosyalarına yani arayüz üzerinden ontolojilerdeki sınıflara, özelliklere ve bireylere erişmek için SQL'e benzeyen standart zengin bir sorgu dili olan SPARQL<sup>14</sup> kullanılmıştır. Eclipse programı ile Java kodları kullanarak aşağıda verilen SPARQL sorguları ile import edilen ontolojiden istenilen veriler çekilmiştir. Bu sorgular sayesinde veriler, bilgisayarlar tarafından anlaşılabilir olarak ontolojilerdeki bazı kavramların arayüz üzerinden listelenmesine olanak sağlanmıştır. Şekil 9'daki sorgu kodu modeldeki süreçlerin SPARQL ile çekilmesini göstermektedir. Bu süreç sorgusunun yanı sıra uygulamalar ve iş ürünlerinin listeleri gibi veriler de modellere ait OWL dosyalarından çekilebilmektedir.

```
String sorgu =
    "PREFIX NS:<http://www.semanticweb.org/rukiye/ontologies/2020/2/untitled-ontology-100#> "
    + "PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#> "
    + "PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#> "
    + "SELECT ?x "
    + "WHERE { ?x      rdfs:subClassOf      NS:Maturity_Level_2_Managed_ProcessArea. }";

    Query query = QueryFactory.create(sorgu);
    // Sorgumuzu calistiriyoruz
    QueryExecution qe = QueryExecutionFactory.create(query, model);
    ResultSet results = qe.execSelect();
    // Sonuclari ekrana yaziyoruz
```

Şekil 9. SPARQL ile OWL dosyasından çekilen süreçlerin Sorgu kodu

Şekil 10'da SPARQL ile OWL dosyasından çekilen süreçlerin arayüzdeki görüntüsü gösterilmektedir.

Process	Projects	Organizations	Assessment
NS:Configuration_Management			
NS:Measurement_and_Analysis			
NS:Process_Monitoring_and_Control			
NS:Process_and_Product_Quality_Assurance			
NS:Project_Planning			
NS:Requirements_Management			
NS:Supplier_Agreement_Management			
Validation			

Şekil 10. SPARQL ile OWL dosyasından çekilen Süreç listesinin Arayüz görüntüsü

Arayüzde üç modele ait yazılım süreç değerlendirme yapma seçeneği oluşturulmuştur. Her modele ait süreç sayfaları oluşturulmuştur. Bu sayfalardan her sürecin kendine has

<sup>14</sup> SPARQL: <https://www.w3.org/TR/rdf-sparql-query/>

notlarının, uygulama sorularına verdiği cevapların, uygulama başarı oranlarının ve ortaya çıkan iş ürünlerinin girilmesi istenmiştir. Değerlendirme sırasında süreç alanlarındaki başarı oranlarını hesaplamak için her bir uygulamaya Tablo 1'deki gibi süreç referans modellerindeki standart notlar verilerek bunların aritmetik ortalaması alınmıştır. Bu işlemlerin sağlanması için oluşturulan Java Kodunun bir kısmı Şekil 28'de gösterilmektedir.

Ontolojileri görselleştirmek için OWLGrEd editörü kullanılmıştır. OWL dosyaları bu editörde açılarak OWLGrEd<sup>15</sup> ile ontolojileri görselleştirip ontolojilerde tutulan formel bilginin dağıtılması yeteneği sağlamıştır. Bu görselleştirici sayesinde, "ontoloji okur-yazarlığı olmayanlar" için bile anlaşılması ve kullanılması kolay görselleştirme imkânı sağlanmıştır.

### **OSPAT(Ontology-Based Software Process Assessment Tool)**

Bu bölümde geliştirilen araçtan ve aracın doğrulanmasından bahsedilmiştir.

#### **Geliştirilen Araç ve Özellikleri**

OSPAT ile CMMI, SPICE ve TMMI modelleri kullanılarak yapılan değerlendirmeler desteklenmektedir. Değerlendirme sonuçlarının ortak bir ontolojide buluşturulması sağlanmıştır. Bu farklı modellerle süreç değerlendirme yapabilmesi aracın esnekliğini göstermektedir. Ayrıca araç değerlendirme sonuçlarını dosyalarda tutarak organizasyonlara raporlar sunar. Aracın üç model ile değerlendirme yapmaya uygun olması üç modele ait değerlendirme sonuçlarını gözlemleyerek karşılaştırma yapma imkânı sağlar. Üç modelin aynı organizasyondaki değerlendirmelerin karşılaştırılabilmesi için her modele ait içerikleri benzeyen ama isimleri farklı olabilen süreçlerin kullanılması gerekmektedir.

25'ten fazla süreç alanı ile değerlendirme yapmaya olanak sağlayan ontoloji tabanlı bu araç, değerlendirmesi yapılan süreçler bazında organizasyonların hangi kalite düzeyinde olduğunu barındırmaktadır. Bu araç, desteklediği modeller için değerlendirme sırasında kullanılması gereken notların girilmesi, başarı yüzdelerinin girilmesi ve iş ürünlerinin girilmesi gibi birçok fonksiyonu gerçekleştirir. Aracın arayüzünde kullanıcılara kolaylık sağlanması için CMMI, SPICE ve TMMI modellerindeki süreç alanları, bir süreç alanında yer alan pratikler ile ilgili kısa açıklamalara yer verilmiştir. Kullanılan SPARQL ile araç arayüzü ve ontolojiler arasında önemli verilerin listelenmesini sağlayarak araca sorgulama işlevi kazandırılmıştır. Sorgular ile modellere ait süreç alanları, süreç alanlarına ait uygulamalar ve uygulamalara ait iş ürünleri listelenmiştir. Bütünlük sağlanması açısından arayüz üzerinden eklenen kavramlar OWL dosyalarına diğer kavramlarla ilişkili olarak eklenmiştir. Ayrıca Protégé programı ile

---

<sup>15</sup> <https://www.w3.org/2001/sw/wiki/OWLGrEd>

oluşturulan ontolojilerin arayüz üzerinden okunabilmesi mümkündür. Bu aracın diğer bir özelliği, değerlendirme ve raporlama sırasında sonuçları tekrar kullanabilmek ve sonraki değerlendirmeler ile kıyaslamalar yapabilmek için verileri güvende tutmasıdır. Yapılan değerlendirmeleri süresiz olarak saklama yeteneği açısından güvenilirdir. Son olarak bu araç uzman olmayan kişiler tarafından da değerlendirme yapmaya olanak sağlaması için basit ve görsel bir kullanıcı arayüzüne sahiptir. Süreç değerlendirme araçlarının kaliteli ve verimli araçlar olması için sahip olması gereken özellikler vardır (Lok and Walker 1997). Bu araç, Lok ve arkadaşının çalışmasında belirlenen kalite özellikleri ile kıyaslandığında, beklenen özelliklerin birçoğuna sahip olduğu görülmüştür.

Tablo 6, Tablo 7 ve Tablo 8’de sırasıyla aracın değerlendirme yapma, yapılan değerlendirmeyi OWL dosyasına ekleme ve silme işlemlerinin kullanıcı senaryoları gösterilmiştir. Geliştirilen araçla bir yazılım organizasyonuna ait süreç değerlendirmesini yapmak için örnek kullanıcı senaryosu oluşturulmuştur.

**Tablo 6.** SPICE Modeli Yazılım Gereksinim Analizi Sürecinde Yapılan Değerlendirmeye Ait Verilerin Girilmesi Kullanıcı Senaryosu

Kullanıcı Senaryosu Adı	Yazılım Gereksinim Analizi Sürecinde yapılan değerlendirmeye ait verilerin girilmesi
Tanım	Ontoloji tabanlı bir yazılım süreç değerlendirme aracı ile bir yazılım organizasyonunun SPICE modelinde Yazılım Gereksinim Analizi Sürecinde yazılım süreç değerlendirmesi yapılır.
Aktörler	Kullanıcı
Ön koşullar	Kullanıcı Uygulama Aracına girerek yeni bir değerlendirme açmalıdır
Ana Senaryo	<ol style="list-style-type: none"> <li>1. Kullanıcı New Assessment butonuna basarak yeni bir değerlendirme açar.</li> <li>2. Açılan sayfada CMMI, SPICE ve TMMI Model seçim butonlarından Yazılım Gereksinim Analizi sürecinde değerlendirme yapmak için SPICE olanı seçer.</li> <li>3. Açılan Model sayfasında hangi süreçte değerlendirme yapacağını seçeceği sayfaya gitmek için FillIn butonuna basar.</li> <li>4. Açılan sayfada SPICE modeline ait süreçlerden olan Yazılım Gereksinim Analizi butonuna basar.</li> <li>5. Açılan süreç sayfasında üst kısımdaki sürece ait genel notları ve bilgileri doldurur.</li> <li>6. Yazılım Gereksinim Analizi sürecinin genel uygulamalarına ait notları, güçlü yanlarını, eksik yanlarını gibi bilgileri doldurur.</li> <li>7. Bu girdiği bilgileri bir dosyaya kaydetmek için sayfanın üstündeki Dosyaya Ekle butonuna basar.</li> <li>8. Dosyaya kaydettikten sonra notlarını girdiği uygulamaların iş ürünlerini de girer.</li> <li>9. İş ürünleri bölümlerini doldurduktan sonra iş ürünleri OWL dosyasına eklemek için sayfanın üstündeki İş ürünlerini Ekle butonuna basar.</li> <li>10. Aynı sayfada her bir uygulama için N,P,L,F butonlarından birine basarak başarı yüzdelerini seçer.</li> </ol>
Alternatif Senaryo	-
Son Koşullar	Verilerin arayüz üzerinden girilmesi gerekir
Referans	Şekil 11

**Tablo 7.** SPICE Modeli Yazılım Gereksinim Analizi Sürecinde Yapılan Değerlendirmeyi OWL Dosyasına Ekleme Kullanıcı Senaryosu

Kullanıcı Senaryosu Adı	Yazılım Gereksinim Analizi Sürecinde yapılan değerlendirme OWL dosyasına ekleme
Tanım	Ontoloji tabanlı bir yazılım süreç değerlendirme aracı ile bir yazılım organizasyonunun SPICE modelinde Yazılım Gereksinim Analizi Sürecinde yapılan yazılım süreç değerlendirmesi OWL dosyasına aktarılır.
Aktörler	Kullanıcı
Ön koşullar	Kullanıcı OWL dosyasına eklemeye başlamadan önce değerlendirilecek sürecin sayfasında verilerini girmelidir.
Ana Senaryo	1. Bütün uygulamaların başarı yüzdelerini girdikten sonra yüzdelerin OWL dosyasına eklenmesi için sayfanın üstünde olan Başarı Yüzdesi(Rating) Ekle butonuna basar. 2. Bütün uygulamaların iş ürünlerini girdikten sonra iş ürünlerinin OWL dosyasına eklenmesi için sayfanın üstünde olan İş Ürünleri Ekle(Work Product ) butonuna basar. 3. Sonra sayfanın üstündeki Assessment Page Butonuna tıklayarak değerlendirmesini yaptığı kuruma ait bilgileri girerek OWL dosyasına kaydeden sayfaya geçer. 4. Açılan sayfada Organization kısmına tıklayarak kurumun ismini, değerlendirmesi yapılan projenin ve sürecin ismini girer. 5. Organizasyon Ekle butonuna basarak OWL dosyasına değerlendirme yapılmış olur.
Alternatif Senaryo	6. Bireyler OWL dosyasına kaydolmamışsa Yazılım Gereksinim Analizi sayfasına gidilir
Son Koşullar	7. Veriler girilerek Adım 1'den sırasıyla işlemler yapılır. Bireyler OWL dosyasına kayıt olmalıdır
Referans	Şekil 11 Şekil 12

**Tablo 8.** Yapılan Değerlendirmeyi Silme Kullanıcı Senaryosu

Kullanıcı Senaryosu Adı	Araç üzerinden SPICE modelinde yapılarak OWL dosyasına kaydedilen değerlendirme silme
Tanım	Ontoloji tabanlı bir yazılım süreç değerlendirme aracı ile daha önce değerlendirme yapılarak eklenen OWL dosyasındaki organizasyon ve değerlendirme verilerini silme
Aktörler	Kullanıcı
Ön koşullar	Sileceği Değerlendirmenin OWL dosyasındaki ismini ve hangi modelde değerlendirme yapıldığını bilmesi gerekir
Ana Senaryo	1. Kullanıcı New Assessment butonuna basarak yeni bir değerlendirme açar. 2. Açılan sayfada CMMI, SPICE ve TMMI Model seçim butonlarından silmek istediği değerlendirme olduğu SPICE modelini seçer 3. Açılan sayfada Prepare sayfasına ait Organization kısmını açar. 4. Bu kısımda kurumun ismini, değerlendirme yapılan projenin ve sürecin ismini girer. 5. Organizasyon Sil butonuna basarak OWL dosyasından değerlendirme silmiş olur.
Alternatif Senaryo	
Son Koşullar	Veriler OWL dosyasından silinmiş olmalıdır.
Referans	Şekil 12

OSPAT ile süreç değerlendirmesi yapılırken hangi süreç referans modeli ile değerlendirme yapılıyorsa ilgili modelin kendi ontolojisi üzerinden süreç değerlendirmesi yapılır. SPICE süreç referans modeli ile süreç değerlendirmesi yapılırken SPICE modeline ait ontoloji, CMMI süreç referans modeli ile süreç değerlendirmesi yapılırken CMMI modeline ait ontoloji, TMMI süreç referans modeli ile süreç değerlendirmesi yapılırken TMMI modeline ait ontoloji kullanılmıştır. Farklı modeller ile yapılan değerlendirmeler kavramlar üzerinden görselleştirmek için farklı zamanlarda aynı proje için hangi değerlendirmeler yapıldığını görebilmek için YGOM ontolojisinde birleştirilmiştir.

### **Aracın Doğrulaması**

Araç geliştirme tamamlandıktan sonra bir devlet üniversitesine ait bir yazılım geliştirme projesinde doğrulama çalışması yapılmıştır. Bu üniversite sahip olduğu yazılım projelerinin büyüklüğü ve bu projelerdeki tecrübeleri sebebiyle yazılım süreç değerlendirmesi yapmak için uygun bulunmuştur. Ayrıca aracın bu üniversitede doğrulanması için gerçekleştirilecek çalışmayla ilgili Etik Kurul Onay Belgesi alınmıştır. Bu üniversiteye ait verilerin gizliliğini sağlamak amacıyla bundan sonra Organization1 ismi ile bahsedilecektir. Organization1 sürekli iyileştirmeyi temel alan bir kalite anlayışında olan fakat henüz yazılım geliştirme süreçleri açısından uluslararası standart modeller ile kendini belgelememiştir. İlk olarak Organization1'in yazılım projelerinde hangi süreçlere yer verildiğine dair ön görüşme yapılmıştır. Görüşme sonrası doğrulama için beş farklı süreç alanında değerlendirmelerin yapılmasına karar verilmiştir.

Bu doğrulama için kullanılan süreç alanlarına karar verilirken üç modelde de ortak süreç alanları olmasına dikkat edilmiştir. Böylece modeller arası karşılaştırma yapılmasına olanak sağlanmıştır. Tablo 9'da üç modelin doğrulama çalışmasında kullanılan ortak süreçleri gösterilmiştir. Bu süreçlerin eşleştirilmesinde Bella ve arkadaşlarının yaptığı makale çalışmasından faydalanılmıştır (Vanamali *et al.* 2008). Bu tabloya göre Yazılım Gereksinim Analizi ve Gereksinim Yönetimi süreçlerinin TMMI modelinde karşılığı olan bir süreç olmadığı için o kısım boş bırakılmıştır.

**Tablo 9.** Doğrulama Çalışmasında Kullanılan Modellerin Süreçlerinin Eşleştirilmesi

GRUPLAR	SPICE	CMMI	TMMI
1.Grup	Proje Yönetimi	Proje planlama	Test Planlama
2.Grup	Yazılım Gereksinim Analizi	Gereksinim Yönetimi	-

Değerlendirmeyi yapmak için geniş kapsamlı Mesleki Yazılım projelerinden biri seçilmiştir. Seçilen projede çalışmış Sistem Analisti ile mülakatlar yapılmıştır. Önce, bu

mülakatlarda değerlendirmeye katılan organizasyon ve analist ile ilgili bilgi toplamak amacıyla demografik sorular sorulmuştur. Tablo 10’da demografik sorular listelenmiştir.

**Tablo 10.** Demografik Sorular

NUMARA	SORULAR	CEVAPLAR
1	Projede yer aldığınız görev nedir?	Sistem Analisti
2	Firmanız yazılım uygulama alanlarından hangisine uymaktadır?	Bilimsel, mühendislik ve mesleki yazılımlar
3	Firmanız hangi tür yapılara uymaktadır?(Devlet, Tamamen Yerli Organizasyon, Tamamen Yabancı Organizasyon)	Devlet
4	Firmanızda çalışan sayısı ne kadardır?	50-100 arası
5	Firmanız son 5 yılda kaç tane yazılım geliştirmiştir? (1-5 arası, 6-10 arası, 11-15 arası, 16-20 arası, 20’den fazla)	11-15 arası
6	Firmanızın yazılım aktivitelerini sıralamak gerekirse nasıl sıraladınız?(Uygulama geliştirmek, bakım, teknik destek)	1.Yeni uygulamalar geliştirmek 2.Bakım 3.Teknik Destek
7	Disiplinli, dokümente edilmiş (formal ) sistem geliştirme metodolojileri kullanıyor musunuz?	Evet
8	Bu metodolojileri kullanıyorsanız, ne kadar süredir kullanmaktasınız?	3 yıldan fazla süredir
9	Formal veya formal olmayan (informal) metodolojiyle hangi yazılım süreci modellerini izlemektesiniz?	Şelale Modeli

Mülakat esnasında demografik sorulardan sonra seçilen beş süreç alanına dair sorular sorularak süreç değerlendirmesine geçilmiştir. Bu sorular seçilen modellere ait süreçlerin literatürde olan standart uygulama sorularıdır (Kalaycı 2007). SPICE modelinin Yazılım Gereksinim Analizi sürecine ait uygulama soruları aşağıda listelenmiştir (ISO 2004).

**Tablo 11.** SPICE Modeli Yazılım Gereksinim Analizi Soruları

SPICE Modeli Yazılım Gereksinim Analizi Soruları
SWE.1.BP1: Bir yazılım gereksinimleri belirtiminde işlevsel ve işlevsel olmayan yazılım gereksinimlerini müşterinin belirtilen ihtiyaçlarını karşılayacak şekilde detaylı olarak belirttiniz mi?
SWE.1.BP2: Doğru uygulama ve izlenebilirliği artırmak için gereksinimleri kategorize edip önceliklendirdiniz mi? Öncelik sırasına göre listeleme yaptınız mı?
SWE.1.BP3: Doğruluğu, teknik fizibiliteyi, doğrulanabilirliği sağlamak ve risk tanımlamasını desteklemek için belirtilen yazılım gereksinimlerini analiz ettiniz mi? Maliyet, program(zamanlama) ve teknik etki açısından analiz yaptınız mı?
SWE.1.BP4: Ortaya çıkan yazılım gereksinimlerin karşılanması için kurumun işletim ortamı uygun mu diye analizini yaptınız mı?
SWE.1.BP5: Bir gereksinimin doğrulanması için nitel ve nicel önlemleri tanımlayan her bir yazılım gereksinimi için doğrulama kriterleri geliştirdiniz mi?

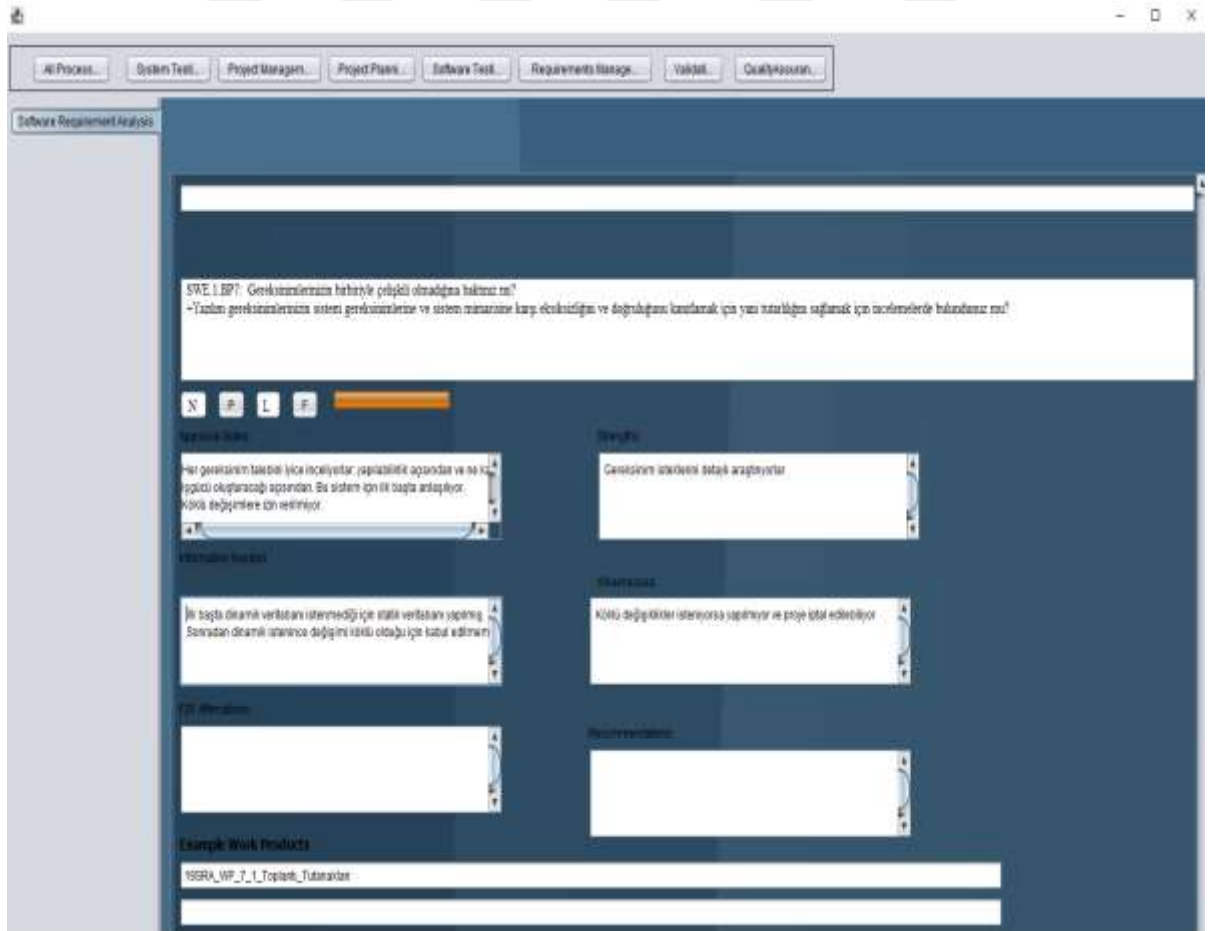
**Tablo 11.** (devamı)

SWE.1.BP6: Sistem gereksinimleri ve yazılım gereksinimleri arasında çift yönlü izlenebilirlik sağladınız mı? İzlenecek gereksinimin kolayca bulunabilir eşsiz bir id numarası var mı?

SWE.1.BP7: Gereksinimlerinizin birbiriyle çelişkili olmadığına baktınız mı? Yazılım gereksinimlerinin sistem gereksinimlerine ve sistem mimarisine karşı eksiksizliğini ve doğruluğunu kanıtlamak için yani tutarlılığını sağlamak için incelemelerde buldunuz mu?

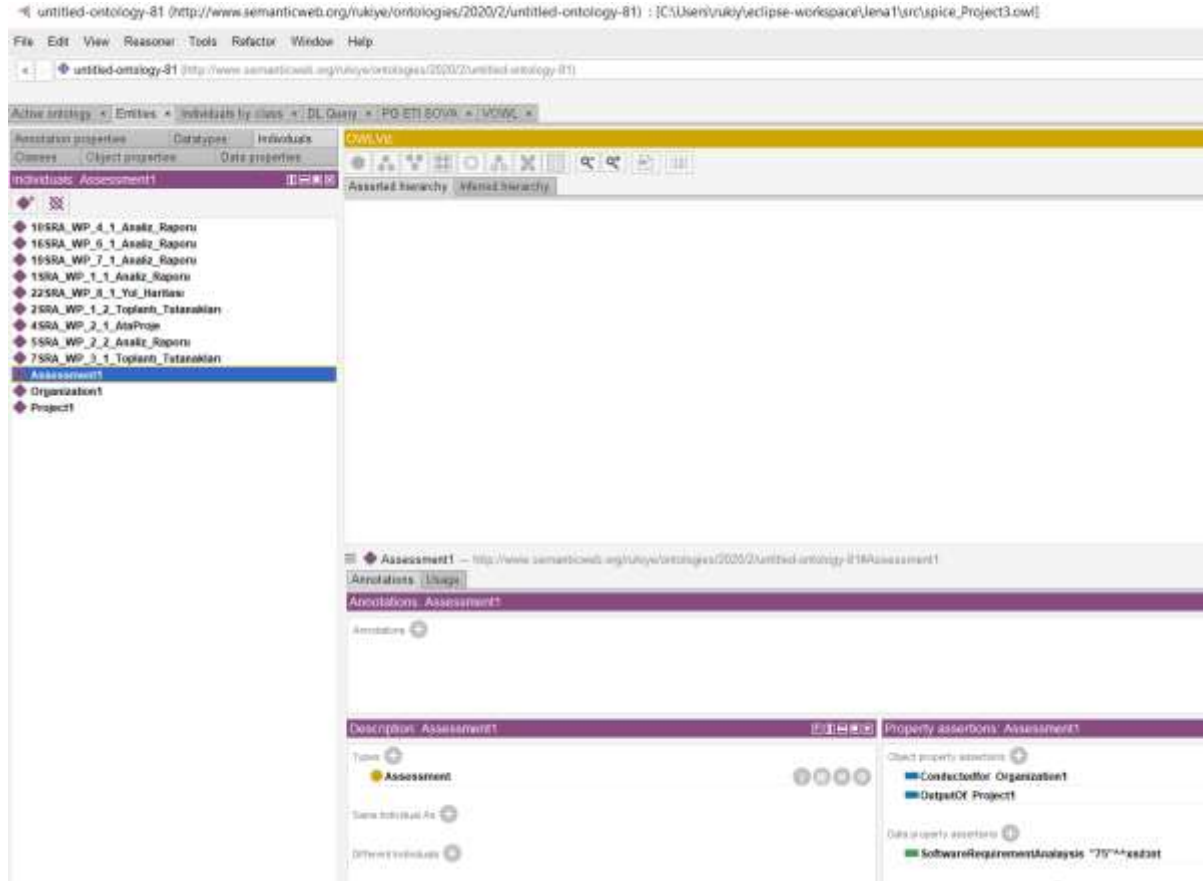
SWE.1.BP8: Yazılım gereksinimlerini tüm ilgili paydaşlara ya da değişikliklerden etkilenen kişilere bu süreçler aracılığıyla yaptığımız değişiklikler ve güncellemeler hakkında bilgilendirdiniz mi?

CMMI modelinin Proje Planlama ve Gereksinim Yönetimi, SPICE modelinin Proje Yönetimi ve TMMI modelinin Test Planlama süreçlerine ait değerlendirme soruları da Ekler kısmında gösterilmiştir. Değerlendirme sırasında projedeki süreçlere ait notlar girilmiştir. Mülakat esnasında seçilen süreç alanlarına ait uygulama sorularına verilen cevaplardan çıkarımlar yapılarak uygulamanın güçlü yanları, zayıf yanları gibi veriler araca girilmiştir. Her bir uygulamaya ait ortaya çıkan iş ürünleri verisi eklenmiştir. Veriler girilip incelemeler yapıldıktan sonra değerlendirmesi yapılan uygulamaların başarı yüzdeleri girilmiştir. Programa göre uygulamaların yüzdelerinin ortalaması sürecin başarı yüzdesini oluşturmuştur. Şekil 11’de mülakattan sonra araçtaki SPICE Modeline ait Yazılım Gereksinim Analizi sürecine girilen başarı yüzdelerinin ve iş ürünlerinin arayüz görüntüsü verilmiştir.



**Şekil 11.** Yazılım gereksinim analizi sürecinin arayüzü

Bu arayüzde veriler girilerek OWL dosyasına çekilen verilerden bireyler ve özellikler eklenmiştir. Şekil 12’de Yazılım Gereksinim Analizi sürecinin başarı oranının %75 olduğu görülmüştür. Değerlendirme yapılan yerin ismini vermemek adına Organization1 diye isimlendirilmiştir. Değerlendirmenin ismi Assessment1 ve değerlendirilen projenin ismi Project1 olarak girilmiştir. Bu süreç sonrası ortaya çıkan iş ürünleri ise arayüz üzerinden birey olarak eklenmiştir.



**Şekil 12.** Değerlendirme sonrası Yazılım Gereksinim Analizi sürecinin Protégé editöründeki OWL dosyasına eklenen bireylerin ve özelliklerin gösterimi

Ekler kısmında Gereksinim Yönetimi sürecine ait görüntüde değerlendirme sonucunda ortaya çıkan başarı yüzdesi ve iş ürünlerinin olduğu arayüz görüntüsü gösterilmiştir. Başarı oranının %70 olduğu görülmüştür. Gereksinim Yönetimi sürecine ait arayüz ile eklenen verilerin OWL dosyasına aktarıldıktan sonra ortaya çıkan iş ürünlerinin ve başarı yüzdesinin Protégé editörü ile gösterimi de mevcuttur.

1. Grup süreç değerlendirmesine göre CMMI modeline ait Gereksinim Yönetimi Süreci ve SPICE modeline ait Yazılım Gereksinim Analizi değerlendirmeleri kıyaslandığında iki sürecin arasında %5 oranında bir fark olduğu görülmüştür. Bu veri iki modelde yapılan değerlendirme sonuçlarının birbirine yakın olduğunu göstermektedir. Ayrıca Organization1 diye adlandırdığımız kurumun iki süreç alanının da yaklaşık olarak büyük bir oranda başarı

sağladığı görülmektedir. Bu süreç alanlarındaki uygulamalardaki başarı oranlarının ne ölçüde olduğu görülmektedir. Bu değerlendirme sonucunda organizasyonun gereksinim yönetimi konusundaki süreç yetkinliği gözlemlenmektedir.

Ekler kısmında Proje Planlama sürecine ait görüntüde değerlendirme sonucunda ortaya çıkan başarı yüzdesi ve iş ürünlerinin olduğu arayüz görüntüsü gösterilmiştir. Başarı oranının %76 olduğu görülmüştür. Proje Planlama sürecine ait arayüz ile eklenen verilerin OWL dosyasına aktarıldıktan sonra ortaya çıkan iş ürünlerinin ve başarı yüzdesinin Protégé editörü ile gösterimi de mevcuttur.

Proje Yönetimi sürecine ait görüntüde değerlendirme sonucunda ortaya çıkan başarı yüzdesi ve iş ürünlerinin olduğu arayüz görüntüsü gösterilmiştir. Başarı oranının %78 olduğu görülmüştür. Proje Yönetimi sürecine ait arayüz ile eklenen verilerin OWL dosyasına aktarıldıktan sonra ortaya çıkan iş ürünlerinin ve başarı yüzdesinin Protégé editörü ile gösterimi de mevcuttur.

Test Planlama sürecine ait görüntüde değerlendirme sonucunda ortaya çıkan başarı yüzdesi ve iş ürünlerinin olduğu arayüz görüntüsü gösterilmiştir. Başarı oranının %42 olduğu görülmüştür. Test Planlama sürecine ait arayüz ile eklenen verilerin OWL dosyasına aktarıldıktan sonra ortaya çıkan iş ürünlerinin ve başarı yüzdesinin Protégé editörü ile gösterimi mevcuttur.

2. Grup süreç değerlendirmesine göre CMMI modeline ait Gereksinim Proje Planlama ve SPICE modeline ait Proje Yönetimi süreç değerlendirmeleri kıyaslandığında iki sürecin arasında %2 oranında bir fark olduğu görülmüştür. Bu veri iki modelde yapılan değerlendirme sonuçlarının birbirine çok yakın olduğunu göstermektedir. Ayrıca Organization1 diye adlandırdığımız kurumun iki süreç alanının da büyük bir oranda başarı sağladığı görülmektedir. Bu süreç alanlarındaki pratikleri yeterli ölçüde uyguladıkları görülmektedir. Bu organizasyonun gereksinim yönetimi konusundaki süreç yetkinliği gözlemlenmektedir. Fakat TMMI modelindeki Test Planlaması süreç değerlendirmesinin uygulamalardaki başarı oranının %42 olması bu süreç alanında oldukça düşük puan alındığını göstermiştir. Bu düşük puan bu organizasyonun Test Planlaması süreç alanındaki süreç pratiklerinde eksikleri olduğunu ve bu süreç alanında iyileştirme adımlarına ihtiyaç olduğunu göstermiştir.

Bu değerlendirme sonuçlarının yani ayrı ayrı süreç referans modelleri ile yapılan süreç değerlendirmelerini ortak ontoloji olan YGOM ontolojisine entegre halinin Protégé editöründeki gösterimi Ekler bölümünde mevcuttur. Bu şekilde, arayüz ile eklenen üç modelin birleştirildiği ontolojide, iş ürünleri ve başarı yüzdeleri görülmektedir

Son olarak ontolojilerin görselleştirilerek doğrulanması sağlanmıştır. Görselleştirilen ontolojiler ile grafik temsilleri karşılaştırılmıştır. Kullanılan OWLGrEd editörü ile kullanıcılara ontoloji öğelerini sınıfları, veri türlerini, özellikleri ve bireylerini görselleştirme imkânı sağlanmıştır. Ekler bölümünde sırasıyla CMMI, SPICE, TMMI ve YGO modellerinin ontolojilerinin görselleştirilmesi verilmiştir. Bu sayede karmaşık ve uzun OWL dosyasını incelemek yerine " ontoloji okur-yazarlığı olmayanlar " için bile anlaşılması ve kullanılması kolay olan görselleştirme editörü ile tek seferde tüm ontolojiyi görme imkânı sağlanmıştır. Kullanılan bu görselleştiricinin uzaklaştırma ve yakınlaştırma yeteneği sayesinde ontolojileri incelemek kolaydır. Editörün yapısı gereği ontolojilerin görselleştirilmesi için UML sınıf diyagramları tarzında yapı sunan OWLGrEd, OWL sınıflarını UML sınıfları olarak, veri özelliklerini sınıf nitelikleri olarak, nesne özelliklerini ilişkilendirmeler olarak, bireyleri ise nesnelere gösterir.

OSPAT aracının arayüzünün objektif değerlendirilmesi için de bir bağımsız bir kullanıcı üzerinde anket yapılmıştır. Bu kullanıcıya araç üzerinde süreç değerlendirmesi yaptıktan sonra Tablo 12’deki sorular sorulmuştur ve cevapları alınmıştır (Charfi *et al.* 2015).

**Tablo 12.** Kullanıcı Arayüzü Anket Soruları ve Cevapları

No	Kullanıcı Arayüzü Anket Soruları	Kesinlikle Katılıyorum	Katılıyorum	Bilmiyorum	Katılmıyorum	Kesinlikle Katılmıyorum
1.	Yapmak istediğinizi gerçekleştirdiniz mi?		X			
2.	Arayüzün kullanımı kolay mıydı?	X				
3.	Kullanıcı arayüzünde bir şey beklediniz ve bulamıyor musunuz?				X	
4.	Her bir süreç alanının ve uygulamanın tanımı iyi tanımlanmış mı?	X				
5.	Araç bir işlemi yaparken sizi doğru yönlendirdi mi?		X			
6.	Arayüz sayfalarının içeriğini okumak kolay mı?	X				

Tablo 12’de verilen cevaplar doğrultusunda aracın ne kadar kullanışlı olduğu doğrulanmıştır. Verilen cevaplara göre aracın kullanımı oldukça kolaydır. Bir arayüzde olması beklenen özellikleri büyük bir oranda karşılamaktadır.

## ARAŞTIRMA BULGULARI VE TARTIŞMA

Bu tez aşağıdaki araştırma sorularına cevap bulmuştur.

1. Birden fazla süreç değerlendirme referans modellerini bir araya getirecek bir ontoloji neleri içermelidir?

Bu çalışma organizasyonlar tarafından popüler olarak kullanılan üç farklı modeli ortak bir ontolojide buluşturarak ontoloji tabanlı genel bir süreç değerlendirme aracı geliştirerek farklı süreçlerde değerlendirme yapma imkânı sunmuştur. Ayrıca üç modelin ortak bir ontolojide buluşması ontolojilerin eşleştirilebilir yapıda olma özelliği sayesinde gerçekleştirilmiştir. Bu ontolojiler ile modellerin ve modellere ait süreçlerin, uygulamaların, iş ürünlerinin ve başarı yüzdelerinin sorgulanabilir olduğu görülmüştür. Ontolojilerin görselleştirilmesi sayesinde ise kullanıcıların ontolojileri anlaması, doğrulaması ve güncellemesi kolaylaşmıştır. Daha önce ontolojilerin biçimsel, makineler tarafından anlaşılabilir ve paylaşılabılır bir dil ile ifade edilmesi gerektiğinden bahsedilmiştir(Gazel 2009). Geliştirilen araç için oluşturulan ontolojilerin bu şekilde ifade edildiği görülmüştür.

2. Birden fazla süreç referans modeli ile süreç değerlendirmesine destek olan bir aracı geliştirmenin avantajları nelerdir? Bu aracın özellikleri neler olmalıdır?

Bu araç ile yazılım projeleri üreten bir üniversitede üç farklı modelde yazılım süreç değerlendirmesinin yapıldığı doğrulanmıştır. Yapılan doğrulama ile geliştirilen aracın hedeflenen değerlendirme için kullanıma uygun olduğu gösterilmiştir. Bu aracın ontoloji tabanlı olma özelliği sayesinde aracın geliştirilmesinde kullanılan yazılım tarafından anlaşılabilir bir araç olduğu görülmüştür. Araçta kullanılan ontolojiler sayesinde modellerin ve süreçlerin güncellemesinde kolaylık sağlanmıştır. Araçta, üç süreç referans modeli için anlamlı sonuçlar elde etmek için farklı süreçler değerlendirilmiştir. Böylece organizasyonlara üç model için hangi seviyede olduğunu görme imkânı sağlanmıştır. Ayrıca bu araç üç model ile değerlendirme yapabildiği için aynı organizasyona ait farklı modellerle yapılan değerlendirme sonuçlarının karşılaştırılabildiği görülmüştür. Bunların yanında farklı organizasyonlara farklı projelerinin süreçlerini değerlendirme imkânı vermiştir. Bu araç kurum dışından bir bilirkişiye ihtiyaç duymadan öz değerlendirme yapmayı sağlamaktadır. Fakat bir kullanıcı olmadan otomatik değerlendirme yapma yeteneği yoktur. Aracın yapılan anket çalışması neticesinde kurumların öz değerlendirmelerini yapmaları için basit ve görsel bir arayüze sahip olduğu görülmüştür.

Aracın yukarıda açıklanan özellikler açısından yapılan doğrulamalar ile beklenen özellikleri karşıladığı görülmüştür. Fakat aracın tasarım açısından kullanımı her ne kadar kolay olsa da işlevsellik açısından iyileştirilebilir. Bunların yanında bu araç aynı anda bir projenin aynı sürecini farklı modellerle senkron değerlendirme yapmasına imkân vermemektedir. Bu araca göre bir modele ait sürecin değerlendirilmesi için diğer modeldeki süreç değerlendirmesinin bitmesi gerekir. Yani paralel değerlendirme yapma özelliği yoktur. Organizasyonlarda süreç değerlendirmesi tek bir ekip tarafından yapılabileceği gibi birden fazla ekip tarafından da gerçekleştirilebilir olduğundan bu özellik eklenmiş olsaydı bu araç ile aynı anda birden çok proje değerlendirebilecekti (Yürüm 2014).



## SONUÇLAR VE ÖNERİLER

Bu çalışmada OSPAT adındaki üç farklı modelde değerlendirme yapmaya olanak sağlayan ontoloji tabanlı yazılım süreç değerlendirme aracı geliştirilmiştir. Bu aracın doğrulanması için referans model tabanlı süreç değerlendirmesi yapan yazılım projeleri üreten bir üniversitede üç farklı modelden seçilen süreç alanları kapsamında beş farklı süreç alanında yazılım süreç değerlendirmesi yapılmıştır. Daha sonra bu çalışmada yaygın modeller ve literatürdeki mevcut araçlar incelenmiştir. Literatürdeki ontoloji tabanlı farklı modellerde değerlendirme yapabilen süreç değerlendirme destek aracı eksikliği çerçevesinde ortaya çıkan araştırma sorularına karşın OSPAT aracı geliştirilmiştir. Bu araç ile üç süreç referans modeline ait ontolojilerden sorgular çekilerek değerlendirmelerin bir arada yapılması sağlanmıştır. Üç modeli tek bir platformda görmek aynı proje üzerinde üç model ile yapılan değerlendirmeleri karşılaştırma imkânı vermiştir. Bu aracın bir süreç alanında değerlendirme yapma, OWL dosyasına girme ve girilen değerlendirmeyi silme özelliklerine ilişkin kullanıcı senaryoları verilmiştir. Bu çalışma ile ontoloji tabanlı bir süreç değerlendirme aracı geliştirmek için; ontolojileri oluşturma programı olarak Protégé programı, geliştirme ortamı olarak Eclipse, kavram tabanı olarak OWL ontolojisi tanımlama dili, sorgulama dili olarak SPARQL, programlama dili olarak Java, ontolojileri yüklemek için ise JENA kütüphanesi kullanılmıştır. Süreç değerlendirmesi yapılan modellerin ontolojilerini, içerisindeki sınıfları, alt-sınıfları, ilişkileri, özellikleri ve eklenen bireyleri, daha kolay anlayabilmek ve bunları bütünüyle görebilmek için OWLGrEd editörü ile görselleştirmeleri sağlanmıştır. Bu araç üç modelin eşleştirilerek bir ontolojide buluşabileceğini gösterdi. Bu modellerde eşleştirilen süreç alanlarının ne seviyede benzer oldukları gösterilmiştir. Ayrıca gelecekte farklı referans modellerin bir ontolojide buluşabileceğini de gösterilmiştir. Oluşturulan ontolojiler açık kaynaklı olarak paylaşılabilir ve tekrar kullanılabilir olduğu için ilgilenen uzmanlar için model bilgisine erişim ve bu bilginin yayılımı açısından kolaylık sağlayacaktır. Bu araç kurum dışından bir bilirkişiye ihtiyaç duymadan öz değerlendirme yapmaya imkân verir. Ayrıca bu araç organizasyonlara düzenli yapacakları süreç değerlendirmelerinin sonuçlarını karşılaştırma yapma imkânı sağlar. Bu sayede organizasyonlara yazılım süreç değerlendirmelerindeki ilerlemelerini görmeleri sağlanmıştır.

Bunların dışında bu çalışmada sadece bir organizasyonda doğrulama yapılmıştır. Organizasyonlara farklı projelerinin süreçlerini değerlendirme imkânı vermiştir. Fakat zaman kısıtından dolayı farklı projelerde doğrulamalar yapılamamıştır. Fakat daha fazla

organizasyonla farklı projelerin deęerlendirmesi aracın doęrulaması aısından daha iyi olabilir. Ayrıca yaygın kullanımları dolayısıyla bu alıřmada seilerek kullanılan üç sre referans modeli dıřında bařka yazılım sre referans modellerine uygun deęerlendirme özellięi de gelecekte bu araca eklenebilir. Fakat dięer sre referans modellerinden SPICE ve CMMI tabanlı btn sre referans modeller de benzer kavramlara sahip olduęundan OSPAT aracı ile sre deęerlendirmesini desteklemeye uygundur. Bu alıřmada CMMI sre referans modelinin v1.3 srmne gre ontolojiler oluřturulmuřtur. Fakat CMMI v2.0 srmnde olan kavramlardaki deęiřiklikler modele entegre edilerek gelecekteki alıřmalarda ontolojiler gncellenebilir. SPICE modelinde de son zamanlarda yapılan deęiřiklikler ile pratikleri deęerlendirme kategorileri 4'ten 6'ya ıkarıldıęı iin gelecekteki alıřmalarda bu deęiřiklikler eklenerek ontolojilerde gncellemeler yapılabilir.



## KAYNAKLAR

- 15504, A.I.I., 2003. Information Technology–Process Assessment.
- Athiththan, K., Rovinsan, S., Sathveegan, S., Gunasekaran, N., Gunawardena, K.S. and Kasthurirathna, D., 2018. An ontology-based approach to automate the software development process, 2018 IEEE International Conference on Information and Automation for Sustainability (ICIAfS). IEEE, pp. 1-6.
- Aytaç, T., 2008. Araçlara Dayalı Kapsamlı Süreç Değerlendirme Sistemi.
- Casey, V. and McCaffery, F., 2012. Medi SPICE and the Development of a Process Reference Model for Inclusion in IEC 62304, ICISOFT.
- Charfi, S., Ezzedine, H. and Kolski, C., 2015. RITA: a useR Interface evaluation framework. J. Univers. Comput. Sci., 21 (4), 526-560.
- Chrissis, M.B., Konrad, M. and Shrum, S., 2011. CMMI for development: guidelines for process integration and product improvement. Pearson Education.
- Çiflikli, E.G. and Coşkunçay, A., 2018. An ontology to support TMMi-based test process assessment, International Conference on Software Process Improvement and Capability Determination. Springer, pp. 345-354.
- Feloni, D. and Braga, R.T.V., 2015. Methodologies for Evaluation and Improvement of Software Processes in the Context of Quality and Maturity Models: a Systematic Mapping, CIbSE, pp. 123.
- Garcia, I. and Pacheco, C., 2009. Toward automated support for software process improvement initiatives in small and medium size enterprises. Software engineering research, management and applications 2009. Springer, 51-58.
- Garousi, V. and van Veenendaal, E., 2021. Test Maturity Model integration (TMMi): Trends of worldwide test maturity and certifications. IEEE Software.
- Gazel, S., 2009. "CMMI-DEV" için bir ontoloji ve CMMI esaslı süreç değerlendirmenin ontoloji tabanlı bir araç ile desteklenmesi, Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü.
- Gazel, S., Sezer, E.A. and Tarhan, A., 2012. An ontology based infrastructure to support CMMI-based software process assessment. Gazi University Journal of Science, 25 (1), 155-164.
- Gazel, S., Tarhan, A. and Sezer, E., 2009. A CMMI ontology for an ontology-based software process assessment tool, proceedings of the 16th EuroSPI2 Conference.
- Grambow, G., Oberhauser, R. and Reichert, M., 2012. Towards automated process assessment in software engineering.
- Gruber, T.R., 1993. A translation approach to portable ontology specifications. Knowledge acquisition, 5 (2), 199-220.
- Hunter, R., Robinson, G. and Woodman, I., 1997. Tool support for software process assessment and improvement. Software Process: Improvement and Practice, 3 (4), 213-223.
- ISO, I., 2004. ISO/IEC 15504-Information technology-Process Assessment. Geneva, Switzerland: Author.

- Kalaycı, O., 2007. Yöneticiler için Doğru Sorular CMMI. Shamrock Süreç İyileştirme ve Yenilikçilik, Kanada.
- Kitson, D.H. and Masters, S.M., 1993. An analysis of SEI software process assessment results: 1987-1991, Proceedings of 1993 15th International Conference on Software Engineering. IEEE, pp. 68-77.
- Lepasaar, M. and Makinen, T., 2002. Integrating software process assessment models using a process meta model, IEEE international engineering management conference. IEEE, pp. 224-229.
- Liao, L. and Leung, H.K., 2014. A Software Process Ontology and Its Application.
- Lok, R.H. and Walker, A., 1997. Automated tool support for an emerging international software process assessment standard, Proceedings of IEEE International Symposium on Software Engineering Standards. IEEE, pp. 25-35.
- Mesquida, A.L., Mas, A., Amengual, E. and Calvo-Manzano, J.A., 2012. IT Service Management Process Improvement based on ISO/IEC 15504: A systematic review. Information and Software Technology, 54 (3), 239-247.
- Messnarz, R., Ross, H.L., Habel, S., König, F., Koundoussi, A., Unterreitmayr, J. and Ekert, D., 2009. Integrated Automotive SPICE and safety assessments. Software Process: Improvement and Practice, 14 (5), 279-288.
- Mollaoğlu, A.A. and Bolatan, G.İ.S., 2020. Yazılım Süreçlerinde Kalite ve Yetenek Olgunluk Modeli. Organizasyon ve Yönetim Bilimleri Dergisi, 12 (1), 77-97.
- Mora, M., Cervantes-Pérez, F., Garrido, L., Wang, F. and Angel-Sicilia, M., 2010. On KM, KMS and ontology-based KMS in the domain of SwE standards and models of processes: A conceptual survey. Intelligent Decision Technologies, 4 (1), 75-98.
- Noy, N.F. and McGuinness, D.L., 2001. Ontology development 101: A guide to creating your first ontology. Stanford knowledge systems laboratory technical report KSL-01-05 and ....
- Proença, D. and Borbinha, J., 2018. Formalizing ISO/IEC 15504-5 and SEI CMMI v1. 3-Enabling automatic inference of maturity and capability levels. Computer Standards & Interfaces, 60, 13-25.
- Rocha, A.R., Montoni, M., Santos, G., Mafra, S., Figueiredo, S., Albuquerque, A. and Mian, P., 2005. Reference model for software process improvement: A brazilian experience, European Conference on Software Process Improvement. Springer, pp. 130-141.
- Rout, T.P., 2003. ISO/IEC 15504—Evolution to an international standard. Software Process: Improvement and Practice, 8 (1), 27-40.
- Rout, T.P. and Tuffley, A., 2007. Harmonizing iso/iec 15504 and cmmi. Software Process: Improvement and Practice, 12 (4), 361-371.
- Soydan, G.H. and Kokar, M., 2006. An OWL ontology for representing the CMMI-SW model, Workshop on Semantic Web Enabled Software Engineering (SWESE). Citeseer.
- Tarhan, A. and Giray, G., 2017. On the use of ontologies in software process assessment: a systematic literature review, Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering, pp. 2-11.
- Tarhan, A., Turetken, O. and Ilisulu, F., 2015. Business process maturity assessment: state of the art and key characteristics, 2015 41st Euromicro Conference on Software Engineering and Advanced Applications. IEEE, pp. 430-437.

- Tayamanon, T., Suwannasart, T., Wongchingchai, N. and Methawachananont, A., 2011. TMM appraisal assistant tool, 2011 21st International Conference on Systems Engineering. IEEE, pp. 329-333.
- Team, C.P., 2006. CMMI for Development, version 1.2.
- Türkyılmaz, İ. and Yaşar, C., 2008. Semantik web teknolojileri. Akademik Bilişim 2008.
- van Veenendaal, E., 2018. Test maturity model integration (tmimi): Guidelines for test process improvement, release 1.2. TMMi Foundation. Ireland.
- Vanamali, B., Bella, F. and Hörmann, K., 2008. From CMMI to SPICE-Experiences on How to Survive a SPICE Assessment Having Already Implemented CMMI, 2008 32nd Annual IEEE International Computer Software and Applications Conference. IEEE, pp. 1045-1052.
- Walker, A. and Lok, H., 1995. SPICE Assessments using the SEAL assessment tool. Software Engineering Applications Laboratory.
- Yolcular, F. and Erdoğan, Ş.Z., 2009. A questionnaire based method for CMMI level 2 maturity assessment. Journal of aeronautics and space technologies, 4 (2), 39-46.
- Yücalar, F., 2006. Kurumsal Bilgi Sistemi uygulamaları için popüler proje yönetim yaklaşımları.
- Yürüm, O.R., 2014. GSPA: Genel bir yazılım süreç değerlendirme aracı, Orta Doğu Teknik Üniversitesi.
- Yürüm, O.R., Top, Ö.Ö., Ertugrul, A.M. and Demirörs, O., 2014. Yazılım Süreç Değerlendirme Araçlarının Karşılaştırılması: Bir Çoklu Durum Çalışması, UYMS.

## ÖZGEÇMİŞ

Kişisel Bilgiler	
<b>Adı Soyadı:</b>	Rukiye BAKIRHAN
<b>Doğum tarihi:</b>	
<b>Doğum Yeri:</b>	
<b>Uyruğu:</b>	
<b>Adres:</b>	
<b>Tel:</b>	
<b>E-mail:</b>	
Eğitim	
<b>Lise:</b>	Gazi Kars Anadolu Lisesi
<b>Lisans:</b>	İstanbul Arel Üniversitesi Mimarlık- Mühendislik Fakültesi Bilgisayar Mühendisliği (İngilizce-% 100 Burslu)
Yabancı Dil Bilgisi	
	İngilizce: İyi Almanca: Biraz