



T.C.
OSMANIYE KORKUT ATA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
YÖNETİM BİLİŞİM SİSTEMLERİ ANA BİLİM DALI

ANLAMSAL AĞ ODAKLI KİŞİSELLEŞTİRİLMİŞ
E-TİCARET ÜRÜN ÖNERİ SİSTEMİ

YÜKSEK LİSANS TEZİ

Hilmi Cihan MERAL

OSMANIYE / 2022

T.C.
OSMANİYE KORKUT ATA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
YÖNETİM BİLİŞİM SİSTEMLERİ ANA BİLİM DALI

ANLAMSAL AĞ ODAKLI KİŞİSELLEŞTİRİLMİŞ E-TİCARET

ÜRÜN ÖNERİ SİSTEMİ

YÜKSEK LİSANS TEZİ

HİLMİ CİHAN MERAL

Danışman: Dr. Öğr. Üyesi Fatih Çağatay BAZ

Jüri Üyesi: Dr. Öğr. Üyesi Ahmet DOĞAN

Jüri Üyesi: Dr. Öğr. Üyesi Mustafa Mikail ÖZÇİLOĞLU

OSMANİYE / 2022

Osmaniye Korkut Ata Üniversitesi Sosyal Bilimler Enstitüsü Müdürlüğüne;

“Anlamsal Ağ Odaklı Kişiselleştirilmiş E-Ticaret Ürün Öneri Sistemi” başlıklı çalışma jürimiz tarafından Ana Bilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Başkan: Dr. Öğr. Üyesi Fatih Çağatay BAZ

(Danışman)

Üye: Dr. Öğr. Üyesi Ahmet DOĞAN

Üye: Dr. Öğr. Üyesi Mustafa Mikail ÖZÇİLOĞLU

ONAY

Yukarıdaki imzaların, adı geçen öğretim elemanlarına ait olduklarını onaylarım.

.../.../2022

Doç. Dr. Ebru GÜHER

Enstitü Müdürü

NOT: Bu tezde kullanılan ve başka kaynaktan yapılan bildirişlerin, çizelge, şekil ve fotoğrafların kaynak gösterilmeden kullanımı, 5846 sayılı Fikir ve Sanat Eserleri

Kanunu’ndaki hükümlere tabidir.



T.C.
OSMANIYE KORKUT ATA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
TEZ ÇALIŞMASI ETİK BEYANI FORMU

FORM
TEZLİ YL-15

SOSYAL BİLİMLER ENSTİTÜSÜ MÜDÜRLÜĞÜNE

Osmaniye

Osmaniye Korkut Ata Üniversitesi Sosyal Bilimler Enstitüsü Tez Yazım kurallarına uygun olarak hazırladığım; başlıklı Yüksek Lisans Tez çalışmamda;

- Sunulan verilerin; gerekli izinleri alınmış ve denetimli laboratuvar koşullarında tarafımdan veya ilgili görevlilerce elde edildiğini ya da izin belgesine dayalı olarak ve kaynak göstermek suretiyle kullanıldığını,
- Kullanılan veriler üzerinde herhangi bir değişiklik veya eksiltme yapılmaksızın etik kurallara uygun olarak işlenip sunulduğunu,
- Maddi veya manevi destek sağlamış olan Kurum, Kuruluş ve kişilere destek türü de belirtilerek, varsa proje protokol numarası ile yoksa ismen Ön Söz/Teşekkür Bölümlerinde yer verildiğini,
- Yararlanılan kaynaklara Tez metni içinde atıf göstermek suretiyle değinildiğini ve bunların Kaynaklar Bölümüne eklendiğini,
- Teknik/Bilimsel Eser niteliği taşıyan Tezin özgün parçalarının bir başka ortamdan kopyalanarak alınmadığını ve bu parçaların bir başka Kurum/Kuruluş bünyesinde akademik amaç veya unvan almak amacıyla hiçbir suretle kullanılmadığını ve bir başkasının kullanımına izin verilmediğini,
- Burada belirttiğim hususların aksinin tespit edilmesi halinde tüm yasal sorumluluğun şahsıma ait olduğunu beyan ederim.

...../..... / 20.....

İmza

(Adı ve Soyadı)
Öğrenci

ÖĞRENCİ NO	
ANABİLİM/ ANA SANAT DALI	
BİLİM/SANAT DALI	
ENSTİTÜ KAYIT TARİHİ/...../ 20.....

Osmaniye Korkut Ata Üniversitesi
Sosyal Bilimler Enstitüsü
80000 Osmaniye / TÜRKİYE

<http://sbe.osmaniye.edu.tr/>

Tel: +90 328 827 10 00/4107-4108
E-Posta: sbe@osmaniye.edu.tr

ÖZET
ANLAMSAL AĞ ODAKLI KİŞİSELLEŞTİRİLMİŞ E-TİCARET ÜRÜN
ÖNERİ SİSTEMİ

HİLMİ CİHAN MERAL

Yüksek Lisans, Yönetim Bilişim Sistemleri Ana Bilim Dalı

Danışman: Dr. Öğr. Üyesi Fatih Çağatay BAZ

Ağustos 2022, 49 sayfa

Anlamsal ağ teknolojisi üzerine yapılan çalışmalar ile birlikte bu teknolojiye olan ilgi zaman içerisinde artış göstermiştir. Günümüzde optimizasyon, kullanıcı deneyimini artırma, iyileştirme gibi amaçlar için kullanılmak üzere verinin önemi artış göstermiştir. Bu artış beraberinde verilerin okunabilirliği ihtiyacını doğurmuştur. Anlamsal ağ bu noktada büyük veri setlerini makinelerinde insanlar gibi yorumlayabileceği biçime sokarak bilginin okunabilirliğini artırmıştır. Bu çalışmada anlamsal ağ teknolojisinin kullanıldığı teknolojik çalışmalardan yola çıkarak alanyazın incelemesiyle birlikte, kullanıcılara isteklerine ilişkin ilgili ürünleri önerebilecek bir elektronik ticaret sistemi geliştirilmiştir. Sistem geliştirmesi aşamasında ilk olarak kabul görmüş ontoloji oluşturma yöntemleri incelenmiş ve ihtiyaç duyulan ontolojiyi oluşturmak için bir yöntem seçilmiştir. Tez kapsamında birinci bölümde anlamsal ağın teknolojik ihtiyaçları karşılama konusunda kabiliyetleri tanıtılmıştır. İkinci bölümde anlamsal ağ ve içerisinde kullanılan teknolojiler açıklanmıştır. Üçüncü bölümde anlamsal ağ ile uygulama geliştirme yapabilmek için ihtiyaç duyulan araçlar anlatılmıştır. Dördüncü bölümde uygulama geliştirme adımları açıklanmıştır. Beşinci bölümde başarılı bir şekilde geliştirilen sistem hakkında bilgi verilmiş ve böylece anlamsal ağ kullanımı örnekenmiştir. Son olarak da tasarlanan sistemi geliştirmek için öneriler paylaşılmıştır.

Anahtar kelimeler: Anlamsal ağ, Ontoloji, Web 3.0, Owl, RDF, SPARQL, Jena, Apache Jena Fuseki.

ABSTRACT**SEMANTIC WEB ORIENTED PERSONALISED
E-COMMERCE PRODUCT SUGGESTION SYSTEM****HİLMİ CİHAN MERAL****Master Thesis, Department of Management Information Systems****Supervisor: Asst. Prof. Fatih Çağatay Baz****August 2022, 49 pages**

The interest to semantic web technology have been increased with studies through time. Importance of the data have been increased nowadays to meet requirement of using to optimisation, user experience improvement etc. This increment have come along with meaningful information requirement. To fulfill these requirements and increase the legibility of data, semantic web have provided machines ability to interpret data just as humans. In this study, an electronic commerce system have been developed to offer more relevant products to users by searching literature and, examples which used semantic web technology. At the stage of development of the system, one of approved ontology development methods has been choosed first. Through the scope of thesis sections firstly, semantic web's ability of fulfilling demands has been introduced. In the second section, semantic web and its technologies within has been explained. Third section have explained, necessary tools to make development in semantic web. In the 4th section information about application development steps have been given. 5th section have included knowledge about successfully developed system. Lastly suggestions have been given to improve designed system.

Keywords: Semantic web, Ontology, Web 3.0, Owl, RDF, SPARQL, Jena, Apache Jena Fuseki.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT	v
ÖN SÖZ	ix
KISALTMALAR	x
TABLolar LİSTESİ	xi
ŞEKİLLER LİSTESİ	xii

BÖLÜM I

GİRİŞ

1.1.Elektronik Ticaret	1
------------------------------	---

BÖLÜM II

KURAMSAL AÇIKLAMALAR VE İLGİLİ ARAŞTIRMALAR

2.1.Anlamsal Ağın Tanımı.....	3
2.2.Anlamsal Ağ'da Ontoloji Kavramı.....	5
2.3.KTÇ (Kaynak Tanımlama Çerçevesi) Nedir?	6
2.4.OWL (İnternet Ontoloji Dili) Dili	8
2.5.URI (Standartlaşmış Kaynak Tanımlayıcı) Nedir?.....	10
2.6.URI ile URL'nin Karşılaştırılması.....	11
2.7.SPARQL Nedir?	13
2.8.SPARQL ile SQL Veritabanı Sorgu Dillerinin Karşılaştırılması	16
2.9.Literatürde Yapılan Çalışmalar	17

BÖLÜM III

ANLAMSAL AĞ İÇİN TEKNOLOJİK ARAÇLAR

3.1.Protégé: Ontoloji Oluşturma Uygulaması.....	19
3.2.Apache Jena Fuseki Sunucusu	20
3.3.Jena API.....	21
3.4.Java Programlama Dili.....	22
3.5.Spring Framework Teknolojisi	23
3.6.Thmeleaf Aracının Kullanımı	24
3.7.Bootstrap Kullanımı.....	24

BÖLÜM IV

YÖNTEM

4.1.Ontoloji Oluşturma	25
4.2.Apache Jena Yerel Sunucu Oluşturma	27
4.3.Spring Projesi Oluşturma.....	28
4.4.Java Kütüphanesine Jena API Ekleme.....	30
4.5.Java Geliştirme Ortamı Oluşturma	31
4.6.Uygulamaya Thymeleaf ve Bootstrap Ekleme	32

BÖLÜM V

UYGULAMA

5.1.Uygulamanın Amacı	33
5.2.Uygulama Arayüzü	34

BÖLÜM VI

SONUÇ VE ÖNERİLER

KAYNAKÇA	37
EKLER.....	39
ÖZGEÇMİŞ.....	49



ÖN SÖZ

Yüksek Lisans eğitimim ve tez çalışmam süresince bilgi ve deneyimlerini paylaşan ve bilimsel katkılarıyla destek olan sayın Dr. Öğr. Üyesi Fatih Çağatay BAZ'a tez çalışmama olan katkılarından ötürü, ayrıca bugünlere gelmemde büyük emekleri olan ve hiçbir zaman desteğini esirgemeyen değerli aileme teşekkürü borç bilirim.



KISALTMALAR

RDF: Resource Description Framework (Kaynak Tanımlama Çerçevesi)

URI: Uniform Resource Identifier (Standartlaşmış Kaynak Tanımlayıcı)

URL: Uniform Resource Locator (Standartlaşmış Kaynak Bulucu)

IRI: Internationalized Resource Identifier (Uluslararasılaştırılmış Kaynak Tanımlayıcı)

XML: Extensible Markup Language (Genişletilebilir İşaretleme Dili)

EDI: Electronic Data Interchange (Elektronik Veri Değişimi)

W3C: World Wide Web Consortium (Dünya Çapında İnternet Konsorsiyumu)

Turtle: Terse RDF Triple Language (Öz Üçlü Kaynak Tanımlama Çerçevesi Dili)

N-Triples: Simplified Version Of Turtle (Basitleştirilmiş Öz Üçlü Kaynak Tanımlama Çerçevesi Dili Sürümü)

SPARQL: SPARQL Protocol and RDF Query Language (SPARQL Protokol ve Kaynak Tanımlama Çerçevesi Sorgulama Dili)

SQL: Structured Query Language (Yapısallaşmış Sorgulama Dili)

OWL: Web Ontology Language (İnternet Ontoloji Dili)

API: Application Programming Interface (Uygulama Programlama Arayüzü)

SWRL: Semantic Web Rule Language (Anlamsal Ağ Kural Dili)

JVM: Java Virtual Machine (Java Sanal Makinesi)

HTML: HyperText Markup Language (Köprümetin Biçimlendirme Dili)

CSS: Cascading Style Sheets (Basamaklı Stil Sayfaları)

TABLULAR LİSTESİ

	Sayfa
Tablo 1. World Wide Web ile anlamsal ağ teknolojisinin mukayesesi	5
Tablo 2. Standartlaşmış Kaynak Tanımlayıcı Örnekleri.....	11
Tablo 3. URI Sözdizim Yapısı	12
Tablo 4. SPARQL'in Sağladığı Faydalar.....	14
Tablo 5. SPARQL Dilinde Yazılmış Temel Sorgular.....	15
Tablo 6. SPARQL ile SQL Arasındaki Farklılıklar	16
Tablo 7. Jena API ile Anlamsal Ağın Karşılaştırılması	21



ŞEKİLLER LİSTESİ

	Sayfa
Şekil 1. Semantik Ağın Mimari Yapısı.....	4
Şekil 2. Semantik Ağın Mimari Yapısı.....	6
Şekil 3. Kaynak Tanımlama Çerçevesi Örneği.....	7
Şekil 4. Kaynak Tanımlama Çerçevesi Modelinin Gösterimi	7
Şekil 5. Uygulamada Kullanılan Kaynak Tanımlama Çerçevesi Bölümü.....	8
Şekil 6. Uygulamada Kullanılan OWL Bölümü	9
Şekil 7. Protégé Ontoloji Geliştirme Uygulamasının Arayüzü.....	20
Şekil 8. Yerel Ağ Üzerinde Apache Jena Fuseki Sunucusunun Arayüzü	20
Şekil 9. Ontolojide Oluşturulan Sınıfların Görünümü.....	25
Şekil 10. Uygulama Ontolojisinin İskelet yapısı	26
Şekil 11. Ontolojide Kullanılan Sınıfların Nesne Özellikleri	27
Şekil 12. Apache Jena Fuseki Sunucusunun Ana Ekran Görünümü	28
Şekil 13. Spring Projesi Oluşturma Sayfasının Görünümü	28
Şekil 14. Uygulama Geliştirme Ortamına Spring Projesi Yükleme Ekranı	29
Şekil 15. Projeye Jena Kütüphanesi Yükleme Ekranı	30
Şekil 16. Uygulamanın Proje Görünümü.....	31
Şekil 17. Thmeleaf ve Bootstrap bağımlılıklarının görünümü	32
Şekil 18. Uygulama Arayüzünün Görüntüsü	34

BÖLÜM I

GİRİŞ

Günümüzde kullanılan e-ticaret sistemlerinde yapılan ürün önerilerinin kullanıcıların tercihlerinden ziyade çerezler aracılığı ile toplanan daha önceki tercihlerini baz alarak yapılması ve bunun bir sonucu olarak yapılan önerilerin alakasız olabildiği gözlemlenmiştir. Bu tez kapsamında java programlama dili ve onunla birlikte spring framework, thymeleaf, bootstrap gibi teknolojileri anlamsal ağ ile geliştirme yapmak için ihtiyaç duyulan apache jena fuseki sunucusu ve jena ile ilgili kütüphaneler ile entegre ederek bir ürün öneri sistemi geliştirilmiştir. Tasarlanan bu sistem ile daha önceki gezinmelerin baz alınarak yapıldığı ürün önerilerinin yerine verilerin anlamlandırılarak daha tutarlı ürün önerilerinin yapılabilmesine olanak sağlanması amaçlanmıştır.

1.1.Elektronik Ticaret

Endüstri 4.0 değişiminin yaşandığı çağımızda pazarlama dünyasında da rekabet giderek artış göstermiştir. Pazarlama alanı günlük aktivitelere farklı şekillerde etkilemeye başlamıştır. Bu yüzden ekonomik gelişim ve büyümede pazarlama önemli bir rol oynar hale gelmiştir. Böylece etkili bir pazarlama sistemi bir ulusun geleceğine katkı sağlayacak hale gelmiştir. Başarılı bir pazarlama yapabilmek için müşteri memnuniyetinin gözetilmesinin gerektiği gözlemlenmiştir. Medeniyette meydana gelen değişimler ile birlikte sosyal ortamda ve insanların davranışlarında değişim gerçekleşmiştir. Bu davranışsal değişimler kullanıcıların hizmet beklentilerini ve tatmin ölçüğünü değiştirmiştir. Böylece kullanıcıların beklentilerini karşılamak ve memnuniyet seviyelerini artırmak için internet ağı üzerinden kolayca alışveriş yapabilmeye olanak sağlayacak elektronik ticaret ortamı gelişim göstermiştir.

Alanyazında yapılan benzer çalışmalarda elektronik ticaretin kısaltması olarak gösterilen E-ticaret, tarafların birbirleriyle fiziksel olarak görüşmesine gerek kalmadan bilgisayar ağları üzerinden elektronik işlemler aracılığı ile ticaret yapabilmeye olanak

sağlayan teknolojiler bütünü olarak tanımlanmıştır (Purwanto, 2022; Peng & Zhang & Wang & Liang, 2019; Qureshi & Fang & Ramsey & McCole & Ibbotson & Compeau, 2009). Teknolojik gelişmelere ve yeni buluşlara duyulan ihtiyaç bilişim teknolojileri, eğitim, sağlık, e-ticaret gibi pek çok alanda artış göstermiştir. Günümüzde yoğun olarak kullanılan ve kavramsal olarak halen gelişmekte olan e-ticaret, geleneksel alışveriş işlemlerini elektronik ve bilgi teknolojilerini kullanarak yükseltmeyi hedeflemiştir (Qin, 2009, s.1).

Son kullanıcıların sayılarının giderek çoğalması, beraberinde işlem sayılarında da artışa sebep olmuştur. Bu artış kullanıcılara sunulacak hizmet konusunda yeni ihtiyaçları doğurmuştur. Fakat gerek teknolojik geliştirmelerin yapılabilmesi için ihtiyaç duyulan bütçenin fazla olması gerekse yeni buluşların ortaya çıkışının uzun sürmesi ve kullanılabilir hale gelmeden önce bir dizi testlerden geçmek zorunda olması ve veri bilimi ile ilgili çalışmalarda sürekli donanımsal veya yazılımsal geliştirme yapmanın uygulanabilir olmayışı hizmet sağlayıcıları alternatif çözüm arayışlarına yöneltmiştir. Optimizasyon ve performans alanında yaşanan gelişmeler ile birlikte bu arayışlara çözüm önerileri sunulurken bu noktada ihtiyaçlar karşılanabilir hale gelmeye başlamıştır. Dökümanların sadece birbirleriyle bağlantılı olmasının yanı sıra bu dökümanların anlamlarıyla birlikte tanınabileceği bir internet ağı oluşturmayı hedefleyen anlamsal ağ (Berners-Lee, 2004) alanında yapılan çalışmalar ile büyük veri setleri üzerinde yapılabilecek işlemlerin kapasitesi ve hızı artış göstermiştir. Böylece elektronik ticaret gibi büyük veri setlerinde hızlı ve doğru işlem yapmanın önemli olduğu alanlarda, optimizasyonu iyi bir şekilde yapılmış biçimde anlamsal ağ teknolojisi kullanılarak donanım veya yazılım yükseltmelerinin beraberinde getireceği yükün hafifletilebilmesine olanak sağlanmıştır.

BÖLÜM II

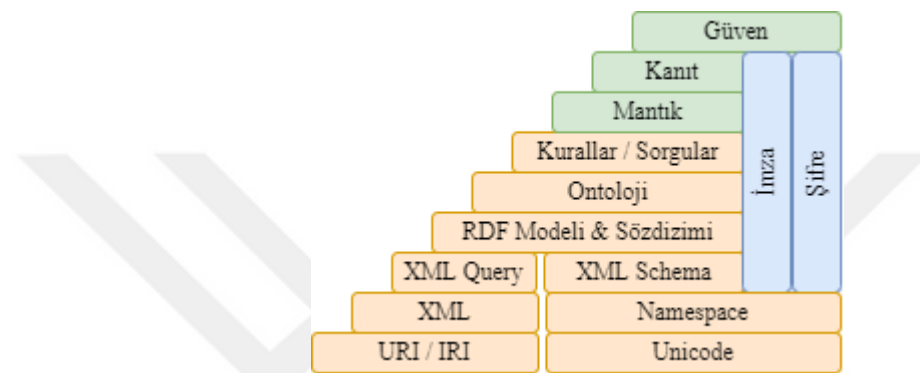
KURAMSAL AÇIKLAMALAR VE İLGİLİ ARAŞTIRMALAR

Çalışmanın bu bölümünde anlamsal ağ teknolojisinin yapısal bileşenlerin tanımı yapılmıştır. Tez kapsamında örnek olarak geliştirilmiş uygulama içerisindeki karşılıkları gösterilmiştir.

2.1.Anlamsal Ağın Tanımı

Semantik sözcüğü özünde sözdizimleriyle ilişkili olup Türk Dil Kurumunda anlam bilimi olarak tanımlanmıştır (TDK, t.y.). Sözdizimleri birçok dil içerisinde bir şeyi nasıl söyleyeceğimizi belirtmiştir. Anlamsal ağ (Semantik ağ) da benzer bir şekilde verilerin birbirleriyle anlamsal bir bağ oluşturarak makineler tarafından da anlaşılabilir olmasına zemin hazırlamıştır. Anlamsal ağ, günümüzde kullandığımız World Wide Web'in kurucusu Tim Berners-Lee tarafından, mevcut internetin bir eklentisi olacak şekilde eldeki verilerin iyi tanımlanmış bir biçimde anlamlandırıldığı böylece insanlar ve bilgisayarların daha iyi bir iş birliği içinde çalışması olarak ifade edilmiştir. Günümüzde anlamsal ağı geliştirmek, genişletmek ve standardize etmek için W3C'de kurulmuş bir ekip tarafından da anlamsal ağı kavramak için tanımlama yapılmıştır. Bu tanıma göre anlamsal ağ bize, bilgiyi uygulamalar, işletmeler ve topluluk sınırları boyunca paylaşarak ve tekrar kullanılabilir hale getirerek ortak bir çerçeve sunmuştur böylece anlamsal ağ internette bulunan bilgilerin makineler tarafından anlamlı ve anlaşılabilir hale geleceği teknolojiler ve standartlar bütünü olarak tanımlanmıştır (Yu, 2014). Anlamsal ağ temelinde felsefi bakış açısı, farklı türlerde tasarım ilkeleri içeren, müşterek çalışan teknoloji gruplarından oluşmuştur. Bu özelliği sayesinde adresler aracılığı ile etiketlenen bilgileri, sadece görünür halde olmasından öteye taşıyarak anlamlı hale getirmiştir (Baz, Meral, 2021). Şekil 1'de anlamsal ağın mimarisinde bulunan teknolojik bileşenler

listelenmiştir. Anlamsal ağın mimari yapısında bulunan teknolojilerden Sorgular, Ontoloji, RDF, URI / IRI çalışmanın ilerleyen bölümlerinde ele alınmıştır. Anlamsal ağın mimari yapısının ilk sırasında bulunan ve daha çok güvenlik ile ilgili olan Güven, Kanıt ve Mantık katmanlarında yapılan çalışmaların yeterli düzeyde olmadığı gözlemlenmiştir. Anlamsal ağ teknolojisiyle yapılacak çalışmaların mantık dahilinde kanıtlarla desteklenerek güvenilebilir hale gelmesi için mimari yapıda bulunan katmanların her birinin özenle hazırlanması ve sunumunun yapılması gerekliliği tespit edilmiştir (Hazer, 2015, s.4-5).



Şekil 1 Semantik ağın mimari yapısı (Hebeler, Fisher, Blace, Perez-Lopez ve Dean, 2009, s. 26)

İnternet, ilk dönemlerinde sadece dökümanlardan oluşan koleksiyonları son kullanıcılara sunmak için bir dizi web sayfası olarak kullanılmıştır. Bu haliyle internet kullanıcıların çeşitli yollarla bilgiye erişebildikleri fakat onunla etkileşime geçemedikleri halen kullanımına devam edilen gazetelere benzemiştir. Günümüzde internet artık çok daha etkileşimli bir mecra haline gelmiştir. Bu interaktif mecra interneti kullanıcıların sadece içerikleri görüntüleyebildiği değil, artık içerik üretebildiği ve bu içeriklerle etkileşime geçebildikleri bir yer haline getirmiştir. İnteraktif internet ortamı sosyal medya araçları üzerinden içerik üretmenin yanı sıra mevcut bulunan içeriklerle de etkileşime geçebilme, internet üzerinden alışveriş yapabilme veya canlı spor müsabakalarını izleyebilme gibi etkinliklerin yapılabilmesine olanak sağlamıştır. Bu noktada anlamsal ağda bilgisayarların da bu etkileşim ağı içerisinde yer alarak mevcut internet ağı üzerinde bulunan bilgileri anlayacak hale gelmesi, böylece büyük ölçekli havuzlarda bulunan bilgiler ile ilgili görevleri el ile yapmak yerine otomatik olarak gerçekleştirmesi hedeflenmiştir. Tablo 1’de anlamsal ağ ve günümüzde kullanılan internetin teknolojik özellikleri açısından kıyaslaması yapılmıştır.

Tablo 1

World Wide Web İle Anlamsal Ağ Teknolojisinin Mukayesesi

Özellik	WWW (World Wide Web)	Anlamsal Ağ
Ana bileşen	Salt görüntülenebilir belge odaklı	Biçimsel İfadeler
Hedef kitle	Etkileşimli ve insan odaklı	Uygulamalar
Bağlantılar	Bilgi ve bilgisayar temelli	Lokasyon ve anlamın ifadesi
Öncelikli kelime haznesi	Biçim talimatları	Mana ve lojik
Lojik	Biçimsel ve standart olmayan	Tanımlama mantığı

Kaynak: Baz ve Meral, 2021.

Bu bağlamda değerlendirince anlamsal ağ, bilgisayarlar ve insanları ortak bir çatı altında toplamıştır.

2.2.Anlamsal Ağ'da Ontoloji Kavramı

Ontolojilerin en basit haliyle tanımı: Kavramsallaşmanın biçimsel hale gelmesi, bir alan içerisinde yaygın olan terimleri tanımlamak ve sunabilmek için yapılan düzgün tanımlamalardır. Ontolojiler belirli bir alandaki bilgileri ifade etmek ve gösterebilmek için kullanılacak terimleri tanımlarlar. Bu özellikleri sayesinde anlamsal ağı fikri olarak tam manasıyla anlayabilmek için kritik bir rol üstlenenmişlerdir. Bu özellikleri kapsamında anlamsal ağı anlamada temel faydalar sağlamışlardır. Ontolojiler,

- Bir alandaki belirleyici konseptler hakkında ortak ve paylaşılabilen tanımlar.
- RDF oluşturmak üzere kullanılacak terimler.
- Tekrar kullanılabilen kaynak bilgileri.
- Bir alandaki açıkluğa kavuşturulmuş varsayımlar.
- Makinelerin anlayabileceği şekilde kodlanmış bilgiler ve anlamlar.

gibi özellikleri kapsamında anlamsal ağı anlamada temel faydalar sağlamışlardır (Yu, 2014, s.136-138). E-Ticaret, internetin sahip olduğu potansiyelin avantajından faydalanan ilk bilgi teknolojileri alanından biri olmuştur. Elektronik veri değişim standartları dahilinde sahip olduğu izinler sayesinde benzeri görülmemiş bir ölçekte iş yapılabilmesine olanak sağlamıştır. Geleneksel olarak kullanılan ticari yapıların internet ortamında uygulanması beraberinde bilgi güvenliği, veri akışı gibi konularda güven ve sorunsuz çalışma gibi ek ihtiyaçlar doğurmuştur. Bu ihtiyaçlar ile birlikte dijital market

üzerinde yapılacak işlemlerin hem akıllı araçlar hem de insan kullanıcılar tarafından toplanması, gösterilmesi, paylaşılması ve bir başlık haline gelebilmesi için kavramsallaştırılmayı gerekli hale gelmiştir. Anlamsal ağ dijital market içerisinde taraflar arasında yapılacak olan bilgi alışverişinin temelini oluşturmayı hedeflemiştir. Ontolojiler vasıtasıyla oluşturulacak olan paylaşılabilir kavramsallaşmış yapılar ile akıllı araçlar ve insan kullanıcılar adına verilerin anlamlarının toplanabildiği ve anlamsal olarak takas edebildiği böylece zenginleşmiş bilgiye erişilebilecek bir ortam oluşturulmak istenmiştir. Ontoloji alanındaki başarılar açık bir sunumla hazırlanmış konseptleri, hem insanlar hem de uygun bir şekilde yapılandırılmış bilgisayar programları tarafından anlaşılabilir hale getirmiştir. RDF (KTÇ) ve OWL gibi W3C organizasyonu tarafından bilişim sistemlerinin birlikte çalışabilirliği adına hem teknik hem de anlamsal seviyede en önemli teknolojiler olduğu belirtilmiştir. Anlamsal ağ odaklı E-ticaret, anlamsal olarak zenginleştirilmiş veri takasları ile çalışan ve bilgi yönetimi konusunda anlamsal ağ teknolojilerinin sistematik uygulamalarından faydalanarak yapılan ticaret olarak tanımlanmıştır (Badr, Chbeir, Abraham ve Hassanien, 2010, s. 245-246).

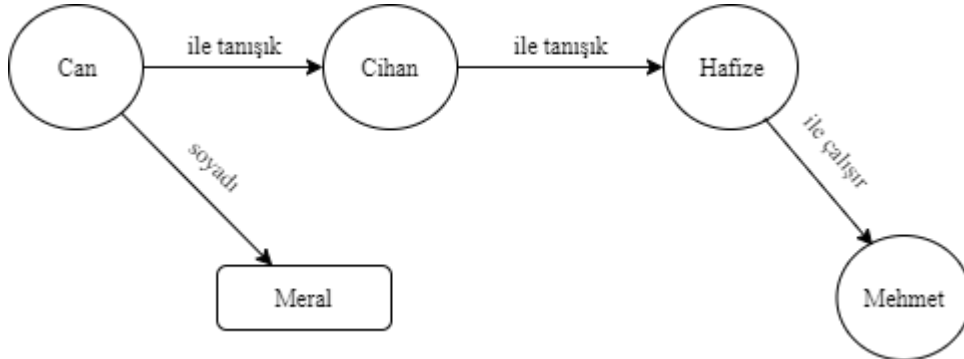
2.3.KTÇ (Kaynak Tanımlama Çerçevesi) Nedir?

Anlamsal ağ içerisinde bilgi üçleme olarak adlandırılan ve ingilizcedeki dil bilgisi kuralına göre özne, yüklem ve nesne olacak şekilde oluşturulmuş cümleler olarak gösterilmiştir (Hebeler vd., 2009). Kaynak tanımlama çerçevesinin tanımı W3C tarafından, bilginin dağıtık halini sunarak bilgisayar uygulamalarının kullanabileceği ve işleyebileceği ölçeklendirilebilir bir standart olarak yapılmıştır. Kaynak tanımlama çerçevesinde bilgiyi küçük parçalara ayırtırmak için her bir parçasının anlamı olacağı basit kuralları olan soyut bir model kullanılmıştır. Bu yapıyla herhangi bir durumu açıklayabilmek basitlik ve esneklikte, bilgisayar uygulamalarının yönetebileceği ve bilgiyi işleyebileceği genel bir metod oluşturmak hedeflenmiştir. Bu soyut model şekil 2'deki anahtar bileşenlerden oluşmuştur.

Özne	Yüklem	Nesne
------	--------	-------

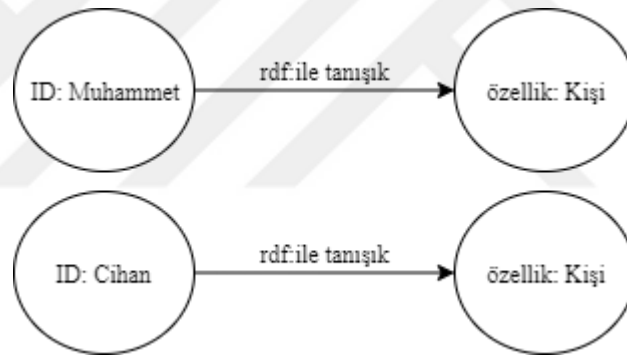
Şekil 2 Semantik ağın mimari yapısı (Yu, 2014).

Bileşenlerden özne ve nesne dünya üzerindeki iki şeyi gösterirken yüklem, bu iki nesne arasındaki ilişkiyi gösteren ve sıralarının kesinlikle değiştirilmemesi gereken cümle olarak isimlendirilen bir format haline gelmiştir (Yu, 2014, s. 24-26).



Şekil 3 Kaynak tanımlama çerçevesi örneği (Baz ve Meral, 2021, s. 416).

Şekil 3’de gösterilen kaynak tanımlama çerçevesinde kullanılabilecek örnek cümleler gösterilmiştir.



Şekil 4 Kaynak tanımlama çerçevesi modelinin gösterimi (Baz ve Meral, 2021).

Şekil 3’de gösterilmiş olan cümlelerin düğümlerinin kendi içerisinde sahip oldukları soyut modelin gösterimi şekil 4’de gösterilmiştir. Özne-yüklem-nesne modelinin tasarlanacağı bir yapıda bilgisayar uygulamalarını, Muhammet ve Cihan öznelerinin “ile tanışık” yüklemi aracılığıyla “özellik: Kişi” nesnesiyle ilişkisine bakarak Hafize öznesinin de herhangi bir nesne tanımlanmamış olsa da “özellik: Kişi” nesnesine sahip olduğu yorumunu yapabilecek hale getirmiştir. Bilginin anlaşılabilir şekilde göstermesindeki sağladığı kolaylığın yanında bilgisayar uygulamaları arasında da bilgi paylaşımını daha kolay hale getirebilmek için farklı kaynak tanımlama çerçeveleri geliştirilmiştir. En yaygın olarak RDF/XML, Turtle ve N-Triples kullanılmıştır.

```

semanticecommerce:Educationalrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Categories.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Electronics
semanticecommerce:Electronicsrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Categories.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Fitness
semanticecommerce:Fitnessrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Sport.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Laptop
semanticecommerce:Laptoprdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:PC.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Notebook
semanticecommerce:Notebookrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Educational.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Outdoor
semanticecommerce:Outdoorrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Sport.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#PC
semanticecommerce:PCrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Electronics.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Phone
semanticecommerce:Phonerdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Electronics.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Products
semanticecommerce:Productsrdf:typeowl:Class.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Sport
semanticecommerce:Sportrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Categories.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#TV
semanticecommerce:TVrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Electronics.

```

Şekil 5 Uygulamada kullanılan kaynak tanımlama çerçevesi bölümü

Uygulamada kullanılmak üzere oluşturulmuş kaynak tanımlama çerçevesinin geri kalan bölümleri ekler bölümünde gösterilmiştir.

2.4.OWL (İnternet Ontoloji Dili) Dili

İnternet ontoloji dilinin amacı sınıflar, özellikler ve bu özelliklerin özel uygulama alanlarındaki iletişimlerini, kaynak tanımlama çerçevesi şemasıyla tıpatıp aynı olacak şekilde tanımlanmıştır. Herhangi bir kaynak açıklanmak istendiğinde bu terimler yayınlanmış kaynak tanımlama çerçevesi belgelerini kullanarak, uygulamaların

muhakeme (Reasoning) işlemleri yaparak kesin olmayan veya bilinmeyen olguları keşfetmek için kullanılmıştır. Kaynak tanımlama çerçevesi şemalarına kıyasla internet ontoloji dili bizlere daha kompleks ve zengin ilişkileri ifade edebilme kabiliyeti sağlamıştır. İnternet ontoloji dili, kaynak tanımlama çerçevesi şeması ile daha iyi ifadeler oluşturmak için geliştirilmiş yapıların sentezi olduğu için ontoloji geliştirmelerinde sıklıkla kullanılmışlardır (Yu, 2014).

```
<?xml version="1.0"?>
<Ontology xmlns="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.semanticweb.org/oku/ybs/semanticcommerce"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  ontologyIRI="http://www.semanticweb.org/oku/ybs/semanticcommerce">
  <Prefix name="" IRI="http://www.semanticweb.org/oku/ybs/semanticcommerce#"/>
  <Prefix name="owl" IRI="http://www.w3.org/2002/07/owl#"/>
  <Prefix name="rdf" IRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#"/>
  <Prefix name="xml" IRI="http://www.w3.org/XML/1998/namespace"/>
  <Prefix name="xsd" IRI="http://www.w3.org/2001/XMLSchema#"/>
  <Prefix name="rdfs" IRI="http://www.w3.org/2000/01/rdf-schema#"/>
  <Prefix name="semanticcommerce" IRI="http://www.semanticweb.org/oku/ybs/semanticcommerce#"/>
  <Declaration><Class IRI="#Article"/>Declaration>
  <Declaration>Class IRI="#Book"/>Declaration>
  <Declaration>Class IRI="#Categories"/>Declaration>
  <Declaration>Class IRI="#Clothes"/>Declaration>
  <Declaration>Class IRI="#Desktop"/>Declaration>
  <Declaration>Class IRI="#Educational"/>Declaration>
  <Declaration>Class IRI="#Electronics"/>Declaration>
  <Declaration>Class IRI="#Fitness"/>Declaration>
  <Declaration>Class IRI="#Laptop"/>Declaration>
  <Declaration>Class IRI="#Notebook"/>Declaration>
  <Declaration>Class IRI="#Outdoor"/>Declaration>
  <Declaration>Class IRI="#PC"/>Declaration>
  <Declaration>Class IRI="#Phone"/>Declaration>
  <Declaration>Class IRI="#Products"/>Declaration>
  <Declaration>Class IRI="#Sport"/>Declaration>
  <Declaration>Class IRI="#TV"/>Declaration>
```

Şekil 6 Uygulamada kullanılan OWL bölümü

Uygulamada kullanılmak üzere oluşturulmuş internet ontoloji dilinin geri kalan bölümleri ekler bölümünde gösterilmiştir.

2.5.URI (Standartlaşmış Kaynak Tanımlayıcı) Nedir?

Standartlaşmış Kaynak Tanımlayıcı soyut veya fiziksel kaynaklara basit ve genişletilebilir anlamlar yüklemeye olanak sağlamıştır. Standartlaşmış kaynak tanımlayıcılar karakteristik olarak aşağıdaki gibi açıklanmıştır.

- Standartlaşmış: Standartlaşma karakterizasyonu, farklı erişim mekanizmalarına sahip olsalar dahi değişik tiplerdeki kaynakları aynı içerik için kullanılabilir hale getirmeye olanak sağlamıştır. Sağladığı bu özellik tanımlayıcıların farklı programlar veya protokollerin tekrar kullanabileceği önceden hazır bulunan, geniş ve yaygın olarak kullanılan kaynak tanımlamalarına erişimine izin vermiştir.
- Kaynak: Bu karakterizasyon standartlaşmış kaynak tanımlayıcı tarafından nitelendirilebilecek ölçekte herhangi bir kaynağı göstermek için kullanılmıştır. Bu kaynaklar internet adresi, sayılar, matematik denklemi, akrabalık gibi soyut konseptler veya insanlar, şirketler, kütüphanedeki kitaplar gibi somut konseptlerden oluşmuştur.
- Tanımlayıcı: Bir tanımlayıcı karakterizasyonu talep edilen bilgiyi tanımlama ölçeğinde bulunan diğer tüm kavramlardan ayırarak somutlaştırmaya olanak sağlamıştır. Bu bağlamda standartlaşmış kaynak tanımlayıcı tarafından yapılan tanımlama özünde bir kaynağı diğer kaynaklardan ayırmak anlamına gelmiştir.

Bu karakterizasyonlar kaynakların doğasına herhangi bir limit koymamıştır. Bunun yerine kaynaklara içeriklerden bağımsız olacak şekilde kullanıcıların küresel ölçekte erişebilecekleri tutarlı bir şekilde yorumlayabilme olanağı sağlamıştır. Örneğin "http://localhost/" adresi buraya başvuran tüm kullanıcılar için aynı yorumlamaya sahip olduğu halde her bir kullanıcı için farklı bir arayüze sahip olmuştur. Standartlaşmış kaynak tanımlayıcıları sayesinde bu yorumlamalar küresel olarak yapılabilir hale gelmiştir. Örnek olarak bir kullanım klavuzunun son kullanıcının "dosya://etc/hosts" dosya sistemi üzerini referans alarak diğer kaynaklardan sıyrarak küresel hale gelmesi sağlanmıştır. Tablo 2'de örnek standartlaşmış kaynak tanımlayıcılar verilmiştir.

Tablo 2

Standartlaşmış Kaynak Tanımlayıcı Örnekleri

Standartlaşmış Kaynak Tanımlayıcı Örnekleri
ftp://ftp.is.co.za/rfc/rfc3986.txt
http://www.ietf.org/rfc/rfc3986.txt
news:comp.infosystems.www.servers.unix

Kaynak: Berners-Lee, Fielding ve Masinter, 2005, s.1-7

Örneklerde görüldüğü üzere standartlaşmış kaynak tanımlayıcılar ile standartlaşmış kaynak bulucular birbirlerine benzedikleri görülmüştür. Sahip oldukları şemalar aracılığıyla buldukları ortandan ayrılan standartlaşmış kaynak tanımlayıcılar bu ayrımın yapılabilmesi için ":", "?", "/", "#" gibi ayraçları kullandığı görülmüştür. Bu bağlamda değerlendirilince her URL bir URI olurken, her URI'nin bir URL olmadığı görülmüştür (Berners-Lee vd., 2005, s. 1-7).

2.6.URI ile URL'nin Karşılaştırılması

Bu çalışma kapsamında standartlaşmış kaynak tanımlayıcı ile standartlaşmış kaynak bulucunun karşılaştırılması hiyerarşi, sözdizimi ve güvenlik konuları üzerinden yapılmıştır. Bu alanda gerek güncel gerekse doğrudan karşılaştırmaların yapıldığı çalışmaların bulunmadığı gözlemlenmiştir.

- **Hiyerarşi:** Standartlaşmış kaynak tanımlayıcıların sözdizimleri soldan sağa doğru önemi azalacak şekilde sıralanarak organize edilmiştir. Bazı URI'lerde görülebilir hiyerarşi şemanın kendisiyle sınırlanmıştır. Bir tanımlayıcının önem derecesine göre yorumlamasını yapabilmek için genellikle eğik çizgi ("/"), soru işareti ("?") ve diyez işareti("#") kullanılmıştır. Bu kullanım benzer yazımlar içerisinde tutarlı ve daha okunabilir olmaya olanak sağlamıştır. Böylelikle bir hiyerarşi ile ilişkili olabilecek şema bağımsız başvurular isimlendirilmiş ve standartlaşmış bir sunum olarak kullanılabilir hale gelmiştir (Berners-Lee vd., 2005, s. 10). Standartlaşmış kaynak tanımlama çerçevelerinde ise, bir kaynak başka bir kaynağı işaret edecek şekilde kullanılabilmiştir. Bu işaretler bazı durumlarda başka bir bağlantıyı gösterirken bazı durumlarda ise eğik çizgi ("/")

işareti kullanılarak kaynağın ilişkisel olarak temelini gösterecek şekilde bir hiyerarşiyi göstermiştir (Berners-Lee, Masinter ve McCahill, 1994, s. 4).

- **Sözdizim:** Standartlaşmış kaynak tanımlayıcılar şema, yetki, adres yolu, sorgular ve parçalara referans olan hiyerarşik sekansa sahip bileşenlerden oluşmaktadır (Berners-Lee vd., 2005, s. 4).

Tablo 3

URI Sözdizim Yapısı

Şema	Yetki	Adres yolu	Sorgu	Parça
rdf:	//ornek.com:8042	/adres/yolu	?isim=can	#meral
urn:	ornek:insan:can:meral			

Kaynak: Berners-Lee vd., 2005, s. 4

Tablo 3’de standartlaşmış kaynak tanımlayıcılara ilişkin örnek ve bileşenler gösterilmiştir. Standartlaşmış kaynak bulucular ise bir kaynağın soyut anlamda adresini göstermeye olanak sağlamıştır. Konumlandırılmış olan bu kaynak üzerinde işlem yapabilmek için sistem erişim, güncelleme, değiştirme, bağlantı gibi belirli karakterizasyonlardan faydalanmıştır. Bu karakterizasyonların içerisinde genellikle sadece erişim metodunun belirtilmesine ihtiyaç duyulmuştur (Berners-Lee, Masinter ve McCahill, 1994, s. 2).

- **Güvenlik:** Standartlaşmış kaynak tanımlayıcılar yapısal olarak doğrudan bir güvenlik tehdidi içerisinde olmamışlardır. Fakat ağ kaynaklarına erişim için sıklıkla kullanılan bir dizi yönergelerin standartlaşmış kaynak tanımlayıcılar içerisinde dikkatli bir şekilde yorumlanması verilere istenmeyen kişiler tarafından erişimi engellemek için gerekli hale gelmiştir. Bu istenmeyen erişimlerin önüne geçebilmek için bilgilerin açık bir metin içerisinde açığa çıkmaması üzerine özen göstermek gerekmiştir (Berners-Lee vd., 2005, s. 43). Standartlaşmış kaynak bulucular da doğrudan bir güvenlik tehdidini yapısal olarak oluşturmamışlardır. Kullanıcılar burada ilgili adreslerin nereye gidecekleri ve devamında ne yapacağı konusunda farkındalığa sahip olması gerekmiştir. Bu farkındalık sağlandığı müddetçe oluşabilecek zararlı teşebbüslerin önüne geçilmiştir (Berners-Lee, Masinter ve McCahill, 1994, s. 20).

2.7.SPARQL Nedir?

SPARQL anlamsal ağ içerisindeki veriler üzerinde sorgulamalar yapabilmek için W3C tarafından 2008'de standart haline getirilmiştir. SPARQL kaynak tanımlama çerçevesinde tutulan veriler üzerinde işlemlere olanak sağlayabilme özelliğiyle anlamsal ağın teknik yapısını oluşturmuştur. İlk haliyle SPARQL kaynak tanımlama çerçevelerine sadece erişim ve onları görüntüleyebilmeye olanak sağlamıştır. Bu durum W3C'nin yaptığı geliştirmeler ile duyurdukları SPARQL 1.1 ile değişmiştir. Yapılan bu güncelleme SPARQL 1.1 Sorgulama (Query), SPARQL 1.1 Güncelleme (Update), SPARQL Protokol (Protocol), SPARQL Servis tanımı (Service Description), SPARQL 1.1 Kaynak tanımlama çerçevesi grafikleri için birleştirilmiş HTTP protokolü (Uniform HTTP Protocol for Managing RDF Graphs), SPARQL 1.1 Zorunluluk rejimleri (Entailment Regimes) ve SPARQL 1.1 Konum yolları (Property Paths) şeklinde büyük parçalardan oluşmuştur. Kaynak tanımlama çerçevesi içerisinde saklanan veriler günümüzde kullanılan veri tabanlarının veri saklama yöntemleriyle benzerlik gösterdiği gözlemlenmiştir. Fakat kaynak tanımlama çerçevesi içerisinde bulunan veriler üzerinde değişiklik yapmak günümüzde tablolar üzerinden verileri işleyebilen veri tabanı sistemlerindeki gibi kolay yapılamadığı görülmüştür. Bu işlemleri yerine getirebilmek için bir dil eklentisi olarak SPARQL 1.1 Güncelleme geliştirilmiştir (Yu, 2014).

Tablo 4

SPARQL'in Sağladığı Faydalar

Kaynak tanımlama çerçevesindeki verilerde basit eşleşmeler yapabilmesi.
Çoklu eşleşmeleri birleştirebilmesi.
Integer, literal gibi veri tiplerini büyüktür, eşittir gibi koşul temelli eşleştirme yapabilmesi.
Verinin aranan kaynaktaki durumunu kontrol edip eğer orada bulunuyorsa sonucu getirme ya da bulunmuyorsa sorgunun hata almaması gibi opsiyonel veri eşleştirmesi yapabilmesi.
Birden fazla kaynak tanımlama çerçevelerinde eşleşen veri setlerini birleştirerek aynı anda sorgulayabilme, sıralayabilmesi ve limitleyebilmesi.
Anı ölçek içerisinde bulunan ifadeleri her zaman aynı anlama geldiğini gösterecek şekilde şeffaf referans yapabilmesi.
Soru cevaplamadaki gücüyle sorgu sonuçlarını sonraki sorgulamalara bir girdi olarak kullanabilmesi.
Mantık programlayabilmesi
Herhangi bir şemayla tanımlı olmasa bile internet üzerindeki bir veriyi desteklemesi.
XML, internet ontoloji dili, kaynak tanımlama çerçevesi gibi farklı veri formatlarını seri haline getirebilmesi.
Sorguların anlamsal muhakemeyi destekleyebilmesi.

Kaynak: Kumar, Kumar ve Kumar, 2011.

Tablo 4'de SPARQL'in fayda sağladığı alanlar gösterilmiştir (Kumar vd., 2011).

Tablo 5

SPARQL Dilinde Yazılmış Temel Sorgular

İfade	Sorgu
Select	PREFIX : <http://www.semanticweb.org/oku/ybs/semanticecommerce#> SELECT ?Laptop FROM <http://www.semanticweb.org/oku/ybs/semanticecommerce> WHERE { ?Laptop a :Laptop }
Insert	PREFIX : <http://www.semanticweb.org/oku/ybs/semanticecommerce#> INSERT DATA {GRAPH <http://www.semanticweb.org/oku/ybs/semanticecommerce>{:Samsung a :Laptop;}}
Delete	PREFIX : <http://www.semanticweb.org/oku/ybs/semanticecommerce#> DELETE WHERE { GRAPH <http://www.semanticweb.org/oku/ybs/semanticecommerce>{:HP a :Laptop;?property ?value}}
Update	PREFIX : <http://www.semanticweb.org/oku/ybs/semanticecommerce#> DELETE WHERE { GRAPH <http://www.semanticweb.org/oku/ybs/semanticecommerce>{:HP a :Laptop;?property ?value}}

Tablo 5'te SPARQL sorgulama dili kullanılarak temel ifadeler gösterilmiştir.

2.8.SPARQL ile SQL Veritabanı Sorgu Dillerinin Karşılaştırılması

SQL, ilişkisel veritabanlarında kontrol, I/O gibi işlemleri kolaylaştırmaya olanak sağlayan bir veritabanı alt dili olarak tanımlanmıştır (Melton, 1996). SQL tabloları her birinin özel bir veri tipine sahip olacak şekilde bir veya daha fazla sütun içerecek şekilde tanımlanmışlardır. Tanımlaması yapılmış bu tablolarda bulunan satırların tabloda bulunan sütunlar ve onların deklare edilmiş veri tipleriyle eşleşme zorunluluğu olduğu görülmüştür. Kaynak tanımlama çerçevesinde ise bunun tam tersine veriler üçlemeler olarak tutulduğu için OWL kullanılarak ifade edilmiş tüm verilerin kaynak tanımlama çerçevesi içerisinde ait olduğu varlığı göstermiştir. Tablo 5'te SPARQL ile SQL arasındaki farklılıklar gösterilmiştir (Kumar vd., 2011).

Tablo 6

SPARQL ile SQL Arasındaki Farklılıklar

Bileşen adı	Açıklama	SQL	SPARQL
Tür adı	Veri tipi	Tam sayı, karakter, isim	Özne, yüklem, nesne, uri
Nitelik adı	Belirgin bir isim	plaka, sehir, isim	Değişkenler: ?plaka ?sehir Düğüm pozisyonları: özne yüklem nesne
Nitelik	Tür adı ile nitelik adının birleşimi	Plaka: Integer, sehir: char, isim: char	?sehir: Özne, y1: Yüklem, ?plaka: Nesne
Demet değeri	Nitelik ve değer çiftleri seti	sehir, plaka, isim 'Cihan'	?sehir: Özne (#1. Özne), y1: Yüklem (#İsim), n1: Nesne ('Cihan')
İstikamet	Nitelik seti	sehir, plaka, isim, tam sayı, karakter, isim	?sehir özne, y1 yüklem, ?plaka nesne

Kaynak: Kumar vd., 2011.

SPARQL her ne kadar SQL'e kıyasla daha esnek bir yapıya sahip olsa da henüz yaygın olarak kullanılmadığı görülmüştür. Buna rağmen kaynak tanımlama çerçevelerinde gerçekleştirebileceği işlemlerin daha açık ve anlaşılabilir olması onu SQL karşısında daha avantajlı hale getirmiştir. Sağladığı bu kolaylık sayesinde SQL'in daha çok veritabanının

oluşturulma şekli ve tablo yapılarına bağlı mantığına karşılık SPARQL sorgulamaları daha çok insan mantığına yakın bir şekilde sıralanmıştır. Teknik bir farklılık olarak da SPARQL SQL'deki standart ekleme, silme, güncelleme işlemlerine ek olarak "ASK" komutu ile doğrudan sorgulamaların sonucunun doğruluk ve yanlılığı hakkında bilgi verme imkanı sağlamıştır (Hazer, 2015, s. 24).

2.9.Literatürde Yapılan Çalışmalar

Hazer (2015) "Anlamsal Ağ Tabanlı İlaç Öneri Sistemi", başlıklı çalışmada anlamsal ağ ile ilgili temel kavramlar ve barındırdığı teknolojiler ile birlikte ilaç sanayisinde etken maddesi eşdeğer düzeyde ilaçların üretilmesi ve böylece hem hastaların ilaçlara erişiminin artması hemde ülke ekonmonisine olumlu katkı sağlanması noktasında"muadil ilaç" ihtiyacını karşılamak üzere eşdeğer etken maddeye sahip olan ilaçların önerilmesini sağlayacak bir uygulama üzerinde çalışılmıştır.

Yu (2014) "A Developer's Guide to the Semantic Web", başlıklı çalışmada anlamsal ağ konseptinin tanıtılması ve bu kapsamında bir uygulama geliştirmek için ihtiyaç duyulabilecek araçlara ve teknolojilere değinilmiştir. Çalışma kapsamında anlamsal ağ ile ilgili araçların kurulumları ve bu araçların birlikte çalışabilmesi için gerekli entegrasyonların yapılması konularına değinilerek bir uygulama geliştirmek için uyulması gereken adımlar gösterilerek örnek bir uygulama geliştirilmiştir.

Hebeler vd. (2009) "Semantic Web Programming", başlıklı çalışmada anlamsal ağın çoklu bilgi ve bilgi servisleri üzerinde sağladığı güçlü ve pratik yaklaşım ile bilginin paylaşılabilme kapasitesini artıracak programlama yapıları ve araçları üzerine değinilmiştir.Bu şekilde günümüzde önemi her geçen gün daha da artan verinin etkili ve verimli bir şekilde kullanılarak doğru bir biçimde değerlendirilmesi hedeflenmektedir.

Badr vd. (2010) "Emergent Web Intelligence: Advanced Semantic Technologies", başlıklı çalışmada internetin başarısının bir göstergesi olan kullanıcıların verileri saklama, işlem yapabilme ve mesafe gözetmeksizin bu verilere erişebilme gibi kriterlerin karşılanabilmesi ve gelişen insan-makine etkileşiminin etkili olabilmesi için kullanılacak anlamsal ağ, yapay zeka ve bilgi temelli servisler gibi anlamsal teknolojiler üzerine değinilmiştir.

Berners-Lee (2004) "Tim Berners-Lee's Semantic Web", başlıklı çalışmada World Wide Web'in kurucusu olan Tim Berners-Lee mevcut internetin bir eklentisi olarak tanımladığı anlamsal ağ konseptinin vizyonu, sahip olduğu teknolojiler ve metodları üzerine değinilmiştir. Çalışma kapsamında anlamsal ağ ile veriler arasında oluşturulabilecek mantıksal ilişkilerin anlamsal ağ çerçevesinde değerlendirmesi yapılmıştır.

Qin (2009) "Introduction to E-commerce", başlıklı çalışmada elektronik ticaretin temellerini ve anahtar yönelimlerini elektronik ticaret endüstrisinde meydana gelen son gelişmeler kapsamında tanıtılması üzerine değinmiştir. Böylece elektronik ticaretin sahip olduğu potansiyel gösterilmiş ve sağlayabileceği fayda ve fırsatlar gösterilmeye çalışılmıştır.

Rodriguez-Muro, Kontchakov, Zakharyashev (2013) "Ontology-Based Data Access: Ontop of Databases", başlıklı çalışmada anlamsal ağ içerisinde kullanılan ontolojilerin veri saklama noktasında teorik temelleri ve ilişkisel veritabanları ile performans değerlendirmesi üzerine çalışılmıştır. Çalışma kapsamında günümüzde kullanılan veri tabanlarının veriler üzerinde gerçekleştirilen standart işlemlerdeki performansı değerlendirilmiş ve sonuçları benzer işlemlerin ontolojiler ile yapılarak elde edilen sonuçları ile değerlendirilmiştir.

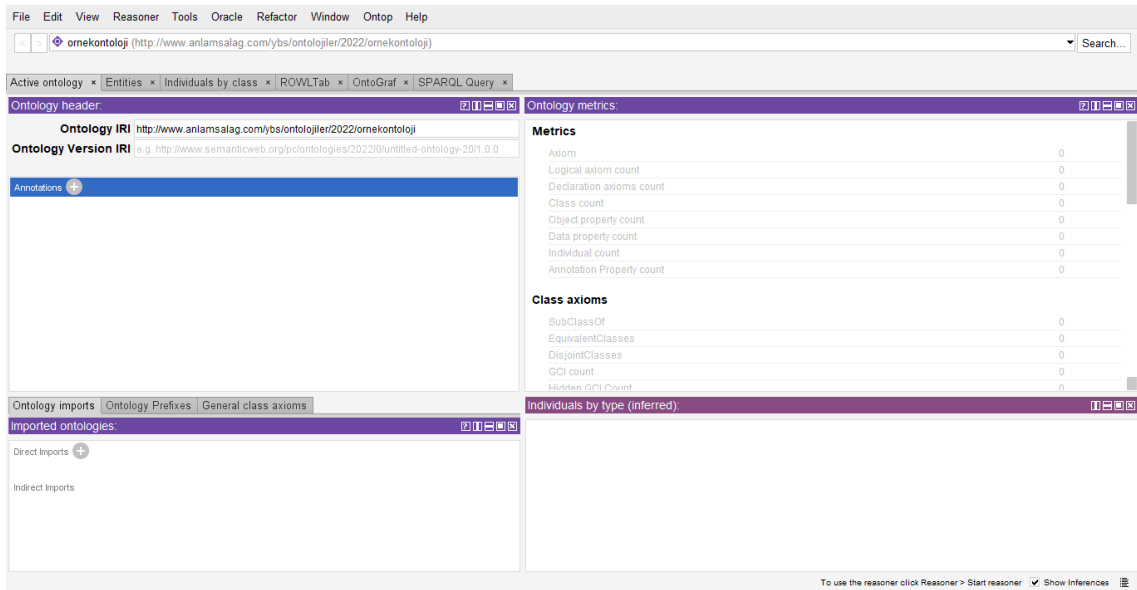
BÖLÜM III

ANLAMSAL AĞ İÇİN TEKNOLOJİK ARAÇLAR

Anlamsal ağ kullanılarak bir uygulama geliştirebilmek için sunucu, ön yüz katmanı, uygulama seviyesi gibi farklı aşamalarda ihtiyaç duyulan çeşitli araçların birlikte çalışmasını gerektirmektedir. Çalışmanın 3. bölümünde anlamsal ağ teknolojisi ile bir uygulama geliştirmek için ihtiyaç duyulabilecek araçlar ve bu araçlar hakkında bilgi verilmiştir.

3.1.Protégé: Ontoloji Oluşturma Uygulaması

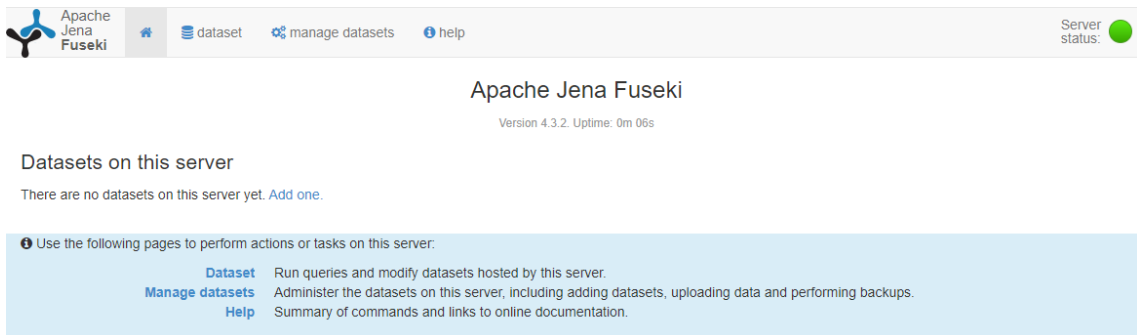
Protégé Java dilinde yazılmış, bilgi temelli, açık kaynak ve ücretsiz oluşuyla günümüzde ontoloji geliştirme ortamının lideri olmuştur. Protégé ilk olarak Stanford üniversitesi tarafından geliştirilmiştir. Günümüzde Protégé yazılım geliştirici ve akademi gibi topluluklar tarafından geliştirildiği gözlemlenmiştir. En son sürümü olan Protégé 5.5.0 Mart 2019'da çıkartılmıştır. Protégé iki ana yolu kullanarak ontolojileri modellemiştir. Bunlardan ilki kullanıcıların ontolojileri açık bilgi tabanlı bağlanabilirlik protokolüne göre kullanabilmesine olanak sağlayan Protégé çerçeve düzenleyici olmuştur. İkincisi ise W3C standartlarına göre ontoloji geliştirilmesine olanak sağlayan Protégé-OWL düzenleyici olmuştur. Bu yönlerinin yanı sıra eklenti mimarisi sayesinde Protégé kullanıcılara daha geniş ölçekte özellikler sağlamıştır. . Bu çalışmada ikinci seçenek üzerinden geliştirme yapılmıştır (Yu., 2014).



Şekil 7 Protégé ontoloji geliştirme uygulamasının arayüzü

3.2. Apache Jena Fuseki Sunucusu

Apache jena fuseki bir işletim sistemi servisi şeklinde Java internet uygulaması olarak çalışan bağımsız bir sunucu olarak tanımlanmıştır. Apache jena fuseki iki biçimden oluşmuştur. Bunlardan biri kullanıcı arayüzüyle birleştirilmiş münferit bir sunucu olmuştur. Diğer biçimi ise daha geniş dağıtımları çalıştırma ortamına uygun bir sunucu olmuştur. Apache jena fuseki kaynak tanımlama çerçevesi depolama ve sorgulamak üzere TDB bileşenini kullanmıştır. TDB tek bir makine üzerinde depolanmak üzere yüksek performanslı kaynak tanımlama çerçevesi olarak kullanılmıştır (Apache Jena Fuseki, t.y.).



Şekil 8 Yerel ağ üzerinde apache jena fuseki sunucusunun arayüzü

3.3.Jena API

Jena Java programlama dilinin içerisinde anlamsal ağın cümlelerini ve yapılarını sınıflar, nesnelere, metodlar ve özellikle gibi kullanışlı programlama eserlerine çevirmek üzere geliştirilmiştir. Anlamsal ağ üzerinde bulunan eserler verilerin sunumuyken, Jena ise hem verinin sunumunu yapıp hemde karşılık gelen metodları oluşturan nesne yönelimli eserlerden oluşmuştur (Hebeler vd., 2009).

Tablo 7

Jena API ile Anlamsal Ağın Karşılaştırılması

Eser	Anlamsal Ağ	Jena Sınıfı	Java Notlar
Özne, nesne, yüklem	URI	Kaynak, özellik	Bir kaynağın özne, yüklem ve nesne olması
İfade	İfade	İfade	Somutlaştırılmış ifadelerin düşünülmesi
Veri	Ontoloji ve örnek veri	Diyagram ve model	Diyagramların modellerin temel yapısını oluşturarak hem ontolojileri hem de örnek verileri bulunması
Sorgu ve sonuçlar	SPARQL ve anlamsal ağ verisi	Sorgu ve sonuç dizileri	İlişkisel veritabanlarıyla analoji bulunması.
Mantıksal muhakeme	Akılcı araç	Akılcı araç	Dahili ve harici mantıksal muhakeme yapılması
Kurallar	SWRL	Akılcı araç	Kural desteğinin özel akılcı araçlar tarafından yapılması
Etkinlik bildirimi	Uygulanabilir değil	Nesne dinleyiciler	Etkinlik bazlı takip yapılması.

Kaynak: Hebeler vd., 2009, s. 270.

Tablo 7’da Jenaya ait verilmiş eserlerin açıklamaları aşağıda yapılmıştır.

- **Kaynak:** Özne, yüklem veya nesne gibi elementlerden oluşan ifadeler.

- **İfade:** Özne, yüklem, nesneden oluşan ve içerdiği bileşenleri kolayca sorgulayabilmeye olanak sağlayan anlamsal ağ üçlemeleri.
- **Diyagram:** Anlamsal ağ üzerindeki verilere ekleme, silme, arama gibi temel operasyonların sürdürülebilmesine olanak sağlayan metod.
- **Model:** Anlamsal ağ verisiyle etkileşime girebilmeyi sunan temel diyagram.
- **Sorgu ve sonuç dizileri:** SPARQL ile yapılan sorguların sonuç dizileri üzerinden yinelenebilmesine olanak sağlayan yapı.
- **Akılcı araç:** Jena'nın kendi başına gerçekleştirebildiklerinin dahili, üçüncü taraf yazılımlar tarafından gerçekleştirilenlerin ise harici olarak tanımlanan ve ontolojilerin tutarlı olmasına yardımcı olan araç.

Jena ontolojileri Java sınıflarına dönüştürerek anlamsal ağı programlanabilir hale getirmiştir (Hebeler vd., 2009, 265-271).

3.4. Java Programlama Dili

Java programlama dili henüz internetin hayatımıza dahil olmadığı 1991 yılında günümüzde Oracle Inc. bünyesinde bulunan Sun Microsystems'de çalışmakta olan James Gosling ve arkadaşları tarafından yeni geliştirilmiş cihazlarda kullanılmak üzere oluşturulmuştur. İnternetin kullanımının artmaya başladığı dönemde ilk Java destekli tarayıcı HotJava geliştirilmiştir. İlerleyen dönemde Java'nın kurumsal ihtiyaçları karşılamak üzere 2. sürümü olan J2EE geliştirilmiştir. Böylece Java artık nesne yönelimli bir programlama dili haline gelerek kurumsal projelerde çalışılabilecek duruma gelmiştir. Sun Microsystems Oracle Inc. tarafından satın alındıktan sonra Java ücretsiz ve açık kaynak kodlu hale dönüşmüştür. Bu şekilde Java'nın gelişimi Java topluluğu işlemleri çatısı altında yürütülmüştür. Java'nın tanımı platformdan bağımsız yani sanal makineler üzerinde çalışan nesne yönelimli bir teknoloji olarak yapılmıştır. Java programlama dilinin taşınabilirlik, Java derleyicisi ve Bytecode, JVM, bellek yönetimi, senkron programlama, dağıtık sistemler ve JavaEE internet uygulamaları, gömülü sistem yapısı ve JavaME gibi temel özelliklere sahip olduğu görülmüştür.

Taşınabilirlik: Java ile yazılmış programların Windows, Linux, Mac OS gibi herhangi bir işletim sisteminde herhangi bir değişiklik yapılmadan çalışabilmesini sağlamıştır.

Java derleyicisi ve Bytecode: Java uzantılı metin belgelerinde bulunan kaynak kodlarının Bytecode'lara dönüşüp, Java'nın platform gözetmeksizin kodlarda herhangi bir değişiklik yapmaya gerek olmaksızın çalışabilmesini sağlamıştır.

JVM: Her platform için farklı ve özel olarak geliştirilen sanal makineler Java bytecode'larının çalışabileceği ortamı oluşturmuştur.

Bellek yönetimi: Java programlama dilinde bellek yönetimi için ekstra bir kod yazımına gerek kalmadan otomatik olarak yönetilmesi olarak tanımlanmıştır.

Senkron programlama: Java'nın çoklu işlem yapabiliyor olması birden fazla hesaplamanın yapılacağı durumlarda herhangi bir bekleme yapmadan eş zamanlı olarak hesaplamayı yapabilmesine olanak sağlamıştır.

Dağıtık sistemler ve JavaEE internet uygulamaları: Ağ üzerinde bulunan farklı sistemler ile haberleşmeye olanak sağlanıp böylece internet sunucularıyla birlikte çalışabilen sunucu tabanlı internet uygulamalarının Java ile geliştirilebilmesi sağlanmıştır.

Gömülü sistem yapısı ve JavaME: Java'nın akıllı televizyonlar, cep telefonları, cihaz üstü gibi ürünlere yazılım geliştirebilmek için Javanın küçültülmüş sürümü olarak geliştirilmiştir.

Bu çalışmada Java programlama dili Jena API, thymeleaf, springboot ve bootstrap gibi teknolojiler ile birlikte geliştirilecek ontolojin görüntülenmesi, düzenlenmesi gibi görevlerde kullanılmıştır (Meral ve Dai, 2014, s.2-8).

3.5.Spring Framework Teknolojisi

Spring framework Java programlama dili ile işletmelere daha kolay uygulama oluşturmak için geliştirilmiştir. Spring framework geniş bir alanda uygulama geliştirme senaryolarına destek verebilmiştir. Büyük işletmelerde geliştirilen uygulamaların geliştirilme süreleri sıklıkla daha uzun sürmüştür. Bu durum da yazılım geliştiricilerin kontrol sahibi olmadıkları ve belirli bir güncelleme döngüsünde olacak sunucularda çalışma zorunluluğu getirmiştir. 2003 yılında geliştirilmeye başlayan spring framework uygulamaların mümkün olabilecek en kısa sürede çalışabilmesi için gerekli ayarlamaları

yapmada sağladığı fayda ile üretime hazır spring temelli uygulamalar üretmeye imkan sağlamıştır (Spring Framework Overview, 2022).

3.6.Thmeleaf Aracının Kullanımı

Thymeleaf sunucu tarafında hem internet hemde bağımsız ortamlarda HTML, XML, JavaScript, CSS ve hatta sadece metinleri bile işleyebilme kapasitesine sahip modern bir Java taslak motoru olarak tanımlanmıştır. Thymeleaf'in ana amacı taslaklar oluşturulurken, taslakları zarif ve üzerlerinde rahatlıkla çalışma yapılabilecek hale getirmek olarak belirlenmiştir. Bunu yapabilmek için tasarım prototipi olarak kullanılacak taslakları etkilemeden doğal şablonlar kullanarak çalışma mantığını taslak dosyalarına aktararak başarmıştır. Thymeleaf şablon modu olarak adlandırılan 6 türde şablonu işleyecek şekilde geliştirilmiştir. Bu şablonlar HTML, XML, metin, JavaScript, CSS ve RAW şeklinde belirtilmiştir (Tutorial: Using Thymeleaf, 2018).

3.7.Bootstrap Kullanımı

Bootstrap kullanıcıların ticari veya kişisel ihtiyaçlar doğrultusunda oluşturacağı internet sitelerini hızlıca tasarlayabileceği açık kaynak kodlu bir araç olarak tasarlanmıştır. İçeriğinde bulundurduğu önceden yazılmış CSS, HTML ve JavaScript kodları sayesinde geliştiricileri bileşenleri tasarlama ve kodlamaya ayıracağı vakitten kurtarmıştır. Bootstrap sahip olduğu ızgara yapısı sayesinde her türlü cihazda boyutunu otomatik olarak ayarlayabilecek şekilde geliştirilmiştir. Ayrıca Chrome, Firefox, Safari, Edge, Opera gibi büyük internet tarayıcılarıyla birlikte iOS, Android, Windows 10 Mobile, Mac, Windows gibi büyük işletim sistemlerinde de ek bir adıma gerek duymaksızın çalışabilecek şekilde tasarlanmıştır. Böylece yazılım geliştiriciler tasarlayacakları sitenin görünümüne harcayacakları vakti kazanmışlardır (A Beginner's Guide to Bootstrap, 2021).

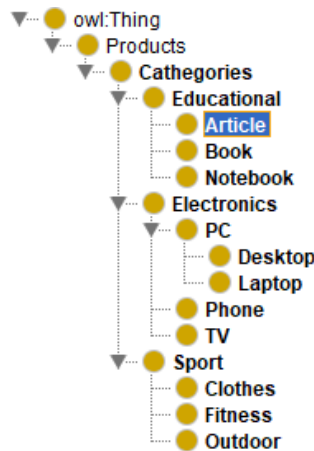
BÖLÜM IV

YÖNTEM

Bu çalışmada bilimsel araştırma yöntemlerini göz önünde bulundurarak kullanılacak yazılımsal teknolojiler ile ilgi yayınlanmış resmi dökümanlar incelenerek bilişim teknolojilerinde kullanılan yöntemler uygulanmıştır ve bununla ilgili yazılım geliştirme, ağ ve sunucu işletimi, veritabanı yönetimi gibi kavramlara yer verilmiştir.

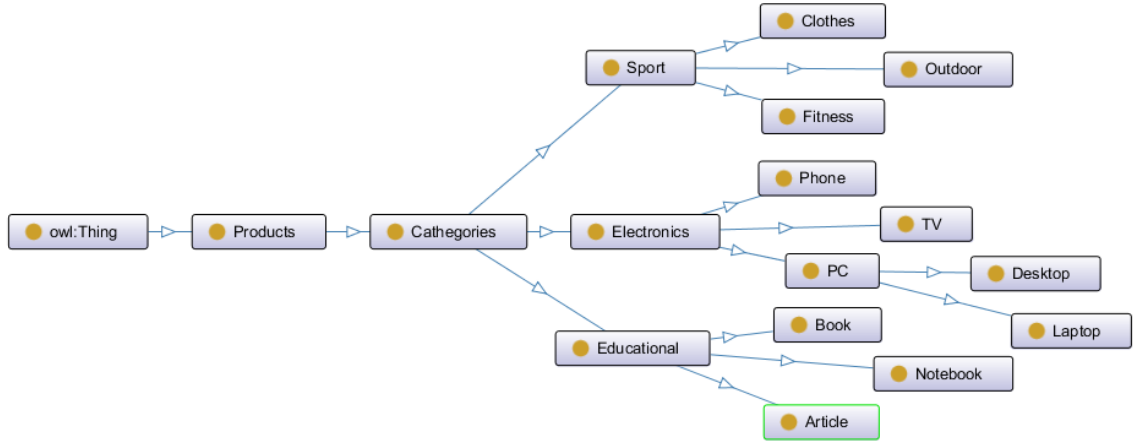
4.1.Ontoloji Oluşturma

İnternet ontoloji dilinin standartlaşmasından bu yana bu dilin kullanım alanı sağlık, biyoloji, coğrafya, astronomi, savunma ve havacılık endüstrisi gibi büyük alanlara genişlemiştir (Yu., 2014, s. 157). Ontoloji temelli veri erişimi özellikle anlamsal ağ gibi büyük ölçekte verinin işlendiği uygulamalar başta olmak üzere, yeni nesil bilgi sistemlerinde ana malzeme olarak ilişkilendirilmiştir. Ontoloji temelli veri erişim paradigmasında bir ontoloji kullanıcılara sorgularında faydalanabilecekleri yüksek seviyede küresel bir şema ve bir sözlük sağlamıştır. Böylece kullanıcılar veri kaynaklarının yapısal detaylarından izole edilebilmiştir (Rodriguez-Muro vd., 2013, s. 558).



Şekil 9 Ontolojide oluşturulan sınıfların görünümü

Şekil 9’da geliştirilen Protégé uygulaması kullanılarak oluşturulan sınıfların hiyerarşik görünümü gösterilmiştir.



Şekil 10 Uygulama ontolojisinin iskelet yapısı

Şekil 10’da Protégé uygulaması kullanılarak oluşturulan ontolojilerin yapısı gösterilmiştir. Bu görünüm sayesinde ontoloji içerisinde bulunan sınıfların birbirleriyle etkileşimleri kontrol edilebilmiştir. Bir ontolojide farklı hiyerarşik çatı altında bulunan sınıfların kendi aralarında etkileşime geçebilmeleri için nesne özellikleri kullanılmıştır.



Şekil 11 Ontolojide kullanılan sınıfların nesne özellikleri

Şekil 11’de sınıfların sahip olduğu nesne özellikleri gösterilmiştir. Örneğin bir bilgisayar ve ona ait bir aksesuarı belirtirken bilgisayar için “hasPCAccesory”, bilgisayara ait olan bir aksesuar için “isPCAccesory” nesne özelliği kullanılmıştır.

4.2.Apache Jena Yerel Sunucu Oluşturma

Protégé ile geliştirilen ontolojiyi veritabanı olarak kullanabilmek üzere gerekli olan sunucu ihtiyacını karşılamak için Apache Jena Fuseki kullanılmıştır. Bu sunucu ile birlikte oluşturulan ontolojiler sunucuya veri setleri olarak yüklenerek farklı uygulamalar ve SPARQL sorguları ile etkileşime geçebilir hale gelmiştir.

Apache Jena Fuseki

Version 4.3.2. Uptime: 16m 16s

Datasets on this server

dataset name	actions
/SemanticWebECommerce	query add data info

Use the following pages to perform actions or tasks on this server:

- [Dataset](#) Run queries and modify datasets hosted by this server.
- [Manage datasets](#) Administer the datasets on this server, including adding datasets, uploading data and performing backups.
- [Help](#) Summary of commands and links to online documentation.

Şekil 12 Apache jena fuseki sunucusunun ana ekran görünümü

Şekil 12’de sunucu üzerinde yeni veri seti ekleme, sorgu yazma, sunucu hakkındaki bilgiler gibi işlemlerin gerçekleştirilebildiği Apache Jena Fuseki sunucusunun ana ekranı gösterilmiştir.

4.3.Spring Projesi Oluşturma

Java programlama dili kullanılarak veri tabanı düzeyinde işlemleri gerçekleştirmek üzere bir Spring Framework kullanılmıştır.



Project

Maven Project Gradle Project

Language

Java Kotlin Groovy

Spring Boot

3.0.0 (SNAPSHOT) 3.0.0 (M2) 2.7.0 (SNAPSHOT) 2.7.0 (M3)

2.6.7 (SNAPSHOT) 2.6.6 2.5.13 (SNAPSHOT) 2.5.12

Project Metadata

Group

Artifact

Name

Description

Package name

Packaging Jar War

Java 18 17 11 8

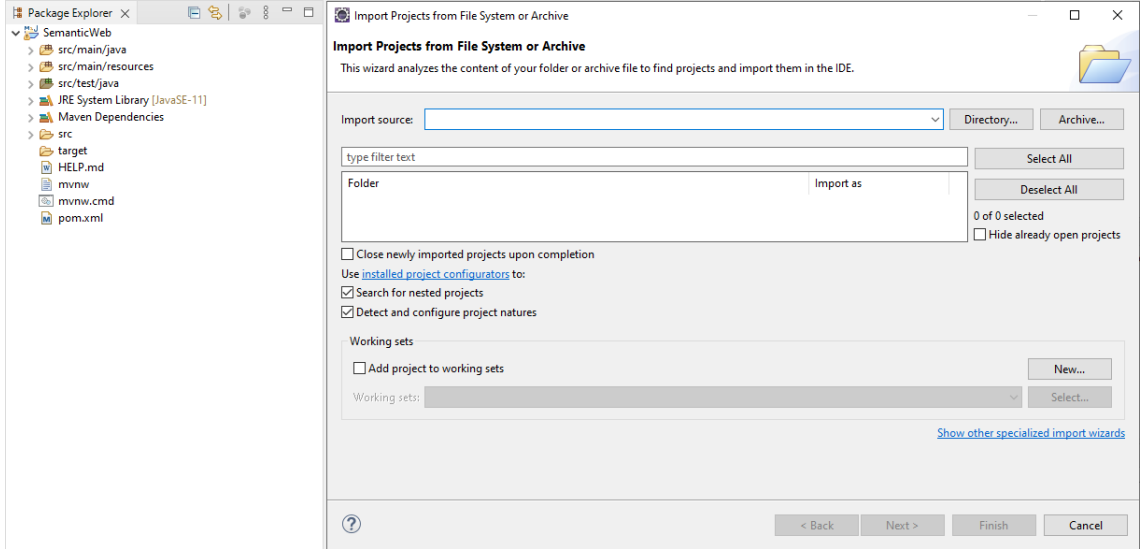
Dependencies

Spring Web WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

Şekil 13 Spring projesi oluşturma sayfasının görünümü

Şekil 13’de uygulama geliştirme aşamasında kullanılmak üzere Spring initializr aracılığı ile oluşturulan Spring projesi hakkındaki bilgiler gösterilmiştir. Proje oluşturulduktan sonra kullanılabilmek için geliştirme yapılacak ortama yüklenerek hazır hale getirilmiştir.

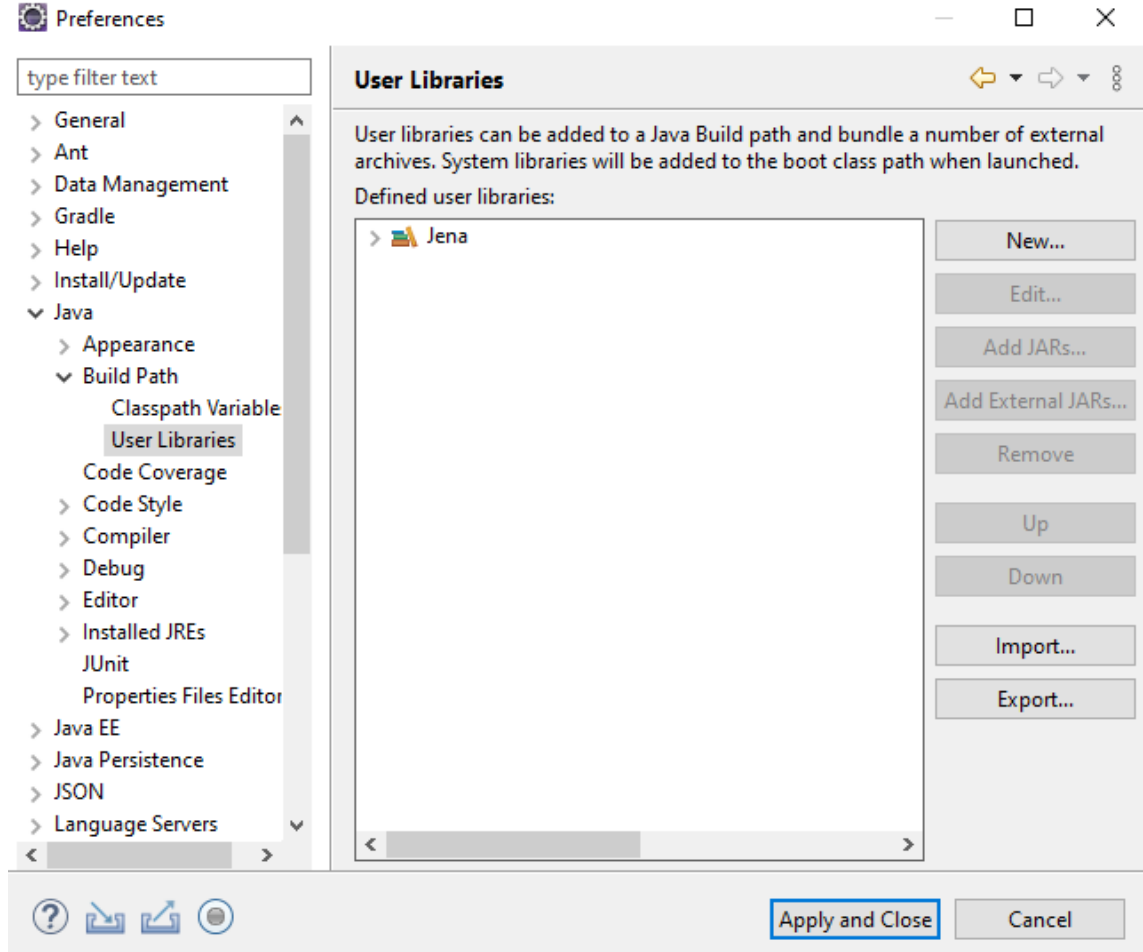


Şekil 14 Uygulama geliştirme ortamına spring projesi yükleme ekranı

Şekil 14’te Spring initializr kullanılarak oluşturulan projenin konumu belirtilerek uygulama geliştirme ortamına yükleme ekranı gösterilmiştir.

4.4. Java Kütüphanesine Jena API Ekleme

Arkaplanda iletişimi sağlamak üzere oluşturulan Spring projesiyle birlikte anlamsal ağ uygulamasında kullanılacak ontoloji üzerinde işlemler yapabilmek için Jena kütüphaneleri projeye yüklenmiştir. Böylece doğrudan Apache Jena Fuseki sunucusu üzerinde kullanılan sorgular uygulama katmanında da kullanılabilir hale gelmiştir.

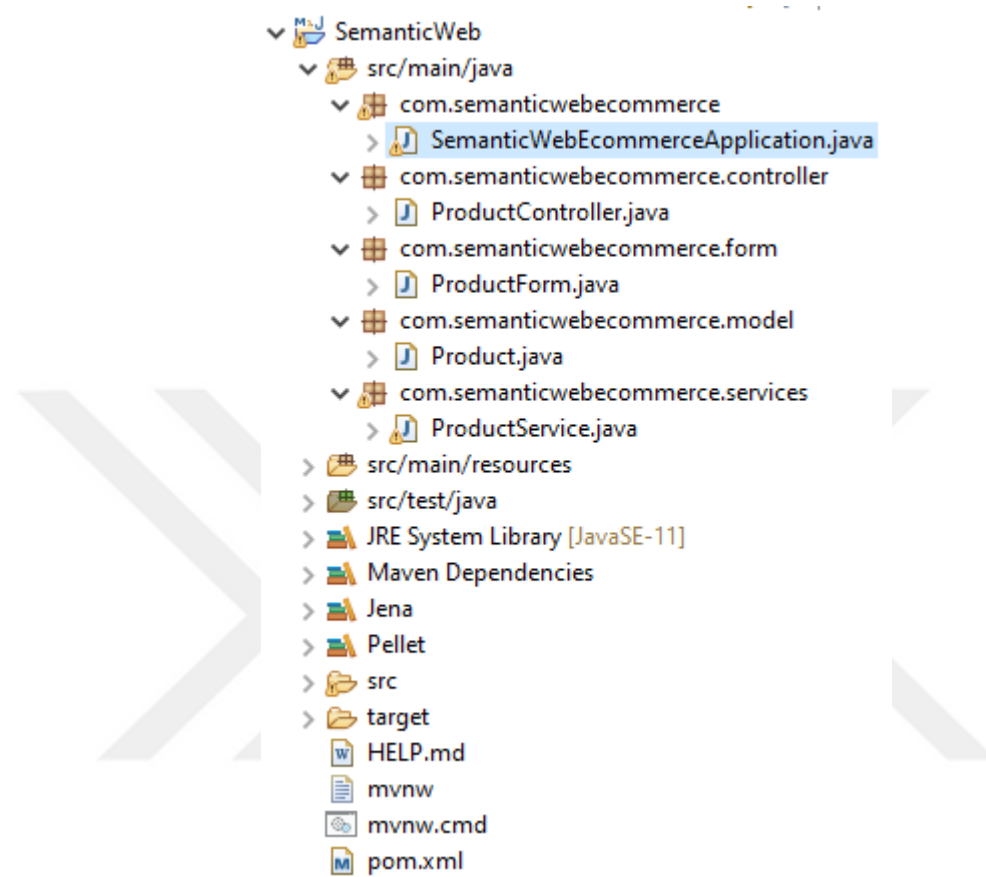


Şekil 15 Projeye jena kütüphanesi yükleme ekranı

Şekil 15’de Eclipse geliştirme ortamının tercihler ekranı kullanılarak uygulama geliştirme esnasında ihtiyaç duyulacak kütüphanelerin projeye eklenmesi gösterilmiştir.

4.5. Java Geliştirme Ortamı Oluşturma

Uygulama katmanında proje yapısını oluşturabilmek için Java geliştirme ortamında MVC yazılım mimarisi kullanılmıştır.



Şekil 16 Uygulamanın proje görünümü

Şekil 16'da MVC mimarisinin kullanıldığı uygulamanın geliştirilme aşamasındaki proje görünümü gösterilmiştir. Herhangi bir uygulama geliştirilirken yapısal formatının görüntülenmesinde mimari desenler kullanılmıştır. MVC mimari deseni, kullanıcı arayüzünden bağımsız olarak sistemin çekirdek işleyişini kontrol eden Model, sunumun yapılacağı View ve uygulamaya gönderilecek girdilerin düzenlendiği Controller olacak şekilde üç büyük bileşenden oluşmuştur (Kalelkar ve Churi, 2014).

4.6.Uygulamaya Thymeleaf ve Bootstrap Ekleme

Geliştirilen uygulamanın ön yüz katmanında kullanılacak nesnelere ve bunlarla ilgili işlemlerin gerçekleştirilebilmesi için projeye Thymeleaf ve Bootstrap eklenmiştir.

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-thymeleaf</artifactId>
</dependency>
<dependency>
  <groupId>org.webjars</groupId>
  <artifactId>bootstrap</artifactId>
  <version>5.1.3</version>
</dependency>
```

Şekil 17 Thymeleaf ve bootstrap bağımlılıklarının görünümü

Şekil 16’da gösterilen “pom.xml” dosyasının içerisine uygulamada kullanılmak istenen thymeleaf ve bootstrap teknolojileri Şekil 17’de gösterilen bağımlılıkları eklenerek projede kullanılabilir hale getirilmiştir.

BÖLÜM V

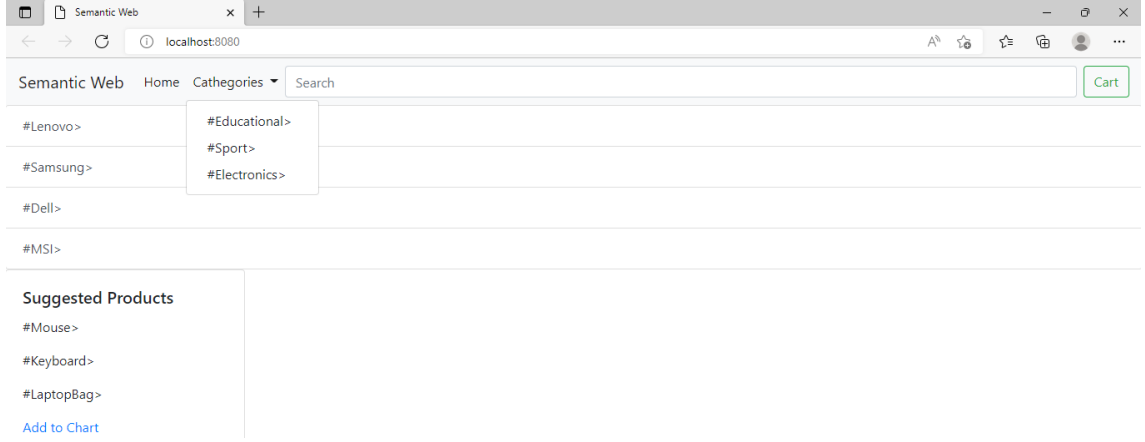
UYGULAMA

Bu çalışma kapsamında anlamsal ağ alanında kabul görmüş yöntemler kullanılarak web 3.0 teknolojisinin kullanıldığı bir uygulama geliştirilmiştir.

5.1.Uygulamanın Amacı

Günümüzde kullanılan elektronik ticaret sistemlerinin kullanıcılarına satın alacakları bir ürünün seçimi sonrasında önerdiği seçeneklerin kullanıcının satın almak üzere olduğu ürün ile ilişkili olmaktan ziyade daha önceki gezinmelerinde göz attığı ürünlerden oluştuğu gözlemlenmiştir. Buna bir çözüm olarak anlamsal ağ teknolojisinde karşımıza çıkan ontolojiler kullanılarak mevcut veritabanı sistemlerinde karmaşık bir yapıya sebebiyet verebilecek tablo ilişkilendirmeleri daha basite indirgenmiştir. Böylece geliştirme aşamasında sisteme performans artışı sağlayarak işletmeleri ek maliyetlerden kurtarmak ve bu geliştirme sonucunda kullanıcılara daha ilgili ürünler önererek kullanıcıların zamandan tasarruf etmeleri hedeflenmiştir.

5.2.Uygulama Arayüzü



Şekil 18 Uygulama arayüzünün görüntüsü

Şekil 18’de uygulama içerisinde Apache Jena Fuseki yerel sunucusuna gönderilen sorguların çıktıları gösterilmiştir.

BÖLÜM VI

SONUÇ VE ÖNERİLER

Yapılan bu çalışma kapsamında anlamsal ağ teknolojisi hakkında bilgiler, bu teknoloji içerisinde kullanılan araçlar ve bu araçların nasıl kullanılacağına dair örnekler paylaşılmıştır. Sürecin ilk aşamasında anlamsal ağ teknolojisi kullanılarak geliştirilecek bir uygulama için ihtiyaç duyulacak araçlar belirlenip bu araçlar ile ilgili kabul görmüş yöntemler araştırılmıştır. Bir ontoloji geliştirmek için öncelikle ontolojiler ile ilgili genel kavramlar öğrenilerek ontoloji hazırlama kuralları incelenmiştir ve geliştirilmek istenen sisteme uygun olacak ontoloji geliştirme dili olarak OWL, geliştirme ortamı olarak da Protégé tercih edilmiştir. Geliştirilen ontolojiye yerel ağ üzerinden erişmek için Apache Jena Fuseki sunucusu kullanılmıştır. Bu sunucu üzerinden ontoloji üzerinde çalışma yapabilmek için Eclipse geliştirme ortamı içerisinde Java programlama dili kullanılarak oluşturulan servislerin SPARQL dilinde yazılmış sorgular kullanılmıştır. Uygulama sunucu katmanında Apache Jena Fuseki çalıştırılmıştır. Ara katmanda Spring Framework ve anlamsal ağ teknolojisini kullanabilmek için gerekli kütüphaneler ile birlikte Java programlama diliyle geliştirmeler yapılmıştır. Ön yüz katmanında ise Java ile geliştirilen servisler üzerinden sunucuya erişebilmek için Thymeleaf, görsel açıdan iyileştirmeler yapabilmek için ise Bootstrap tercih edilmiştir. Uygulama ve ontoloji içerisinde kullanılan sınıf, değişken gibi ifadelerde, veri tabanında sorgulama yaparken oluşabilecek karakter tabanlı sorunları gidermek için İngilizce dili kullanılmıştır. Rehberlerin eski ve az olmasından ötürü uygulama geliştirme aşamasında harcanan zamanın önemli bir bölümü, anlamsal ağ teknolojisinde kullanılacak araçlar ile java programlama dilinin entegre bir şekilde çalışması için harcanmıştır. Bu tez kapsamında anlamsal ağ ile bir uygulama geliştirilmek istendiğinde ihtiyaç duyulacak araçlar ve bunların kullanılması noktasında yardımcı olmaya çalışılmıştır. Gelişen teknoloji, her geçen gün yeni yöntemler ve yaklaşımların ortaya çıkmasına olanak sağlamıştır. Bu nedenden dolayı bu çalışma kapsamında kullanılan araçların, uygulama geliştirilecek dönem içerisinde bulunan en son veya stabil çalışan bir sürümünü seçmek geliştiricilere kolaylık sağlayacaktır. Ontoloji geliştirilirken ortak bir dilin kullanılması dünya çapında kullanılması halinde

farklı dillerde kullanılan karakterlerden ötürü oluşabilecek karışıklıkları önlemede faydalı olacaktır.



KAYNAKÇA

- A Beginner's Guide to Bootstrap (2021), <https://www.codecademy.com/>
Erişim tarihi: 15.03.2022
- Apache Jena Fuseki (t.y.) <https://jena.apache.org/documentation/fuseki2/>
Erişim tarihi: 07.01.2022
- Badr, Y., Chbeir, R., Abraham, A., ve Hassanien, A. E. (2010) *Emergent Web Intelligence: Advanced Semantic Technologies*. Springer
- Baz, F. Ç. ve Meral, H. C. (2021). Web 3.0 – Anlamsal Ağ Çalışmalarının Farklı Alanlarda Kullanımı: Bir Uygulama Geliştirme Örneği. *Osmaniye Korkut Ata Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 4(3), 408-421.
- Berners-Lee, T. ve Masinter, L. & McCahill, M. (1994). *Uniform Resource Locators* (URL), https://www.researchgate.net/publication/224001129_Uniform_Resource_Locators_URL_Online Erişim tarihi: 15.09.2022
- Berners-Lee, T. ve Fielding, R. and Masinter, L. (2005). Uniform Resource Identifier (URI): Generic Syntax, <https://www.ietf.org/rfc/rfc3986.txt>
Erişim tarihi: 15.09.2022
- Berners-Lee, T. (2004). Tim Berners-Lee's Semantic Web, https://www.researchgate.net/publication/307845029_Tim_Berners-Lee's_Semantic_Web Erişim tarihi: 15.09.2022
- Hazer, B. (2015). “*Anlamsal Ağ Tabanlı İlaç Öneri Sistemi*”. Yüksek lisans tezi, Erciyes Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı.
- Hebeler, J., Fisher, M., Blace, R., Perez-Lopez, A. ve Dean, M. (2009). *Semantic Web Programming*. Wiley inc.
- Kalelkar, M. ve Churi, P. ve Kalelkar, D. (2014). Implementation of model-view-controller architecture pattern for business intelligence architecture. *International Journal of Computer Applications*, 102(12).
- Kumar, A. P. ve Kumar, A. ve Kumar, V. N. (2011). A comprehensive comparative study of SPARQL and SQL. *International Journal of Computer Science and Information Technologies*, 2(4), 1706-1710.
- Melton, J. (1996). Sql language summary. *Acm Computing Surveys (CSUR)*,

28(1), 141-143

- Meral, E. ve Dai, N. ve Dalıcı, R. (Ed.). (2014). Eclipse ile Java, 1, Dikeyksen, Mutlu Basım Yayın.
- Peng, L. ve Zhang, W. ve Wang, X. ve Liang, S. (2019). Moderating effects of time pressure on the relationship between perceived value and purchase intention in social E-commerce sales promotion: Considering the impact of product involvement. *Information & Management*, 56(2), 317-328.
- Purwanto, A. (2022). What Is The Role Of Customer Behavior For Electronic E-Commerce And Modern Market Visit Intention? *Journal of Information Systems and Management (JISMA)*, 1(6), 46-57
- Qin, Z. (2009). *Introduction to E-commerce*. Tsinghua University Press, 4-8.
- Qureshi, I. ve Fang, Y., Ramsey, E., McCole, P., Ibbotson, P. ve Compeau, D. (2009). Understanding online customer repurchasing intention and the mediating role of trust—an empirical investigation in two developed countries. *European Journal of Information Systems*, 18(3), 205-222.
- Rodriguez-Muro, M., Kontchakov, R. ve Zakharyashev, M. (2013). *Ontology-Based Data Access: Ontop of Databases*. Springer
- Spring Framework Overview (2022). <https://docs.spring.io/spring-framework/>
Erişim tarihi: 30.01.2022
- Tutorial: Using Thymeleaf (2018). <https://www.thymeleaf.org/> Erişim tarihi: 15.03.2022
- Türk Dil Kurumu (t.y.). <https://sozluk.gov.tr/>, Erişim tarihi: 04.12.2021
- Yu, L. ve Gerstner, R. (2014). *A Developer's Guide to the Semantic Web*. Springer .

EKLER

Uygulama geliştirme aşamasında kullanılan OWL ve RDF yapıları

Ek 1

Uygulamada Kullanılan Kaynak Tanımlama Çerçevesi Bölümü

```

owl:inverseOfsemanticcommerce:isLaptopAccesory.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#hasNotebookAccesory
semanticcommerce:hasNotebookAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticcommerce:isNotebookAccesory.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#hasOutdoorAccesory
semanticcommerce:hasOutdoorAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticcommerce:isOutdoorAccesory.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#hasPCAccesory
semanticcommerce:hasPCAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticcommerce:isPCAccesory.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#hasPhoneAccesory
semanticcommerce:hasPhoneAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticcommerce:isPhoneAccesory.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#hasSportAccesory
semanticcommerce:hasSportAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticcommerce:isSportAccesory.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#hasTVAccesory
semanticcommerce:hasTVAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticcommerce:isTVAccesory.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#isArcicleAccesory
semanticcommerce:isArcicleAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#isBookAccesory
semanticcommerce:isBookAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#isClothAccesory
semanticcommerce:isClothAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#isDesktopAccesory
semanticcommerce:isDesktopAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#isEducationalAccesory
semanticcommerce:isEducationalAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#isElectronicAccesory
semanticcommerce:isElectronicAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticcommerce#isFitnessAccesory

```

Uygulamada Kullanılan Kaynak Tanımlama Çerçevesi Bölümü

```

@prefix:<http://www.semanticweb.org/oku/ybs/semanticecommerce#>.
@prefixowl:<http://www.w3.org/2002/07/owl#>.
@prefixrdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefixxml:<http://www.w3.org/XML/1998/namespace>.
@prefixxsd:<http://www.w3.org/2001/XMLSchema#>.
@prefixrdfs:<http://www.w3.org/2000/01/rdf-schema#>.
@prefixsemanticecommerce:<http://www.semanticweb.org/oku/ybs/semanticecommerce#>.
@base<http://www.semanticweb.org/oku/ybs/semanticecommerce>.
<http://www.semanticweb.org/oku/ybs/semanticecommerce>rdf:typeowl:Ontology.
#####
#ObjectProperties
#####
###http://www.semanticweb.org/oku/ybs/semanticecommerce#hasArcicleAccesory
semanticecommerce:hasArcicleAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticecommerce:isArcicleAccesory.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#hasBookAccesory
semanticecommerce:hasBookAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticecommerce:isBookAccesory.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#hasClothAccesory
semanticecommerce:hasClothAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticecommerce:isClothAccesory.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#hasDesktopAccesory
semanticecommerce:hasDesktopAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticecommerce:isDesktopAccesory.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#hasEducationalAccesory
semanticecommerce:hasEducationalAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticecommerce:isEducationalAccesory.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#hasElectronicAccesory
semanticecommerce:hasElectronicAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticecommerce:isElectronicAccesory.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#hasFitnessAccesory
semanticecommerce:hasFitnessAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;
owl:inverseOfsemanticecommerce:isFitnessAccesory.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#hasLaptopAccesory
semanticecommerce:hasLaptopAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty;

```

Uygulamada Kullanılan Kaynak Tanımlama Çerçevesi Bölümü

```

semanticecommerce:isFitnessAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#isLaptopAccesory
semanticecommerce:isLaptopAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#isNotebookAccesory
semanticecommerce:isNotebookAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#isOutdoorAccesory
semanticecommerce:isOutdoorAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#isPCAccesory
semanticecommerce:isPCAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#isPhoneAccesory
semanticecommerce:isPhoneAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#isSportAccesory
semanticecommerce:isSportAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#isTVAccesory
semanticecommerce:isTVAccesoryrdf:typeowl:ObjectProperty;
rdfs:subPropertyOfowl:topObjectProperty.
#####
#Classes
#####
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Article
semanticecommerce:Articlerdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Educational.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Book
semanticecommerce:Bookrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Educational.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Cathegories
semanticecommerce:Cathegoriesrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Products.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Clothes
semanticecommerce:Clothesrdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:Sport.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Desktop
semanticecommerce:Desktoprdf:typeowl:Class;
rdfs:subClassOfsemanticecommerce:PC;
owl:disjointWithsemanticecommerce:Laptop.
###http://www.semanticweb.org/oku/ybs/semanticecommerce#Educational

```

Uygulamada Kullanılan Kaynak Tanımlama Çerçevesi Bölümü

```
#####
#Generalaxioms
#####
[rdftypeowl:AllDisjointClasses;
owl:members(semanticcommerce:Article
semanticcommerce:Book
semanticcommerce:Notebook
)].
[rdftypeowl:AllDisjointClasses;
owl:members(semanticcommerce:Clothes
semanticcommerce:Fitness
semanticcommerce:Outdoor
)].
[rdftypeowl:AllDisjointClasses;
owl:members(semanticcommerce:Educational
semanticcommerce:Electronics
semanticcommerce:Sport
)].
[rdftypeowl:AllDisjointClasses;
owl:members(semanticcommerce:PC
semanticcommerce:Phone
semanticcommerce:TV
)].
###GeneratedbytheOWLAPI(version4.5.9.2019-02-01T07:24:44Z)https://github.com/owlcs/owlapi
```



Uygulamada Kullanılan OWL Bölümü

```

<Declaration>ObjectProperty IRI="#hasArcicleAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasBookAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasClothAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasDesktopAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasEducationalAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasElectronicAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasFitnessAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasLaptopAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasNotebookAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasOutdoorAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasPCAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasPhoneAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasSportAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#hasTVAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isArcicleAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isBookAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isClothAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isDesktopAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isEducationalAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isElectronicAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isFitnessAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isLaptopAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isNotebookAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isOutdoorAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isPCAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isPhoneAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isSportAccesory"/>Declaration>
<Declaration>ObjectProperty IRI="#isTVAccesory"/>Declaration>
<SubClassOf>Class IRI="#Article"/>Class IRI="#Educational"/>
</SubClassOf>SubClassOf>Class IRI="#Book"/>Class IRI="#Educational"/>
</SubClassOf>SubClassOf>Class IRI="#Cathegories"/>Class IRI="#Products"/>

```

```

</SubClassOf>SubClassOf>Class IRI="#Clothes"/>Class IRI="#Sport"/>
</SubClassOf>SubClassOf>Class IRI="#Desktop"/>Class IRI="#PC"/>
</SubClassOf>SubClassOf>Class IRI="#Educational"/>Class IRI="#Categories"/>
</SubClassOf>SubClassOf>Class IRI="#Electronics"/>Class IRI="#Categories"/>
</SubClassOf>SubClassOf>Class IRI="#Fitness"/>Class IRI="#Sport"/>
</SubClassOf>SubClassOf>Class IRI="#Laptop"/>Class IRI="#PC"/>
</SubClassOf>SubClassOf>Class IRI="#Notebook"/>Class IRI="#Educational"/>
</SubClassOf>SubClassOf>Class IRI="#Outdoor"/>Class IRI="#Sport"/>
</SubClassOf>SubClassOf>Class IRI="#PC"/>Class IRI="#Electronics"/>
</SubClassOf>SubClassOf>Class IRI="#Phone"/>Class IRI="#Electronics"/>
</SubClassOf>SubClassOf>Class IRI="#Sport"/>Class IRI="#Categories"/>
</SubClassOf>SubClassOf>Class IRI="#TV"/>Class IRI="#Electronics"/>
</SubClassOf>DisjointClasses>Class IRI="#Article"/>Class IRI="#Book"/>
  <Class IRI="#Notebook"/>DisjointClasses>DisjointClasses>
  <Class IRI="#Clothes"/>Class IRI="#Fitness"/>Class IRI="#Outdoor"/>
</DisjointClasses>DisjointClasses>Class IRI="#Desktop"/>Class IRI="#Laptop"/>
</DisjointClasses>DisjointClasses>Class IRI="#Educational"/>Class IRI="#Electronics"/>
  <Class IRI="#Sport"/>DisjointClasses>DisjointClasses>Class IRI="#PC"/>
  <Class IRI="#Phone"/>Class IRI="#TV"/>DisjointClasses>SubObjectPropertyOf>
  <ObjectProperty IRI="#hasArcicleAccesory"/>
  <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
  <ObjectProperty IRI="#hasBookAccesory"/>
  <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
  <ObjectProperty IRI="#hasClothAccesory"/>
  <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
  <ObjectProperty IRI="#hasDesktopAccesory"/>
  <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>

```

Uygulamada Kullanılan OWL Bölümü

```

    <ObjectProperty IRI="#hasEducationalAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
    <ObjectProperty IRI="#hasElectronicAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
    <ObjectProperty IRI="#hasFitnessAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
    <ObjectProperty IRI="#hasLaptopAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
    <ObjectProperty IRI="#hasNotebookAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
    <ObjectProperty IRI="#hasOutdoorAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
    <ObjectProperty IRI="#hasPCAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
    <ObjectProperty IRI="#hasPhoneAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
    <ObjectProperty IRI="#hasSportAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
    <ObjectProperty IRI="#hasTVAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
    <ObjectProperty IRI="#isArcicleAccesory"/>

```

Uygulamada Kullanılan OWL Bölümü

```

    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
  </SubObjectPropertyOf>SubObjectPropertyOf
    <ObjectProperty IRI="#isBookAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
  </SubObjectPropertyOf>SubObjectPropertyOf
    <ObjectProperty IRI="#isClothAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
  </SubObjectPropertyOf>SubObjectPropertyOf
    <ObjectProperty IRI="#isDesktopAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
  </SubObjectPropertyOf>SubObjectPropertyOf
    <ObjectProperty IRI="#isEducationalAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
  </SubObjectPropertyOf>SubObjectPropertyOf
    <ObjectProperty IRI="#isElectronicAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
  </SubObjectPropertyOf>SubObjectPropertyOf
    <ObjectProperty IRI="#isFitnessAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
  </SubObjectPropertyOf>SubObjectPropertyOf
    <ObjectProperty IRI="#isLaptopAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
  </SubObjectPropertyOf>SubObjectPropertyOf
    <ObjectProperty IRI="#isNotebookAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
  </SubObjectPropertyOf>SubObjectPropertyOf
    <ObjectProperty IRI="#isOutdoorAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
  </SubObjectPropertyOf>SubObjectPropertyOf
    <ObjectProperty IRI="#isPcAccesory"/>
    <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>

```

Uygulamada Kullanılan OWL Bölümü

```

</SubObjectPropertyOf>SubObjectPropertyOf>
  <ObjectProperty IRI="#isPhoneAccesory"/>
  <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
  <ObjectProperty IRI="#isSportAccesory"/>
  <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>SubObjectPropertyOf>
  <ObjectProperty IRI="#isTVAccesory"/>
  <ObjectProperty abbreviatedIRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>InverseObjectProperties>
  <ObjectProperty IRI="#hasArcicleAccesory"/>
  <ObjectProperty IRI="#isArcicleAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasBookAccesory"/>
  <ObjectProperty IRI="#isBookAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasClothAccesory"/>
  <ObjectProperty IRI="#isClothAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasDesktopAccesory"/>
  <ObjectProperty IRI="#isDesktopAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasEducationalAccesory"/>
  <ObjectProperty IRI="#isEducationalAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasElectronicAccesory"/>
  <ObjectProperty IRI="#isElectronicAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasFitnessAccesory"/>
  <ObjectProperty IRI="#isFitnessAccesory"/>
</InverseObjectProperties>InverseObjectProperties>

```

Uygulamada Kullanılan OWL Bölümü

```
<ObjectProperty IRI="#hasLaptopAccesory"/>
<ObjectProperty IRI="#isLaptopAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasNotebookAccesory"/>
  <ObjectProperty IRI="#isNotebookAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasOutdoorAccesory"/>
  <ObjectProperty IRI="#isOutdoorAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasPCAccesory"/>
  <ObjectProperty IRI="#isPCAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasPhoneAccesory"/>
  <ObjectProperty IRI="#isPhoneAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasSportAccesory"/>
  <ObjectProperty IRI="#isSportAccesory"/>
</InverseObjectProperties>InverseObjectProperties>
  <ObjectProperty IRI="#hasTVAccesory"/>
  <ObjectProperty IRI="#isTVAccesory"/>
</InverseObjectProperties></Ontology>
```

ÖZGEÇMİŞ

KİŞİSEL BİLGİLERİ

Adı ve Soyadı : Hilmi Cihan MERAL

EĞİTİM BİLGİLERİ

Yüksek Lisans: Osmaniye Korkut Ata Üniversitesi, Sosyal Bilimler Enstitüsü,
Yönetim Bilişim Sistemleri Anabilim Dalı, 2020-2022.

Lisans : Osmaniye Korkut Ata Üniversitesi, İktisadi ve İdari Bilimler Fakültesi,
Yönetim Bilişim Sistemleri Bölümü, 2013-2017.

YAYINLAR

Baz, F. Ç. ve Meral, H. C. (2021). Web 3.0 – Anlamsal Ağ Çalışmalarının Farklı Alanlarda Kullanımı: Bir Uygulama Geliştirme Örneği. *Osmaniye Korkut Ata Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 4(3), 408-421.

İŞ DENEYİMİ

Yıl	Kurum	Görev
2020-Halen	Pbs Bilgi Teknolojileri ve Bilişim Hizmetleri A.Ş.	SAP BASIS Danışmanlığı



T.C.
OSMANİYE KORKUT ATA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
TEZ ÇALIŞMASI BENZERLİK RAPORU FORMU

FORM
TEZLİ YL-16

ÖĞRENCİ BİLGİLERİ

Adı ve Soyadı	Hilmi Cihan Meral
Öğrenci Numarası	2021507103
Ana Bilim/ Ana Sanat Dalı	Yönetim Bilişim Sistemleri
Bilim / Sanat Dalı	Yönetim Bilişim Sistemleri
Danışman Unvanı, Adı-Soyadı	Dr. Öğr. Üyesi Fatih Çağatay Baz
Tez Başlığı (Türkçe)	Anlamsal Ağ Odaklı Kişiselleştirilmiş E-Ticaret Ürün Öneri Sistemi

SOSYAL BİLİMLER ENSTİTÜSÜ MÜDÜRLÜĞÜNE

Yukarıda başlığı belirtilen tez çalışmamın a) Kapak sayfası, b) Giriş, c) Ana bölümler ve d) Sonuç kısımlarından oluşan toplam 49 sayfalık kısmına ilişkin, 16/09/2022 tarihinde Sosyal Bilimler Enstitüsü tarafından Turnitin adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı % 10 'dur.

Filtreleme Tip 1 (maksimum %30)

Filtreleme Tip 2 (maksimum %10)

- 1- Kabul/Onay ve Bildirim sayfaları hariç,
2- Kaynakça dâhil,
3- Alıntılar dâhil.

- 1- Kabul/Onay ve Bildirim sayfaları hariç,
2- Kaynakça hariç,
3- Alıntılar dâhil,
4- 5 Kelimeden daha az örtüşme içeren metin kısımları hariç.

Osmaniye Korkut Ata Üniversitesi Sosyal Bilimler Enstitüsü Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Uygulama Esasları'nı inceledim ve bu Uygulama Esasları'nda belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Tarih ve İmza
Öğrenci

Danışman Onayı
UYGUNDUR

Enstitü Onayı
UYGUNDUR

Unvanı, Adı Soyadı
(İmzası)

Adı Soyadı
(İmzası)

AÇIKLAMALAR

- Lisansüstü tezler, savunma öncesinde intihal program raporu ile birlikte Enstitüye teslim edilir.
- İntihal raporu ile ilgili olarak etik kurallar dâhilindeki benzerlik oranları ilgili Enstitü Yönetim Kurulu tarafından belirlenir. (Enstitü Yönetim Kurulu tarafından tezin, intihal kapsamı dışında değerlendirilmesi için TURNITIN'den alınan raporda "benzerlik oranı"nın, "alıntılar dahil" en fazla %10, "alıntılar dahil" % 30'u geçmemesi şeklinde kabul edilmiştir).