



T.C.  
EGE ÜNİVERSİTESİ  
Fen Bilimleri Enstitüsü



**ÇOK-ETMENLİ SİSTEMLER İÇİN  
GELİŞTİRİLEN ALANA-ÖZGÜ MODELLEME  
DİLLERİNİN SİSTEMATİK  
DEĞERLENDİRİLMESİNİ SAĞLAYAN BİR ARAÇ**

**Yüksek Lisans Tezi**

Ömer Faruk ALACA

Uluslararası Bilgisayar Anabilim Dalı

İzmir  
2019



T.C.  
EGE ÜNİVERSİTESİ  
Fen Bilimleri Enstitüsü

**ÇOK-ETMENLİ SİSTEMLER İÇİN  
GELİŞTİRİLEN ALANA-ÖZGÜ MODELLEME  
DİLLERİNİN SİSTEMATİK  
DEĞERLENDİRİLMESİNİ SAĞLAYAN BİR ARAÇ**

Ömer Faruk ALACA

Danışman : Doç. Dr. Geylani KARDAŞ

Uluslararası Bilgisayar Anabilim Dalı  
Bilgi Teknolojileri

İzmir  
2019




Ömer Faruk ALACA tarafından YÜKSEK LİSANS tezi olarak sunulan “Çok-Etmenli Sistemler için Geliştirilen Alana-Özgü Modelleme Dillerinin Sistematik Değerlendirilmesini Sağlayan bir Araç” başlıklı bu çalışma EÜ Lisansüstü Eğitim ve Öğretim Yönetmeliği ile EÜ Fen Bilimleri Enstitüsü Eğitim ve Öğretim Yönergesi'nin ilgili hükümleri uyarınca tarafımızdan değerlendirilerek savunmaya değer bulunmuş ve 26.08.2019 tarihinde yapılan tez savunma sınavında aday oybirliği/oyçokluğu ile başarılı bulunmuştur.

**Jüri Üyeleri:**

**Jüri Başkanı** : Doç. Dr. Geylani KARDAŞ  
**Raportör Üye** : Prof. Dr. Oğuz DİKENELİ  
**Üye** : Prof. Dr. Onur DEMİRÖRS

**İmza**

  
.....  
  
.....  
  
.....



## EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

### ETİK KURALLARA UYGUNLUK BEYANI

EÜ Lisansüstü Eğitim ve Öğretim Yönetmeliğinin ilgili hükümleri uyarınca Yüksek Lisans Tezi olarak sunduğum “Çok-Etmenli Sistemler için Geliştirilen Alana-Özgü Modelleme Dillerinin Sistemik Değerlendirilmesini Sağlayan bir Araç” başlıklı bu tezin kendi çalışmam olduğunu, sunduğum tüm sonuç, doküman, bilgi ve belgeleri bizzat ve bu tez çalışması kapsamında elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara atıf yaptığımı ve bunları kaynaklar listesinde usulüne uygun olarak verdiğimi, tez çalışması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını, bu tezin herhangi bir bölümünü bu üniversite veya diğer bir üniversitede başka bir tez çalışması içinde sunmadığımı, bu tezin planlanmasından yazımına kadar bütün safhalarda bilimsel etik kurallarına uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul edeceğimi beyan ederim.

26 / 08 / 2019

Ömer Faruk ALACA





## ÖNSÖZ

Çok-etmenli sistemlere ait yazılımların geliştirilmesini kolaylaştırmak amacıyla etmen tabanlı yazılım mühendisliği araştırma alanı içerisinde özellikle 2000’li yılların başından itibaren birçok modelleme dili ve alana-özü dil önerilmiştir. Ancak bu dillerin etmen yazılımlarını oluşturmada sağladığı verimliliği değerlendirmek ve diğer alternatifler ile karşılaştırmak için çok az sayıda çalışma ortaya konmuştur. Bu alanda çalışanlar arasında ortak bir yaklaşımın oluşturulması, farklı diller arasında ihtiyaca uygun dilin belirlenebilmesi ve dillerin gelişim süreçlerinde belirli metrikleri baz alarak yol haritası belirleyebilmek için sistematik bir değerlendirme yaklaşımı zorunluluk haline gelmektedir.

Yukarıda sözü edilen ihtiyaçlardan hareketle bu tez çalışmanın gerekliliği ve çerçevesi belirlenmiştir. Bu doğrultuda yapılan literatür incelemesi, araştırma, geliştirme ve laboratuvar çalışmaları sonucu bu tezde etmen modelleme dillerinin değerlendirilmesini sağlayacak bir araç sunulmuştur.

İZMİR

26/08/2019

Ömer Faruk ALACA



**ÖZET****ÇOK-ETMENLİ SİSTEMLER İÇİN GELİŞTİRİLEN  
ALANA-ÖZGÜ MODELLEME DİLLERİNİN SİSTEMATİK  
DEĞERLENDİRİLMESİNİ SAĞLAYAN BİR ARAÇ**

ALACA, Ömer Faruk

Yüksek Lisans Tezi, Uluslararası Bilgisayar Anabilim Dalı

Tez Danışmanı: Doç. Dr. Geylani KARDAŞ

Ağustos 2019, 95 sayfa

Çok-etmenli sistemlerin (MAS) geliştirilmesi için Etmen-tabanlı Yazılım Geliştirme (AOSE) araştırma alanında birçok alana-özü modelleme dili (DSML) önerilmiştir. Bu MAS DSML'lerinin MAS DSML geliştiricisi ve kullanıcısı gibi paydaşların çeşitli ihtiyaçlarını karşılayacak yapıda olması ve MAS geliştirmeyi ne derece etkin ve kolay hale getirdiğinin belirlenmesi gerekmektedir. Ancak literatürdeki MAS DSML çalışmaları incelendiğinde bunların özellikle dil bileşenleri, üretilen çıktılar ve DSML'in etmen tabanlı yazılımın tasarımı ve inşası sırasında kullanımını içeren bir değerlendirmesinin hiç yapılmadığı ya da çok yüzeysel bir şekilde yerine getirildiği görülmektedir. AOSE alanında yukarıda sözü edilen eksikliği gidermek amacıyla bu tezde MAS DSML'lerinin belirlenen bir takım metriklere göre sistematik olarak niceliksel ve niteliksel açıdan çevrimiçi değerlendirilmesini destekleyen bir web tabanlı yazılım aracı geliştirilmiştir.

Araç içerisinde niceliksel değerlendirme sırasında ilk olarak değerlendirilen MAS DSML'inin üstmodeli bir referans üstmodel ile kıyaslanarak dilin MAS alanını kapsayıcılığı belirlenmektedir. Çoklu durum çalışması temelli niceliksel değerlendirmede ise dilin MAS yazılımı geliştirme süresi ve çıktı üretim performansı göz önüne alınmaktadır. Son olarak dili kullanan MAS geliştiricilerinin çevrimiçi bu araç üzerinden cevapladıkları bir anket ile dilin belirli metriklere göre niteliksel değerlendirmesi yapılmaktadır. Örnek uygulama olarak AOSE alanında iyi bilinen Prometheus/PDT dilinin geliştirilen bu araç üzerinden niceliksel ve niteliksel değerlendirilmesi de yine tez kapsamında yerine getirilmiştir.

**Anahtar sözcükler:** Çok-etmenli sistem, Alana-özü modelleme dili, Etmen-tabanlı yazılım mühendisliği.



**ABSTRACT****A TOOL SUPPORTING THE SYSTEMATIC EVALUATION OF  
DOMAIN-SPECIFIC MODELING LANGUAGES  
FOR MULTI-AGENT SYSTEMS**

ALACA, Ömer Faruk

MSc in International Computer

Supervisor: Assoc. Prof. Dr. Geylani KARDAŞ

August 2019, 95 pages

In Agent-oriented software engineering (AOSE) research field, various domain-specific modeling languages (DSMLs) are proposed for the development of Multi-agent Systems (MASs). These DSMLs are supposed to meet software development requirements of both MAS DSML developers and MAS DSML users. Moreover, they should be evaluated to determine how they both speed up and facilitate MAS development. However, many MAS DSML studies does not consider such an evaluation of language components and generated artifacts over using the language while design and implementation of agent software. In order to fill this gap in AOSE research, a web-based tool is developed in this thesis. The tool provides the systematic evaluation of MAS DSMLs from both quantitative and qualitative aspects according to a series of pre-defined metrics.

During the quantitative evaluation inside the tool, MAS domain coverage is determined by comparing a DSML's metamodel with a reference MAS metamodel. In the second part of the quantitative evaluation, which is constructed on a multi-case study, the software development time and the artifact generation performance are taken into account. Finally, qualitative evaluation of the language is performed in which MAS developers are requested to answer a questionnaire and assess the language according to some quality metrics. Use of the proposed online MAS DSML evaluation process is exemplified in the thesis with the evaluation of Prometheus/PDT, a well-known tool in AOSE.

**Keywords:** Multi-agent system, Domain-specific modeling language, Agent-oriented software engineering.



**İÇİNDEKİLER**

|   | <u>Sayfa</u> |
|---|--------------|
| ÖNSÖZ.....  | v            |
| ÖZET .....  | vii          |
| ABSTRACT .....  | ix           |
| İÇİNDEKİLER DİZİNİ.....   | xi           |
| ŞEKİLLER DİZİNİ .....   | xiv          |
| TABLolar DİZİNİ.....  | xvii         |
| SİMGELER VE KISALTMALAR DİZİNİ.....                                 | xviii        |
| 1.GİRİŞ.....  | 1            |
| 2.ALTYAPI.....  | 3            |
| 2.1 Etmen ve Çok Etmenli Sistemler.....                             | 3            |
| 2.2 Model GÜdümlü Mühendislik ve Alana-özgü Modelleme Dilleri ..... | 4            |
| 2.3 FAML Üstmodeli .....  | 5            |
| 2.4 Prometheus/PDT.....   | 6            |
| 2.5 Araçlar ve Teknolojiler.....                                    | 8            |

**İÇİNDEKİLER (devam)**

|   | <u>Sayfa</u> |
|---|--------------|
| 3. İLGİLİ ÇALIŞMALAR.....   | 10           |
| 3.1 MAS Üstmodelleme ve Model Güdümlü Geliştirme Çalışmaları..... | 10           |
| 3.2 MAS DSL/DSML Çalışmaları.....                                 | 11           |
| 4. DEĞERLENDİRME ÇERÇEVESİ.....                                   | 18           |
| 4.1 Genel Bakış.....  | 18           |
| 4.2 Çoklu Durum Çalışması.....                                    | 19           |
| 4.3 Niceliksel Değerlendirme Yaklaşımı.....                       | 19           |
| 4.3.1 Referans üstmodel ile kıyaslama .....                       | 20           |
| 4.3.2 Durum çalışması analizi.....                                | 23           |
| 4.4 Niteliksel Değerlendirme Yaklaşımı.....                       | 24           |
| 5. AGENTDSM-EVAL DEĞERLENDİRME ARACI.....                         | 27           |
| 5.1 AgentDSM-Eval Uygulaması .....                                | 27           |
| 5.1.1 Değerlendirme formu sayfası.....                            | 27           |
| 5.1.2 Değerlendirme sonuç sayfası .....                           | 31           |



**İÇİNDEKİLER (devam)**

|   | <u>Sayfa</u> |
|---|--------------|
| 6. DEĞERLENDİRME ARACININ KULLANIMI ÖRNEĞİ..... | 38           |
| 6.1 Üstmodel Analizi.....                       | 38           |
| 6.2 Değerlendirme Süreci.....                   | 38           |
| 6.2.1 Seçilen durum çalışmaları.....            | 39           |
| 6.3 Değerlendirme Sonuçları.....                | 41           |
| 6.3.1 Niceliksel değerlendirme sonuçları.....   | 41           |
| 6.3.2 Niteliksel değerlendirme sonuçları.....   | 44           |
| 6.3.3 Çıkarımlar ve sonuç.....                  | 53           |
| 7. SONUÇ.....                                   | 54           |
| KAYNAKLAR DİZİNİ.....                           | 57           |
| TEŞEKKÜR.....                                   | 65           |
| ÖZGEÇMİŞ.....                                   | 66           |
| EKLER.....                                      | .....        |

## ŞEKİLLER DİZİNİ

| <u>Şekil</u>   | <u>Sayfa</u> |
|--|--------------|
| 2.1 Prometheus Metadolojisi Süreçleri ((Padgham and Winikoff, 2005)'dan alınmıştır.).....  | 7            |
| 2.2 PDT Aracı Arayüzü ((Padgham and Winikoff, 2005)'dan alınmıştır.) ...                   | 8            |
| 5.1 Onay metni ekranı.....   | 27           |
| 5.2 Kişisel bilgi girişi ekranı.....   | 28           |
| 5.3 Geliştirme süreleri giriş ekranı .....   | 29           |
| 5.4 Proje dosyaları yükleme ekranı .....   | 29           |
| 5.5 Kalite anketi ekranı.....  | 30           |
| 5.6 Dil genel bilgileri ekranı .....   | 31           |
| 5.7 Referans üstmodeli ile kıyaslama (genel).....  | 32           |
| 5.8 Referans üstmodeli ile kıyaslama (ayrıntılı).....                                      | 32           |
| 5.9 Geliştirme süreleri analizi.....   | 33           |
| 5.10 Niteliksel analiz anketi genel sonuçları .....  | 33           |
| 5.11 Niteliksel analiz anketi karakteristik sonuçları (puan bakış açısı).....              | 34           |
| 5.12 Niteliksel analiz anketi karakteristik sonuçları (alt karakteristik bakış açısı)..... | 35           |
| 5.13 Niteliksel analiz anketi alt karakteristik sonuçları .....                            | 35           |

**ŞEKİLLER DİZİNİ (devam)**

| <u>Şekil</u>  | <u>Sayfa</u> |
|---|--------------|
| 5.14 Modelleme çıktıları sonuçları.....   | 36           |
| 5.15 Kod üretimi performansı sonuçları.....   | 37           |
| 6.1 PDT üstmodelinin referans üstmodeli ile kıyaslama sonuçları .....                                 | 41           |
| 6.2 PDT geliştirme süreleri sonuçları.....  | 42           |
| 6.3 PDT dil elemanları kullanım sıklığı sonuçları.....  | 43           |
| 6.4 PDT dilinde kod üretim performansı sonuçları .....  | 44           |
| 6.5 PDT niteliksel değerlendirme fonksiyonel uygunluk sonuçları (puan bakış açısı) .....              | 45           |
| 6.6 PDT niteliksel değerlendirme fonksiyonel uygunluk sonuçları (alt karakteristik bakış açısı) ..... | 46           |
| 6.7 PDT niteliksel değerlendirme kullanılabilirlik sonuçları (puan bakış açısı).....                  | 46           |
| 6.8 PDT niteliksel değerlendirme kullanılabilirlik sonuçları (alt karakteristik bakış açısı).....     | 47           |
| 6.9 PDT niteliksel değerlendirme güvenilirlik sonuçları (puan bakış açısı) 48                         |              |
| 6.10 PDT niteliksel değerlendirme güvenilirlik sonuçları (alt karakteristik bakış açısı).....         | 48           |
| 6.11 PDT niteliksel değerlendirme anlamlılık sonuçları (puan bakış açısı)...                          | 49           |
| 6.12 PDT niteliksel değerlendirme anlamlılık sonuçları (alt karakteristik bakış açısı).....           | 50           |

**ŞEKİLLER DİZİNİ (devam)**

| <u>Şekil</u>   | <u>Sayfa</u> |
|--|--------------|
| 6.13 PDT niteliksel değerlendirme uygunluk sonuçları (puan bakış açısı) ...                      | 51           |
| 6.14 PDT niteliksel değerlendirme uygunluk sonuçları (alt karakteristik bakış açısı).....        | 51           |
| 6.15 PDT niteliksel değerlendirme MAS geliştirme sonuçları (puan bakış açısı).....               | 52           |
| 6.16 PDT niteliksel değerlendirme MAS geliştirme sonuçları (alt karakteristik bakış açısı) ..... | 52           |
| 6.17 PDT niteliksel değerlendirme sonuçları.....   | 53           |

**TABLolar DİZİNİ**

| <u>Tablo</u>   | <u>Sayfa</u> |
|--|--------------|
| 4.1 FAML üstmodeli tasarım zamanı kavramları (Beydoun et al., 2009).....                           | 21           |
| 4.2 FAML üstmodeli çalışma zamanı kavramları (Beydoun et al., 2009) .....                          | 22           |
| 4.3 AgentDSM-Eval aracında yer alan niteliksel analiz karakteristik ve alt karakteristikleri ..... | 25           |



## SİMGELER VE KISALTMALAR DİZİNİ

### Kısaltmalar

|               |  |
|---------------|--|
| A & A         | Etmenler ve Yapılar (ing. Agents & Artifacts)  |
| ACL           | Etmen İletişim Dili (ing. Agent Communication Language)  |
| AgentDSM-Eval | Etmen Alanına Özel Modelleme Değerlendirmesi (ing. Agent Domain-specific Modeling Evaluation)        |
| AOSE          | Etmen Tabanlı Yazılım Mühendisliği (ing. Agent Oriented Software Engineering)                        |
| BDI           | Kanı-İstek-Hedef (ing. Belief-Desire-Intention)  |
| DMM           | Afet Yönetimi Üstmodeli (ing. Disaster Management Metamodel)   |
| DSL           | Alana-özü Dil (ing. Domain-specific Language)  |
| DSML          | Alana-özü Modelleme Dili (ing. Domain-specific Modeling Language)                                    |
| ERE           | Acil Müdahale Ortamı (ing. Emergency Response Environment)   |
| FAML          | FAME Etmen Tabanlı Modelleme Dili (ing. FAME Agent-oriented Modelling Language)                      |
| FAME          | Etmen Tabanlı Metot Mühendisliği için Çerçeve (ing. Framework for Agent-Oriented Method Engineering) |
| FIPA          | Akıllı Fiziksel Etmenler Kuruluşu (ing. Foundation for Intelligent Physical Agents)                  |
| FQAD          | Nitel Değerlendirme Çerçevesi (ing. Framework for Qualitative Assessment)                            |

**SİMGELER VE KISALTMALAR DİZİNİ (devam)**Kısaltmalar

|     |   |
|-----|---|
| GPL | Genel Amaçlı Dil (ing. General Purpose Language)                    |
| IDK | INGENIAS Geliştirme Aracı (ing. INGENIAS Development Kit)           |
| MAS | Çok-etmenli Sistem (ing. Multi-agent System)                        |
| MDA | Model Güdümlü Mimari (ing. Model Driven Architecture)               |
| MDD | Model Güdümlü Geliştirme (ing. Model Driven Development)            |
| M2M | Modelden Modele Dönüşüm (ing. Model to Model Transformation)        |
| M2T | Modelden Metine Dönüşüm (ing. Model to Text Transformation)         |
| OCL | Nesne Kısıt Dili (ing. Object Constraint Language)                  |
| OMG | Nesne Yönetim Grubu (ing. Object Management Group)                  |
| OWL | Web Ontoloji Dili (ing. Web Ontology Language)                      |
| PDT | Prometheus Tasarım Aracı (ing. Prometheus Design Tool)              |
| SWS | Anlamsal Web Servis (ing. Semantic Web Service)                     |
| TAO | Etmenlerin ve Nesnelerin İşlenmesi (ing. Taming Agents and Objects) |





## 1. GİRİŞ

Yazılım etmenleri (ing. software agents) ve bunların oluşturduğu Çok-etmenli Sistemler (Multi-agent Systems - MAS) özerk ve dağıtık yapıdaki sistemlerin modellenmesi ve geliştirilmesinde etkin olarak kullanılmaktadır. Ancak, etmenlerin özerk, karşıt-eylemli ve pro-aktif yapıları etmen tabanlı yazılım sistemlerinin geliştirilmesini diğer yazılım sistemlerine göre daha zor bir hale getirmektedir. Bununla birlikte, etmenlerin davranış modelleri ve etmen organizasyonları içerisindeki etkileşimleri farklı etmen ortamlarının değişken ihtiyaçları göz önüne alındığında daha karmaşık ve uygulaması zor bir yapıya bürünmektedir. İçsel karmaşıklıkları, dağıtık yapıları ve açık sistemler olmaları sebebiyle MAS'ların geliştirilmesinde yazılım kodlaması öncesinde daha yüksek soyutlama seviyelerinde çalışmak büyük öneme sahiptir.

MAS geliştirme için yukarıda bahsedilen problemlerini çözmek amacıyla, Etmen-yönelimli Yazılım Mühendisliği (Agent-oriented Software Engineering - AOSE) alanında çalışan araştırmacılar, içerisinde etmen modellerinin temel bileşenlerini ve ilişkilerini barındıran çeşitli üstmodeller (ing. metamodel) tanımlamaktadırlar. Araştırmacılar, bu üstmodelleri bazı sözdizim ve anlamlarla zenginleştirerek aynı zamanda etmen geliştirmeyi kolaylaştıracak alana-özgü modelleme dilleri (Domain-specific modeling Language - DSML) de önermektedirler (Kardas and Gomez-Sanz, 2017). Bu dillerin çoğu etmen içsel davranış modelini, diğer etmenlerle etkileşimleri, etmen dışındaki diğer ortam kaynaklarının kullanılmasını, vb. içeren çeşitli bakış açıları (ing. viewpoint) üzerinden etmen yazılımlarının hem statik hem de dinamik modellenmesini sağlamaktadır.

Kullanılması önerilen mevcuttaki etmen DSML'lerinin yazılım geliştirmede görev alan MAS DSML geliştiricisi ve MAS DSML kullanıcısı gibi paydaşların çeşitli ihtiyaçlarını karşılayacak yapıda olması ve MAS geliştirmeyi ne derece etkin ve kolay hale getirdiğinin belirlenmesi gerekmektedir. Ancak literatürdeki MAS DSML çalışmaları (örneğin SEA\_ML (Challenger et al., 2014), DSML4MAS (Hahn, 2008), MAS-ML (Goncalves et al., 2015), JADEL (Bergenti, 2014)) incelendiğinde bunların özellikle dil bileşenleri, üretilen çıktılar ve DSML'in etmen tabanlı yazılımın tasarımı ve inşası sırasında kullanımını içeren bir değerlendirmesinin hiç yapılmadığı ya da çok yüzeysel bir şekilde yerine getirildiği görülmektedir. AOSE alanında yukarıda sözü edilen eksikliği

gidermek amacıyla bu tezde MAS DSML'lerinin belirlenen bir takım metriklere göre değerlendirilmesini sağlayacak bir web tabanlı yazılım aracı geliştirilmiştir.

Tez çalışması, TÜBİTAK Araştırma Destek Programları Başkanlığı (ARDEB) tarafından desteklenen 115E591 no' lu "AgentDSM-Eval: Çok-Etmenli Sistem Alana-özgü Modelleme Dilleri için bir Değerlendirme Çerçevesinin Geliştirilmesi" isimli Ar-Ge projesi kapsamında yürütülmüştür. Bu projede MAS DSML'lerinin hem niceliksel analizi hem de niteliksel değerlendirmesini sağlayacak bir değerlendirme çerçevesi geliştirilmektedir. Tezde önerilen web tabanlı araç AgentDSM-Eval adı verilen bu değerlendirme çerçevesine göre MAS DSML'lerinin özellikle durum çalışması temelli değerlendirmesinin çevrimiçi yapılmasına imkan vermekte ve dil değerlendirme sonuçlarını sunan görsel raporlamaları sağlamaktadır.

Tezin kalan kısmı şu şekilde organize edilmiştir: Bölüm 2'de etmen, MAS, alana özgü diller ve modelleme dilleri, FAME Etmen-tabanlı Modelleme Dili (FAME Agent-oriented Modelling Language - FAML) üstmodeli ve Prometheus Tasarım Aracı (Prometheus Design Tool - PDT) gibi çalışmanın altyapısını oluşturan konular ve tez çalışmasında kullanılan teknolojiler tanıtılmıştır. İlgili çalışmalar Bölüm 3'te incelenmiştir. Bölüm 4'te AgentDSM-Eval değerlendirme çerçevesi tanıtılmıştır. Bölüm 5'te ise AgentDSM-Eval değerlendirme çerçevesinin çevrimiçi web aracı tanıtılmıştır. Bölüm 6'da AgentDSM-Eval değerlendirme çerçevesi ve web aracı kullanarak PDT dili değerlendirilmiş ve sonuçları sunulmuştur. Son bölümde tezde elde edilen sonuçlara ve ileriye dönük yapılabilecek çalışmalara yer verilmiştir.

## 2. ALTYAPI

Bu bölümde çalışmanın altyapında yer alan Etmenler, Çok-etmenli Sistemler, Model Güdümlü Yazılım Geliştirme, Alan-özü Modelleme Dilleri, FAML üstmodeli ve Prometheus/PDT aracı kısaca tanıtılmaktadır. Ayrıca, çalışmada kullanılan yazılım teknolojisi ve araçlar da takip eden altbölümlerde tanıtılmaktadır.

### 2.1 Etmen ve Çok-etmenli Sistemler

Etmen yönelimli programlama paradigması ilk olarak 1990'da yapay zeka alanında Shoham tarafından önerilmiştir (Shoham, 1991). Nesneye yönelik programlama paradigmasının bir uzantısı olarak kabul edilir. Nesne yönelimli paradigmada, sistem nesnelere modellenir ve nesnelere arasındaki etkileşim metotlar aracılığı ile sağlanır. Herhangi bir görevin gerçekleştirilmesi için bir nesnenin çağırılması gerekir. Etmen yönelimli programlama paradigmasında, etmenler nesnelere daha karmaşık yapılar olarak kurgulanmıştır.

Etmenler, kullanıcıları adına belirli görevleri yerine getirebilecek özerk yazılım bileşenleri olarak kabul edilir ve yapay zeka kökenli bileşenlerdir. Algılayıcılar aracılığı ile çevre etkileşimi yapabilme yetenekleri vardır (Russell and Norvig, 2013). Akıllı etmenlerin geliştirilmesi çalışmaları hem yapay zeka hem de yazılım mühendisliği alanları kapsamındadır. Bir etmen, inançlardan, seçimlerden, yeteneklerden vb. yapılardan oluşan bir duruma sahiptir. Ayrıca etmenler bulunduğu çevre içerisinde kendisine atanan görevleri yerine getirebilmek için esnek, özerk eylemlerde bulunabilen bilgisayar sistemleri olarak da tanımlanabilmektedir (Woolridge, 2002; Badica et al., 2011). Özerk, karşıt-eylemler ve proaktif yapılarına ek olarak etmenler sosyal yeteneklere de sahiptirler. Problemleri çözebilmek adına diğer etmen ve insanlarla etkileşimde bulunabilirler. İşbirlikçi bir yaklaşımla genel problemlerin çözümü için diğer etmenlerle ortak çalışabilirler.

Problemlerin çözümünde etmenlerin sosyal yeteneklerinden faydalanmak ve birbirleri ile etkileşimde bulunmalarını sağlamak için akıllı etmenler Çok Etmenli Sistemler (MAS) olarak gruplandırılır. Bir MAS'ta etmenler bilgisayar sistemleri aracılığı ile mesajlaşarak etkileşime girerler. Bu etkileşimler işbirlikçi ya da bencil bir yapıda olabilir (Sycara, 1998). MAS yaklaşımı karmaşık dağıtık sistemlerin

modellenmesi için hem kullanışlı hem de uygulanabilir soyutlamalar ve teknolojiler sunmaktadır.

MAS yazılımlarının tasarımını ve gerçekleştirmesini kolaylaştırmak için tanımlanmış çeşitli AOSE metodolojileri vardır. Tropos (Giunchiglia et al., 2002), Passi (Henderson-Sellers et al., 2006), Prometheus (Padgham and Winikoff, 2003), Gaia (Woolridge et al., 2000), Adelfe (Bernon et al., 2003) bu MAS metodolojilerinden bazılarıdır. Ayrıca bu metodolojilerden hareketle geliştiricilerin MAS modellemesini desteklemek için IDK (Pavon et al., 2005), MAS-ML (Goncalves et al., 2015), JADEL (Bergenti, 2014), SEA\_ML (Challenger et al., 2014) gibi çeşitli MAS DSML'leri de sunulmaktadır.

## **2.2 Model GÜdümlü Mühendislik ve Alana-özgü Modelleme Dilleri**

Yazılım sistemlerinin zamanla büyük ve karmaşık bir yapı kazanması geleneksel yöntemler ile bu sistemleri tasarlamayı ve geliştirmeyi zorlaştırmaktadır. Bu durum yeni alan mühendislikleri ve yazılım geliştirme tekniklerine ihtiyaç duyulmasına sebep olmuştur. Bu karmaşıklıkla baş etmenin olası yollarından biri modelleri kullanarak soyutlama düzeyini arttıran Model GÜdümlü Geliştirme (Model Driven Development - MDD)'dir. Bu yaklaşımla yazılım geliştirme süreçlerinde ana odağın kodlamadan modellemeye evrilmesi amaçlanmaktadır. Bu doğrultuda alan üstmodellerine, sistem modellerine, modelden modele dönüşümlere ve yazılım kodlarının modelden otomatik üretimine ihtiyaç duyulmaktadır.

MDD'nin somutlaştırılmasını sağlayan Model GÜdümlü Mimari (Model Driven Architecture - MDA) (Frankel, 2003) aynı zamanda MDD üretkenliğini arttırmakta (Kos et al., 2011) ve geliştirme giderlerini azaltmaktadır (Vallecillo, 2008). Modelleme, doğası gereği daha üst düzey bir soyutlama sağladığı için yazılım geliştirme süreçlerinde kodlara dayalı işlemlerden daha az karmaşıklık içermektedir. Bir başka deyişle MDD yazılım geliştiricilerin probleme odaklanmalarını sağlamak için daha üst düzey bir soyutlama sağlamaktadır (Schmidt, 2006).

MDD yaklaşımının temel kavramları OMG (Object Management Group - OMG) MDA standardına göre üstmodeller, modeller ve dönüşümler olarak açıklanabilir. Üstmodeller, kavramları ve bu kavramlar arasındaki ilişkileri (örneğin; birleşme, kalıtım vb.) belirli bir alan için tanımlamaktadır. Modeller ise üstmodellerin tanımlamalarını benimseyen üstmodel örnekleri olarak

açıklanabilir. Dönüşümler, üstmodeli benimseyen bir kaynak modeli, bir diğer üstmodele dayalı hedef modele dönüştürmeyi sağlamaktadır. OMG MDA standardına göre modelden modele ve modelden koda olmak üzere 2 temel dönüşüm bulunmaktadır.

MDD amaçlarını gerçekleştirmek için farklı yaklaşımlar kullanır. Bu yöntemlerden biri Alana-özgü Modelleme Dilleridir. Bu dillerin kullanımı sırasında modeller hem görsel bir biçimde oluşturulduğundan hem de bu modeller yazılım kodları yerine ana yapıları temsil ettiğinden çeşitli uygulama alanları için yazılım geliştirme kolaylaşmaktadır. Ayrıca sunulan grafiksel sözdizimle tasarımda anlatımlılık ve kullanım kolaylığı sağlanmaktadır (Schmidt, 2006; Gray et al., 2007). Bir DSML'in geliştirilmesi genellikle dil modelinin tanımlanmasıyla başlar (Strembeck and Zdun, 2009). Bir başka deyişle, alana ait kavramlar ve soyutlamaların hedef alanı (dil modeli) yansıtması için tanımlanmaları gerekmektedir. Sonrasında ise, dil kavramları arasındaki ilişkilerin de tanımlanması gerekmektedir. Her ikisi de modelleme dilinin soyut bir sözdizimini oluşturur. Genellikle, bir dil modeli üstmodelle tanımlanır. Alan soyutlamaları ve ilişkileri, grafiksel somut sözdizimde sunuma gereksinim duyar ve bu somut sözdizim genellikle son kullanıcı için bir modelleme aracı ile desteklenmektedir.

### 2.3 FAML Üstmodeli

Önceki bölümde de açıklandığı gibi, yazılım mühendisliğinde MDD yaklaşımı modellemenin sağladığı soyutlamalar ile karmaşıklık problemini çözmeyi hedeflemektedir. Bu doğrultuda etmen yazılımlarının geliştirilmesi için literatürde bir çok DSML önerilmiştir. Yazılım mühendisliğinde herhangi bir araştırma alanı ile ilgili çeşitliliğe sahip olunması başlangıçta avantajlı bir durum olarak görülse de bir süre sonra çeşitlilik araştırma grupları arasında bilgi alışverişi ve sonuçların paylaşılması açısından engel teşkil etmektedir. Etmen DSML'leri özelinde de bu soruna çözüm olarak Beydoun ve arkadaşlarının yaptığı çalışma ile FAML olarak adlandırılan genel bir üstmodel sunulmuştur (Beydoun et al., 2009).

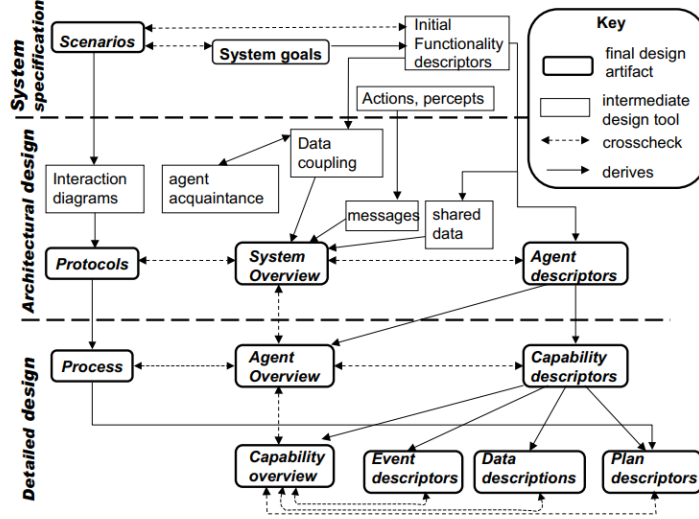
FAML üstmodeli, mevcut üstmodellerin birleştirilmesi ve genel bir üstmodelin ortaya konulması amacını taşımaktadır. Aynı grubun daha önce gerçekleştirdiği etmen odaklı modelleme dili olan FAME (Framework for Agent-Oriented Method Engineering) çerçevesini (Beydoun et al., 2006) temel almaktadır. Farklı DSML'lerin üstmodellerini inceleyerek ve kapsamlı bir doğrulama süreci ile FAML üstmodeli ortaya konulmuştur.

FAML üstmodeli Tasarım Zamanı (ing. design time) ve Çalışma Zamanı (ing. run time) olmak üzere 2 katman ve her bir katmanın sahip olduğu etmen-harici (ing. agent external) ve etmen-dahili (ing. agent-internal) olmak üzere 2 kapsamdan oluşmaktadır. İlk olarak, AOSE prensiplerinden hareketle bir başlangıç üstmodeli oluşturulmuştur. Ardından Adelfe, Passi, Gaia, Ingenias ve Tropos gibi etmen tabanlı metodolojilerle kıyaslama ve doğrulama çalışmaları yapılarak üstmodelin nihai versiyonu sunulmuştur (Beydoun et al., 2009). Bu üstmodel tez ürünü AgentDSM-Eval aracında etmen DSML'lerinin söz dizimlerinin etmen kavramlarını ne ölçüde karşıladıklarını belirleme sırasında referans model olarak kullanılmıştır. FAML üstmodelinde tasarım zamanı için 26, çalışma zamanı için 21 kavram tanımlanmaktadır. Bu kavramlar ve tanımlamaları tezin Değerlendirme Çerçevesi bölümünde yer alan Tablo 4.1 ve Tablo 4.2'de görülmektedir.

## 2.4 Prometheus/PDT

Tezde geliştirilen AgentDSM-Eval aracının sunduğu özelliklerin değerlendirilmesi bu aracın AOSE'de iyi bilinen Prometheus/PDT metodolojisi ve modelleme dilinin değerlendirilmesinde kullanılması üzerinden gerçekleştirildiğinden bu bölümde Prometheus/PDT hakkında kısaca bilgi verilmektedir. Prometheus AOSE metodolojisi (Padgham and Winikoff, 2005) bir etmen sisteminin analiz ve tasarım aşamalarının tüm ihtiyacını karşılayarak süreci basitleştirmeyi hedeflemektedir. Detaylı bir analiz ve tasarım imkanı sunan bu metodolojide etmenlerin tasarımı hedef ve plan odaklı olarak gerçekleştirilmektedir. Herhangi bir MAS için ihtiyaç analizinden tasarım aşamasına kadar olan süreci içeren bu metodoloji aynı zamanda "Prometheus Design Tool" (PDT) isimli bir yazılım aracını da etmen geliştiricilerine sunmaktadır.

Prometheus metodolojisi 3 temel süreç üzerine kuruludur. Bunlar sistem tanımlama, mimari tasarım ve detaylı tasarım süreçleridir. Şekil 2.1'de bu süreçler ve alt süreçleri görülmektedir. Sistem Tanımlama süreci genel olarak sistemi açıklayan bir süreç olarak özetlenebilir. Sistemin hedefi, kullanım senaryoları ve fonksiyonelliği bu süreçte şekillenir. Etmenlerin proaktif ve hedef yönelimli olmalarından hareketle sürecin hedefleri belirleyerek başlaması öngörülmüştür. Belirlenen her hedef alt hedeflerin çıkarılmasını ve kullanım durumlarının saptanmasını kolaylaştırmaktadır. Hedefleri belirledikçe hedefler arasında grupların oluşması sağlanmakta ve bu gruplar üzerinden ise sistemin fonksiyonallikleri belirlenmektedir.



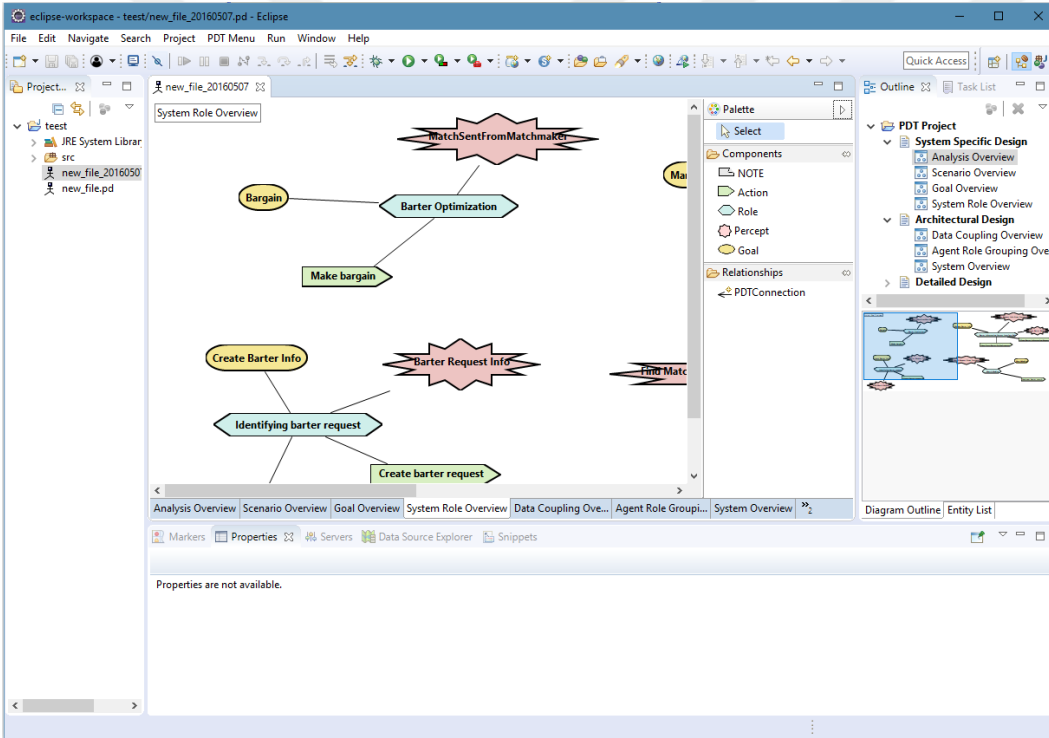
Şekil 2.1: Prometheus Metadolojisi Süreçleri ((Padgham and Winikoff, 2005)'dan alınmıştır.)

Sistem özelliklerini belirleme süreci tamamlandıktan sonra mimari tasarım sürecine geçilmektedir. Bu süreçte temel odak noktaları sistemdeki etmen tiplerine karar vermek, etmenler arası iletişim protokollerini netleştirmek ve genel sistem yapısını belirlemek olarak özetlenebilir. Sistemde bulunacak etmen tiplerine karar vermek belki de bu süreçte yapılan en önemli işlem olarak düşünülebilir. Prometheus'ta bir veya daha fazla fonksiyonlute birleştirilerek bir etmen türü oluşturulur. Elde edilen etmen tipleri bağlantı diyagramı (ing. coupling diagram) ve etmen tanışıklık diyagramı (ing. agent acquaintance diagram) kullanılarak modellenir. Ayrıca bu süreçte etkileşim diyagramı ve etkileşim protokolleri kullanılarak iki etmen arası iletişimin özellikleri de netleştirilmektedir. Son olarak sistem yapısını netleştirmek için genel sistem yapısı diyagramı çıkarılır. Bu diyagram, sistemdeki etmen tiplerini, sistemin sınırlarını ve arayüzlerini sadece eylem ve algılamalar açısından değil, aynı zamanda sistemin dışındaki veri ve kod açısından da belirlemektedir.

Prometheus metodolojisinin son aşaması ise detaylı analiz aşamasıdır. Bu süreçte yetenekleri açısından etmenlerin iç tasarımları, etkileşim protokolleri için süreçler ve olaylar, planlar ve veriler açısından yeteneklerin ayrıntıları belirlenir. Etmenlerin iç mimarileri belirlenirken etmen genel görünüm diyagramları ve yetenek tanımlayıcı diyagramlar kullanılır. Yeteneklerin ayrıntılandırılması işleminde ise yeteneğe genel bakış diyagramları ve çeşitli tanımlayıcılar kullanılır. Anahtar odak, hedeflere ulaşmak ve uygun kapsamı sağlamak için plan kümelerini geliştirmektir.

Prometheus metodolojisi ile etmen yazılımlarının geliştirilmesini destekleyen PDT aracı her alt süreç için çeşitli etmen modellerinin oluşturulabildiği bir etmen DSML'i olarak kabul edilebilir. PDT ile geliştirilen bu görsel modellerden kod üretimi yapılabilmektedir. Kod üretimi sonucu etmen geliştirme platformlarından biri olan JACK (Howden et al., 2001) için kod üretimi gerçekleştirilmektedir.

PDT aracı Java programlama dili ile hazırlanmış ve Eclipse entegre geliştirme ortamına (Eclipse, 2019) bir plug-in olarak kurulabilecek şekilde sunulmaktadır. Aracın arayüzü Şekil 2.2'de görülmektedir. Hedefler, planlar, algılamalar, eylemler ve protokoller gibi belirli etmen varlıklarını modelleyebilecek bir ortam sağlamaktadır.



Şekil 2.2: PDT Aracı Arayüzü ((Padgham and Winikoff, 2005)'den alınmıştır.)

## 2.5 Araçlar ve Teknolojiler

Tez çalışması kapsamında geliştirilen web tabanlı AgentDSM-Eval aracının hazırlanması için Google Firebase (Firebase, 2019) sunucusuz arka uç teknolojisi ve VueJs (VueJS, 2019) ön yüz teknolojisi kullanılmıştır.

Google Firebase, herhangi bir sunucu yönetimi ve organizasyonu gereksiz olan arka uç ihtiyaçlarını sağlayan hizmetler içeren bir teknolojidir. Bu



ürün kapsamında “Cloud Firestore NoSQL” veritabanı, “Cloud Storage” döküman depolama altyapısı ve dinamik ön-yüz projelerinin dağıtılabildiği “Firebase Hosting” hizmeti sağlamaktadır.

AgentDSM-Eval web uygulamasının ön yüzü VueJS Javascript kütüphanesi ile gerçekleştirilmiş, ayrıca Apex ChartJS kütüphanesinden de verilerin görselleştirilmesi sürecinde yararlanılmıştır. VueJS bileşenler (ing. component) üzerine kurulu bir yapı sunmaktadır. Bileşenler arasında javascript yapıları ile iletişim sağlanarak basit ve hızlı geliştirme yapılmasına zemin hazırlamaktadır. Ayrıca farklı kütüphanelerin entegre edilmesi ile geliştirme sürecini hızlandırılmaktadır. Örneğin AgentDSM-Eval web uygulamasında verilerin farklı grafiklerle görsel olarak sunulması için Apex ChartJS kütüphanesi VueJS projesine eklenmiştir. Apex ChartJS, modern ve interaktif grafiklerin sunulduğu bir javascript kütüphanesidir. Çizgi, alan, kolon, pasta grafik gibi çeşitli grafiklerin gerçekleştirimini sağlamaktadır.

### 3. İLGİLİ ÇALIŞMALAR

Bu tez konusu ile ilgili literatürdeki mevcut çalışmalar aşağıdaki iki alt başlıkta verilmiştir. İlk olarak AOSE'deki MAS üstmodelleme ve MDD çalışmaları incelenmiştir. Daha sonra etmenlerin geliştirilmesi için önerilen çeşitli alana özgü diller anlatılmıştır.

#### 3.1 MAS Üstmodelleme ve Model Güdümlü Geliştirme Çalışmaları

MAS'lar için literatürde yer alan genel amaçlı ilk üstmodelin AALAADIN (Ferber and Gutknecht, 1998) olduğu söylenebilir. Aynı çalışmada bu üstmodel kavramlarının "MadKit" adı verilen platform üzerinde gerçekleştirimi yerine getirilmiştir. Öte yandan bazı araştırmacılar MAS geliştirme metodolojilerine özel bazı etmen üstmodellerini tanımlamayı tercih etmiştir. Bunlardan Bernon et al. (2005), ADELFE (Bernon et al., 2003), Gaia (Zambonelli et al., 2003) ve PASSI (Cossentino and Potts, 2002) MAS geliştirme metodolojileri için üstmodelleri verdikten sonra her metodolojiye ait önemli bileşenleri ve özellikleri birleştiren yeni bir üstmodeli tanıtmışlardır. Benzer bir çalışma SODA metodolojisi (Omicini, 2001) için (Molesini et al., 2005)'te yerine getirilmiştir. Bu çalışmada SODA metodolojisinin etkileşim ve sosyal bakış açılarının modellenmesi ve bunlara ait bir üstmodelin çıkarılması hedeflenmiştir. Sunulan üstmodeller metodolojilerin genişletilebilmesi ve iyi bir model analizi için önemlidir. Çünkü bu çalışmalar ilgili metodolojilerin kavramlarını ve ilişkilerini formal bir şekilde temsil etmektedir. Üstelik Pavon et al. (2006) bu üstmodellerin farklı etmen geliştirme metodolojilerini karşılaştırmak için kullanılabildiklerini de vurgulamaktadır. Bununla beraber bu tip çalışmalar genel MAS modellemesine uygun değildir. Çünkü belli bir metodolojiye ait kavramların kullanılmasına zorlarlar ve sadece o metodolojiye özgü çerçevelere kod üretilmesini gerçekleştirirler. Ayrıca bu çalışmaların birçoğu üstmodellerin bir MDD içerisinde nasıl kullanılabileceğine dair bir yaklaşımı da barındırmamaktadırlar (Kardas, 2013).

Bir diğer önemli MAS üstmodeli (Hahn et al., 2009)'da tanıtılmıştır. Bu üstmodelde, etmen modelleme kavramları yedi MAS bakış açısında toplanmıştır. Bu bakış açıları, MAS, etmen, davranışsal, organizasyon, rol, etkileşim ve çevre olarak tanımlanmıştır. Benzer şekilde (Kardas et al., 2009) etmen sistemlerini özellikle Anlamsal Web ortamında geliştirmek ve çalıştırılabilir MAS'lar ortaya çıkartmak için bir etmen üstmodeli tanımlamışlardır. Omicini et al. (2008) ise etmenler için "artifact" kavramını göz önüne alan Etmenler ve Yapılar (Agents

& Artifacts - A&A) üstmodelini tanıtmıştır. A&A üstmodelinde etmenler bir sistemin amaçları ve görevleri için pro-aktif varlıklar olarak modellenirken yapılar servis ve fonksiyonları sunan ve böylece ilgili MAS için ortamı oluşturan karşıt-eylemli varlıklar şeklinde temsil edilmektedirler. Öte yandan, bir önceki altyapı bölümünde de üzerinde durulan FAML üstmodeli (Beydoun et al., 2009) etmen sistemleri için geliştirilmiş çeşitli üstmodellerin bir sentezi sayılabilir. Bu çalışmada tasarım ve çalışma zamanı aşamaları belirlenmiş ve bu aşamalarda tanımlanan kavramların doğrulaması MAS geliştirme metodolojileriyle sağlanmıştır.

### 3.2 MAS DSL/DSML Çalışmaları

Bir önceki bölümde sözü edilen üstmodelleme çalışmalarından yola çıkılarak son yıllarda MAS geliştirme için literatürde çeşitli DSML çalışmalarına yoğunlaşıldığı görülmektedir. Bunlara ilk örnek olarak yine bir önceki Altyapı bölümünde ayrıntılı olarak incelenen Prometheus/PDT (Padgham and Winikoff, 2005) MAS DSML'i verilebilir. Prometheus metodolojisini temel alan bu modelleme dili, PDT isimli bir araçla MAS modelleme sürecini desteklemektedir.

INGENIAS (Pavon et al., 2005) MAS'ların modellenmesi, analizi, tasarımı, doğrulanması ve kod üretimi için çeşitli faaliyetlerin tanımlandığı bir AOSE metodolojisi olsa da sonradan bu metodoloji MDD'yi destekleyecek bir biçimde revize edilmiş (Pavon et al. 2006) ve etmenlerin görsel olarak geliştirilmesi için INGENIAS Geliştirme Kiti (INGENIAS Development Kit - IDK) ortaya konulmuştur. IDK ile etmenlerin MAS bakışaçıları ile görsel modellenmesi ve bu modellerden sistem implementasyonları şablonlarının otomatik elde edilmesi mümkün olduğundan sunulan ortam bir DSML'in uygulaması olarak kabul edilebilir. Nesneye yönelik programlama için bir geliştiricinin sınıflar, arayüzler, nesnelere ve benzeri kavramlarla çalışması gerekmektedir. Benzer şekilde bir MAS üzerinde çalışan bir geliştirici için de bu kavramlara ek olarak etmen, organizasyon amaç, görev, kaynak gibi etmen odaklı yazılım kavramları önem taşımaktadır. INGENIAS tüm bu kavramların analiz aşamasından uygulama aşamasına kadar kullanılabilirliğini arttırmayı amaçlamaktadır. Etmen tabanlı sistem geliştirme döngüsünde tüm bu kavramların kullanımı doğal olarak karmaşıklığa sebep olmaktadır ve bunu yönetmek için INGENIAS 5 bakışaçısı altında tüm bu kavramları organize etmiştir. Bu bakışaçıları, Organizasyon, Etmen, Hedefler/Görevler, Etkileşim ve Ortam olarak belirlenmiştir (Pavon et al. 2005).

Alman Yapay Zeka Araştırma Merkezi'nde (DFKI) görev alan bir grup etmen araştırmacısının bütünleşik bir MAS DSML'ini tanımladığı görülmektedir. DSML4MAS adı verilen bu dil ilk olarak (Hahn, 2008)'de tanıtılmıştır. DSML'e ait somut sözdizim ve formal anlam ise sırasıyla (Warwas and Hahn, 2008) ve (Hahn and Fischer, 2009)'da yer almaktadır. Ortaya konan bu DSML'e ait soyut sözdizim bir MAS'ın farklı bakışaçılarını (örneğin MAS, etmen, organizasyon ve rol gibi) dikkate alan ve genel etmen üstvarlıklarını içeren platform bağımsız bir üstmodelden türetilmiştir. Somut sözdizimi ortaya koymak için alan kavramlarının grafiksel notasyonu ve bunun soyut sözdizim elemanları ile olan bağlantıları verilmiştir.

Bir diğer ilgili çalışma olarak ise MAS-ML modelleme dili göz önüne alınabilir. MAS-ML modelleme dili (Goncalves et al., 2015) "Taming Agents and Objects" (TAO) (Silva et al., 2002) kavramlarını modellemek için tasarlanmış bir modelleme dilidir. Önceden tanımlanmış planlar tarafından yönlendirilen hedef odaklı proaktif etmenlerin modellenmesi bu modelleme dilinin temel hedefi olarak özetlenebilir. MAS-ML tarafından tanımlanan yapısal diyagramlar Rol Diyagramı, Sınıf Diyagramı ve Organizasyon Şeması'dır. Dinamik diyagramlar ise Dizi Diyagramı ve Etkinlik Diyagramıdır. Tüm bu diyagramları kullanarak, TAO'da tanımlanan varlıkların yapısal ve dinamik yönlerini modellemek mümkündür.

TAO çerçevesinin daha sonraki çalışmalarda yeni kavram tanımlamaları ile kapsamı genişletilmiştir. Bu duruma paralel olarak da MAS-ML modelleme dili de MAS-ML 2.0 adıyla bu yeni çerçeveye uyum sağlamaktadır. TAO konsepti bir etmeni özerk, uyarlanabilir ve interaktif olarak tanımlamaktadır. Bu etmenler hedef ve inanç gibi yapısal özelliklere; eylem ve plan gibi davranışsal özelliklere sahip olmaktadır. TAO çerçevesinin genişletilmiş yeni versiyonunda ise etmenlerin iç mimarilerini de özelleştirebilen yeni kavramlar tanımlanmıştır. Bu doğrultuda TAO çerçevesi için yeni etmen tanımı; özerk, uyarlanabilir ve interaktif olan; davranışsal ve yapısal özellikleri iç mimarisi tarafından daha önceden tanımlanmış unsurlardır. Tanımlanan bu iç mimariler incelendiğinde aslında yeni sürüm TAO ve bunu üstmodel alan MAS-ML dilinin daha çok doğrudan Russell and Norvig (2013)'in etmenler için tanımladığı mimarileri desteklemek için ortaya konulduğunu göstermektedir.

MAS-ML 2.0 Eclipse entegre geliştirme platformu eklentisi olarak bir modelleme aracı sunmaktadır. Model geliştiricileri için kolay bir kullanıcı

arayüzüne sahiptir. TAO çerçevesine göre kavramların modellendiği bir ortam sunan modelleme aracı (Goncalves et al., 2015)'den öğrenildiği kadarıyla MAS modellerinden otomatik kod dönüşümünü henüz tam olarak gerçekleştirememektedir.

Kanı-İstek-Hedef (Belief-Desire-Intention - BDI) mimarisine dayanan yazılım etmenlerinin AOSE'de sık kullanılan JADE platformu (Bellifemine et al., 2007) üzerinde geliştirilmesi için BDI4JADE adlı bir çerçeve önerilmiştir (Faccin and Nunes, 2015). Bu çerçevede tasarım faaliyetleri temelde öğrenme tabanlı bir plan seçimi mantığının modellenmesine dayanmaktadır. BDI4JADE'de modelgüdümlü etmen yazılımı geliştirmek için "Sam" isimli bir araç tanıtılmıştır (Faccin and Nunes, 2017). Aracın MAS geliştirilmesi için görsel bir sözdizim sunduğu ve otomatik kod dönüşümü sağladığı görülmektedir.

BDI4JADE'de bir etmeni modellemek için önerilen grafiksel gösterimi kullanmak, iki ana çıktı oluşmasını sağlamaktadır. Birincisi, modellenmiş etmeni tasvir etmek için kullanılan etmen diyagramıdır. İkincisi ise, öğrenme tabanlı plan seçme tekniğiyle ilgili etmene özgü tercih değerleri ve özellikler gibi belirli bilgileri içeren bir etmen modeli dosyasıdır. Bu çıktıların her biri tek bir etmeni tanımlar. Farklı etmenleri modellemek, farklı etmen diyagramları ve model dosyalarına sahip olmak anlamına gelmektedir.

Sam aracı, Graphiti çerçevesine (Eclipse, 2013) ve Xpand şablon diline (Eclipse, 2014) dayanan bir Eclipse eklentisi olarak geliştirilmiştir. Sam ile önerilen gösterimleri kullanarak etmen modellerinin örneklenmesini destekleyen grafiksel bir düzenleyici ve modelden metne dönüşüm sürecini otomatikleştiren bir kod üretme özelliği sunulmaktadır.

JADEL, JADE'in (Bellifemine et al., 2007) Java sınıfları açısından sağladığı soyutlamalar için etkili somut bir sözdizim sunmak amacıyla geliştirilmiş bir dildir. Etmenler, davranışlar ve iletişim ontolojileri; JADEL'in desteklediği ana soyutlamalardır (Bergenti, 2014). JADE tarafından sağlanan soyutlamaların sadece küçük bir kümesi JADEL tarafından desteklenmektedir. Çünkü JADEL geliştiricinin yüksek bir soyutlama seviyesinden projenin temel prensiplerini basitçe belirlemesini hedeflemektedir. Örneğin, JADEL, geliştiricinin, IEEE FIPA "Agent Communication Language" (ACL) (FIPA, 2002) mesajlarını göndermek ve almak için kullanılacak yeni bir "codec" bileşeni uygulamasına izin vermez. Geliştirici mevcut "codec" setlerinden bir mesaj

“codec”i seçebilir. Bununla birlikte gerçekleştirilen incelemelerde JADEL’in küçük bir soyutlama kümesiyle etmen sistemi düzeyinde özetlenebilir ve nispeten tutarlı bir görünüm sunduğu görülmüştür. JADEL’in kapsamı dışında kalan bir diğer durum ise etmenlerin ve MAS’ın çalıştırılabilir hale getirilmesidir. Geliştiricinin JADEL soyutlamaları sonucu elde ettiği MAS görünümünü JADE üzerinde komut dosyaları aracılığıyla etkinleştirmesi gerekmektedir. Özet olarak JADEL’in JADE platformu üzerinde uygulama geliştirmeye yarayan bir DSL olduğu söylenebilir.

JADEL incelendiği kadarıyla geliştiricinin hem sistemin tasarımında hem de uygulamasında doğru etmen tabanlı soyutlamaları kullanabilmesini sağlamayı amaçlamaktadır. Aynı zamanda, etmen temelli modelleme ve uygulama arasındaki boşluğun azaltılmasını ve sonuçların genel kalitesinin artırılmasını hedeflemektedir. JADEL DSL’ine ait MAS geliştirme aracı Xtext (Eclipse, 2010) çerçevesi üzerine kuruludur (Bergenti et al. 2017). Xtext kullanımı JADEL dilbilgisinin oluşturulması ve derleyicinin uygulanması gibi aşamalarda kolaylıklar sağlamıştır. Dili geliştiren grubun yaptığı çalışmalar sonucu JADEL şu anda kaynak kodlarını, anlamsal olarak eşdeğer Java kaynak kodlarına çeviren ve bu kodların JADE üzerinde kullanılmasını sağlayan bir derleyici tarafından desteklenmektedir. Bu derleyici, Java kaynak kodunun efektif üretilmesini desteklemek için Xtext altyapısını kullanmaktadır. Nihai çalışma sonucu JADEL’in JADE eklentisi olarak sunulması planlanmaktadır.

SEA\_ML (Challenger et al., 2014), geliştiricilerin etmen sistemlerini platformdan bağımsız bir düzeyde modellemesini ve ardından hedef MAS uygulama platformlarında modellenmiş MAS'nin yürütülmesi için gerekli olan kodları ve ilgili belgeleri otomatik olarak elde etmesini sağlayan bir MAS modelleme dilidir. Etmen geliştiricilerin çeşitli uygulama alanlarında çalışan etmen sistemlerini inşa etmesi ve geliştirmesi için kullanışlı ve uygun bir ortam sağlar. MAS uzmanlarını geliştirme esnasında modellerde görsel olarak ince ayar yapabileceği şekilde destekleyen SEA\_ML, bir etmen sisteminin tüm yönlerini tek bir etmenin iç görünümünden karmaşık MAS organizasyonuna kadar kapsamaktadır. Bu özelliklere ek olarak, SEA\_ML, semantik veriyi değerlendirebilen ve Anlamsal Web'in (ing. semantic web) (Shadbolt et al., 2006) anlamsal web servisleri (Semantic Web Service - SWS) (Sycara et al., 2003) gibi tanımlanmış varlıklarıyla işbirliği yapabilen özerk etmenlerin model tabanlı tasarımını ve geliştirilmesini de desteklemektedir. Bu özellik SEA\_ML’i mevcut MAS DSML’lerden farklı kılmaktadır. Bu bağlamda, SEA\_ML, etmen

üstmodelleri için etmen, plan, etkileşim, vb. bilinen bakış açılarına Anlamsal Web'i gözeterek yeni bir bakış açısı kazandırmaktadır. SEA\_ML'de etmen, etmen bilgi tabanları, platform ontolojileri, SWS ve etmenler ile SWS arasındaki etkileşimleri modellemek mümkündür.

SEA\_ML, hem MAS modelleme hem de oluşturulmuş modellerden kod üretimi için kullanılabilir. SEA\_ML örnekleri, JADDEX (Pokahr et al., 2005) etmen platformu için inşa edilecek sistemin çalıştırılabilir yazılım kodlarını ve etmenlerin etkileşimde olacağı servislerin Web Ontoloji Dili (Web Ontology Language – OWL) (W3C, 2004) ile uyumlu SWS tanımlama dokümanlarını elde etmek için bir dizi M2M (Model-to-Model) ve M2T (Model-to-Text) dönüşümünün girdisi olarak verilir. SEA\_ML modellerinin bütünlüğünü ve geçerliliğini otomatik olarak kontrol etmek de mümkündür (Getir et al., 2014).

SEA\_ML dili ve bu dil ile modelleme yapılmasını sağlayan IDE açık kaynaklı olarak sunulmaktadır. SEA\_ML IDE'de SEA\_ML'in sunduğu bakış açılarının her birinde etmen modelleme imkanı bulunmaktadır.

Acil durum ortamlarına (Emergency Response Environment - ERE) müdahale sistemlerinin etmen tabanlı modellenmesi için (HoseinDoost et al., 2017)'de ERE-ML isimli bir DSML önerilmiştir. ERE-ML dili bu bölümde incelenen MAS-ML dili (Goncalves et al., 2015) ile TAO çerçevesinin genişletilmesi sonucu ortaya konan bir DSML'dir. Bu genişletme işlemi için afet yönetiminin dört aşamasını kapsayan DMM (Disaster Management Metamodel) üstmodeli (Othman et al., 2014) kullanılmıştır. Afet yönetiminin 4 aşaması hafifletme (ing. mitigation), hazırlıklı olma (ing. preparedness), müdahale etme (ing. response) ve kurtarmadır (ing. recovery). Önerilen bu DSML'de, ERE modellemesi için görev konsepti eklenmiş ve afet kavramı, hafifletme aşamasından itibaren baz alınmıştır. İlk olarak TAO çerçevesine acil durumlara uygun olacak şekilde görev ve afet kavramları eklenmiştir. Ayrıca bazı ilişkilerle TAO çerçevesi genişletilmiştir (yaratma, bölme, atama, azaltma gibi). Aynı şekilde TAO çerçevesini baz alan MAS-ML dilinde de bu doğrultuda değişiklikler meydana gelmiştir. Her yeni kavram için MAS-ML soyut ve somut sözdiziminde yeni tanımlamalar yapılmış ve ardından dilin anlamsal modeline bu kavramlar eklenmiştir.

EMT, ERE-ML DSML'i kullanılarak MAS modellemesi yapmak üzere geliştirilen bir araçtır. NorMAS-ML (Freire et al. 2013) aracının bir uzantısı

olarak geliştirilen bu araç ERE-ML üstmodeline göre model doğrulaması yapabilmektedir. Doğrulama için kullanılan kısıtlar OMG Nesne Kısıt Dili'nde (Object Constraint Language - OCL) yazılmıştır. Ayrıca yazılan dönüşüm şablonları sayesinde modelden metne dönüşüm de gerçekleştirilmektedir. Buna ek olarak oluşturulan kodu yürütmek için bir etmen platformu ERE-ML dağıtımı içerisinde yer almaktadır. Bu platform ERE kavramlarının eklenmesi ile elde edilen genişletilmiş bir JAMDER (Lopes et al., 2011) versiyonudur.

Yakın zamanda BDI etmen mimarisine göre kurgulanacak etmenlerin MDD'si için DSML4BDI adı verilen bir dil (Kardas et al., 2018) geliştirilmiştir. Dil aynı zamanda etmenlerin mantığa dayalı olarak programlamasını sağlayan AgentSpeak (Rao, 1996) tanımlamalarını modelleme ile oluşturabilmekte ve AgentSpeak için bir yorumlayıcı olan Jason platformu (Bordini et al., 2007) için otomatik kod üreten bir işletimsel semantik sunmaktadır. Son olarak Sredevic et al. (2018) Siebog adını verdikleri dağıtık MAS ortamında çalışacak akıllı etmenlerin geliştirilmesi için bir DSL tanıtmaktadırlar. ALAS isimli bu dili kullanarak yazılım geliştiriciler Siebog platformuna özel kodları hazırlamakta; bu kodlar tanımlanan bir dönüştürücü üzerinden ilgili platformda etmenlerin çalıştırılması için gerekli Java kodlarına dönüştürülmektedir.

Yukarıda değinilen tüm MAS DSML çalışmalarının özellikle etmen üstmodellemeye, model dönüşümlerine ve araç desteğine yoğunlaştığı ve bu açılardan MAS'ların model güdümlü geliştirilmesine önemli katkıları bulunduğu söylenebilir. Ancak bu çalışmaların (Faccin and Nunes, 2017) ve (Kardas et al., 2018) dışında hiçbirisi sundukları DSML'lerin özelliklerini ve kullanımını göz önüne alan herhangi bir değerlendirme içermemektedir. Çoğunlukla basit durum çalışmaları üzerinden sadece dilin ve destekleyen yazılım araçlarının kullanımı örneklenmektedir. Tek durum senaryosu üzerinden yapılan bazı değerlendirme gayretlerinin ise herhangi bir sistematik içermediği, iddia edilen kullanım kolaylığının diğer durum senaryolarında da tekrar edilebilirliğinin açık olmadığı, kullanıcı geribildirimlerini içeren deneysel (ing. empirical) bir yaklaşımın izlenmediği ve/veya dilin ve bu dili kullanmanın niceliksel / niteliksel herhangi bir ölçümünün yapılmadığı görülmektedir.

MAS DSML'lerinin değerlendirilmesi ile ilgili gerçekleştirilen ilk çalışmanın (Challenger et al., 2016) olduğu söylenebilir. Sunulan yaklaşım farklı boyutlar ve kriterler içeren bir değerlendirme çerçevesine dayanmaktadır. Çoklu durum çalışması prensibine göre dört farklı durum çalışması belirlenerek



uygulanmıştır. Bir grup durum çalışmalarını gerçekleştirmek için SEA\_ML'i araç olarak kullanırken, diğer grup genel amaçlı modelleme dilleri ile geliştirme yapmıştır. Değerlendirme çerçevesi, dil, çalıştırma ve nitelik isimli üç temel boyutu benimser. İlk iki boyut ve ölçütleri, değerlendirmenin niceliksel yönü olarak ele alırken, diğeri değerlendirmenin niteliksel yönüyle ilgilenir. Dil boyutu, her birinin değerlendirilen belirli ölçütleri kapsadığı dil elemanları ve model dönüşümlerinin alt boyutlarını içerir. Dil elemanları altboyutu, soyut sözdizim, somut sözdizim ve statik semantik ölçütlerini içerir. Model dönüşüm altboyutları sırasıyla, modelden modele dönüşümü ve modelden metne dönüşümü temsil etmektedir. Esas itibarıyla, değerlendirme çerçevesinin bu ölçütleri, değerlendirilen DSML'nin yapısını karakterize edip incelemeyi amaçlamıştır. Bu değerlendirme yaklaşımı sonrasında MAS DSML'lerinin birlikte işlevliğini göz önüne alan (Kardas et al., 2017) ve DSML4BDI dili için de (Kardas et al., 2018) izlenmiştir.

MAS DSML değerlendirme ile ilgili yakın zamanda karşılaşılan bir diğer çalışmada (Faccin and Nunes, 2017) Sam aracının kullanımı değerlendirilmiştir. Bu çalışmada Sam ampirik olarak araç kullanılabilirliği, kullanıcı memnuniyeti ve kullanım kolaylığı açısından değerlendirilmiştir. Gönüllülerden oluşan bir grubun aracı kullanımı sağlanmış ve gönüllülerden yararlılık, memnuniyet ve kullanım kolaylığı çerçevesinde hazırlanmış bir anketin cevaplanması istenmiştir. Ayrıca gönüllülerden, Sam aracını kullanırken yaşadıkları en olumsuz ve olumlu üç yönü içeren bir liste sunmaları talep edilmiştir.

Mevcut MAS DSML değerlendirme çalışmaları (Challenger et al., 2016; Kardas et al., 2017; Faccin and Nunes, 2017; Kardas et al., 2018) genellikle yine aynı çalışmayı yürütenlerin geliştirdikleri bir MAS DSML'ini değerlendirmeyi göz önüne almıştır ve bu çalışmalarda değerlendirmelerin yapılması için bir yazılım aracı desteği bulunmamaktadır. Bu tezde geliştirilen yazılım aracı hem değerlendirmenin farklı metriklerle niteliksel ve niceliksel değerlendirmesini sağlamakta hem de bir referans üstmodeline göre dilin sözdiziminin MAS'ları destekleme seviyesi yönünden incelenmesine klavuzluk etmektedir. Ayrıca web tabanlı bu araç kendi veri deposundaki mevcut bilgiler kullanılarak birden fazla MAS DSML'inin karşılaştırmalı değerlendirmesine de zemin hazırlamaktadır.

## 4. DEĞERLENDİRME ÇERÇEVESİ

Bu bölümde tezde geliştirilen web tabanlı AgentDSM-Eval aracının kullandığı MAS değerlendirme çerçevesi anlatılmaktadır.

### 4.1 Genel Bakış

MAS DSML'lerinin kullanım kolaylığı, araç zenginliği, alan hakimiyeti, etkinliği, etmen yazılım bileşenlerinin üretkenliği, vb. açılardan ve hem nitelik hem de nicelik yönünden değerlendirilmesini ve karşılaştırılmasını sağlayacak AgentDSM-Eval için Challenger et al.'ın (2016) sunduğu değerlendirme çerçevesi temel alınarak bir MAS DSML değerlendirme yaklaşımı oluşturulmuştur. AgentDSM-Eval'de değerlendirme süreci için Çoklu-durum Çalışması yaklaşımı benimsenmektedir. Sunulan kavramsal modele göre Değerlendirme Yaklaşımı (ing. Evaluation Approach) bir Değerlendirme Çerçevesini (ing. Evaluation Framework) kullanarak girdi olarak verilen MAS DSML'i değerlendirmektedir. Challenger et al.'ın (2016) değerlendirme çerçevesine ileriki bölümlerde ayrıntılandırılacak şekilde katkılar sunularak AgentDSM-Eval isimli yeni çerçeve ortaya konulmaktadır.

Challenger et al.'daki (2016) değerlendirme metrikleri tezde revize edildikten sonra ayrıca AgentDSM-Eval ile MAS DSML'lerinin niceliksel değerlendirilmesi aşamasına referans bir üstmodelle kıyaslama özelliği de eklenmiştir. Referans üstmodel olarak tezin Altyapı bölümünde tanıtılan ve MAS DSML'lerin üstmodellerini genelleştirmeyi hedefleyen FAML üstmodeli seçilmiştir. Ayrıca Challenger et al. (2016) niteliksel değerlendirme sürecini az sayıda açık uçlu soru ile değerlendirmeyi tercih etmiştir. Ancak metin bazlı cevaba dayanan ve az sayıda soru içeren bir değerlendirme MAS DSML sayısı arttıkça kapsamlı bir çıkarımda bulunmayı engellemektedir. Alana özgü dillerin niteliksel açıdan değerlendirilmesi ile ilgili (Wile, 2004), (Kahlaoui et al., 2008) ve (Kahraman and Bilgen, 2015) gibi çalışmalar literatürde göze çarpmaktadır. Çeşitli metriklerle, anket temelli bir DSL değerlendirme yaklaşımı içeren Kahraman ve Bilgen'in (2015) çalışması diğer çalışmalara göre gelişmişlik ve güncellik yönünden ön plana çıkmaktadır. Bu nedenle, tezde geliştirilen AgentDSM-Eval aracında niteliksel değerlendirme için Kahraman ve Bilgen'in (2015) Nitel Değerlendirme için Bir Çerçeve (Framework for Qualitative Assessment – FQAD) çalışmasında yer alan metrikler MAS DSML'ler için uyumlu hale getirilerek kullanılmıştır. Ayrıca FQAD'da varolan metriklere MAS

DSML'lere özgü yeni metrikler eklenmiştir. Böylece, AgentDSM-Eval aracı ile etmen odaklı MAS DSML'lerin sistematik değerlendirilmesi sağlanmakta; tüm bu MAS DSML önerileri ve yeni DSML'lerin niteliksel ve niceliksel açıdan değerlendirilmesi gerçekleştirilmektedir.

## 4.2 Çoklu Durum Çalışması

AgentDSM-Eval'de değerlendirme süreci olarak Challenger et al.'da (2016) tanıtilan Çoklu-durum Çalışması benimsenmiştir. Çoklu-durum Çalışması Hazırlık, Çalıştırma ve Analiz evrelerinden oluşan bir protokole uygun olarak işletilmektedir.

Hazırlık evresinde değerlendirme sürecinde kullanılacak olan MAS kullanımı durum çalışması senaryoları belirlenmektedir. Ayrıca değerlendirmeyi gerçekleştirecek olan MAS geliştiricileri grubu belirlenmektedir. Bu gruba ilgili MAS DSML eğitimi verilerek grubun yaklaşık olarak benzer bir bilgi seviyesine ulaştırılması sağlanmaktadır.

Çalıştırma evresinde, öncelikle, belirlenen durum çalışması senaryosunu ve AgentDSM-Eval için geliştirilen aracı tanıtmak için bir açılış bilgilendirmesi yapılır. Ardından değerlendirici ekip durum senaryosunun gereksinimlerini karşılayan sistemi geliştirirler. Geliştirme sonrasında tüm değerlendiricilerin Agent-DSM Eval web aracını kullanarak veri girişi yapması sağlanır.

Son olarak, Analiz evresi ise bu çalışma ile beraber otomatize edilen aşamayı temsil etmektedir. Değerlendiriciler veri girişini sağladıklarında web uygulaması otomatik olarak verileri analiz etmekte ve niceliksel ve niteliksel sonuçları raporlamaktadır.

Yukarıdaki sürece bağlı olarak, durum çalışmalarının, söz konusu MAS DSML'inin kullanımının deneyimlenmesini sağladığı belirtilebilir. Bunun yanı sıra, durum çalışmaları, MAS DSML işletimsel özelliklerinin toplanmasını da sağlar.

## 4.3 Niceliksel Değerlendirme Yaklaşımı

AgentDSM-Eval bünyesinde niceliksel değerlendirme süreci 3 ana bölümden oluşmaktadır. Değerlendirilmek istenen MAS DSML öncelikle referans üstmodel olarak seçilen FAML üstmodeli ile kıyaslanmaktadır. Geliştirme süreleri

ve geliştirme çıktıları analizi ise değerlendirme sürecinde gerçekleştirilen durum çalışması üzerinden sürelerin ve çıktıların karşılaştırmalı değerlendirmesini içermektedir.

### 4.3.1 Referans üstmodel ile kıyaslama

AgentDSM-Eval’de bir MAS DSML’inin referans bir üstmodelle göre kapsayıcılığının ne seviyede olduğunun belirlenmesi de gerçekleştirilmektedir. Önceki çalışmalarda (örneğin Challenger et al., 2016; Kardas et al., 2018) MAS DSML’leri için belli düzeylerde etmen iç yapısı, davranış modeli ve MAS bünyesinde etmenler arasındaki etkileşim için kavramlar ve ilişkiler tanımlanmıştır. Ancak bunların özellikle karşılaştırmalı analizini desteklemek için AgentDSM-Eval’de bu tanımlamalara ek olarak değerlendirmeye alınacak her MAS DSML’inin üstmodelinin analizini bir referans üstmodelle göre yapabilme özelliği de sunulmaktadır. Böylece her bir dil bu referans modele göre değerlendirildikten sonra örneğin dilin BDI etmen modelini ne oranda sağladığı ile ilgili olarak MAS geliştiricilere niceliksel bir bilgi sunabilmektedir. Referans üstmodeli için kapsayıcı ve genel bir üstmodel sunması nedeniyle FAML üstmodeli (Beydoun et al., 2009) tercih edilmiştir. FAML üstmodeli tasarım zamanı ve çalışma zamanı adı verilen iki katmana sahiptir. Her katmanın ise iki kapsamı vardır: “agent-internal” ve “agent-external”. Bu iki katman ve iki kapsama göre 4 farklı durumu ifade eden 4 farklı üstmodel FAML’da tanımlanmıştır.

FAML’daki her üstmodel için belirlenen kavramların tanımlamaları bulunmaktadır. Ortaya konulan 4 üstmodel için 2 düzeyde kavram listesi sunulmaktadır. Bunlar tasarım zamanı kavramları ve çalışma zamanı kavramlarıdır. Bu kavramlar Tablo 4.1 ve 4.2’de görülmektedir. Anlam karmaşıklığını önlemek amacıyla tablolarda kavramlar orjinal İngilizce isimleri ile kullanılmıştır. AgentDSM-Eval’de bu kavramlar ile üzerinde çoklu durum çalışması yürütülen ve AgentDSM-Eval ile değerlendirilmesi yapılmakta olan bir MAS DSML’inin soyut sözdizimindeki varlıklar eşleştirilmekte ve FAML ile ilgili DSML’in ortaklıkları / benzerlikleri sayısal olarak hesaplanmaktadır. Bir başka MAS DSML’i de bu şekilde değerlendirildikten sonra bu iki MAS DSML’inin karşılaştırılmasında bu dillerin soyut sözdizimlerinin FAML ile eşleştirilmesinde elde edilen sonuçlar da göz önüne alınabilecektir.

Tablo 4.1: FAML üstmodeli tasarım zamanı kavramları (Beydoun et al., 2009).

| FAML Tasarım Zamanı Kavramları |   |
|--------------------------------|---|
| Kavram                         | Tanım   |
| Action Specification           | Herhangi bir önkoşul ve sonkoşul dahil, bir eylemin belirtilmesi.                                   |
| Agent Definition               | Bir etmenin yaratılmasından hemen sonra başlangıç durumunun belirtilmesi.                           |
| Environment Statement          | Çevre hakkında Boole ifadenin belirtilmesi  |
| Facet Action Specification     | Görünüm değişimi veya yeni değer yazımı açısından görünüm eylemi tanımlaması                        |
| Facet Definition               | Adı, veri tipi ve erişim modu dahil olmak üzere verilen bir görünüm yapısının belirlenmesi.         |
| Functional Requirement         | Sistem kullanıcılarına katma değer sağlayan gereksinim.   |
| Interaction Protocol           | Sistemde meydana gelen iletişim modellerinin belirlenmesi.  |
| Mental State Specification     | İlk zihinsel durumun spesifik inançlar ve temsilci hedefleri açısından şartnamesi                   |
| Message Action Specification   | Bir mesaj eyleminin mesaj şeması ve kullanılacak parametreler açısından şartnamesi                  |
| Message Schema                 | Sistem içerisinde oluşabilecek belirli bir mesaj türünün yapısının ve anlamının belirlenmesi.       |
| Non-Functional Requirement     | Oluşturulacak sistemde herhangi bir sınırlama, kısıtlama ya da dayatma gerekliliğinin belirlenmesi. |
| Ontology                       | Belirli bir alanın yapısal modeli.  |
| Organization Definition        | Bir sistem hedefine yönelik işbirliği yapan roller ve temsilciler koleksiyonunun belirlenmesi.      |
| Plan Resource Specification    | Plan belirtiminde kullanılan kaynakların tanımlaması  |
| Plan Specification             | Eylem tanımlamasının organize koleksiyonu   |
| Policy                         | Belirli bir ortamda gerçekleşmesi beklenen olayların düzenini belirten bir kural.                   |
| Requirement                    | Bir sistemin uygulanması gereken özellik.   |
| Resource Specification         | Adı, makul gösterimleri olan ve edinilen, paylaşılan veya üretilen kaynak şartnamesi.               |
| Role                           | Belirli bir sistemdeki bazı etmenlerin beklenen davranış biçiminin belirlenmesi.                    |
| Role Compatibility             | Kaynak rolün belirli bir amaç için hedef rolle bağdaşmadığı rol ilişkisi.                           |
| Role Dependency                | Kaynak rolün belirli bir amaç için hedef rolüne bağlı olduğu rol ilişkisi.                          |
| Role Relationship              | Belirli bir amaç için iki rol arasındaki sosyal ilişki.   |
| Service                        | Bir etmenin katılabileceği tek, tutarlı aktivite bloğu.   |
| System                         | Etmen odaklı yazılım geliştirme projesinin son ürünü.   |
| System Goal                    | Sistemin gerçekleştirmeye çalıştığı çevre durumunun tanımlaması                                     |
| Task                           | Sistemin gerçekleştirebileceği bir davranış parçasının belirlenmesi.                                |

FAML'ın referans üstmodel olarak kullanılması sonucunda değerlendirilecek DSML'lerin MAS programlama perspektifinde yeterlilikleri tespit edilirken öncelikle yukarıda anlatılan FAML ve dil sözdizimi kavramları arasındaki eşleşmeler tamamlanmalıdır. Bu aşamada eşleştirme işlemi değerlendirici tarafından FAML üstmodelinde yer alan kavramların tanımlamaları doğrultusunda doğrulanmalıdır. Eşleşmenin doğrulanması sonrası FAML üstmodeli ile değerlendirilen üstmodelin yakınlığı sayısal olarak belirlenebilecektir.

Tablo 4.2: FAML üstmodeli çalışma zamanı kavramları (Beydoun et al., 2009).

| FAML Çalışma Zamanı Kavramları |   |
|--------------------------------|---|
| Kavram                         | Tanım   |
| Action                         | Temel etmen davranışı birimi.   |
| Agent                          | Çok özerk, yerleşmiş, yönlendirilmiş ve ilişkisel bir varlık.                                 |
| Agent Goal                     | Bir etmen tarafından takip edilen bir durumu temsil eden çevre bildirimi.                     |
| Belief                         | Bir etmen tarafından tutulan ve belirli bir zaman diliminde doğru sayılan bir çevre bildirimi |
| Communication                  | Birden fazla mesajın bileşimi   |
| Environment                    | Bir etmenin bulunduğu dünya.  |
| Environment History            | Çevre başlangıcı ile verilen herhangi bir an arasında meydana gelen olayların sırası.         |
| Environment Statement          | Çevre hakkında bir açıklama   |
| Event                          | Çevre geçmişini değiştiren bir şeyin oluşumu.   |
| Facet                          | Etmenlerin etkileşime girebileceği ortamın mülkiyeti.   |
| Facet Action                   | Belirli bir görünüm değişikliğine yol açan eylem.   |
| Facet Event                    | Bir görünümün değeri değiştiğinde gerçekleşen olay.   |
| Mental State                   | Bir etmen tarafından belirli bir zaman diliminde tutulan etmen hedefleri ve inançları.        |
| Message                        | Belirli bir mesaj şemasına uyan etmenler arasındaki iletişim birimi.                          |
| Message Action                 | Mesajın gönderilmesine yol açan eylem.  |
| Message Event                  | Bir mesaj gönderildiğinde gerçekleşen olay.   |
| Organization                   | Bir sistem hedefine doğru işbirliği yapan özel rollere sahip etmenler topluluğu.              |
| Plan                           | Belirli bir etmen hedefine ulaşmak için gerçekleştirilebilecek organize eylemler topluluğu.   |
| Resource                       | Adı olan, makul temsilleri olan ve edinilen, paylaşılan veya üretilen bir ortam varlığı.      |
| Role                           | Belirli bir sistemdeki bazı etmenlerden beklenen bir davranış biçiminin belirlenmesi.         |
| System                         | Etmen odaklı yazılım geliştirme projesinin son ürünü.   |

### 4.3.2 Durum çalışması analizi

Durum çalışması analizi değerlendirme süreci için belirlenen bir MAS geliştirme (durum) çalışması temel alınarak elde edilen sonuçları içermektedir. Durum çalışmaları geliştirme süreleri ve geliştirme çıktıları olmak üzere 2 temel çıktıya sahiptir.

Geliştirme Süreleri değerlendirici grubun belirlenen bir durum çalışması için problem analizi, modelleme ve tasarım, geliştirme ve test işlemlerine ayırdıkları süreleri ifade etmektedir. Ölçülen bu süreler değerlendirici sayısı arttıkça her bir parametre için çeşitli sonuçlar çıkarmaya elverişli bir veriyi sağlamaktadır.

MAS DSML'ler öncelikli olarak modelleme ve tasarım süresi ile geliştirme süresini kısaltmayı hedeflemektedirler. Geliştiricilere basit ve hızlı modelleme yapmasını sağlayan anlaşılır bir arayüz sunarak modelleme süresini kısaltabilirler. Modelleme çıktısı ile yapılan başarılı bir kod üretimi ile de durum çalışması gereksinimlerini mümkün olduğunca çok sağlayarak geliştiricilerin daha az delta kod yazmalarını sağlayabilirler. Bu yaklaşımdan hareketle modelleme ve tasarım süresi ile gerçekleştirme süresi ilişkisi analiz edilebilir. Bir MAS DSML'inin gelecek versiyonları için başarı kriterleri belirlenebilir. Değerlendirme sürecinde geliştirme süreleri ilgili durum analizine özel olarak incelenebileceği gibi tüm durum çalışmalarını kapsayan ortalama sürelerin de analiz edilebilmesinde kullanılabilir.

Geliştirme çıktıları açısından analiz ise 2 farklı perspektiften gerçekleştirilmektedir. Öncelikle, durum çalışması sonucu elde edilen modelleme çıktısını incelenerek MAS DSML'e ait dil elemanlarının kullanım sıklıkları belirlenmektedir. Bu değerlendirme MAS DSML için dil elemanları açısından hangi noktalarda geliştirme yapılması gerektiği ile ilgili bir sonuç sunmaktadır. Daha az kullanılan veya kullanılmayan dil elemanları belirlenerek, dilin gelecek versiyonlarında bu noktalarda iyileştirmeler yapılabileceği sonucu çıkarılabilecektir. Değerlendirme durum çalışması özelinde yapılabileceği gibi tüm durum çalışmalarını kapsayan bir ortalama sonucu da sunabilmektedir.

MAS DSML'ler modelleme çıktıları kullanarak kod üretimi de yapmaktadır. Kod üretiminin performansının yüksek olması değerlendiricinin daha az delta kod ekleyerek durum çalışmasının gerekliliklerini yerine getirmesini sağlamaktadır. Dilin kod üretimi performansı ölçülmesi gereken bir parametre

olarak önplana çıkmaktadır. Değerlendirme çerçevesi kapsamında üretilen kodlar ile nihai kodlar kıyaslanarak dilin kod üretim performansı analiz edilmektedir. Diğer değerlendirme sonuçlarında olduğu gibi bu sonuçlar da durum çalışması özelinde olabileceği gibi tüm durum çalışmalarını kapsayan ortalama bir sonucu da sunmaktadır.

#### 4.4 Niteliksel Değerlendirme Yaklaşımı

Niteliksel değerlendirme için anket aracılığı ile kullanıcılardan veri toplama yöntemi tercih edilmiştir. Niteliksel analiz için uygun metriklerin belirlenmesinde FQAD isimli nitelik analizi çalışması (Kahraman and Bilgen, 2015) temel alınmıştır. FQAD çalışmasında yer alan metrik ve kavramlar MAS DSML'lerin değerlendirilmesi için güncellenmiş ve özelleştirilmiştir. FQAD, DSL'lerin niteliksel açıdan değerlendirilmesi için formal bir yaklaşım içermektedir. Bu yaklaşıma göre herhangi bir DSL'in değerlendirilmesi için kıstas olabilecek bir takım karakteristikler belirlenmiş ve tanımlanmıştır. Belirlenen her bir karakteristik, alt karakteristiklere sahiptir. Herhangi bir karakteristiğin o dil için başarılı olarak değerlendirilmesi bu alt karakteristiklerin başarılı olmasına bağlıdır.

FQAD'de yer alan metrikler genel yapı itibarıyla DSL'leri değerlendirmek üzerine tasarlanmıştır. Bu doğrultuda öncelikli olarak tezde bu alt karakteristikler üzerinde çalışılmış ve bunların MAS DSML'lere uygun şekilde tekrar düzenlenmesi gerçekleştirilmiştir. Ayrıca MAS geliştirme perspektifinden yeni alt karakteristikler de (örneğin dilin MAS geliştirmeyi kolaylaştırmasını değerlendiren veya dilin MAS kavramlarını kapsama düzeyini bildiren karakteristikler) eklenerek etmen modelleme için kullanılan bir dilin niteliksel olarak daha somut şekilde değerlendirilmesi hedeflenmiştir. AgentDSM-Eval kapsamında da alt karakteristikler FQAD'e benzer şekilde farklı karakteristikleri temsil etmektedir. Tablo 4.3'te bu karakteristik ve alt karakteristikler yer almaktadır. Değerlendirme 6 karakteristik açısından gerçekleştirilmektedir. Fonksiyonel Uygunluk, Kullanılabilirlik, Güvenilirlik, Anlamlılık, Uygunluk ve Çok Etmenli Sistem Geliştirme kategorilerinde 24 alt karakteristik ile ilgili görüşleri değerlendiricilere sorulmaktadır. Değerlendiriciler olumsuzdan olumluya olacak şekilde 1-5 arası ölçekte puan verebilmektedirler.



Tablo 4.3: AgentDSM-Eval aracında yer alan niteliksel analiz karakteristik ve alt karakteristikleri

| Karakteristik                 | No | Alt Karakteristik   |
|-------------------------------|----|---|
| Fonksiyonel Uygunluk          | 1  | Alanın tüm kavramları ve senaryoları etmen modelleme dilinde ifade edilebilmektedir. (tamlık)   |
|                               | 2  | Etmen modelleme dili alana özgü spesifik uygulamalar için uygundur (örneğin bir algoritmayı ifade etmek için). (uygunluk)   |
| Kullanılabilirlik             | 3  | Dilin anlaşılması için gereken çaba azdır. (anlaşılabilirlik)   |
|                               | 4  | Dilin kavram ve sembollerinin öğrenilmesi ve hatırlanması kolaydır. (öğrenilebilirlik)  |
|                               | 5  | Dil, kullanıcıların görevlerini minimum sayıda adım ile gerçekleştirmelerine yardımcı olma becerisine sahiptir.   |
|                               | 6  | Kullanıcılar, etmen modelleme dilinin ihtiyaçlar için uygun olup olmadığını belirleyebilir. (benzerlik, kullanıcı algılama)   |
|                               | 7  | Etmen modelleme dili işletmeyi ve kontrol etmeyi kolaylaştıran özelliklere sahiptir. (çalışabilirlik)   |
|                               | 8  | Etmen modelleme dilinin iyi görünümü (çekiciliğe sahip) sembolleri vardır.  |
|                               | 9  | Dil, programın sunumunun kompaktlığı için mekanizmalar sağlar. (yoğunluk)   |
| Güvenilirlik                  | 10 | Etmen modelleme dili, kullanıcıları hata yapmaya karşı korur; kullanıcının hata yapmasını önler. (model kontrolü)   |
|                               | 11 | Etmen modelleme dili, doğru öğeleri ve bunlar arasındaki doğru ilişkileri içerir (Öğeleri arasındaki beklenmedik etkileşimleri önler). (doğruluk)   |
| Anlamlılık                    | 12 | Problem çözme stratejisi kolayca programa eşlenebilir.  |
|                               | 13 | Etmen modelleme dili her bir kavramı ifade etmek için tek bir iyi yol önermektedir. (benzersiz)   |
|                               | 14 | Dildeki her yapı, alanında tam olarak tek bir kavramı temsil etmek için kullanılır. (ortogonal)   |
|                               | 15 | Dil yapıları önemli alan kavramlarına karşılık gelir. Etmen modelleme dili, önemli olmayan alan kavramlarını içermez.   |
|                               | 16 | Etmen modelleme dili, çakışan öğeler içermez.   |
|                               | 17 | Etmen modelleme dili, gerekli olandan daha karmaşık veya detaylı olmayan yapılar içermez; doğru soyutlama seviyesindedir.   |
| Uygunluk                      | 18 | Model geliştirmek için etmen modelleme dilini kullanmak geliştirme sürecine uygundur; çünkü aşamalar ve roller geliştirme sürecinin bir parçası olarak kullanılır.                              |
|                               | 19 | Etmen modelleme dili, alanla uyumludur. Etki alanında belirli bir uygulamayı gerçekleştirmek için gerekli hiçbir değişiklik olmadan etki alanının diğer öğeleriyle çalışma yeteneğine sahiptir. |
| Çok Etmenli Sistem Geliştirme | 20 | Etmen modelleme dili, MAS geliştirmeyi kolaylaştırır.   |
|                               | 21 | Etmen modelleme dili, Kanı-İstek-Hedef (ing. Belief-Desire-Intention) (BDI) etmenlerini geliştirme için uygundur.   |
|                               | 22 | Etmen modelleme dili IDE'sinin kullanımı kolaydır.  |
|                               | 23 | Etmen modelleme dili, genel MAS yapısını modellemek için yeterince güçlüdür.  |
|                               | 24 | Etmen modelleme dili, kullanırken karşılaşılan zorlukları ele alır.   |

MAS geliştirme perspektifinden somut sonuçlar elde etmek için bu tez çalışması kapsamında eklenen metrikler “Çok Etmenli Sistem Geliştirme” karakteristiği başlığı ile Tablo 4.3’te görülmektedir. MAS geliştirme sürecine özel, kullanıcı deneyimlerini kategorize etmeyi amaçlayan bu karakteristik 5 alt karakteristik içermektedir. Metriklerden ilki (Tablo 4.3 – no. 20) kullanıcıların MAS geliştirme sürecinde etmen modelleme dilinin soyutlama ve araç yetenekleri ile süreci ne ölçüde kolaylaştırdığını değerlendirmeyi amaçlamaktadır. Diğer önemli bir konu ise etmen modellemede BDI mimarisi desteğinin kullanıcı deneyimi açısından ölçülmesidir. Tablo 4.3’te yer alan 21 numaralı alt karakteristik ile etmen modelleme dilinin BDI mimarisini gerçekleştirme başarısının ölçülmesi hedeflenmektedir. 22 numaralı metrikte ise etmen geliştirme aracının kullanım kolaylığı üzerinde durulmaktadır. Genel MAS mimarisinin geliştirilmesi sürecinde etmen modelleme dilinin yeteneklerinin değerlendirilmesi 23. alt karakteristikte, modelleme sürecinde karşılaşılan zorluklarla ilgili dilin sağladığı desteğin ölçülmesi ise 24. alt karakteristikte metrik olarak sunulmuştur.

## 5. AGENTDSM-EVAL DEĞERLENDİRME ARACI

Bu bölümde, önceki bölümde tanıtılan değerlendirme çerçevesine uygun olarak MAS DSML'lerinin değerlendirmesine çevrimiçi imkan veren web uygulaması tanıtılmaktadır. Tez kapsamında hazırlanan AgentDSM-Eval web uygulamasına aşağıdaki adres üzerinden erişilebilir:

- <https://agent-dsml-evaluation-tool.web.app>


### 5.1 AgentDSM-Eval Uygulaması

AgentDSM-Eval değerlendirme çerçevesinin web uygulaması temelde 2 web sayfası üzerinden gerçekleştirilmiştir. Bunlar değerlendirme sonuçları sayfası ve değerlendirme formu sayfası olarak tanımlanabilir.

#### 5.1.1 Değerlendirme formu sayfası

Değerlendirme formu sayfası Çoklu-durum çalışması esnasında değerlendiricilerin veri girişi yaptıkları sayfa olarak tanımlanabilir. Bu sayfa sayesinde değerlendirici deneyimlediği MAS DSML ile ilgili değerlendirme verilerini sisteme girmektedir.

Bu sayfa 5 bölümden oluşmaktadır. İlk bölüm Şekil 5.1'de görülen onay metnini içermektedir. Değerlendirmeye katılan katılımcılar bu metni onayladıkları sürece değerlendirmeye dahil olabilirler.

 **Letter of Consent**

This experimental work is conducted within the EGE University International Computer Institute Software Engineering Laboratory (SerLAB).

All information stated as part of this experiment is confidential and will be kept as such. Associate Professor Geylani Kardeş and Ömer Faruk Alaca are responsible for this experiment and can be contacted at:  
Associate Professor Geylani Kardeş: <http://ube.ege.edu.tr/~kardas/>  
Ömer Faruk Alaca: [omerfarukalaca@gmail.com](mailto:omerfarukalaca@gmail.com)

We would like to emphasize that:

1. Your participation is entirely voluntary;
2. You are free to refuse to answer any question;
3. You are free to withdraw at any time.

The experiment will be kept strictly confidential and will be made available only to members of the research team of the study or, in case external quality assessment takes place, to assessors under the same confidentiality conditions. Data collected in this experiment may be part of the final research report, but under no circumstances will your name or any identifying characteristic be included in the report.

I accept the terms addressed above:

Yes  No

Şekil 5.1: Onay metni ekranı

Onay metni ardından değerlendirmeye katılan katılımcıları analiz etmek için kişisel verilerin toplandığı bölüm gelmektedir. Bu bölümde kullanıcının email adresi (değerlendirme analizinde gösterilmemektedir, sadece ileriki süreçte kullanıcı etkileşimi için alınmaktadır.), cinsiyeti, yaşı, çalışma alanı, tamamladığı eğitim seviyesi, MAS deneyimi ve şu anki çalışma durumu ile ilgili bilgiler alınır. Bu bölüm ile ilgili kullanıcı arayüzü Şekil 5.2’te görülmektedir.

## Personal Data

---

**Email**

abc@xyz.com

---

**Gender**

Male

Female

**Age**

17-22

23-25

26+

---

Please select your field of studies.

Computer Science/Engineering

Electronic Engineering

Other

**Completed Education**

Bachelor of Science

Master of Science

Doctor of Philosophy

Postdoctoral

---

**Previous Experience With Multi-Agent Systems (MAS)**

I've learned MAS in the context of a course.

I've used MAS in a professional context

I know what MAS is but never used it

I've never heard of it.

**Current occupation**

Student

Researcher

Worker

Şekil 5.2: Kişisel bilgi girişi ekranı

Değerlendiriciden kişisel bilgileri de alındıktan sonra niceliksel ve niteliksel analiz için gerekli veriler girilmeye başlanmaktadır. Öncelikle değerlendiriciden geliştirme sürecinde problem analizi, modelleme ve tasarım, geliştirme ve test için geçirdiği süreleri dakika cinsinden girmesi istenmektedir (Şekil 5.3). Bu alanlar sadece numerik veri girilebilen ve boş bırakılmayan alanlar olarak tasarlanmıştır.

**Development Times**

Please enter the time for each phase during the evaluation process. All fields must be in **minutes** and as an **integer**

**Problem Analysis** (minutes)

**Modelling - Design** (minutes)

**Implementaion** (minutes)

**Error detection - Testing** (minutes)

Şekil 5.3: Geliştirme süreleri giriş ekranı

Bir sonraki bölümde durum çalışmasının gerçekleştirimi sürecinde elde edilen 3 ana çıktının sisteme yüklenmesi istenmektedir. Bu çıktılar modelleme proje dosyası, modelleme dosyası ile MAS DSML tarafından üretilen kodları içeren temel proje dosyası ve son olarak temel proje dosyası üzerinde değerlendiricinin delta kodlarını da içeren ve durum çalışmasının gereksinimlerini yerine getiren tam proje dosyasıdır. Değerlendiriciden bu üç dosyayı “zip” formatında Şekil 5.4’te yer alan arayüz aracılığı ile yüklemesi istenmektedir.

**Project Files Uploads**

Please upload the project files you developed during the evaluation process in this section. In this process you must have 3 types of folders. These files and their descriptions are listed below.

- Modelling Artifacts:** The model project folder that you developed using the agent modelling tool.
- Generated Project:** Project folder generated by agent modeling tool.
- Full Project:** Project folder that also including your personal codes.

Please note that upload the folders in **.zip format**.

**Modelling Artifacts** (Please rename your zip file to *ModellingProject.zip*)

**Generated Project** (Please rename your zip file to *GeneratedProject.zip*)

**Full Project** (Please rename your zip file to *FullProject.zip*)

Şekil 5.4: Proje dosyaları yükleme ekranı

Son bölümde ise niteliksel analiz çalışması için kalite anketi yer almaktadır. Bu bölümde değerlendiricilerden deneyimlerini 1: çok kötü, 5: çok iyi olmak üzere 1 ile 5 arasında belirtmeleri istenmektedir. Kalite anketi ile ilgili kullanıcı arayüzü Şekil 5.5’te görülmektedir. Buradaki sorular daha önce Tablo 4.3’te verilen kriterleri yansıtmaktadır.

## Quality Survey

Please evaluate Agent DSML in terms of the following norms. You can give a score between 1 and 5 to the norms.

1. Very Bad
2. Bad
3. Average
4. Good
5. Very Good

- 1- All concepts and scenarios of the domain can be expressed in the Agent DSML (completeness)  
 1  2  3  4  5
- 2- Agent DSML is appropriate for the specific applications of the domain (e.g. to express an algorithm) (appropriateness)  
 1  2  3  4  5
- 3- The required amount of effort for understanding the language is small (comprehensibility)  
 1  2  3  4  5
- 4- The concepts and symbols of the language are easy to learn and remember (learnability)  
 1  2  3  4  5
- 5- Language has capability to help users achieve their tasks in a minimum number of steps  
 1  2  3  4  5
- 6- Users can recognize whether the Agent DSML is appropriate for their needs (likeability,userperception)  
 1  2  3  4  5
- 7- Agent DSML has attributes that make it easy to operate and control the language (operability)  
 1  2  3  4  5
- 8- Agent DSML has symbols that are good-looking (attractiveness)  
 1  2  3  4  5
- 9- The language provides mechanisms for compactness of the representation of the program. (compactness)  
 1  2  3  4  5
- 10- Agent DSML protects users against making errors. The DSL avoids the user to make mistakes. (model checking)  
 1  2  3  4  5
- 11- Agent DSML includes right elements and correct relations between them (Agent DSML prevents the unexpected interactions between its elements) (correctness)  
 1  2  3  4  5
- 12- A problem solving strategy can be mapped into a program easily  
 1  2  3  4  5
- 13- The Agent DSML that provides one and only one good way to express every concept of interest (unique)  
 1  2  3  4  5
- 14- Each Agent DSML construct is used to represent exactly one distinct concept in the domain (orthogonal)  
 1  2  3  4  5
- 15- The language constructs correspond to important domain concepts. Agent DSML does not include domain concepts that are not important.  
 1  2  3  4  5
- 16- Agent DSML does not contain conflicting elements.  
 1  2  3  4  5
- 17- Agent DSML is at the right abstraction level such that it is not more complex or detailed than necessary  
 1  2  3  4  5
- 18- Using Agent DSML to develop models fits in the development process, since it is used as part of a development process with phases and roles.  
 1  2  3  4  5
- 19- Agent DSML is compatible with the domain. Agent DMSL has capability to operate with other elements of the domain with no modification required to perform a specific application in the domain.  
 1  2  3  4  5
- 20- Agent DMSL make MAS development easier.  
 1  2  3  4  5
- 21- Agent DSML is appropriate for BDI (Belief-Desire-Intention) agent development.  
 1  2  3  4  5
- 22- IDE of Agent DSML is easy to use.  
 1  2  3  4  5
- 23- Agent DSML is powerful enough to model the general MAS structure.  
 1  2  3  4  5
- 24- Agent DSML handles any encountered difficulties while using it.  
 1  2  3  4  5

SUBMIT

Şekil 5.5: Kalite anketi ekranı

Değerlendirici tüm bölümleri tamamladıktan sonra Gönder (ing. Submit) butonu ile değerlendirme sonuçlarını sisteme göndermektedir. Bu işlemin ardından sistemsal bir sorundan dolayı veri kaybını engellemek adına değerlendiricinin cevaplarını içeren bir pdf dökümanı kullanıcının bilgisayarına otomatik olarak indirilmektedir.

### 5.1.2 Değerlendirme sonuç sayfası

Değerlendirme sonuç sayfası, her MAS DSML'i için özel olarak bulunan ve değerlendirme sonuçlarını ayrıntılı olarak içeren bir sayfadır. Bu sayfa dil ile ilgili genel bilgiler, referans model karşılaştırması, geliştirme süreleri analizi, anket sonuçları ve durum çalışması analizi olmak üzere 5 bölümden oluşmaktadır.

Dil ile ilgili genel bilgiler bölümü, dilin adı, versiyonu, ilgili web bağlantısı ve dili geliştiren ekip bilgisini barındıran bölümlerden oluşmaktadır. Ayrıca bu bölümde dili değerlendirmek için durum çalışması seçilip yönlendiren bir buton da yer almaktadır. İlgili kullanıcı arayüzü Şekil 5.6'da görülmektedir.

**Identification of Prometheus/PDT**

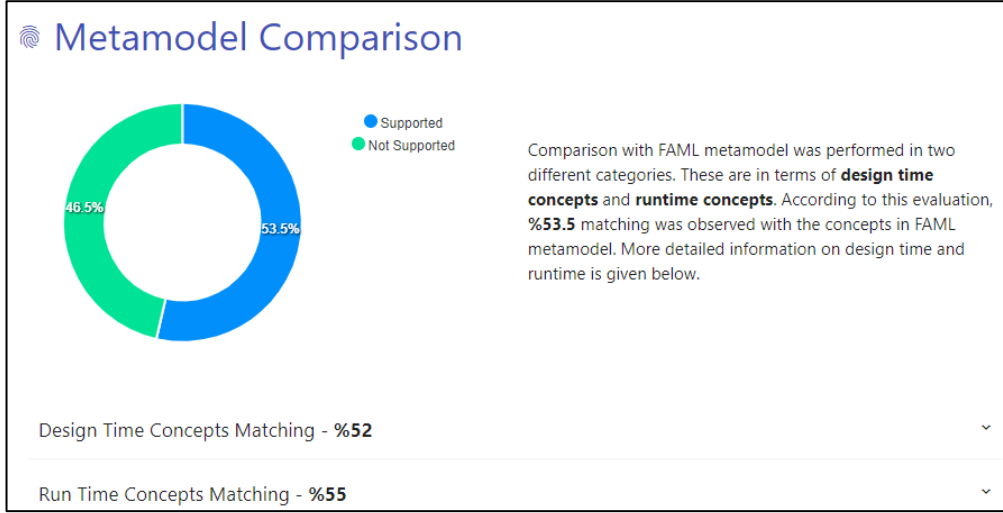
|                        |   |
|------------------------|---|
| <b>Language Name :</b> | Prometheus/PDT  |
| <b>Version :</b>       | 0.5.1   |
| <b>Owner :</b>         | RMIT University Agents Group  |
| <b>URL :</b>           | <a href="https://sites.google.com/site/rmitagents/software/prometheusPDT">https://sites.google.com/site/rmitagents/software/prometheusPDT</a> |

If you want to make evaluation for Prometheus/PDT language. Please select a case study and press the evaluate button.

Case Study Start Evaluation

Şekil 5.6: Dil genel bilgileri ekranı

Dil ile ilgili genel bilgilerin ardından referans model ile kıyaslama sonuçları yer almaktadır. Referans model olarak seçilen FAML üstmodeli için tasarım zamanı ve çalışma zamanı perspektifinde 2 ayrı kıyaslama yapılmaktadır. Bu bölümde öncelikle bu iki perspektiften yapılan kıyaslamaların ortalama sonucu donut grafiği ile gösterilmektedir. Desteklenen ve desteklenmeyen kavramların yüzdesi verilmektedir (Şekil 5.7).



Şekil 5.7: Referans üstmodeli ile kıyaslama (genel)

Donut grafiğın altında ise tasarım zamanı ve çalışma zamanı için ayrıntılı sonuçların verildiği, kapatılıp açılabilen menüler yer almaktadır (Şekil 5.8). Bu menülerde desteklenen kavramlar yeşil, desteklenmeyen kavramlar kırmızı ile ifade edilmektedir. Ayrıca bu bölümde kavramların yer aldığı alan da açılıp kapanma özelliğine sahiptir ve kavramların FAML tanımlamalarını içermektedir.

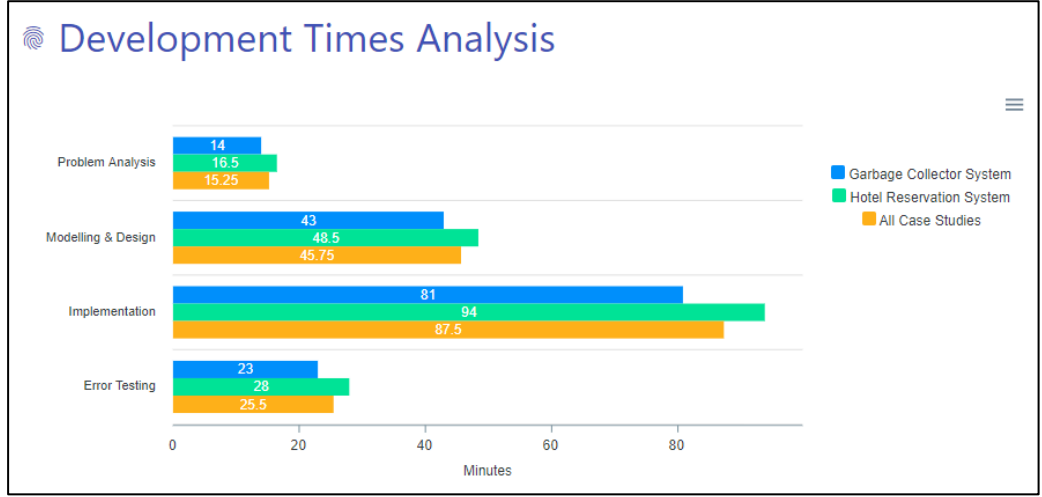
Design Time Concepts Matching - **%52**

| FAML Concept  | Prometheus/PDT Concept |
|---|------------------------|
| > Action Spesification  | Action                 |
| > Agent Definition  | Agent                  |
| ∨ Environment Statement   | N/A                    |
| <b>FAML Definition:</b> A Boolean statement about the environment |                        |
| > Facet Action Spesification                                      | Action                 |
| > Task  | N/A                    |
| > Facet Definition  | N/A                    |

Şekil 5.8: Referans üstmodeli ile kıyaslama (ayrıntılı)

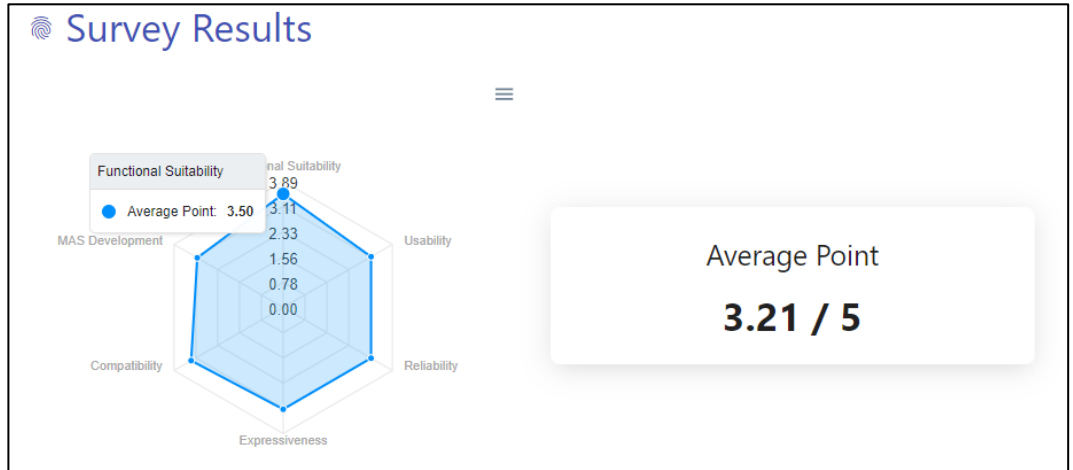
Geliştirme sürelerinin analizi ise yatay çubuk grafiği ile sunulmaktadır. Geliştirme süreleri grafiğinin bir örneği Şekil 5.9'de görülmektedir. Grafik geliştirme sürecini oluşturan problem analizi, modelleme ve tasarım, geliştirme ve test süreçlerini dakika cinsinden ayrı ayrı görüntülemektedir. Grafiğın sağında yer alan menü aracılığı ile herhangi bir durum çalışması grafiğe eklenip çıkarılabilmekte, tüm durum çalışmaları açısından sonuç görüntülenebilmektedir.





Şekil 5.9: Geliştirme süreleri analizi

Geliştirme sürelerinin analiz sonuçlarının ardından web uygulamasında niteliksel analiz için gerçekleştirilen anket çalışmasının sonuçları gösterilmektedir. Bu bölüm 2 alt bölüm olarak ele alınabilir. AgentDSM-Eval değerlendirme çerçevesinin tanıtıldığı bir önceki bölümde detaylandırıldığı gibi niteliksel analizde metrikler 6 farklı karakteristik üzerinden sunulmaktadır. Sözedilen karakteristiklerin 1-5 arası aldığı ortalama puanlar radar grafiğinde gösterilmektedir. Radar grafiğin üzerinde istenilen karakteristik üzerine gelinerek bu karakteristiğe ait ortalama puan da ayrıca görüntülebilmektedir (Şekil 5.10). Radar grafiğin sağında ise bu 6 karakteristik için elde edilen puanlardan hareketle toplam ortalama puan verilmektedir.

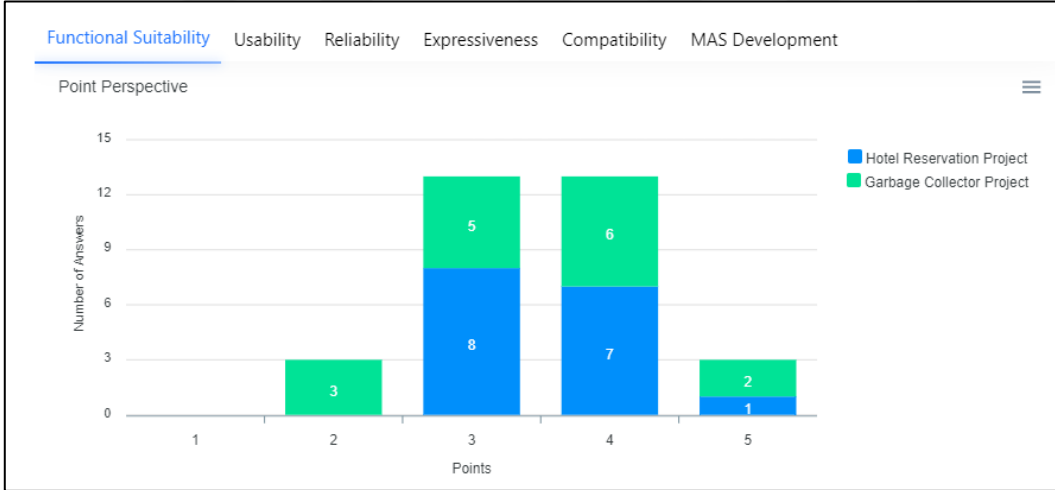


Şekil 5.10: Niteliksel analiz anketi genel sonuçları

Anket sonuçları ile ilgili ayrıntılar ise genel sonuçların altında sekme menüsü ile sunulmaktadır (Şekil 5.11). İncelenmek istenen karakteristik seçilerek

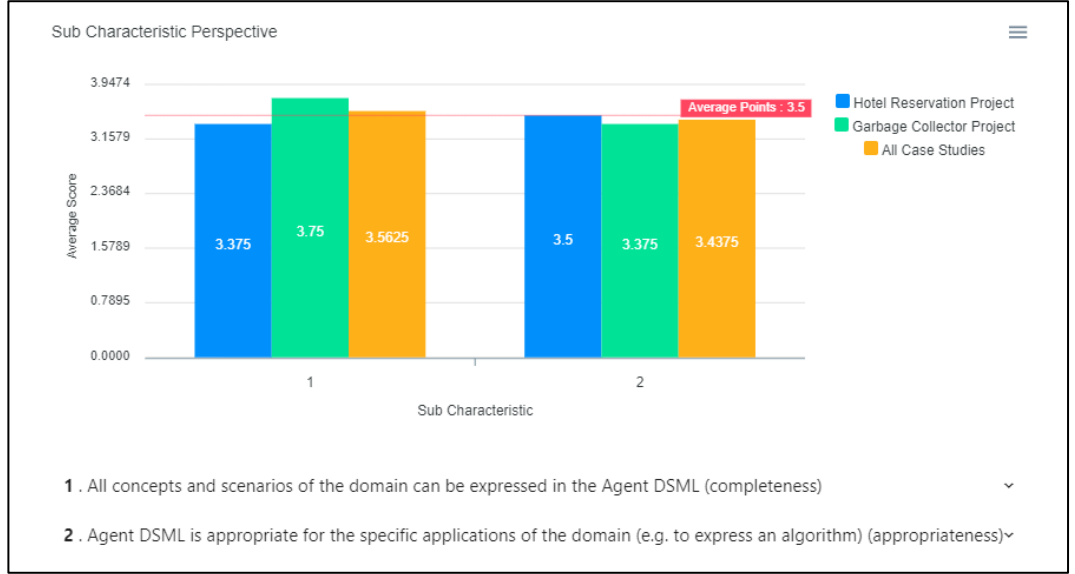
sonuçlar görüntülenebilmektedir. Sekme menüsü her bir karakteristik için genel sonuçları 2 farklı bakış açısı ile sunmaktadır. Bunlar puan bakış açısı ve alt karakteristik bakış açılarıdır. Her iki bakış açısı için de dikey çubuk grafiği ile sonuçlar gösterilmektedir.

Puan bakış açısı; varolan durum çalışmalarını kümülatif gösterecek şekilde ilgili karakteristik için değerlendiricilerin sorulara verdiği puanların dağılımını göstermektedir (Şekil 5.11). Çubuk grafiğinin dikey eksen (y eksen) cevap sayısını yatay eksen (x eksen) ise 1 ile 5 arasındaki puanları göstermektedir. Böylece bu grafik ile herhangi bir karakteristik için 1 ile 5 puan arasında kaç cevabın bulunduğu görüntülenmektedir. Bu grafik ilgili karakteristik için tüm alt karakteristik sonuçlarını içermektedir. Grafiklerde herhangi bir durum çalışmasını görüntülemek/gizlemek için ise sağda yer alan menü kullanılabilir (Şekil 5.11).



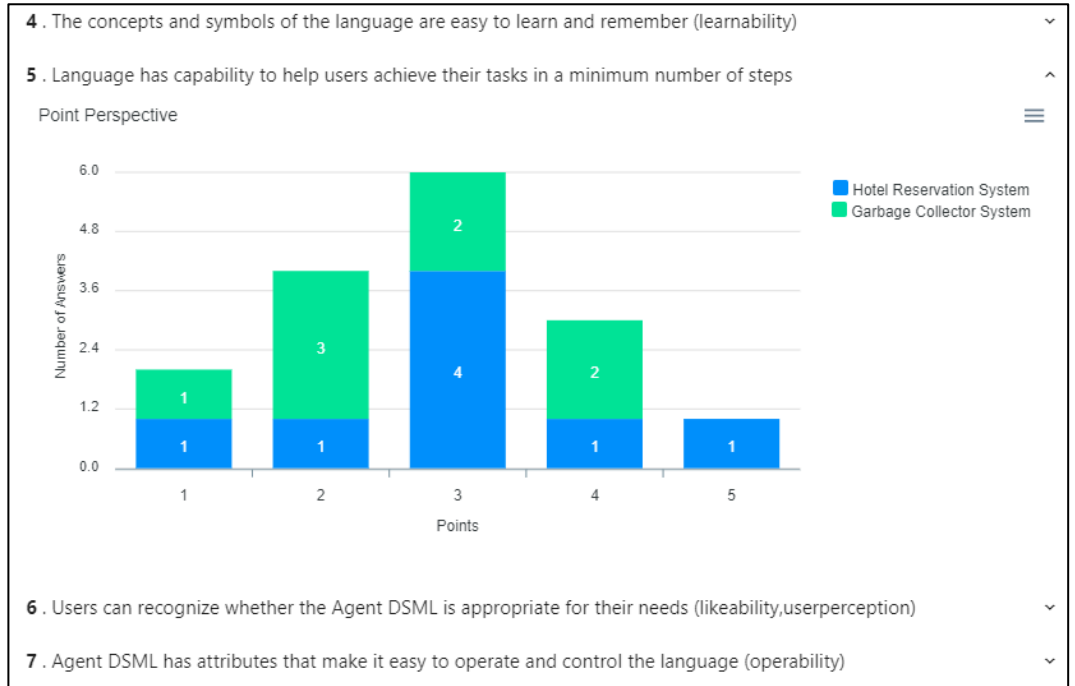
Şekil 5.11: Niteliksel analiz anketi karakteristik sonuçları (puan bakış açısı)

Alt karakteristik bakış açısı ise alt karakteristikler özelinde verilen puanların ortalamasını sunan bir grafik sağlamaktadır (Şekil 5.12). Çubuk grafiğinin dikey eksen (y eksen) ortalama puanı yatay eksen (x eksen) ise alt karakteristikleri göstermektedir. Ayrıca ilgili karakteristiğin genel ortalama puanı da çubuk grafiği yatay şekilde bölerek ortalamaya göre alt karakteristiklerin durumlarının okunmasını kolaylaştırmaktadır. Her bir alt karakteristik için varolan durum çalışmaları ayrı ayrı ve kümülatif olacak şekilde gösterilmektedir ve puan bakış açısı grafiğinde olduğu gibi istenen durum çalışması sağda yer alan menü aracılığı ile görüntülenip gizlenebilmektedir (Şekil 5.12).



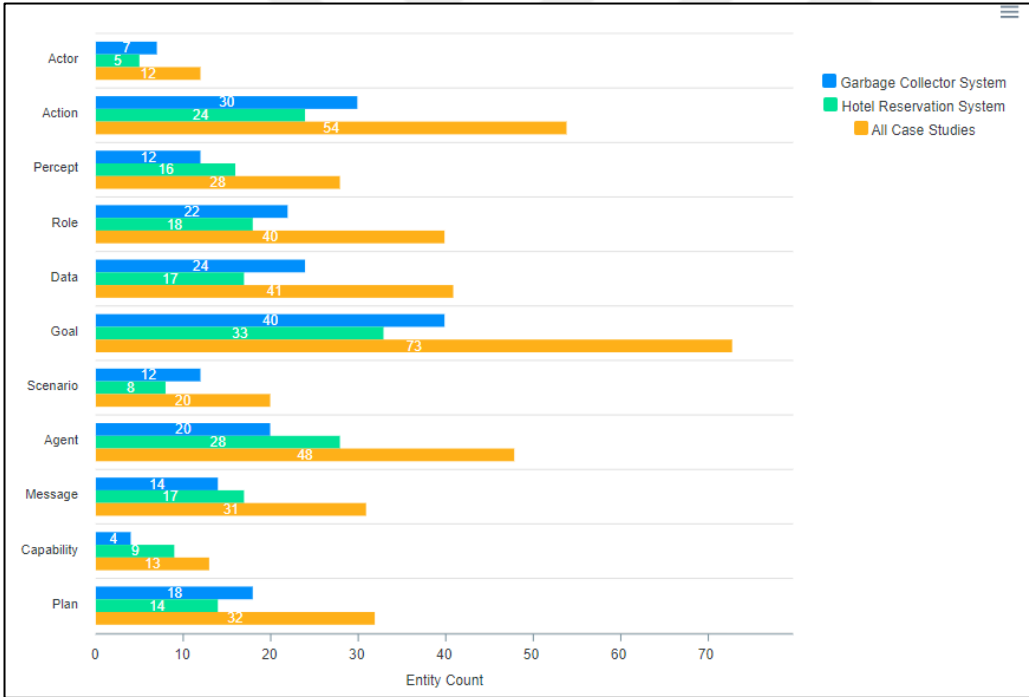
Şekil 5.12: Niteliksel analiz anketi karakteristik sonuçları (alt karakteristik bakış açısı)

Karakteristiklerle ilgili sonuçların görüntülediği sekme menüsünün en altında ise o karakteristik için belirlenen alt karakteristikler yer almaktadır. Alt karakteristiklerin her biri arayüzden seçilebilme özelliğine sahiptir ve seçildiğinde o alt karakteristik özelinde elde edilen sonuçlar gösterilmektedir. Şekil 5.13'te görüldüğü gibi bu grafik de dikey çubuk grafiği olarak sunulmuştur ve puan bakış açısına göre sonuçların gösterildiği grafikte aynı özellikleri taşımaktadır.



Şekil 5.13: Niteliksel analiz anketi alt karakteristik sonuçları

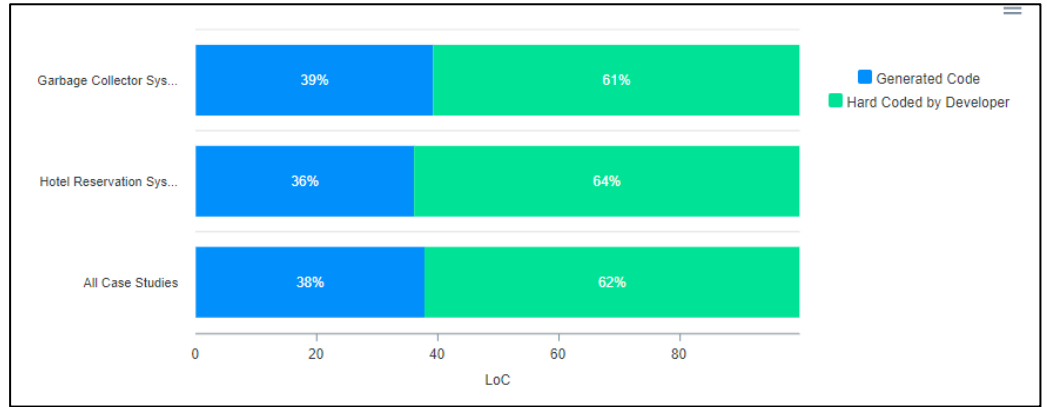
Dil değerlendirmesi ile ilgili son bölüm ise AgentDSM-Eval web aracında durum çalışmaları analizi başlığı altında sunulmaktadır. Bu bölüm de 2 alt bölüme ayrılarak gösterilmektedir. İlk bölüm değerlendiricilerin durum çalışması için MAS geliştirirken elde ettiği modelleme çıktılarının analiz sonuçlarını sunmaktadır. Şekil 5.14'te kullanıcı arayüzü görülmektedir. Modelleme ve tasarım çalışması sonucu elde edilen modelleme çıktısı dilin elemanlarının kullanımları ile ilgili verileri sağlamaktadır. Bu verilerden hareketle o dilin elemanlarının kullanım kıyaslamaları yatay çubuk grafiği ile sunulmaktadır. Yatay çubuk grafiğinin dikey eksen (y eksen) dilin elemanlarını gösterirken, yatay eksen (x eksen) ise bu elemanların kullanım sayılarını göstermektedir. Dilin elemanlarının kullanım sayılarını daha spesifik olarak inceleyebilmek için her bir durum çalışması için ayrı ayrı sonuçlar çubuk grafiğinde sunulmuştur. Tüm durum çalışmalarını içeren sonuç da aynı şekilde çubuk grafiğinde gösterilmektedir. Diğer çubuk grafiği gösterimlerinde olduğu gibi bu grafikte de sağda yer alan menüden istenilen durum çalışması gösterilip/gizlenebilmektedir.



Şekil 5.14: Modelleme çıktıları sonuçları

Durum çalışmaları değerlendirmesinin ikinci bölümünde kod üretim performansı ile ilgili sonuçlar görüntülenmektedir. Kod üretimi performansı gösterimi için Şekil 5.15'te gösterildiği gibi yatay çubuk grafiği kullanılmıştır. Çubuk grafiğinin yatay eksen (x eksen) değerlendirici tarafından geliştirilen ve

MAS DSML tarafından otomatik üretilen kod satırı yüzdesini, dikey eksen (y-ekseni) ise durum çalışmalarını göstermektedir.



Şekil 5.15: Kod üretimi performansı sonuçları

## 6. DEĞERLENDİRME ARACININ KULLANIMI ÖRNEĞİ

Bu bölümde tez çalışmasında geliştirilen AgentDSM-Eval web aracının kullanımı örnek bir MAS DSML'inin değerlendirilmesi üzerinden sunulacaktır. Bu çalışma için daha önce Bölüm 2.4.'te anlatılan Prometheus AOSE metodolojisinin uygulanması için bir DSML konumunda olan PDT kullanılmıştır. Bölümün devamında öncelikle PDT'nin üstmodel analizi anlatılmıştır. Ardından AgentDSM-Eval'in Değerlendirme Çerçevesi bölümünde tanıtılan çoklu durum çalışması üzerinden AgentDSM-Eval aracı kullanılarak PDT DSML'inin değerlendirmesinin nasıl gerçekleştirildiği sunulacaktır. Son olarak elde edilen değerlendirme sonuçları incelenecektir.

### 6.1 Üstmodel Analizi

AgentDSM-Eval aracında bir MAS DSML'ini değerlendirmek için gerçekleştirilmesi gereken ilk görev, söz konusu dil ile ilgili bilgilerin sisteme girilerek dilin web aracında dilin görüntülenmesini ve değerlendirici grubun web aracı üzerinden dile erişmesini sağlamaktır. Dil ile ilgili genel bilgilere ek olarak dilin üstmodelinde yer alan dil elemanlarının sisteme girilmesi gerekmektedir. PDT'nin üstmodel dokümanı "parse" edilebilir bir formatta bulunmamaktadır. İlgili üstmodel PDT'nin dağıtım web sayfasında sadece .pdf formatında bir diyagram olarak sunulmuştur.

Bu çalışmada öncelikle PDT üstmodelinin üstmodel pdf dökümanı üzerinden PDT dilinin dil elemanları belirlenmiştir. Bu çalışma sonucunda 11 dil elemanı tespit edilmiştir. Bu dil elemanları Actor, Action, Percept, Role, Data, Goal, Scenario, Agent, Message, Capability ve Plan'dır. Üstmodel analizi için gerekli aşama belirlenen bu dil elemanları ile FAML üstmodelinde tasarım zamanı ve çalışma zamanı açısından belirlenen kavramları anlamsal olarak eşleştirmektir. Bu eşleştirme sonuçları sisteme girilerek dilin web aracı üzerinden görüntülenmesi ve değerlendirmeye açık olması sağlanmaktadır. PDT için gerçekleştirilen üstmodel analizi sonuçları daha sonra Değerlendirme Sonuçları bölümünde web aracı görüntüleri de kullanılarak ayrıntılandırılacaktır.

### 6.2 Değerlendirme Süreci

Tezin 4. bölümünde anlatılan değerlendirme çerçevesine uygun bir şekilde 2 farklı problem için MAS geliştirmeyi içeren durum çalışmalarında PDT dilinin gönüllü değerlendiriciler tarafından kullanımı sağlanmıştır. Değerlendirmeyi

gerçekleştiren grup Ege Üniversitesi Uluslararası Bilgisayar Enstitüsü'nde (UBE) Bilgi Teknolojileri alanında lisansüstü eğitim alan kişiler arasından seçilmiştir. Bu kişilerin %62'si 22-25 yaş aralığında yer alırken, kalan %38'i ise 26 yaşından büyüktür, ayrıca değerlendirme grubunda 5 Türk vatandaşı, 1 Libya, 1 Irak ve 1 Somali vatandaşı yer almaktadır.

Değerlendirme grubunda yer alan tüm kişiler lisansüstü eğitimlerine yüksek lisans veya doktora seviyesinde devam etmektedir. Ayrıca tamamı MAS ve etmen modelleme dilleri konusunda UBE'de aldıkları "UBI 622 Çok-etmenli Sistemler" doktora dersi kapsamında ön bilgiye sahip olarak değerlendirme sürecine katılmışlardır. Ancak değerlendiricilerden hiç biri değerlendirilecek olan PDT dilini daha önce kullanmamıştır.

Değerlendirme süreci öncesinde bu gruba 60 dk.'lık bir PDT eğitimi verilmiş ve ardından 1 hafta süre ile üzerinde çalışarak araç ile ilgili deneyim kazanabilecekleri bir mini proje geliştirmeleri sağlanmıştır.

### 6.2.1 Seçilen durum çalışmaları

Değerlendirme süreci 2 farklı oturumda 2 farklı senaryo ile gerçekleştirilmiştir. Bu oturumların ilkinde "Çöp Toplama Sistemi" projesi, ikinci oturumda ise "Otel Rezervasyon Sistemi" projesi grup tarafından gerçekleştirilerek değerlendirilmiştir. Çöp Toplama Sistemi durum çalışması daha az gereksinim içeren, daha az sayıda etmen elemanı kullanılarak gerçekleştirilebilecek basit bir yapıda kurgulanırken, Otel Rezervasyon Sistemi durum çalışması nispeten daha karmaşık ve zorluk derecesi daha yüksek bir şekilde kurgulanmıştır. Bu şekilde değerlendirme sürecinde farklı zorluk derecelerinde değerlendiricilerin yaşadıkları deneyimleri gözlemlemek amaçlanmıştır. Seçilen durum çalışmalarının içeriği ve beklenenler aşağıdaki açıklanmıştır.

**1.Çöp Toplama Sistemi:** Belirli bir ortamdaki farklı tiplerdeki çöplerin toplanmasını amaçlayan bir MAS'ın analiz, tasarım ve uygulaması yapılacaktır. Sistemin analiz ve tasarımı PDT aracı kullanılarak gerçekleştirilecek; JACK kodları otomatik üretilerek proje tamamlanacaktır. Sistemde 2 tip etmen bulunacaktır. Yönetici etmen ortamda bulunan çöpler (plastik, kâğıt ve cam olmak üzere 3 tip çöp olacaktır.) hakkında Toplayıcı etmenlere mesaj atacaktır. Toplayıcı etmenler ise hangi çöpu toplayabileceklerini ifade eden kanılarına göre

Yönetici etmene ilgili çöpü toplayabileceğini veya toplamayacağını belirten bir cevap mesajı gönderecektir. Çöpü toplayabileceği veya toplamayacağı cevapları iki ayrı plan yardımıyla Yönetici etmene gönderilecektir. Eğer bir çöpün toplanabileceği mesajı gelmiş ise Yönetici etmen ilgili çöpün mesajı yollayan etmen tarafından toplanmış olduğunu varsayacaktır. Ortamdaki çöpler Yönetici etmenin çalışmaya başlaması sırasında statik olarak oluşturulabilir.

**2. Otel Rezervasyon Sistemi:** Farklı şehirlerde yer alan farklı tipteki oteller için rezervasyon senaryolarını gerçekleştirmeyi amaçlayan bir MAS'ın analiz, tasarım ve uygulaması yapılacaktır. Sistem JACK platformunda çalışacaktır. Sistemin analiz ve tasarımı PDT aracı kullanılarak gerçekleştirilecek; JACK kodları otomatik üretilerek proje tamamlanacaktır. Sistemde kullanıcıları temsil etmek için Customer Agent, otelleri temsil etmek için Hotel Agent ve bu iki etmeni buluşturmak için Matchmaker Agent yer almalıdır. Programın başlamasıyla beraber öncelikle Hotel Agent'lar temsil ettiği otelin adını, lokasyonunu (İzmir, İstanbul vb.) ve yıldız sayısını Matchmaker Agent'a bildirerek kaydolmalıdır. Customer Agent ise Matchmaker Agent'tan lokasyon, yıldız sayısı ve maksimum ücret gibi bilgilerle uygun otelleri talep etmelidir. Matchmaker Agent gelen talebe uygun olan etmenlerin bilgilerini Customer Agent'a döndürerek kullanıcılar ile otellerin iletişimi için uygun ortamı oluşturmalıdır. Customer Agent uygun otel listesinden hareketle otelleri sorgulayarak kendisi için en uygun oteli aramalıdır. Contract Net Protokolü'ne göre otel etmenleri önceden tanımlanan bir süre içerisinde (ör: 30 sn) bu sorgulara cevap verebilir veya vermeyebilir. Customer Agent gelen cevaplardan hareketle kendisi için en hesaplı olan otelden oda rezervasyonunu gerçekleştirmelidir. Sadece tek bir otel etmeni olumlu cevap dönerse bu otele rezervasyon otomatik olarak gerçekleştirilmeli; birden fazla etmeden olumlu cevap gelmesi durumunda ise en ucuz olan teklif kabul edilerek rezervasyon yapılmalıdır.

Değerlendiriciler yukarıdaki durum çalışması metinlerini göz önünde bulundurarak laboratuvar ortamında durum çalışmalarının analiz tasarım ve gerçekleştirim aşamalarını tamamlamış ve ardından AgentDSM-Eval web aracını kullanarak değerlendirme sürecine dahil olmuşlardır.



## 6.3 Değerlendirme Sonuçları

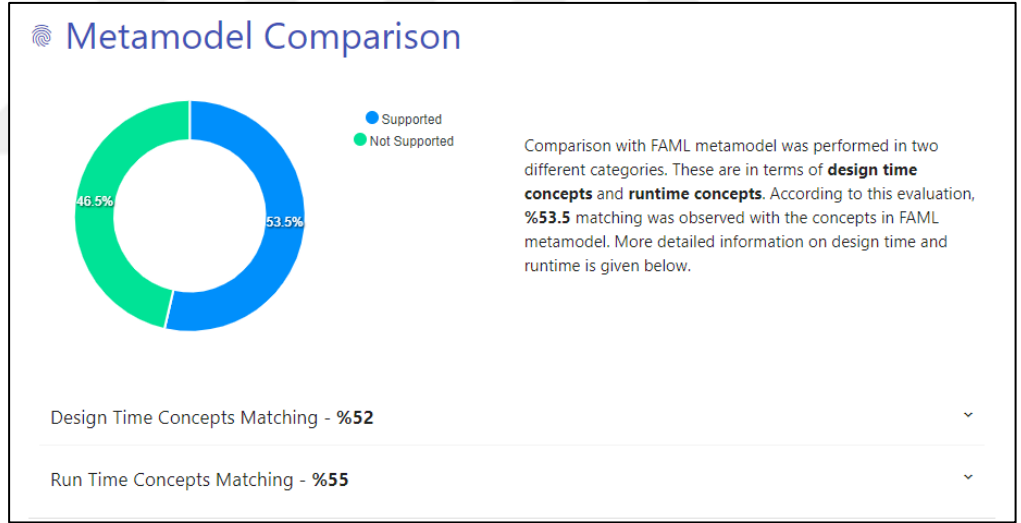
PDT DSML'i ile ilgili yapılan değerlendirme sonuçları niceliksel ve niteliksel değerlendirme sonuçları olmak üzere 2 ana başlıkta sunulacaktır.

### 6.3.1 Niceliksel değerlendirme sonuçları

Niceliksel değerlendirme sonuçları dilin üstmodeli ve durum çalışmaları üzerinden elde edilen sayısal verilerin analizidir.

#### 6.3.1.1 Referans üstmodel kıyaslama sonuçları

Referans üstmodel olarak seçilen FAML üstmodeli ile PDT üstmodelinin kıyaslama sonuçları Şekil 6.1'de görülmektedir. Tasarım zamanı ve çalışma zamanı kavramları açısından kıyaslamamanın genel sonucu bu iki üstmodel arasında %53.5'luk bir eşleşme saptanmıştır. Bir diğer deyişle FAML üstmodelindeki kavramların %53.5'u PDT üstmodelinde desteklenmektedir.



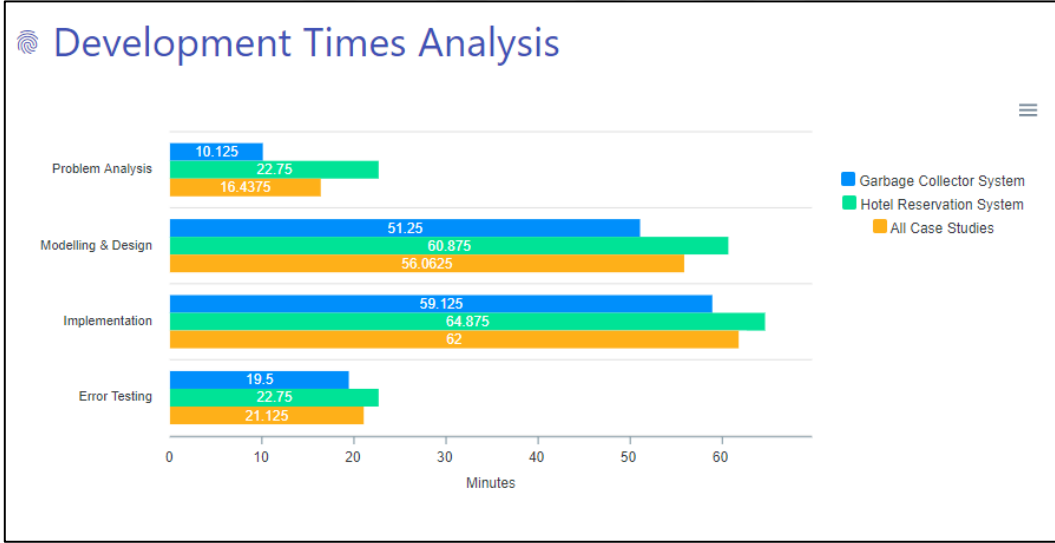
Şekil 6.1: PDT üstmodelinin referans üstmodeli ile kıyaslama sonuçları

Tasarım zamanı açısından sonuçları incelediğimizde desteklenme oranı %52, çalışma zamanı dil elemanları açısından ise %55 olarak hesaplanmıştır. Desteklenmeyen dil elemanları ve tanımlamaları tasarım zamanı açısından Ek 1'de, çalışma zamanı açısından Ek 2'de görülmektedir.

### 6.3.1.2 Durum çalışması analiz sonuçları

Durum çalışmaları ile ilgili analiz sonuçları geliştirme süreleri ve geliştirme çıktılarına dair sonuçları göstermektedir. Sonuçlar AgentDSM-Eval aracına girilen girdiler üzerinden otomatik olarak kullanıcılara gösterilmektedir.

Geliştirme süreleri ile ilgili sonuçlar Şekil 6.2’de görülmektedir. Çöp Toplama Sistemi ile ilgili sonuçlara bakılacak olursa değerlendiricilerin yaklaşık ortalama 10 dakikada problem analizi ve 51 dakikada modelleme aracını kullanarak modelleme ve tasarım yaptıkları görülmektedir. PDT aracının modelden ürettiği JACK projesi kodlarının tamamlanması ile yaklaşık ortalama 59 dk’da çöp toplama sisteminin uygulama süreçleri bitmiştir. Değerlendiriciler ardından ortalama 19,5 dakikalık bir test ve hataların düzeltilmesi işlemi gerçekleştirmişlerdir. Otel Rezervasyon Sistemi için geliştirme sürelerine baktığımızda ise yaklaşık olarak 23 dakikada problem analizi yapıldığı, 60 dakikada modelleme ve tasarımın tamamlandığı, 65 dakikada gerçekleştirimin tamamladığı ve 23 dakikada ise test ve hata düzeltme gerçekleştirildiği görülmektedir.



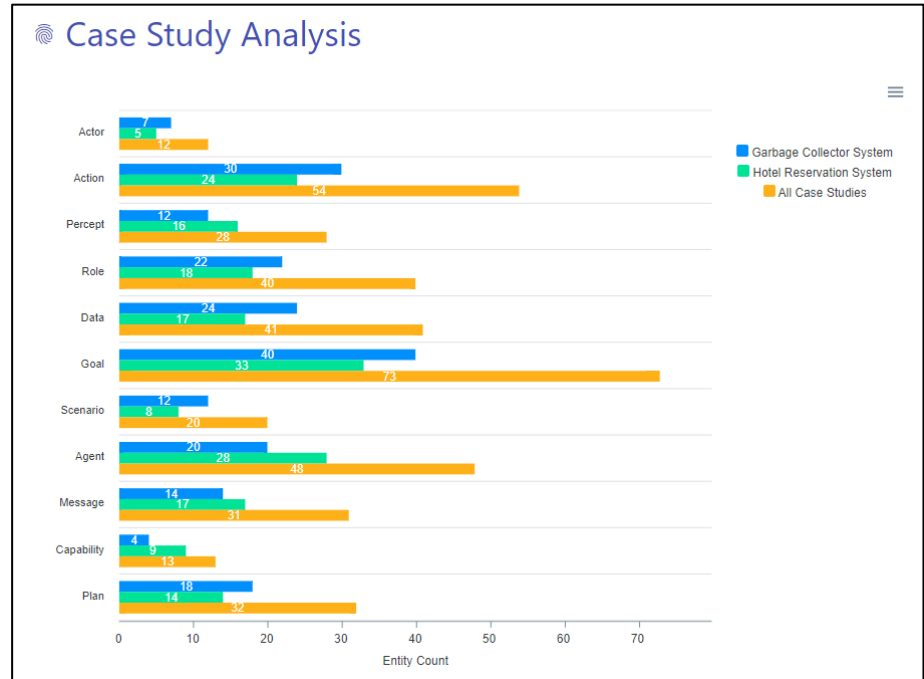
Şekil 6.2: PDT geliştirme süreleri sonuçları

Geliştirme süreleri için tüm durum çalışmaları açısından ortalama yaklaşık 16 dakikalık problem analizi, 56 dakikalık modelleme ve tasarım, 62 dakika gerçekleştirim ve yaklaşık 21 dakikalık test işlemleri gerçekleştirilmiştir. Modelleme süresi modelleme aracının kullanım kolaylığı ile ilgili, gerçekleştirim süresi ise kod üretim başarısı ile ilgili geri bildirimler içermektedir. Bu açıdan bakıldığında PDT aracı gelecek versiyonlarda modelleme ve gerçekleştirim

sürelerini daha da azaltacak kullanıcı arayüzü ve kod üretimi yeniliklerine gidebilir. AgentDSM-Eval aracı kullanılarak benzer durum çalışmaları yeni versiyonlarda tekrar edilip gerçekleştirilen çalışmaların başarıları ölçülebilir.

Geliştirme çıktıları ise 2 farklı sonuç oluşturmaktadır. PDT aracı ile geliştirilen MAS modeli AgentDSM-Eval aracında analiz edilerek PDT dil elemanlarının kullanım sıklığı kıyaslanmaktadır. Tüm değerlendiricilerin model dökümanları analiz edildiğinde dil elemanları ile ilgili elde edilen sonuç Şekil 6.3’de görülmektedir.

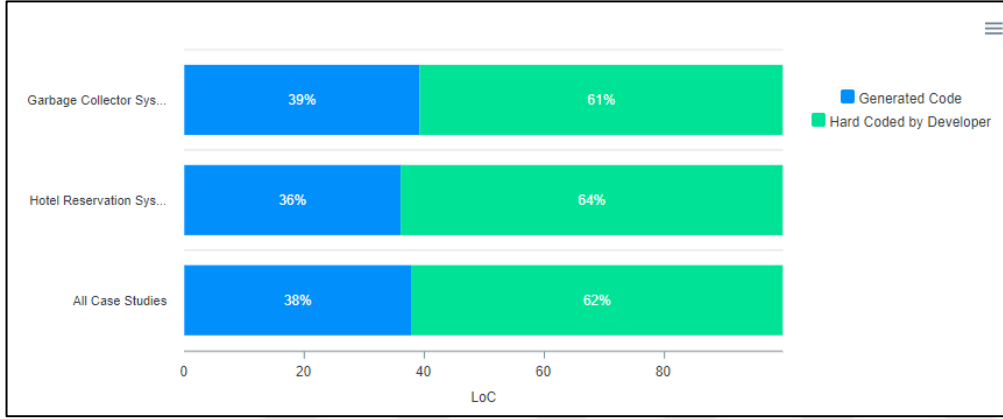
Çöp toplama sistemi durum çalışması özelinde sonuçlara bakılacak olursa Goal ve Action dil elemanlarının diğer elemanlara göre daha yoğun olarak kullanıldığı görülmektedir. Buna karşın Capability ve Actor dil elemanları ise diğer elemanlara göre oldukça az kullanılmıştır. Otel Rezervasyon sistemi açısından sonuçlar incelendiğinde; Goal, Agent ve Action dil elemanlarının yoğunlukla kullanıldığı, Actor, Scenario ve Capability dil elemanlarının ise nispeten az kullanıldığı görülmektedir. Öte yandan dil elemanlarının kullanımı ile ilgili sonuçlara durum çalışması özelinde bakmak yanıltıcı sonuçlara da sebep olabilmektedir. Durum çalışmasının zorluğu veya gereklilikleri göz önünde bulundurularak sonuçlar yorumlanmalıdır. Dil geliştiricileri sonuçları değerlendirerek az kullanılan veya ihtiyaçtan fazla kullanılan dil elemanlarını tespit edip dilin gelecek versiyonlarında bu yönde bir güncelleme yapabilirler.



Şekil 6.3: PDT dil elemanları kullanım sıklığı sonuçları

Tüm durum çalışmaları açısından dil elemanlarının kullanımına bakıldığında Goal, Action ve Agent'in yoğun olarak kullanıldığı; Capability ve Actor dil elemanlarının ise nispeten düşük bir kullanımının olduğu görülmektedir. Bu durum Capability ve Actor'ün MAS geliştiriciler için yeterince açık ve anlaşılır olmadığı sonucunu göstermektedir.

Geliştirme çıktılarının ortaya koyduğu bir diğer sonuç ise kod üretim performansı ile ilgilidir. Her bir durum çalışması ile ilgili ve tüm durum çalışmalarının ortalama kod üretim performansları Şekil 6.4'te görülmektedir.



Şekil 6.4: PDT dilinde kod üretim performansı sonuçları

Çöp toplama sistemi için PDT dili değerlendiricilerin modelleme çalışmaları üzerinden %39 oranında bir kod üretim performansı göstermiştir. Otel Rezervasyon Sistemi için ise bu oran %36 olarak tespit edilmiştir. Tüm durum çalışmaları açısından bakılacak olursa ortalama kod üretim performansı yüzde %37.5 olarak ölçülmüştür. Bu sonuçlar PDT aracının kod üretim performansının takibini kolaylaştırmakta; gelecek versiyonlar için benzer durum çalışmaları ile kod üretim performansının takip edilebilir olmasını sağlamaktadır.

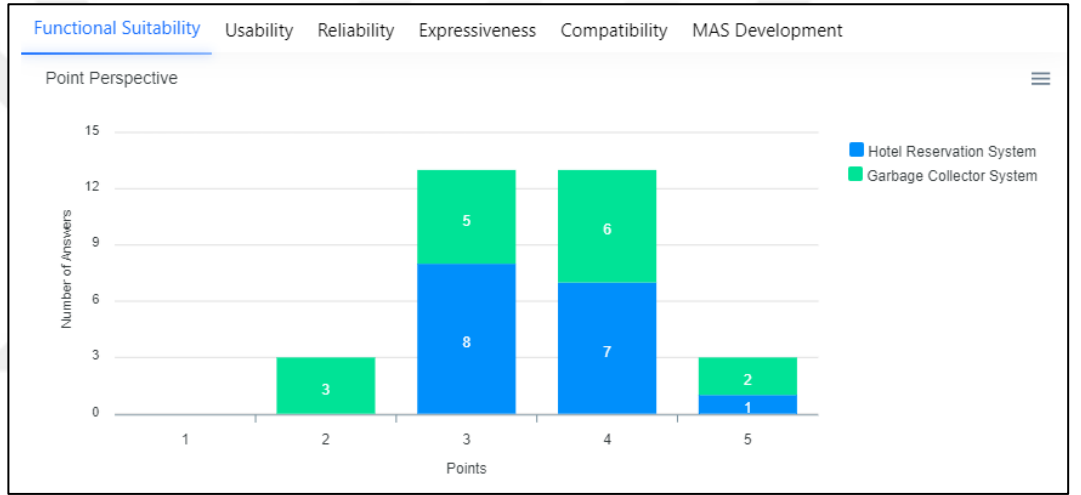
### 6.3.2 Niteliksel değerlendirme sonuçları

PDT aracının niteliksel analizi değerlendiricilerin durum çalışmalarını gerçekleştirdikten sonra cevapladıkları değerlendirme anketi sonuçları üzerinden elde edilmektedir. Değerlendirme çerçevesinin tanıtıldığı Bölüm 4'te belirtildiği gibi niteliksel analiz için 6 farklı karakteristik belirlenmiştir. Bu karakteristikler için farklı alt karakteristikler tanımlanmış ve bu tanımlanan alt karakteristikler için değerlendiricilerden 1 ile 5 arasında bir başarı puanı vermesi istenmiştir. Her bir karakteristik için AgentDSM-Eval aracı içerisinde hesaplanan puanlar ve

yorumlamaları aşağıda verilmiştir. Sonraki altbölümde ise genel bir değerlendirme yapılmaktadır.

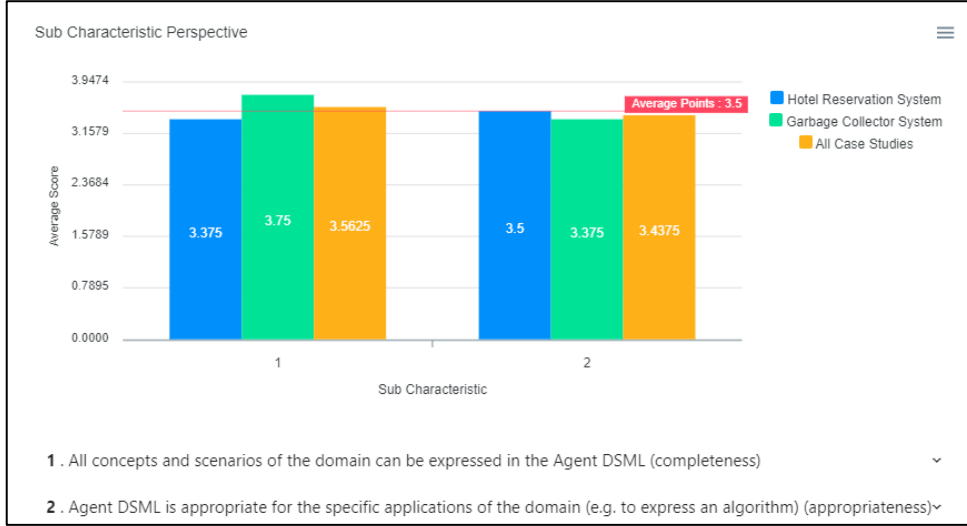
### 6.3.2.1 Fonksiyonel uygunluk değerlendirme sonuçları

Değerlendirme sürecinde fonksiyonel uygunluk için 2 metrik açısından başarımın değerlendirilmesi istenmektedir. Bu metrikler Tablo 4.3'teki 1 ve 2 no'lu alt karakteristiklerdir. Değerlendiriciler puanlama bakış açısından bakıldığında PDT dilinin fonksiyonel uygunluğu için her iki durum çalışması toplamında hiç 1 puan vermezken, 3 cevapta 2 puan, 13 cevapta 3 ve 4 puan ve 3 cevapta 5 puan vermiştir (Şekil 6.5). Puan bakış açısından bakıldığında 3-4 puan ağırlıklı bir dağılım görülmektedir.



Şekil 6.5: PDT niteliksel değerlendirme fonksiyonel uygunluk sonuçları (puan bakış açısı)

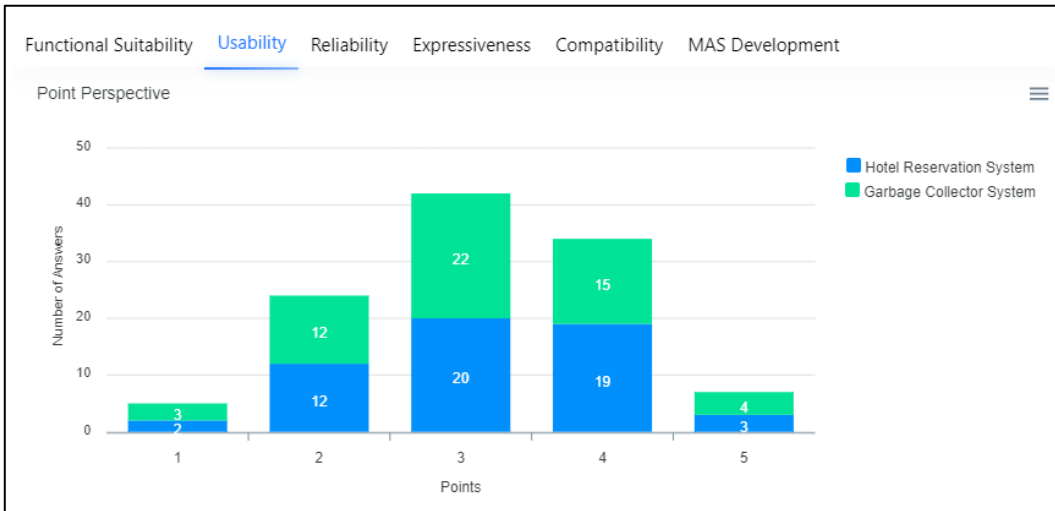
Alt karakteristikler bakış açısıyla bakıldığında Şekil 6.6'da görüldüğü gibi birinci alt karakteristik tüm durum çalışmalarında ortalama 3.56 puan alırken, ikinci alt karakteristik 3.44 puan almıştır. Bu sonuçlara göre, değerlendiriciler fonksiyonel uygunluk açısından PDT'nin alana özgü kavramları ifade yeteneğini ve spesifik uygulamalar için uygunluğunu ortalama üstü olarak değerlendirmektedir. PDT dili fonksiyonel uygunluk açısından değerlendiricilerden 5 üzerinden 3.50 puan olarak genel anlamda fonksiyonel bulunmuştur (Şekil 6.6). Uygulanan durum çalışmaları açısından bakıldığında sonuçların durum çalışmasına göre fazla bir değişkenlik göstermediği görülmüştür.



Şekil 6.6: PDT niteliksel değerlendirme fonksiyonel uygunluk sonuçları (alt karakteristik bakış açısı)

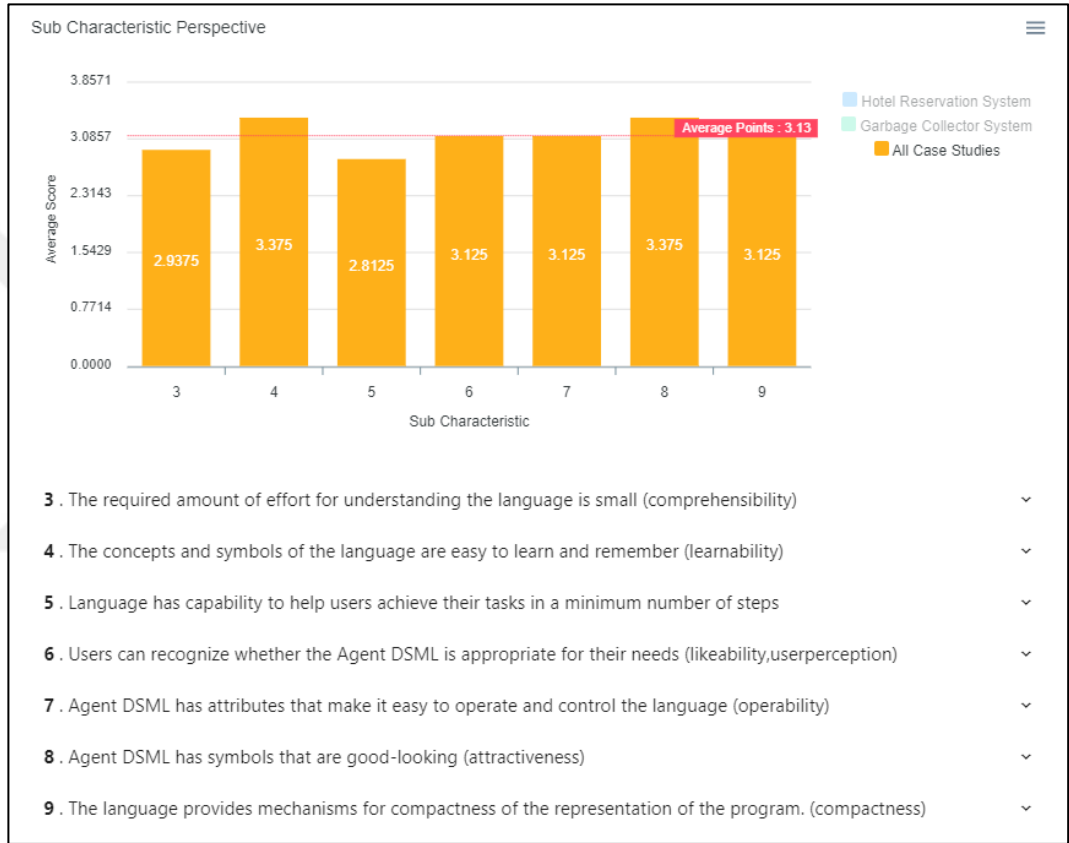
### 6.3.2.2 Kullanılabilirlik değerlendirme sonuçları

Etmen modelleme dilinin kullanılabilirlik açısından değerlendirilebilmesi için değerlendiricilerden Tablo 4.3'te no. 3 ve 9 arasında listelenen toplam 7 alt karakteristiği puanlaması istenmiştir. Öğrenilebilirlik, yoğunluk, anlaşılabilirlik gibi farklı açılardan dilin kullanılabilirliğini ölçen bu ifadeler değerlendiriciler genel olarak 2-4 puan aralığında başarı puanı vermişlerdir. Bu kategorideki metrikler 42 cevapta 3 ve 34 cevapta 4 puan olarak ortalama üstü bir kullanım deneyimi trendi göstermiştir. Tüm sorular için verilen cevapların kümülatif grafiği Şekil 6.7'de görülmektedir. 8 katılımcıdan 2 farklı durum çalışması için bu 7 alt karakteristiği puanlaması istendiğinden toplam 112 cevap elde edilmiştir.



Şekil 6.7: PDT niteliksel değerlendirme kullanılabilirlik sonuçları (puan bakış açısı)

Şekil 6.8’de ise alt karakteristikler açısından ortalama puanlara ait grafik görülmektedir. 3. ve 5. ifadeler ankette ortalama (3 puan) altında puan olarak PDT’nin ilgili özellikler açısından başarısız olduğu sonucunu ortaya koymaktadır. Bu metriklerin genel sorguladığı özellikler olan anlaşılabilirlik ve minimum adımda hedefe ulaşma konularında PDT dilinin gelişmeye ihtiyacı olduğu sonucuna ulaşılmaktadır. Diğer metriklerde ise 3 üzeri puan alınarak ortalama üstü bir sonuca erişilmiş ve kullanılabilirlik açısından PDT 3.13 puan ortalamasına sahip olmuştur.



Şekil 6.8: PDT niteliksel değerlendirme kullanılabilirlik sonuçları (alt karakteristik bakış açısı)

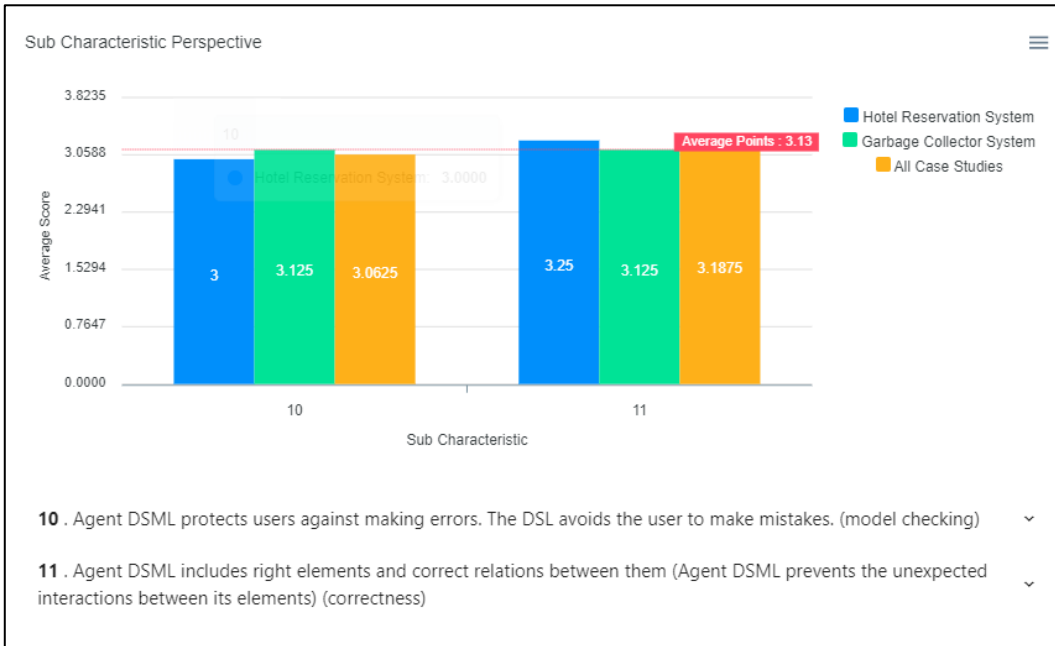
### 6.3.2.3 Güvenilirlik değerlendirme sonuçları

PDT’nin doğruluk ve model kontrolü açısından değerlendirilmesi için Tablo 4.3’te listelenen 10. ve 11. metrikler değerlendiricilere deneyimlerini puanlamaları için yöneltilmiştir. Değerlendiricilerin çoğunluğu güvenilirlik açısından PDT dilini ortalama üstü puanlamışlardır. Verilen cevaplar sonucu oluşan cevap sayısı grafiği Şekil 6.9’daki gibidir. 23 cevabın 3 ve üzeri başarı puanı ile güvenilirliği ortalama üstü değerlendirdiği sonuçlarda, 9 cevapta ise ortalama altı bir değerlendirme gerçekleşmiştir.

Güvenilirliğe alt karakteristikler açısından bakıldığında ise etmen modelleme dilinin hata kontrolü yeteneği (Tablo 4.3 – 10. metrik) ve dil elemanları arası beklenmedik ilişkileri önleme yeteneği (Tablo 4.3 – 11. metrik) ortalamanın üzerinde bir başarı puanı alarak sırası ile 3.06 ve 3.19 puan almışlardır (Şekil 6.10). Hata kontrolü yeteneği ile ilgili metriğin ortalamanın çok az üstünde olması gelecekte durum çalışması veya değerlendirici sayısı arttıkça bu karakteristiğin ortalama altına düşebileceği riskini göstermektedir. Bu nedenle dil geliştiricileri tarafından bu karakteristik özelinde gelecekte çalışma yapılması gerekliliği öngörülmektedir.



Şekil 6.9: PDT niteliksel değerlendirme güvenilirlik sonuçları (puan bakış açısı)

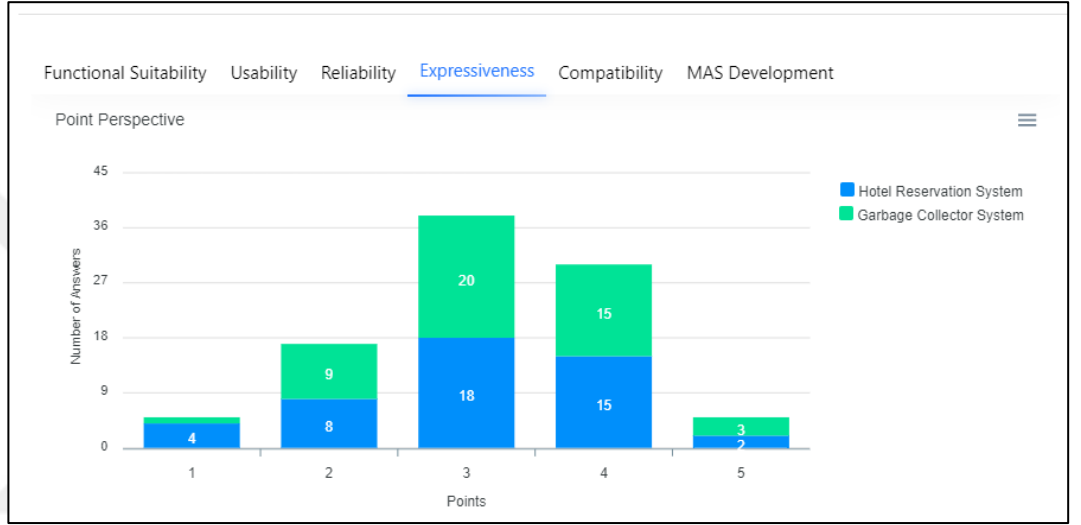


Şekil 6.10: PDT niteliksel değerlendirme güvenilirlik sonuçları (alt karakteristik bakış açısı)



### 6.3.2.4 Anlamlılık değerlendirme sonuçları

PDT DSML'ini anlamsal açıdan değerlendirmek için değerlendiricilere Tablo 4.3'te no. 12 ve 17 arasında listelenen toplam 6 alt karakteristik ile ilgili görüşleri sorulmuştur. 2 durum çalışması için 6 metrik 8 kişi tarafından cevaplanmış ve toplam Anlamlılık açısından yöneltilen bu ifadelerle 38 tane 3 puan ve 30 tane 4 puan ağırlığında cevap verilerek PDT ortalama üstü olarak değerlendirilmiştir. 5'er cevaplama az sayıda kişi ise PDT'yi bu karakteristik için çok iyi ve çok kötü olarak değerlendirmektedir (Şekil 6.11).



Şekil 6.11: PDT niteliksel değerlendirme anlamlılık sonuçları (puan bakış açısı)

Bu karakteristiğe alt karakteristikler bakış açısından bakılacak olursa tüm metrikler için ortalama 3.14 puan verildiği görülmektedir (Şekil 6.12). Sadece 12. ve 13. metriklerde anlamlılık açısından 3 puanın altında bir değerlendirme ile PDT'nin yetersiz olma durumu göze çarpmaktadır. Buna göre PDT'nin analizden programlamaya geçiş süreci için iyileştirmeye ihtiyacı vardır. Ayrıca modelleme kavramlarının semantiğinin tekil olması için büyük ihtimalle dil notasyonunun güncellenmesi gerekmektedir.

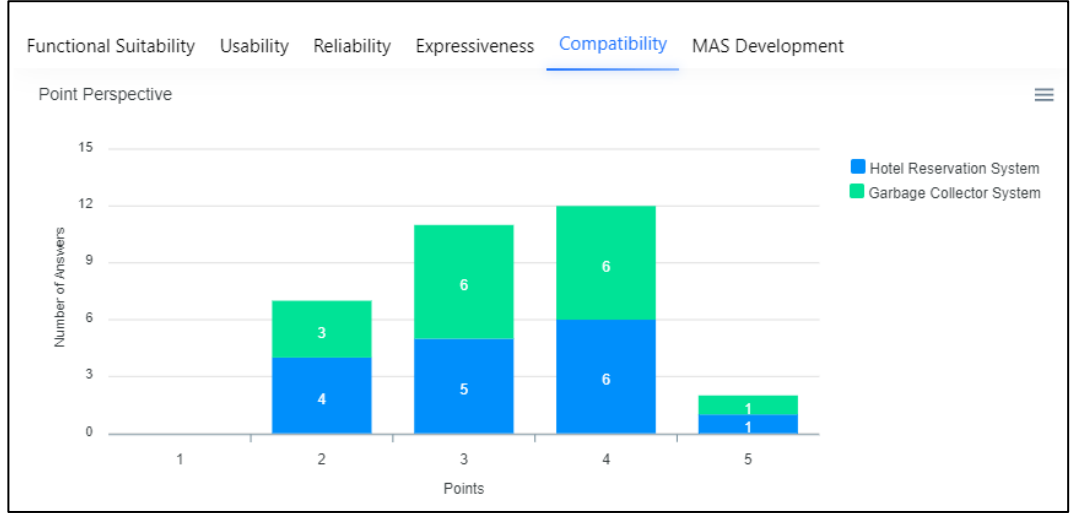


Şekil 6.12: PDT niteliksel değerlendirme anlamlılık sonuçları (alt karakteristik bakış açısı)

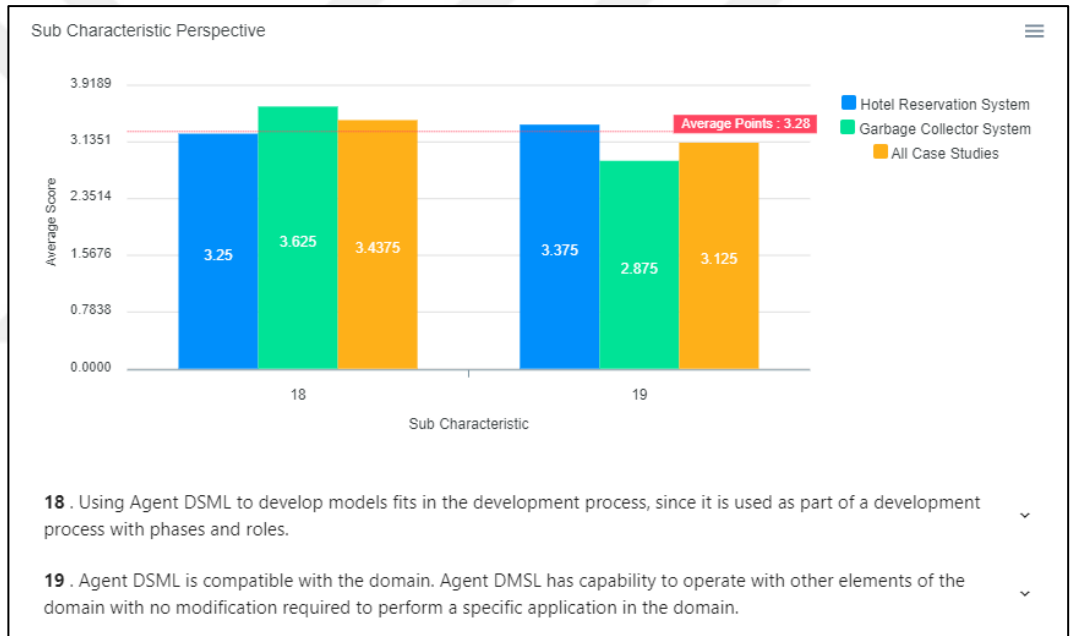
### 6.3.2.5 Uygunluk değerlendirme sonuçları

PDT'nin MAS alanına ve geliştirme sürecine uygunluğu Tablo 4.3'te yer alan 18. ve 19. metrikler için değerlendiricilerin verdiği yanıtlara göre incelenmiştir. Uygunluk için verilen puanlar sonucunda oluşan puan bakış açısından sonuç grafiği Şekil 6.13'de görülmektedir. İki durum çalışması için de hiç bir değerlendiriciden herhangi bir metrik için 1 puan alınmazken, sadece 2 cevapta ise PDT dilinin uygunluk karakteristiği çok başarılı bulunmuştur. Genel dağılımda değerlendirici grup 2-4 puan aralığında bir değerlendirme yaparken, yönelim ortalama üstü olarak seyretmiştir.

Alt karakteristikler için uygunluk sonuçlarına bakıldığında ise uygunluk karakteristiğinin PDT için ortalama puan değeri AgentDSM-Eval aracında 3.28 olarak hesaplanmıştır (Şekil 6.14). Bu sonuçla PDT'nin desteklediği geliştirme sürecinin MAS modelleme için değerlendiriciler tarafından orta üstü derecede uygun bulunduğu söylenebilir. Dilin yine değerlendiriciler tarafından etmen alanı ile de uyumlu bulunduğu ortaya konmaktadır.



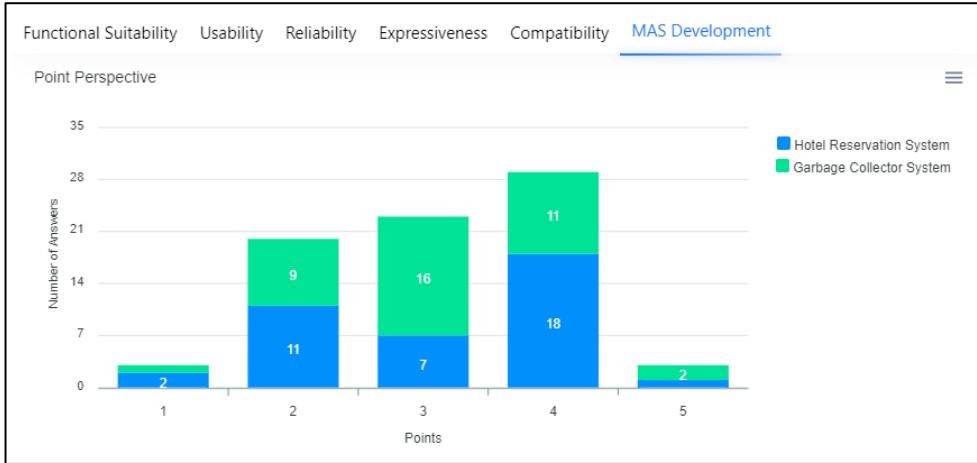
Şekil 6.13: PDT niteliksel değerlendirme uygunluk sonuçları (puan bakış açısı)



Şekil 6.14: PDT niteliksel değerlendirme uygunluk sonuçları (alt karakteristik bakış açısı)

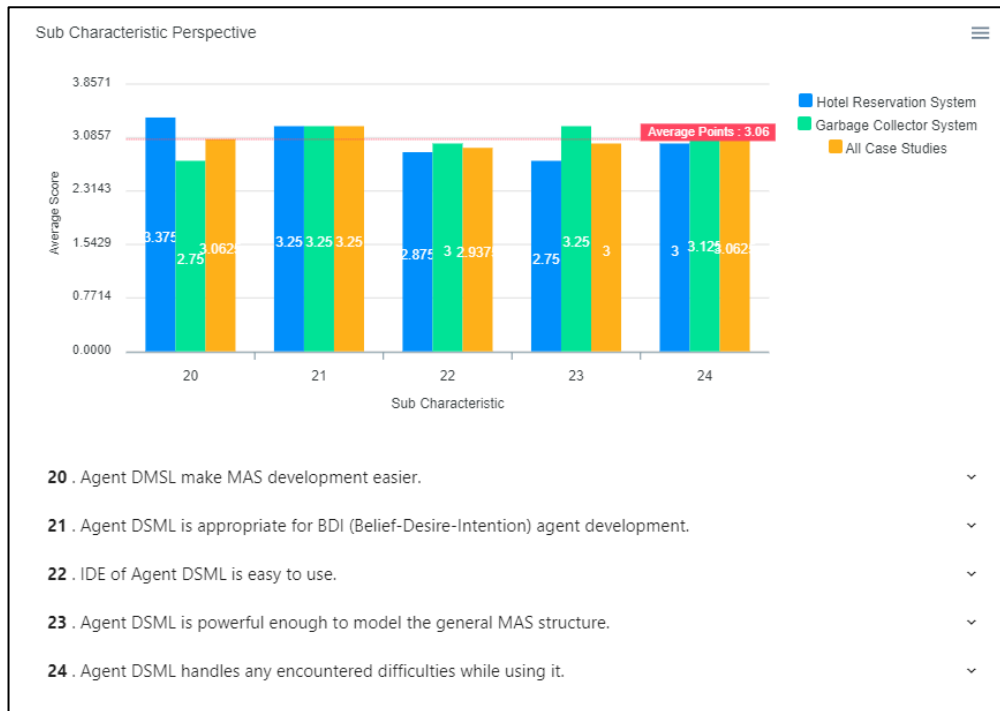
### 6.3.2.5 MAS geliştirme değerlendirme sonuçları

PDT'nin MAS geliştirmeyi kolaylaştırması, genel MAS desteği, BDI etmenlerini geliştirme, vb. özellikler açısından niteliksel değerlendirmesi için değerlendiricilere Tablo 4.3'te no. 20 ve 24 arasında listelenen toplam 5 metrik sorulmuştur. Değerlendiriciler bu karakteristik için puan bakış açısında ağırlıklı olarak 2-4 puan aralığında cevaplar verirken az sayıda cevap 1 ve 5 puan ile uç noktalarda yer almaktadır. 2-4 puan aralığında ise ortalama üstü bir eğilim mevcuttur. İlgili puan grafiği Şekil 6.15'te yer almaktadır.



Şekil 6.15: PDT niteliksel değerlendirme MAS geliştirme sonuçları (puan bakış açısı)

Her bir metrik için hesaplanan ortalama puanlar Şekil 6.16'daki grafikte verilmiştir. Alt karakteristikler açısından 3.06 ortalama puan oluşmaktadır. IDE kullanımı ve modelleme gücü açısından değerlendirme puanı (Tablo 4.3 - 22 ve 23. metrik) diğer metriklere göre ortalamanın altında kalmaktadır. Genel olarak bu kategoride PDT 3 üzeri bir puan elde ettiği için geliştiriciler tarafından MAS geliştirmeyi destekleyen bir DSML olarak belirlenmiştir. Şekil 6.16'da 21. soruya verilen ortama puan (3.25) incelendiğinde; geliştiricilerin özellikle PDT'yi BDI etmen modellemeyi ve geliştirmeyi destekleme açısından bu kategorideki diğer özelliklere göre daha uygun buldukları görülmektedir.

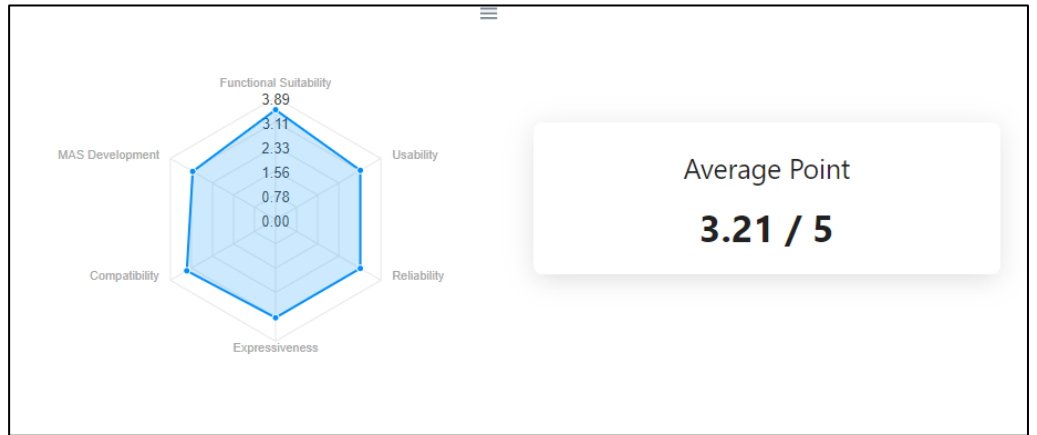


Şekil 6.16: PDT niteliksel değerlendirme MAS geliştirme sonuçları (alt karakteristik bakış açısı)

### 6.3.3 Çıkarımlar ve sonuç

AgentDSM-Eval değerlendirme çerçevesi prensipleri doğrultusunda web aracını kullanarak PDT değerlendirmesi gerçekleştirilmiş; niteliksel ve niceliksel açıdan sonuçlar elde edilmiştir. Bu sonuçlara göre dilin geliştiricilerinin FAML üstmodelindeki kavramlardan hareketle üstmodeli MAS birinci derece varlıklarını kapsayacak şekilde geliştirmeleri gerekmektedir. Yüzde 53.5'lik dil elemanı desteği oranının yükselmesi dilin MAS geliştiricileri tarafından daha kolay ve hızlı öğrenilmesini ve daha fazla benimsenmesini sağlayacaktır. Ayrıca bu sonuçlar dilin kod üretim performansı, kullanıcı arayüzünün kolaylaştırılması, dil elemanlarının anlaşılabilirliğinin ve kullanılabilirliğinin artırılması gibi konularda strateji geliştirilmesinin önünü açmaktadır.

Niteliksel analiz için metriklere verilen cevaplara genel olarak bakılacak olursa; PDT dili ile ilgili 6 karakteristik ve bu 6 karakteristikten elde edilen ortalama genel puan AgentDSM-Eval aracı içerisinde Şekil 6.17'deki gibi gösterilmektedir. Burada yer alan radar grafiği üzerinde ilgili karakteristik seçilerek karakteristiğin ortalama puanı görülebilmekte, radar grafiğin altında yer alan menüden karakteristikler seçilerek bir önceki bölümde verilen detaylı grafikler ve sonuçlar görüntülenebilmektedir. PDT değerlendiricilerden ortalama 3.21 puan almıştır. Karakteristikler açısından bakılırsa; 3.50 puan ortalaması ile fonksiyonel uygunluk kategorisi öne çıkarken, 3.06 ile çok etmenli sistem geliştirme diğer kategorilerin gerisinde kalmıştır. Genel olarak PDT 5 puanlık skalada tüm kategorilerde 3 puan almış; bir diğer deyişle göz önüne alınan niteliksel özellikler açısından ortalama bir kaliteye sahip olduğu çıkarılmasında bulunulmuştur.



Şekil 6.17: PDT niteliksel değerlendirme sonuçları

## 7. SONUÇ

Bu tezde MAS DSML'lerinin sistematik olarak niceliksel ve niteliksel açıdan çevrimiçi değerlendirilmesini destekleyen AgentDSM-Eval isimli bir web aracı sunulmuştur. AgentDSM-Eval aracı dil değerlendirmesini niceliksel ve niteliksel perspektiften çoklu durum çalışması yaklaşımı ile gerçekleştirmektedir. Tezde önerilen bu aracın kullanımı ise Prometheus/PDT isimli bir MAS DSML'inin değerlendirilmesi üzerinden örneklenmiştir.

AgentDSM-Eval aracının dayandığı değerlendirme çerçevesi MAS DSML'lerinin kullanım kolaylığı, araç zenginliği, alan hakimiyeti, etkinliği, etmen yazılım bileşenlerinin üretkenliği, vb. açılardan ve hem nitelik hem de nicelik yönünden değerlendirilmesini amaçlamaktadır.

Niceliksel değerlendirme üstmodel düzeyinde ve durum çalışmaları düzeyinde olmak üzere iki farklı açıdan ele alınmaktadır. Üstmodel düzeyinde değerlendirme için mevcut bir takım değerlendirme metriklerine ek olarak ilk kez bu tezde bir referans üstmodel ile kıyaslama yaklaşımı getirilmiştir. Referans üstmodel olarak MAS DSML'ler için genel bir üstmodel konumunda olan FAML üstmodeli tercih edilmiştir. FAML üstmodelinin tasarım zamanı ve çalışma zamanı kavramları üzerinden değerlendirilen dilin elemanları incelenmekte, desteklenen kavramlar yüzdesel olarak belirlenmektedir. Bu değerlendirme çıktısı MAS DSML geliştiricileri için üstmodellerinin kapsamını değerlendirebilecekleri bir done sağlamaktadır. Durum çalışmaları düzeyinde değerlendirme ise geliştirme süreleri ve geliştirme çıktıları olmak üzere 2 farklı değerlendirme sunmaktadır. Geliştirme süreleri problem analizi, modelleme ve tasarım, gerçekleştirme ve test-hata ayıklama sürelerini içermektedir. Geliştirme sürecinin bu aşamaları dakika cinsinden girdi olarak alınmakta ve değerlendiricilerin tamamı üzerinden ortalama süreler elde edilmektedir. MAS DSML'leri kolay ve hızlı modelleme yapılabilecek ve kod üretim performansı yüksek araçlar sunmayı hedeflemektedirler. AgentDSM-Eval aracı ile geliştirme sürelerinin analizi sayesinde modelleme ve tasarım süreleri ile geliştirme süreleri takip edilebilecek ve araç ile ilgili yapılan iyileştirmelerin etkileri süre bazında takip edilebilecektir. Geliştirme çıktıları açısından değerlendirme ise modelleme çıktısı ve kod üretim performansı olmak üzere 2 farklı durum için sonuç üretir. Modelleme çıktıları MAS geliştiricilerinin ortaya koydukları modellerde kullandıkları dil elemanlarını veri olarak sunmaktadır. Bu veri ile MAS DSML'inde hangi dil elemanları ne sıklıkla kullanılıyor analiz edilebilmektedir. Bu değerlendirme yaklaşımı daha

önce literatürde yer alan çalışmalarda bulunmamaktadır. MAS DSML geliştiricilerinin dil elemanları ile ilgili somut kullanım verilerine ulaşmasına ve buna göre MAS DSML'lerini güncelleyebilmelerine olanak sağlamaktadır. Kod üretim performansı analizi ise MAS DSML'inin otomatik kod üretim verimini ölçmeyi hedeflemektedir. Üretilen kodlar üzerine MAS geliştiricilerinin ekledikleri delta kodları inceleyerek web aracı kod üretim yüzdesini belirlemektedir.

Niteliksel değerlendirme değerlendirici grubun durum çalışması üzerinden değerlendirilmesi yapılmakta olan MAS DSML'i ile ilgili elde ettikleri deneyimi derecelendirmesine olanak tanımaktadır. Değerlendirici grup söz konusu MAS DSML'i ile yazılım geliştirmeyi deneyimlendikten hemen sonra niteliksel metriklerle ilgili bir anketi AgentDSM-Eval aracı içerisinde cevaplayarak deneyimini somut bir veri haline getirmektedir. Bu değerlendirme anketi için FQAD'da (Kahraman and Bilgen, 2015) yer alan metrikler temel alınmıştır. Bu metrikler tezde MAS DSML'lerin değerlendirilmesi için uyumlu olacak şekilde güncellenmiş ve MAS DSML'lere özgü bir metrik grubu daha eklenerek katkıda bulunulmuştur.

Tez çalışması sürecinde bir takım teknik sorunlarla da karşılaşmıştır. Literatürde çok sayıda MAS DSML'i önerilmektedir. Ancak önerilen bu MAS DSML'leri için baskın veya ortak kullanılan bir üstmodel geliştirme teknolojisi yoktur. Bir başka deyişle MAS üstmodellerinin belgelendirilmesi ve serileştirilmiş hallerinin işlenmesi için bir standart bulunmamaktadır. Bu sebeple web uygulamasına entegre edilecek genel bir üstmodel yorumlama modülü henüz mümkün olamamaktadır. Bu problemin sonucu olarak değerlendirilmek istenen diller için üstmodel analizinin uzmanlar tarafından yapılarak AgentDSM-Eval aracına eklenmesi zorunluluğu doğmuştur. Ancak bu analiz ve eklemenin her bir MAS DSML'i için sadece bir defa yapılması yeterlidir. Daha sonraki bu MAS DSML kullanılarak gerçekleştirilecek tüm durum çalışmaları için bu üstmodel entegrasyonunun tekrarına ihtiyaç duyulmamaktadır.

AgentDSM-Eval çalışması sistemde yer alan diller için sistematik değerlendirme sonuçlarını buldurmaktadır. Bu durum gelecekte sistemde yer alan dillerin kıyaslanabilmesinin altyapısını oluşturur. Böylece probleme uygun olan MAS DSML seçimi için bu dillerin kıyaslanması mümkün olacaktır. İleriye yönelik olarak AgentDSM-Eval aracına daha farklı MAS DSML'lerinin üstmodellerinin dahil edilmesi ve bu MAS DSML'leri kullanılarak gerçekleştirilen

MAS geliştirme çalışmalarına ait sonuçların yüklenmesi sağlanabilir. Böylece hem bu yeni MAS DSML'lerinin AgentDSM-Eval'in dayandığı değerlendirme çerçevesine göre sistematik değerlendirilmesi sağlanmış olur hem de yeni MAS DSML'leri için aynı çalışmalar yerine getirildikçe bu dillerin karşılaştırmalı değerlendirmeleri sağlanabilir. Ayrıca üstmodel kıyaslama için AgentDSM-Eval şu an sadece FAML üstmodelini desteklemektedir. Gelecekte, farklı genel üstmodeller eklenebilir veya sistemde yer alan dillerden seçim yaparak üstmodel kıyaslaması yapılabilir. Üstmodel kıyaslama yaklaşımı şu an sadece dil elemanları için gerçekleştirilmektedir. Üstmodellerle ilgili teknolojik kısıtların çözülmesi ile kıyaslama işlemi dil elemanları ve ilişkiler üzerinden daha derinlemesine gerçekleştirilebilir.





## KAYNAKLAR DİZİNİ

- Badica, C., Budimac, Z., Burkhard, H. D. and Ivanovic, M.,** 2011, Software agents: Languages, tools, platforms, *Computer Science and Information Systems*, 8(2), 255-298pp.
- Bellifemine, F. L., Caire, G. and Greenwood, D.,** 2007, *Developing Multi-Agent Systems with JADE*, USA, Wiley & Sons.
- Bergenti, F.,** 2014, An Introduction to the JADEL Programming Language, In *proc. 26th IEEE International Conference on Tools with Artificial Intelligence*, 974-978pp.
- Bergenti, F., Iotti, E., Monica, S. and Poggi, A.,** 2017, Agent-oriented model-driven development for JADE with the JADEL programming language, *Computer Languages, Systems & Structures*, 50, 142-158pp.
- Bernon, C., Cossentino, M., Gleizes, M. P., Turci, P. and Zambonelli, F.,** 2005, A Study of some Multi-Agent Meta-Models, *Lecture Notes in Computer Science*, 3382, 62-77pp.
- Bernon, C., Gleizes, M. P., Peyruqueou, S. and Picard, G.,** 2003, ADELFE: A Methodology for Adaptive Multi-agent Systems Engineering, *Engineering Societies in the Agents World III*, Springer, 156-169pp.
- Beydoun, G., Gonzalez-Perez, C., Henderson-Sellers, B. and Low, G.C.,** 2006, Developing and Evaluating a Generic Metamodel for MAS Work Products, *Software Eng. for Multi-Agent Systems IV: Research Issues and Practical Applications*, A. Garcia, R. Choren, C. Lucena, P. Giorgini, T. Holvoet, and A. Romanovsky, eds., 126-142pp.
- Beydoun, G., Low, G. C., Henderson-Sellers, B., Mouratidis, H., Gomez-Sanz, J. J., Pavon, J. and Gonzalez-Perez, C.,** 2009, FAML: A Generic Metamodel for MAS Development, *IEEE Transactions on Software Engineering*, 35(6), 841-863pp.
- Bordini, R.H., Hubner, J.F. and Wooldridge, M.,** 2007, *Programming Multi-Agent Systems in AgentSpeak Using Jason*, USA, John Wiley & Sons.

**KAYNAKLAR DİZİNİ (devam)**

- Challenger, M., Demirkol, S., Getir, S., Mernik, M., Kardas, G. and Kosar, T.,** 2014, On the use of a domain-specific modeling language in the development of multiagent systems, *Engineering Applications of Artificial Intelligence*, 28, 111-141pp.
- Challenger, M., Kardas, G. and Tekinerdogan, B.,** 2016, A systematic approach to evaluating domain-specific modeling language environments for multi-agent systems, *Software Quality Journal*, 24(3), 755-795pp.
- Cossentino, M. and Potts, C.,** 2002, A CASE tool supported methodology for the design of multi-agent systems, *International Conference on Software Engineering Research and Practice (SERP'02)*.
- Eclipse,** 2010, "Xtext programming framework", <https://www.eclipse.org/Xtext/>, (son erişim tarihi: Ağustos 2019).
- Eclipse,** 2013, "Graphiti - a Graphical Tooling Infrastructure", <http://www.eclipse.org/graphiti/>, (son erişim tarihi: Ağustos 2019).
- Eclipse,** 2014, "Xpand Code Generation Language", <http://wiki.eclipse.org/Xpand>, (son erişim tarihi: Ağustos 2019).
- Eclipse,** 2019, "Eclipse IDE", <https://eclipse.org/ide/>, (son erişim tarihi: Ağustos 2019)
- Faccin, J. and Nunes, I.,** 2015, BDI-agent plan selection based on prediction of plan outcomes, In proc. *International Conference on Web Intelligence and Intelligent Agent Technology*, 166-173pp.
- Faccin, J. and Nunes, I.,** 2017, A Tool-Supported Development Method for Improved BDI Plan Selection, *Engineering Applications of Artificial Intelligence*, 62, 195-213pp.
- Ferber, J. and Gutknecht, O.,** 1998, A Meta-Model for the Analysis and Design of Organizations in Multi-Agent Systems, In proc. the 3rd *International Conference on Multi-Agent Systems*, Paris, France, 128-135pp.
- FIPA,** 2002, "Agent Communication Language (ACL)", <http://www.fipa.org/>, (son erişim tarihi: Ağustos 2019).

**KAYNAKLAR DİZİNİ (devam)**

- Firestore**, 2019, “Google Firestore”, <https://firebase.google.com/> (son erişim tarihi: Ağustos 2019)
- Frankel, D.**, 2003, Model Driven Architecture: Applying MDA to Enterprise Computing: The Complete Book, Wiley Publishing, US, 328p.
- Freire, E., Junior, R. and Cortes, M.**, 2013, A modeling environment for normative multi-agent systems, In proc. 15th International Conference on Enterprise Information Systems, 451–458pp.
- Getir, S., Challenger, M. and Kardas, G.**, 2014, The formal semantics of a domain-specific modeling language for semantic web enabled multi-agent systems, International Journal of Cooperative Information Systems, 23(3), 1-53pp.
- Giunchiglia, F., Mylopoulos, J. and Perini, A.**, 2002, The Tropos Software Development Methodology: Processes, Models and Diagrams, AAMAS Conference 2002, 162-173pp.
- Goncalves, E. J. T., Cortes, M. I., Campos, G. A. L., Lopes, Y. S., Freire, E. S. S., da Silva, V. T., de Oliveira, K. S. F. and de Oliveira, M. A.**, 2015, MAS-ML2.0: Supporting the modelling of multi-agent systems with different agent architectures, Journal of Systems and Software, 108, 77-109pp.
- Gray, J., Tolvanen, J. P., Kelly, S., Gokhale, A., Neema, S. and Sprinkle, J.**, 2007, Domain-specific modeling, Handbook of Dynamic System Modeling, 500p.
- Hahn, C.**, 2008, A Domain Specific Language for Multiagent Systems, In Proceedings of the 7th Autonomous Agents and Multiagent Systems Conference (AAMAS’08), Estoril, Portugal, 233-240pp.
- Hahn, C. and Fischer, K.**, 2009, The Formal Semantics of the Domain Specific Modeling Language for Multi-agent Systems, Lecture Notes in Computer Science, 5386, 145-158pp.

**KAYNAKLAR DİZİNİ (devam)**

- Hahn, C., Madrigal-Mora, C. and Fischer, K.,** 2009, A Platform-Independent Metamodel for Multiagent Systems, *Autonomous Agents and Multi-Agent Systems*, 18(2), 239-266pp.
- Henderson-Sellers, B., Debenham, J., Tran, Q. N. N., Cossentino, M. and Low, G.,** 2006, Identification of Reusable Method Fragments from the PASSI Agent-Oriented Methodology, *Agent Oriented Information Systems III*, Springer, 95-110pp.
- Hoseindoost, S., Adamzadeh, T., Zamani, B. and Fatemi, A.,** 2017, A model-driven framework for developing multi agent systems in emergency response environments, *Software and Systems Modeling*, 1-28pp.
- Howden, N., Rönnquist, R., Hodgson, A. and Lucas, A.,** 2001, JACK intelligent agents-summary of an agent infrastructure, In proc. the 5th International Conference on Autonomous Agents, Montreal, Canada, 1-6pp.
- Kahlaoui, A., Abran, A. and Lefebvre, É.,** 2008, DSML success factors and their assessment criteria, *Metrics News* 13(1), 43-51pp.
- Kahraman, G. and Bilgen, S.,** 2015, A framework for qualitative assessment of domain-specific languages, *Software & Systems Modeling*, 14(4), 1505–1526pp.
- Kardas, G.,** 2013, Model-driven development of multi-agent systems: a survey and evaluation, *The Knowledge Engineering Review*, 28(4):479-503pp.
- Kardas, G., Bircan, E. and Challenger, M.,** 2017, Supporting the platform extensibility for the model-driven development of agent systems by the interoperability between domain-specific modeling languages of multi-agent systems, *Computer Science and Information Systems*, 14(3), 875-912pp.
- Kardas, G., Goknil, A., Dikenelli, O. and Topaloglu, N. Y.,** 2009, Model driven development of semantic web enabled multi-agent systems, *International Journal of Cooperative Information Systems*, 18(2), 261-308pp.

**KAYNAKLAR DİZİNİ (devam)**

- Kardas, G. and Gomez-Sanz, J. J.**, 2017, Special issue on model-driven engineering of multi-agent systems in theory and practice, *Computer Languages, Systems & Structures*, 50, 140-1pp
- Kardas, G., Tezel, B. T. and Challenger, M.**, 2018, Domain-specific modelling language for belief-desire-intention software agents, *IET Software*, 12(4), 356-364pp.
- Kos, T., Kosar, T., Knez, J. and Mernik, M.**, 2011, From DCOM Interfaces to Domain-Specific Modeling Language: A Case Study on the Sequencer, *Computer Science and Information Systems*, 8(2), 361-378pp.
- Lopes, Y.S., Goncalves, E.J.T., Cortes, M.I. and Freire, E.S.S.**, 2011, Extending JADE framework to support different internal architectures of agents, In proc. 9th European Workshop on Multi-agent Systems, 1–15pp.
- Molesini, A., Denti, E. and Omicini, A.**, 2005, MAS meta-models on test: UML vs. OPM in the SODA case study, *Multi-Agent Systems and Applications Iv, Proceedings*. M. Pechoucek, P. Petta and L. Z. Varga, 3690, 163-172pp.
- Omicini, A.**, 2001, SODA: societies and infrastructures in the analysis and design of agent-based systems, First international workshop, *AOSE 2000 on Agent-oriented software engineering*. Limerick, Ireland, Springer-Verlag New York, Inc., 185-193pp.
- Omicini, A., Ricci, A. and Viroli, M.**, 2008, Artifacts in the A&A meta-model for multi-agent systems, *Autonomous Agents and Multi-Agent Systems* 17(3), 432-456pp.
- Othman, S.H., Beydoun, G. and Sugumaran, V.**, 2014, Development and validation of a Disaster Management Metamodel (DMM), *Information Processing & Management*, 50(2), 235-271pp.
- Padgham, L. and Winikoff, M.**, 2003, Prometheus: A Methodology for Developing Intelligent Agents, *Agent Oriented Software Engineering III*, Springer, 174-185pp.

**KAYNAKLAR DİZİNİ (devam)**

- Padgham, L. and Winikoff, M.**, 2005, Prometheus: A practical agent-oriented methodology, Agent-oriented Methodologies. Henderson-Sellers, B. Giorgini, P. (Eds.). Idea Group Publishing, 107-135pp.
- Pavon, J., Gomez, J. J. and Fuentes, R.**, 2006, Model Driven Development of Multi-Agent Systems, Lecture Notes in Computer Science, 4066, 284-298pp.
- Pavon, J., Gomez-Sanz, J.J. and Fuentes, R.**, 2005, The INGENIAS Methodology and Tools, Agent-oriented Methodologies. Henderson-Sellers, B. Giorgini, P. (Eds.). Idea Group Publishing, 236-276pp.
- Pokahr, A., Braubach, L. and Lamersdorf, W.** 2005. Jadex: A BDI reasoning engine, Multi-agent programming, Springer, 149-174pp.
- Rao, A. S.**, 1996, AgentSpeak(L): BDI agents speak out in a logical computable language, In proc. the 7th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, 42-55pp.
- Russell, S. and Norvig, P.**, 2013, Artificial Intelligence: A Modern Approach, Pearson, Belgium, 1152pp.
- Schmidt, D. C.**, 2006, Model-driven engineering, Computer 39(2), 25-31pp.
- Shadbolt, N., Hall, W. and Berners-Lee, T.**, 2006, The Semantic Web revisited, IEEE Intelligent Systems, 21(3), 96-101pp.
- Shoham, Y.**, 1991, Agent oriented-programming, Artificial Intelligence, Elsevier, 60, 51-92pp.
- Silva, V., Garcia, A., Brandao, A., Chavez, C., Lucena, C. and Alencar, P.**, 2002, Taming Agents and Objects in Software Engineering, Lecture Notes in Computer Science, 2603, 1-26pp.
- Sredejovic, D., Vidakovic, M. and Ivanovic, M.**, 2018, ALAS: agent-oriented domain-specific language for the development of intelligent distributed non-axiomatic reasoning agents. Enterprise Information Systems, 12 (8-9), 1058-1082pp.

**KAYNAKLAR DİZİNİ (devam)**

- Strembeck, M. and Zdun, U.,** 2009, An approach for the systematic development of domain-specific languages, *Software-Practice & Experience*, 39(15), 1253-1292pp.
- Sycara, K. P.,** 1998, Multiagent systems, *Ai Magazine*, 19(2), 79-92pp.
- Sycara, K., Paolucci, M., Ankolekar, A. and Srinivasan, N.,** 2003, Automated discovery, interaction and composition of Semantic Web Services, *Journal of Web Semantics*, 1(1), 27-46pp.
- Vallecillo, A.,** 2008, A journey through the secret life of models. Perspectives Workshop: Model Engineering of Complex Systems (MECS), Citeseer.
- VueJS,** 2019, “VueJS”, <https://vuejs.org/>, (son erişim tarihi: Ağustos 2019)
- Warwas, S. and Hahn, C.,** 2008, The concrete syntax of the platform independent modeling language for multiagent systems, *Agent-based Technologies and applications for enterprise interoperability*, held in conjunction with the 7th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), Estoril, Portugal.
- Wile, D.,** 2004, Lessons learned from real DSL experiments, *Science of Computer Programming* 51(3): 265-290pp.
- Woolridge, M.,** 2002, *An Introduction to Multiagent Systems*, JOHN WILEY & SONS, London, 343pp.
- Woolridge, M., Jennings, N.R. and Kinny, D.,** 2000, *The Gaia Methodology for Agent-Oriented Analysis and Design*, *Autonomous Agents and Multi Agent Systems*, Springer, 285-312pp.
- World Wide Web Consortium (W3C),** 2004, “World Wide Web Consortium - OWL-S: Semantic markup for web services”, <http://www.w3.org/Submission/OWL-S/>, (son erişim tarihi: Ağustos 2019).
- Zambonelli, F., Jennings, N. R. and Wooldridge, M.,** 2003, Developing Multiagent Systems: The Gaia Methodology, *ACM Transactions on Software Engineering and Methodology*, 12(3), 317-370pp.





## TEŞEKKÜR

Öncelikle bu tez konusu üzerinde bana çalışma imkanı sunan ve ayrıca danışmanım olan Doç. Dr. Geylani Kardeş'a, deneyimleri, bilgileri ve önerileriyle araştırma ve geliştirmeye yönlendirmesi ve sağladığı kaynaklar ile destek olmasından dolayı teşekkürü bir borç bilirim.

Ayrıca tez süresince TÜBİTAK projesi kapsamında beraber çalışma fırsatı yakaladığım ve desteklerini esirgemeyen Dr. Moharram Challenger ve Ar. Gör. Barış Tekin Tezel'e, yoğun ve zorlu dönemlerde duygusal desteğiyle yanımda olan eşim ve aileme teşekkür ederim.

“115E591” numaralı “AgentDSM-Eval: Çok-Etmenli Sistem Alana-öзgü Modelleme Dilleri için bir Değerlendirme Çerçevesinin Geliştirilmesi” isimli proje kapsamında yüksek lisans çalışmamı maddi olarak destekleyen Türkiye Bilimsel ve Teknolojik Araştırma Kurumu'na (TÜBİTAK) teşekkürlerimi sunarım.

26 / 08 / 2019

Ömer Faruk ALACA

## ÖZGEÇMİŞ

**İsim Soyisim** : Ömer Faruk ALACA  
**Uyruk** : TC  
**Doğum Tarihi** : 09.01.1991  
**E-Posta** : omerfarukalaca@gmail.com

### Education

**Yüksek Lisans** : 2015-2019 , EGE Üniversitesi, Fen Bilimleri Enstitüsü,  
 Uluslararası Bilgisayar Enstitüsü, Bilgi Teknolojileri  
**Lisans** : 2010-2015, EGE Üniversitesi, Mühendislik Fakültesi,  
 Bilgisayar Mühendisliği

### Yayınlar

1. Cakmaz, Y. E., **Alaca, O. F.**, Durmaz, C., Akdal, B., Tezel, B., Challenger, M. and Kardas, G. (2017) “Engineering a BDI Agent-based Semantic e-Barter System”, In proceedings of the 2nd International Conference on Computer Science and Engineering (UBMK 2017), October 5-8, 2017, Antalya, Turkey, IEEE Conference Publications, pp. 1072-1077, DOI: 10.1109/UBMK.2017.8093474.
2. Konaray, S. K., Aras, G., Akcekoce, H., **Alaca, O. F.**, Challenger, M. and Kardas, G. (2017) “Development of a Semantic Web based Activity Recommender Software”, In proceedings of the 11th Turkish National Software Engineering Symposium (UYMS 2017), October 18-20, 2017, Antalya, Turkey, CEUR Workshop Proceedings, , vol. 1980, pp. 65-77 (in Turkish).
3. Challenger, M., Tezel, B. T., **Alaca, O. F.**, Tekinerdogan, B. and Kardas, G. (2018) “Development of semantic web-enabled BDI multi-agent systems using SEA\_ML: an electronic bartering case study”, Applied Sciences, vol. 8, no. 5, pp. 1-32, DOI: 10.3390/app8050688.
4. Silva, J., Barisic, A., Amaral, V., Goulao, M., Tezel, B. T., **Alaca, O. F.**, Challenger, M. and Kardas, G. (2018) “Comparing the Usability of two Multi-Agents Systems DSLs: SEA\_ML++ and DSML4MAS - Study Design”, In proceedings of the 3rd International Workshop on Human Factors in Modeling (HuFaMo 2018), held in conjunction with ACM/IEEE 21st International Conference on Model Driven Engineering Languages and Systems (MODELS 2018), October 15, 2018, Copenhagen, Denmark, CEUR Workshop Proceedings.
5. Miranda, T., Challenger, M., Tezel, B. T., **Alaca, O. F.**, Barisic, A., Amaral, V., Goulao, M. and Kardas, G. (2018) “Improving the Usability of a MAS DSML”, In proceedings of the 6th International Workshop on Engineering Multi-Agent Systems (EMAS 2018), held in conjunction with the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), July 14, 2018, Stockholm, Sweden, Lecture Notes in Artificial Intelligence, vol. 11375, pp. 55-75, DOI: 10.1007/978-3-030-25693-7\_4 (**best paper award**).

## İş Tecrübesi

- **Backend Developer (Mart 2019 – Temmuz 2019)**  
*Radity / Zürich, İsviçre (Uzaktan)*
  - Python Django, PostgreSQL, C# .Net Core, Firebase teknolojilerini kullanarak web uygulamalarının arka-uç mimarilerinin tasarımı ve geliştirilmesi üzerine çalışılmıştır.
  
- **Yazılım Geliştirici ve Araştırmacı ( Nisan 2017 – Ağustos 2018)**  
*EGE Üniversitesi, Uluslararası Bilgisayar Enstitüsü / İzmir, Türkiye*
  - Proje numarası 115E591 olan “AgentDSM-Eval: Developing a Framework on Evaluating Domain-specific Modeling Languages for Multi-agent Systems” isimli TÜBİTAK ve Portekiz Bilim ve Teknoloji Vakfı'nın (FCT) desteklediği projede görev alınmıştır.
  
- **Yazılım Geliştirici ve Araştırmacı ( Ağustos 2014 – Eylül 2015)**  
*UNIT Information Technologies R&D Co. Ltd. / İzmir, Türkiye*
  - .net çerçevesi kullanılarak çeşitli iş akışı yönetim sistemi bileşenleri geliştirilmiştir.

## **EKLER**

**Ek 1:** PDT MAS DSML'i için gerekleřtirilen FAML üstmodeli kıyaslaması sonucunda desteklenen ve desteklenmeyen tasarım zamanı dil elemanları ve tanımlamaları.

**Ek 2:** PDT MAS DSML'i için gerekleřtirilen FAML üstmodeli kıyaslaması sonucunda desteklenen ve desteklenmeyen alıřma zamanı dil elemanları ve tanımlamaları.



**Ek 1: PDT MAS DSML'i için gerçekleştirilen FAML üstmodeli kıyaslaması sonucunda desteklenen ve desteklenmeyen tasarım zamanı dil elemanları ve tanımlamaları.**

Design Time Concepts Matching - %52

| FAML Concept  | Prometheus/PDT Concept |
|---|------------------------|
| > Action Spesification  | Action                 |
| > Agent Definition  | Agent                  |
| ∨ Environment Statement   | N/A                    |
| <b>FAML Definition:</b> A Boolean statement about the environment   |                        |
| > Facet Action Spesification  | Action                 |
| ∨ Task  | N/A                    |
| <b>FAML Definition:</b> Spesification of a piece of behaviour that the system can perform.                              |                        |
| ∨ Facet Definition  | N/A                    |
| <b>FAML Definition:</b> Spesification of the structure of a given facet, including its name, data type and access mode. |                        |
| ∨ Functional Requirement  | N/A                    |
| <b>FAML Definition:</b> Requirement that provides added value to the users of the system.                               |                        |
| ∨ Interaction Protocol  | N/A                    |
| <b>FAML Definition:</b> Spesification of patterns of communications that occur in the system.                           |                        |
| > Mental State Spesification  | Capability             |
| > Message Action Spesification  | Message                |
| ∨ Non-Functional Requirement  | N/A                    |
| <b>FAML Definition:</b> Requirement about any limits, constraints or impositions on the system to be built.             |                        |
| ∨ Ontology  | N/A                    |
| <b>FAML Definition:</b> Structural model of a given domain.   |                        |
| ∨ Organization Definition   | N/A                    |
| <b>FAML Definition:</b> Spesification of a collection of roles and agents co-operating towards a system goal.           |                        |
| > Plan Resource Spesification   | Plan                   |
| > Plan Spesification  | Plan                   |
| ∨ Policy  | N/A                    |
| <b>FAML Definition:</b> A rule that specifies an arrangement of events expected to occur in a given environment.        |                        |
| ∨ Requirement   | N/A                    |
| <b>FAML Definition:</b> Feature that a system must implement.   |                        |

|   |                        |     |
|---|------------------------|-----|
| ▼ | Resource Specification | N/A |
|---|------------------------|-----|

**FAML Definition:** A resource specification specifies something that has a name, may have reasonable representations and that can be acquired, shared or produced.

|   |                    |        |
|---|--------------------|--------|
| > | Role               | Role   |
| > | Role Compatibility | Role   |
| > | Role Dependency    | Role   |
| > | Role Relationship  | Role   |
| > | Service            | Action |
| ▼ | System             | N/A    |

**FAML Definition:** Final product of an agent oriented software development project.

|   |             |     |
|---|-------------|-----|
| ▼ | System Goal | N/A |
|---|-------------|-----|

**FAML Definition:** A specification of a state of the environment that the system tries to achieve.

|   |                |         |
|---|----------------|---------|
| > | Message Schema | Message |
|---|----------------|---------|



**Ek 2: PDT MAS DSML'i için gerçekleştirilen FAML üstmodeli kıyaslaması sonucunda desteklenen ve desteklenmeyen çalışma zamanı dil elemanları ve tanımlamaları.**

Run Time Concepts Matching - %55

| FAML Concept  | Prometheus/PDT Concept |
|---|------------------------|
| > Action  | Action                 |
| > Agent   | Agent                  |
| > Agent Goal  | Agent + Goal           |
| ∨ Belief  | N/A                    |
| <b>FAML Definition:</b> An environment statement held by an agent and deemed as true in a certain timeframe               |                        |
| ∨ Communication   | N/A                    |
| <b>FAML Definition:</b> Composition of more than one message  |                        |
| ∨ Environment   | N/A                    |
| <b>FAML Definition:</b> The world in which an agent is situated.  |                        |
| ∨ Environment History   | N/A                    |
| <b>FAML Definition:</b> The sequence of events that have occurred between the environment start-up and any given instant. |                        |
| ∨ Environment Statement   | N/A                    |
| <b>FAML Definition:</b> A statement about the environment   |                        |
| > Event   | Scenario               |
| ∨ Facet   | N/A                    |
| <b>FAML Definition:</b> Property of the environment with which agents can interact.                                       |                        |
| ∨ Facet Action  | N/A                    |
| <b>FAML Definition:</b> Action that results in the change of a given facet.   |                        |
| ∨ Facet Event   | N/A                    |
| <b>FAML Definition:</b> Event that happens when the value of a facet changes.   |                        |
| > Mental State  | Percept                |
| > Message   | Message                |
| > Message Action  | Message + Action       |
| > Message Event   | Message + Scenario     |
| ∨ Organization  | N/A                    |
| <b>FAML Definition:</b> A collection of agents with specified roles co-operating towards a system goal.                   |                        |
| > Plan  | Plan                   |
| > Resource  | Data                   |
| > Role  | Role                   |
| ∨ System  | N/A                    |
| <b>FAML Definition:</b> Final product of an agent-oriented software development project.                                  |                        |