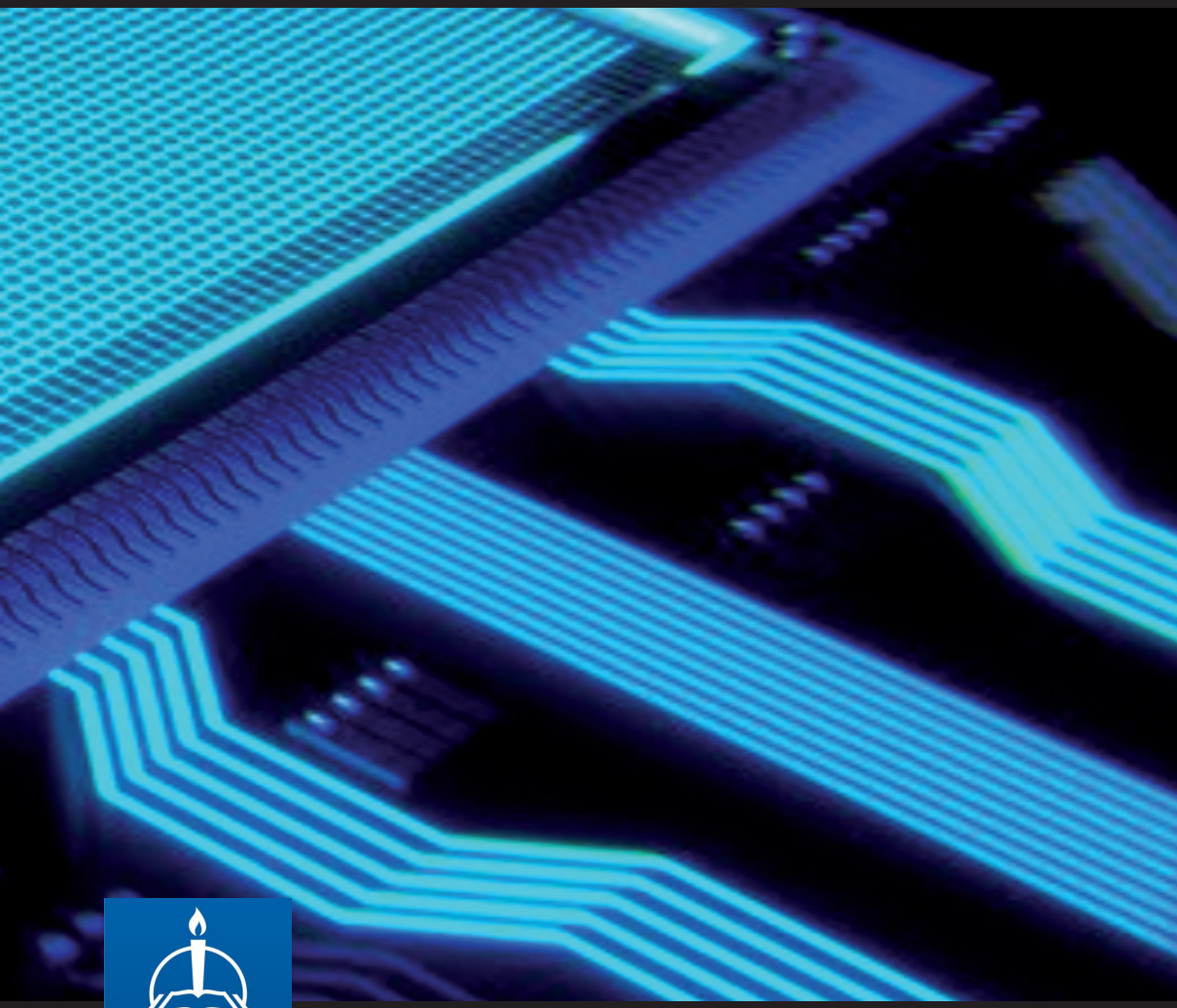
BİLİŞİM SİSTEMİ GELİŞTİRMEDE İHTİYAÇ BELİRLEME

Dr. Öğr. Üyesi Tijen ÖVER ÖZÇELİK
Prof. Dr. Orhan TORKUL



İKSAD
Publishing House

BİLİŞİM SİSTEMİ GELİŞTİRMEDE İHTİYAÇ BELİRLEME

Dr. Öğr. Üyesi Tijen ÖVER ÖZÇELİK[1]

Prof. Dr. Orhan TORKUL[2]

[1] Sakarya Üniversitesi, Mühendislik Fakültesi, Endüstri Müh., Sakarya, tover@sakarya.edu.tr

[2] Yalova Üniversitesi, Mühendislik Fakültesi, Endüstri Müh., Yalova, torkul@yalova.edu.tr

*Bu kitap Tijen ÖVER ÖZÇELİK'in "İhtiyaç Belirlemede Endüktif-ROC Temelli Bir Model" adlı Doktora tezinden üretilmiştir.

Copyright © 2019 by iksad publishing house
All rights reserved. No part of this publication may be reproduced, distributed or transmitted in any form or by
any means, including photocopying, recording or other electronic or mechanical methods, without the prior written permission of the publisher, except in the case of
brief quotations embodied in critical reviews and certain other noncommercial uses permitted by copyright law. Institution of Economic Development and Social
Researches Publications®

(The Licence Number of Publicator: 2014/31220)
TURKEY TR: +90 342 606 06 75
USA: +1 631 685 0 853
E mail: iksadyayinevi@gmail.com
www.iksad.net

It is responsibility of the author to abide by the publishing ethics rules.

Iksad Publications – 2019©
ISBN: 978-625-7954-15-0
Cover Design: İbrahim Kaya
December / 2019
Ankara / Turkey
Size = 16 x 24 cm

İÇİNDEKİLER

BÖLÜM 1

BİLİŞİM İHTİYAÇLARININ BELİRLENMESİNDE TEMEL KAVRAMLAR

1 - 12

BÖLÜM 2.

LİTERATÜR TARAMASI

13 - 58

BÖLÜM 3.

İHTİYAÇ BELİRLEME İÇİN BİR MODEL

59 - 147

BÖLÜM 4.

MODELİN DEĞERLENDİRİLMESİ/SONUÇ

149 - 162

KAYNAKLAR

163 - 169

BÖLÜM 1

BİLİŞİM İHTİYAÇLARININ BELİRLENMESİNDE TEMEL KAVRAMLAR

GİRİŞ

Bilişim sistemi, organizasyonun tüm fonksiyonlarını yerine getirmeyi ve operasyonları desteklemeyi amaçlayan bütünleşik bir bileşenler topluluğu olarak tanımlanmaktadır. Bilginin çok hızlı üretildiği ve kullanıldığı bir çağda organizasyonların hızlı değişimlere uyum sağlayarak varlıklarını sürdürmesi ve rekabet edebilmesi için etkin bilişim sistemlerine sahip olmaları gerekir.

Bilişim sistemleri işletmede ki tüm süreçlerde yöneticilerin karar vermesinde yardımcı bir araç konumundadır. Bilişim sistemleri sayesinde verimlilik arttırılabilir, maliyetler aşağı çekilebilir, yeni ürünler, hizmetler, süreçler ve pazarlar geliştirilerek rakiplere karşı üstünlük sağlanabilir. Dolayısıyla bu sistemlerin süreçlerde oluşabilecek sorunları en aza indirgeyici etki yapması beklenmektedir. Dünyamızın artan bir oranda bilgi tabanlı olması ile bilginin önemi gün geçtikçe artmakta ve doğru bilgiye doğru zamanda ulaşmak bireyler ve toplumlar açısından büyük avantajlar sağlamaktadır. Özellikle küreselleşen dünyada bilgi önemli bir rekabet unsurudur. Bu rekabet ortamında ayakta kalabilmek ve bilgiden gereğince faydalanabilmek ancak teknolojik gelişmeleri izleyerek ve uygun teknolojiyi kullanarak gerçekleşebilmektedir.

Bilişim sistemleri organizasyonlar için ne kadar gerekli ve önemli ise geliştirilme işi de o kadar karmaşık ve zordur. Çünkü organizasyonlarda çok sayıda paylaşımcı mevcuttur ve dinamik organizasyonel çevrelerde ihtiyaçlar hızla değişmektedir. Bu durum sistem geliştirmede analiz

safhasının ilk ve öncelikli aşaması olan ihtiyaç belirleme işini zorlaştırmaktadır.

Sistem geliştirme projelerinde harcanan çaba ve maliyetin en yüksek olduğu safha analiz safhasıdır. Birçok yapılan akademik çalışma ve endüstriyel gözlem araştırmalarında sistem problemlerinin %56'sının belirlenen ihtiyaçların yetersizliğinden kaynaklandığı gözlenmiştir. Ayrıca bakıma yönelik çalışmalarında %82'sinin yetersiz belirlenen ihtiyaçlardan kaynaklandığı savunulmaktadır.

Bilişim sistemleri geliştirmede "bilişim ihtiyaçları belirleme" sürecinin genellikle çok önemli bir aşama olarak gösterilmesi de ihtiyaç belirlemeye verilen önemin bir göstergesi olarak alınabilir. Bu problemin aşılabilmesi amacıyla literatürde geliştirilmiş olan birçok ihtiyaç belirleme, bilişim ihtiyaçları belirleme, ihtiyaç sunum teknikleri veya ihtiyaç geliştirme teknikleri vardır. Bu teknikler yardımıyla kullanıcıların gereksinimleri ve beklentileri karşılanarak bir ihtiyaç kümesi oluşturulur. Eğer ihtiyaçlar kümesi doğru bir şekilde oluşturulmamışsa, geliştirilen bilişim sisteminin yetersizliğinin telafisi nedeniyle harcanan çaba ve maliyet, geliştirme maliyetlerinden çok daha fazla olacaktır.

Birçok farklı alan için ihtiyaç belirleme üzerine çalışma bulunmaktadır. Günümüzde organizasyonlar bilişim tabanlı olmaya başlamış ve stratejilerini bilişimin üzerine odaklamaya başlamıştır [1].

Bilişim sistemlerine yönelik literatürde; ihtiyaç belirleme ve geliştirme metodolojileri göz önünde bulundurulduğunda 1970'lerden önce sistem

geliştiriciler bilgisayar temelli bilişim sistemlerini geliştirirken sürecin teknik kişiler tarafından yapılması gerektiğini savunmuşlar ve sadece imalata yönelik alanlarda bunu gerçekleştirmişlerdir. Oysa günümüzde bilişim ihtiyaçları sadece imalat ortamlarında değil tüm alanlarda mevcuttur.

Bilgisayar bütünleşik sistemlerin geliştirilmesinde önemli bir yere sahip olan konular, 80'lerden sonra düzenlenen çeşitli konferanslarda tartışılarak daha farklı bakış açıları ve teorilerin gelişmesine neden olmuştur. Geçen 50 yıl içinde analistler ve bilgisayar bilimciler organizasyonel bilgi ve rutinler üzerine yoğunlaşarak bu tür konulara açıklık getirmişler ve bu sayede bu konulara yönelik çalışan kişilerin daha verimli ve etkin olmasını sağlamışlardır.

Bilişim sistemi geliştirme çalışmalarında son yıllarda sistem geliştirme yaşam döngüsü gibi geleneksel metotların yanında modern olarak adlandırılan ve geleneksel metotlar temel alınarak geliştirilen, prototipleme, nesneye yönelik analiz/tasarım (OOA/D) ve hızlı uygulama geliştirmede (JAD) kullanılmaktadır. Gerek geleneksel gerekse modern metodolojilerle bilişim sistemi geliştirmede, öncelikli süreçler vardır. Bu süreçlerin başında da ihtiyaç tanımlama veya ihtiyaç belirleme gelmektedir.

Kısacası bilişim sistemlerinin geliştirilmesi sürecinde öncelikli öğelerden biri en uygun bilişim sisteminin tasarımı ve ihtiyaçlarının belirlenmesi gelmektedir. Bu bağlamda bu bölümde öncelikle kitap

çerçevesinde kullanılacak bazı temel kavramların tanımlamaları verilmektedir.

Veri

Veri kelimesi Latince datum kelimesinden türetilmiştir. Veri tam anlamıyla; verilen veya elde edilen bir sayı, durum yada resim olabilir. Eğer bilgi üretimi göz önünde bulundurulacak olsa, veri bir hammadde olarak düşünülebilir. Kısaca sayısal veya mantıksal her türlü değer bir veridir [2].

Bilgi

Bilgi, verinin işlenmiş ve bir anlam ifade eden halidir. Kısaca eldeki verileri süreçlerden geçirip manipüle edilerek elde edilen anlamlı ve faydalı çıktılardır [2].

Bilişim sistemi (BS)

Literatürde, bilişim sistemleriyle ilgili birçok tanım bulunmakta ve bu tanımların aşağı yukarı tümü özde aynı anlamı taşımaktadır. Aşağıda bu tanımlardan bir kaçı verilmektedir.

- Karar verme sürecini kolaylaştırmak, planlama, kontrol ve operasyonel fonksiyonların etkinliğini artırmak amacıyla üst yönetimin ihtiyaç duyduğu bilgiye; doğru, anlamlı şekilde ve zamanında ulaşmasını sağlayan sistemlerdir [2].

- Çeşitli kaynaklar aracılığı ile elde edilen verinin, üst yönetimin karar verebilmesi için gerekli bilgi haline dönüştürülmesini sağlayan bilgisayar bütünleşik sistemlerdir [3].

- Organizasyonun her seviyesi arasında ki bilgi akışının en üst seviyede gerçekleşmesini sağlayan bilgisayar bütünleşik sistemlerdir [4].

Bütün bu tanımlardan hareketle, bir sentez yapmamız gerekirse, bilişim sistemleri; organizasyonların her kademesinde ihtiyaç duyulan bilgilere cevap verebilmek amacıyla, gerekli verileri toplayan, işleyen, saklayan ve doğru zamanda doğru yere ulaşmasını sağlayan böylece de yöneticilerin bu küreselleşen ortamda gerekli rekabet performansını sergilemelerini sağlamak amacıyla onları raporlarla destekleyen sistemlerdir diyebiliriz.

İhtiyaç

İhtiyaç, sistemin gereklerini karşılamak için sistemin sergilemesi gereken özelliktir. İhtiyaçlar; sistemin paydaşları ile ilgili bilgilerdir. Bir sistem veya uygulamanın içermesi gereken özelliklere ihtiyaç denir [5].

İhtiyaç analizi

Bu kavram, bazen ihtiyaç analizi, bazen bilişim ihtiyaçlarını belirleme veya ihtiyaç belirleme bazen de ihtiyaç çıkarma veya ihtiyaç

mühendisliği olarak karşımıza çıkmakta ve aşağıda görüldüğü şekillerde tanımlanmaktadır [6].

- İhtiyaç analizi, bir sitemdeki veri, bilgi (bilişim), işlem, kullanıcı gibi elemanları ve bu elemanların birbirleriyle olan ilişkilerini tanımlamadır. İhtiyaç analizinde sistem analisti sistemde güncelleme, geliştirme veya değişiklik yapmaya gerek duyar. Bu başlangıç analizinden elde edilen değerli bilgi, sistemde neyin niçin çalıştığı veya çalışmadığı şeklindedir. Kısacası, insanların ne yaptığı ve bunu yaparken hangi bilgiyi kullandıklarıyla ilgilidir [5].

- Bilişim ihtiyaçlarını belirleme bir sistem analisti tarafından sistemin son kullanıcılarından hareketle bir bilişim sistemi için ihtiyaçları ve gerekleri öğrenme sürecidir. Tam olarak belirlenen kısıtlar ve özellikler kümesidir [7].

- Bir sistem için, ihtiyaçlar kümesinin keşfedilmesi, dökümanlanması ve bakımı üzerine tüm faaliyetleri kapsar. Bir problem alanının organizasyonel ihtiyaçlarını, keşfetme, yakalama, sınıflandırma, modelleme ve doğrulama sürecidir. Bu sürecin çıktısı, problem alanı için tam, doğru, tutarlı ve kavramsal modelin en küçük bir kümesidir [8].

- İhtiyaç mühendisliği yazılım mühendisliğinin bir alt disiplinidir.Yazılım mühendisliği, yazılım ürünlerini oluşturma sürecindeki yönetim ve geliştirme için gerekli teknikleri, araçları ve metotları içerir. Sistem ve sistem süreçlerinde; analizi,

dökümanlama ve kontrolle ilgili hizmet ve kısıtlı tanımlama ihtiyaçlarıdır. Bu hizmet ve kısıtlar ise ihtiyaç mühendisliği olarak adlandırır. Bu sürecin çıktısı, tamdır, doğrudur, tutarlıdır ve problem alanı için kavramsal bir modelin en küçük bir kümesidir. Bir sistem için, ihtiyaçlar kümesinin keşfedilmesi, dökümanlanması ve bakımı üzerine tüm faaliyetleri kapsar [9, 10].

Sistem analizi

Sistem analizi bilişim sisteminin ne yapması gerektiğini ayrıntılı olarak anlama ve belirleme sürecidir. Sistem gereklerinin belirlenmesi, ya da daha yaygın olarak kullanılan adlarıyla, *sistem çözümleme* ya da *sistem analizi*, bilişim sistemlerinin geliştirilmesinde, en önemli aşamadır. Çünkü bundan sonraki aşamaların tümü bunu baz alarak gerçekleşecektir [11].

İhtiyaç sınıfları

İhtiyaçların sağlıklı bir şekilde belirlenmesi, ihtiyaç kümesinin tamlığı ihtiyaçların sınıflandırılmasından geçer. İhtiyaçlar literatürde çeşitli şekillerde sınıflandırılmıştır. En yaygın sınıflandırmalar;

- Fonksiyonel/Fonksiyonel olmayan ihtiyaçlar,
- Kullanıcı ihtiyaçları/İşletme ihtiyaçları,
- Amaç/Süreç/Görev/Bilişim seviyeli ihtiyaçlar şeklinde görülmektedir.

Fonksiyonel ihtiyaçlar: Fonksiyonel sistem ihtiyaçları bir organizasyonun yaptığı işe yönelik olarak sistemin ne yapması gerektiğini (iş ihtiyaçlarını, işletme ihtiyaçlarını) destekleyen türdeki ihtiyaçlardır. Kullanıcıların amaçlarına ulaşma isteği ve kullanıcıların yapmak istedikleri işlere yönelik bilişim gereksinimlerinin yani; yeni ürün ve hizmetlere ait bilgi girişinin, işlemenin ve geliştirme biçiminin belirlenmesidir. Fonksiyonel sistem ihtiyaçları için öncelikli bilgi kaynağı sistemin çeşitli paylaşımcılarıdır. Genelde, paylaşımcılar 3 kategoride toplanır: Sistemin kullanıcıları, müşteriler, teknik personel. Sistem paylaşımcılarını belirlemek sistem ihtiyaçlarına karar vermedeki en önemli adımlardan biridir Kısaca kullanıcıların gerçekleştireceği faaliyetlerin özetlerini içeren işler yani onların karşılaştığı kısıtlar ve tercihlerdir [5].

Fonksiyonel olmayan ihtiyaçlar: Fonksiyonel olmayan ihtiyaçlar; kullanılacak hizmet ve ürüne ait çevrenin tanımı, kişisel tercihleri, kişilerin karşılaştığı engelleri, özel gereksinimleri, deneyimleri ve önceki bilgiler gibi kullanıcı karakteristiklerini tanımlayan ihtiyaçları içerir. Örneğin elektronik ürün bilgileri ve hizmetlere ait yeni işletme modülleri, ihtiyaçların güvenliği ve gizliliği gibi. Daha sonra gelebilecek fonksiyonel olmayan ihtiyaçlar ise maliyet kısıtlarından türetilenlerdir [12]. Fonksiyonel olmayan ihtiyaçlar genellikle kısıtlar veya kalite ihtiyaçları olarak da adlandırılır. Bu ihtiyaçlar daha da sınıflandırılacak olursa; performans, güvenlik, güvenirlik, bağdaşabilirlik ve daha bir çok ihtiyaçtan bahsedilebilir [5].

Kullanıcı ihtiyaçları: Kullanıcıların amaçlarına ulaşabilmesi, geliştirebilmesi veya problemleri aşabilmesi için gerekenlerdir. Kullanıcı gereksinimlerini sağlamak, tatmin edebilmek amacıyla gerek duyulan her bir fonksiyon, kısıt veya diğer özelliklerdir. Kullanıcı ihtiyaçları kullanıcı ve müşterilerin bakış açılarından tanımlanarak sistemin kullanıcılarından çıkartılır [13].

İşletme ihtiyaçları: İşletme ihtiyaçları organizasyonun amaçları ve hedefleridir. Hedef ve amaçlar ise; tüm kanunları, kararları, bütçeye ilişkin kararları, yönetim talimatlarını, kullanım kitapçıklarını (kılavuzları), misyonu (üstlenilen görevleri) kapsar [13].

Amaç seviyeli ihtiyaçlar: Amaç seviyeli ihtiyaçlar, sistemin geliştirilme nedeni ve organizasyonel amaçların anlaşılmasına odaklıdır.

Süreç seviyeli ihtiyaçlar: Süreç seviyeli ihtiyaçlar, gerçek işletme faaliyetleri üzerine odaklıdır.

Görev seviyeli ihtiyaçlar: Görev seviyeli ihtiyaçlar, çevrede gelişen olaylardan işlerin nasıl etkilendiğini ve işletme faaliyetlerini yerine getirmede ihtiyaç duyulan özel durumlar üzerine odaklanır.

Bilişim seviyeli ihtiyaçlar: Bilişim seviyeli ihtiyaçlar ise tüm alanın veri ilişkilerinin bir bütün halinde anlaşılması ve onlar arasında ki ilişkiler üzerinedir [14, 15].

İhtiyaç belirleme teknikleri

İhtiyaçları belirlemede çeşitli metotlar, teknik veya araçlar kullanılabilir. Metot, ihtiyaç belirleme için bilginin analizi ve toplanmasında farklı süreçler olarak tanımlanmaktadır. Fakat bunun yanında hangi metodun veya tekniğin uygulanacağıyla ilgili özel bir bilgi vermezler. Örneğin bilgi/ihtiyaç toplamayla ilgili olarak; mülakat (standartlaştırılmış sorular), inceleme, gözlem ve doküman toplama verilebilir. İhtiyaçları modelleme/sunum için; ne-eğer analizi, senaryo oluşturma, veri akış diyagramları, kavramsal haritalar vb.leri verilebilir. Genellikle bu strateji, metot ve araçların seçiminde analistler bunların kombinasyonundan faydalanırlar. Bunu da kendi bilgi ve tecrübelerinden hareketle gerçekleştirirler [16, 17, 18].

BÖLÜM 2.

LİTERATÜR TARAMASI

Bu bölümde; sistem geliştirme çalışmalarında analiz safhasının ilk aşaması olan ihtiyaç belirleme süreci, bu süreçte karşılaşılan problemler ve bu problemlerin çözümüne yönelik literatürde var olan strateji/bakış açısı/teknikler ve ihtiyaçların sınıflandırılmasına yönelik çalışmalar incelenmiştir.

Günümüz organizasyonlarının etkinliği ve verimliliği organizasyon için geliştirilen sistemin başarısıyla doğru orantılıdır. Geliştirilen sistemin başarısı da, kullanıcıların gereksinimlerine, beklentilerine ne derece cevap verdiği ile ölçülür. Kısaca geliştirilen sistemin, kullanıcıların ihtiyaçlarını karşılama oranı sistemin başarısını gösterir.

Bu nedenle bilişim sistemi geliştirmede "bilişim ihtiyaçları belirleme/ihtiyaç belirleme" sürecinin genellikle çok önemli bir aşama olarak gösterilmesi bunun bir göstergesi olarak alınabilir [16, 18, 19]. Bir çok araştırma sistem geliştirmedeki hataların, yetersiz veya doğru olarak belirlenmeyen ihtiyaçlardan kaynaklandığı üzerinde yoğunlaşmıştır [20].

Bilişim sistemi geliştirme çalışmalarının ilk safhası olarak bilinen analiz safhasında yukarıda da söz edildiği gibi eksik/yetersiz ve gereğinden fazla ihtiyaç belirlemeden kaynaklanan problemler ortaya çıkmakta ve bu problemler literatürde; ihtiyaç belirleme, bilişim ihtiyaçlarının belirlenmesi veya ihtiyaç belirleme süreci başlıkları altında incelenmektedir.

İhtiyaç belirleme sürecindeki problemlerin aşılabilmesi amacıyla literatürde geliştirilmiş birçok ihtiyaç belirleme, bilişim ihtiyaçları

belirleme, ihtiyaç sunum veya ihtiyaç geliştirme teknikleri bulunmaktadır. Bu teknikler literatürde bazen aşılması gereken probleme bazen modern veya geleneksel oluşlarına bazen de süreç içinde kullanımlarına göre sınıflandırılmıştır. Bu bölümde gerçekleştirilen literatür araştırması bu yaklaşım çerçevesinde ele alınmıştır.

İhtiyaç Belirleme

Browne ve Rogich ihtiyaç belirlemeyi; sistem geliştiriciler/analistler tarafından arzu edilen sistem için gereksinim duyulan fonksiyonlara yönelik bilginin toplanması ve modellenmesi süreci olarak tanımlamışlardır [8]. Bir diğer tanıma göre, arzu edilen sistem için; kullanıcı gereksinimlerinin, problemlerinin anlaşılarak çözümlenmesi amacıyla sistem analistlerinin bilgi toplaması, bir diğerine göre ise; genellikle kullanıcı merkezli bir yaklaşımla sistem ve kullanıcı gereksinimlerinin karşılanması şeklindedir [6].

Literatürde ihtiyaç belirleme; ihtiyaç analizi, bilişim ihtiyaçları analizi, bilişim ihtiyaçlarının belirlenmesi, kullanıcı ihtiyaçları, ihtiyaç mühendisliği, bilgi kazanımı, sistem ihtiyaçları, bilgi çıkarımı vb. birçok farklı adla anılmaktadır [6, 20].

Literatürde, ihtiyaç belirlemenin, sistem geliştirme sürecinin en iyi tanımlanması gereken safhası olduğu [5] ve bilişim sistemi geliştirme faaliyetlerinin en zoru olduğu söylenmektedir [15]. Yine birçok araştırmacı tarafından sistem geliştirme çalışmalarında ki

başarısızlıklara en önemli sebep olarak ihtiyaç belirleme sürecinde oluşan problemler gösterilmektedir [16, 18, 19].

Genellikle sistem geliştirme çalışmalarının teslimi sonrasında çıkan problemlerin çoğunun sistem ihtiyaçlarının yetersiz veya yanlış olarak belirlenmesinden kaynaklandığı tartışılır [21]. İhtiyaç belirleme sürecinden kaynaklanan bu hatalar, sistem geliştirme sürecinin sonra ki safhalarına aktarıldığında [22], giderilmesi çok daha büyük maliyetlere neden olmakta ve bilişim sistemlerinin bakım maliyetlerinin yaklaşık olarak üçte ikisini oluşturmaktadır [14]. Marasco'ya göre de ihtiyaç belirleme sürecinde ki hatalar nedeniyle U.S'deki işletmelerin proje çalışmalarındaki yıllık maliyet kayıpları 30 milyon Dolar'ın üzerindedir [23].

Bilişim ihtiyaçları analizinde/ihtiyaç belirlemede kullanıcı gereksinimlerinin karşılanması çok önemlidir. Çünkü sistemin asıl sahibi o sistemin kullanıcılarıdır. Bu nedenle ihtiyaç belirleme sürecinde sistemin içerdiği kullanıcılardan hareketle ihtiyaçlar belirlenmelidir. Kullanıcı ihtiyaçlarını doğru ve tam olarak belirlemede ki eksiklikler, yeni sitemin başarısında önemli rol oynar. Eğer sistem uygulamaya konduğunda kullanıcıların ihtiyaçlarına cevap vermiyorsa bu sistem geliştirme hatalarına ve başarısızlıklarına neden olur [6, 14, 23, 24].

Bu nedenle, sistem geliştirme çalışmalarında, bilişim ihtiyaçları belirleme sırasında meydana gelen eksiklikler (yetersiz veya hatalı ihtiyaç belirleme) daha sonraki safhalara aktarıldığında "domino

etkisi” göstermekte, hata ve problemler ard arda birbirini izlemektedir. Bunun sonunda;

- Maliyetler artmakta,
- Projenin tamamlanması gecikmekte,
- Son kullanıcı grubu ve geliştiriciler/analistler arasında düş kırıklığı meydana gelmekte,
- Proje alanında potansiyel değişiklikler oluşmakta,
- Tamamlanmamış ve daha az etkin bir bilişim sistemi oluşmaktadır [7].

İhtiyaç mühendisliği teriminin bazı çalışmalarda ihtiyaç belirleme anlamında kullanıldığını literatüre dayanarak birinci bölümde temel kavramlarda açıklanmıştır. İhtiyaç mühendisliği, bir bilişim sisteminin geliştirilmesi sırasında ihtiyaçlar kümesinin çıkartılması, doğrulanması ve bakım süreçlerini kapsamaktadır. İhtiyaç mühendisliği sürecinde, ihtiyaç mühendisleri (veya sistem analistleri) ihtiyaç mühendisliği safhasında ki çıktıların doğru olmasına, uygulama sürecinin uygunluğuna veya bilişim sistemi hatalarından sakınmaya yardımcı olacak tekniklere gereksinim duyar. İhtiyaç mühendisliği araştırmaları geleneksel olarak ihtiyaçların toplanması, çağrıştırılması ve geçerliliğinin sağlanması amacıyla kullanılan metotlar üzerine odaklanmaktadır [9].

İhtiyaç belirleme problemleri

Bilişim sistemi geliştirme süreci ne denli zorsa, ihtiyaç belirleme de bu süreçte karşılaşılan problemler nedeniyle oldukça zordur. İhtiyaç belirleme sürecinde ki bu problemler ihtiyaçların doğru ve eksiksiz olarak belirlenmesine neden olmaktadır [19]. Kısacası doğru ve tam bilişim ihtiyaçlarının belirlenmesi oldukça zordur ve literatürde bunun bir çok sebebi vardır.

Davis, ihtiyaç belirleme sürecinde bazı problemler engeller olduğundan bahsetmiş ve bu engellerin bazı metotlarla aşılabileceği üzerine çalışmalar gerçekleştirmiştir [18, 25].

Bu problemler Davis tarafından aşağıda görülen üç başlık altında toplanmıştır;

- Bilişim işlemcileri ve problem çözücüler gibi insanların sahip olduğu kısıtlar,

- Bilişim ihtiyaçlarının karmaşıklığı, çeşitliliği ve

- İhtiyaçları belirlemede analistler ve kullanıcılar arasındaki bilgi birikimi farklılıkları nedeniyle oluşan iletişim problemleri olarak verilmiştir [15, 25]

Davis'in ihtiyaçlar kümesini doğru ve eksiksiz olarak belirlemeye yönelik saptamalarını Shi ve diğerleri de desteklemektedir [26].

Sawyer ve Kotonya, bilişim sistemi geliştirme çalışmalarında analistler ve kullanıcılar arasındaki iletişim zorluklarının, ihtiyaçların yanlış ve

yetersiz belirlenmesine ve bilişim sistemi geliştirme projelerinin başarısızlıkla sonuçlanmasına neden olduğunu söylemişlerdir [5, 8].

Larsen ve Nauman ihtiyaç belirleme sürecinde kavramsal kısıtlardan kaynaklanan problemler, Valuesk ve Fryback ise analist ve sistem kullanıcıları arasında ki iletişimin geliştirilmesi konuları üzerine çalışmışlardır [27, 28].

Browne ve Ramesh'de bilişim ihtiyaçlarını belirleme sürecinde; bilginin toplanması, sunulması ve doğrulanması aşamalarında Davis ve diğerlerinden farklı olarak; İhtiyaçların belirlenmesinde kullanıcıların isteksizliğinden bahsetmişlerdir [29, 30].

Tüm bu problemlere rağmen sistematik bir metot veya metotlar kullanılarak bu problemlerin çözülebileceği düşünülmekte, Rogich ve Browne' nin çalışması buna bir örnek olarak verilmektedir [29].

İhtiyaçların belirlenmesinin geliştirilmesi, bilişim sistemlerinin kazanç, maliyet ve müşteri memnuniyeti açısından daha verimli ve etkin bilişim sistemi geliştirilmesine neden olacağı söylenmekle birlikte bilişim ihtiyaçlarının belirlenmesi problemlerinin birden bire ve tümüyle ortadan kaldırılması mümkün görülmemektedir. Bu zor bir süreçtir. Bu nedenle ihtiyaç belirleme sürecinin geliştirilmesi için, insanlardan kaynaklanan kavramsal kısıtların aşılması, bilginin toplanması, sunulması ve doğrulanması aşamalarının daha iyi anlaşılması sağlanarak bir model kapsamında ve basamak basamak yapılmalıdır [29].

İhtiyaç belirleme süreçleri

Günümüzde, ihtiyaçları belirleme süreci ve safhalarına yönelik problemlere oldukça yoğun bir ilgi vardır. Örneğin; Vigden bilişim ihtiyaçları analizinde paylaşımcı yaklaşımlarıyla ilgilenmiş [31], Darke ve Shanks ihtiyaç belirleme de kullanıcı bakış açısı olarak adlandırdıkları bir görüşü incelemiş [12], Flynn ve Davarpanah Jazi'de ihtiyaçlar için kullanıcıdan hareketle ihtiyaç belirleme bakış açısını önermişlerdir [32]. Problemler ihtiyaç belirleme sürecinin her aşamasında ortaya çıkmaktadır [33].

Birçok araştırmacı "bilişim ihtiyaçları belirleme" / IRD sürecini ihtiyaçları yakalama ve ihtiyaçları modelleme olmak üzere iki ana kısma ayırır [34, 35], bu yaklaşımdan hariç tutulabilecekler de vardır [29, 36, 37].

Örneğin, Browne ve Ramesh ise ihtiyaç belirleme sürecini üç kısma ayırmış ve bu safhaları ;

- Bilginin toplanması, (Bu aşamada analistler kullanıcılardan veya alan uzmanlarından bilgi toplama işiyle uğraşırlar),
- Bilginin sunulması, (modellenmesi) (Bu aşamada ihtiyaçlar bazı modelleme araçları aracılığı ile sunulur) ve
- Doğrulanması şeklinde tanımlamışlardır (Analistler kullanıcılar aracılığı ile belirledikleri ihtiyaçların doğru ve tam olduğunu kesinleştirirler) [29]

Vilatari' de ihtiyaçların belirlenmesini üç aşamada incelemiş;

- İhtiyaçları çıkartma safhasında ihtiyaçların kullanıcılar tarafından açık olarak ifade edilmesi,

- Sunum safhasında çağrıştırılan ihtiyaçların sistem analisti tarafından modellenmesi,

- Doğrulama ve geçerlilik safhasında; modellenen ihtiyaçların kullanıcılar tarafından gerçekten doğru olup olmadığının test edilmesi olduğunu söylemiştir [37, 38].

Bir diğer çalışmada yeni bir bilişim sisteminin tasarlanması ve geliştirilmesinde, ihtiyaç belirlemenin en önemli safhalardan biri olduğu ve bilişim ihtiyaçları analizinin üç aşamalı bir süreç olduğu söylenmiştir.

- Birinci aşama; mevcut bir sistemin özelliklerinin tanımlanması,

- İkinci aşama; yeni bilişim sistemi için bilişim ihtiyaçlarının belirlenmesi amacıyla veri toplama,

- Üçüncü aşama ise; arzu edilen en iyi bilişim ihtiyaçları kümesinin seçimi veya ağırlıklandırılmasıdır [39].

Nuseibeh ve Easterbrook, ise ihtiyaç analizinin seviyelerini aşağıda ki gibi tanımlamışlardır;

- İhtiyaçların ortaya çıkarılması,

- Çıkarılan ihtiyaçların modellenmesi ve analizi,

- İhtiyaçların birbirleri ile ilişkisi,

- İhtiyaç modellerinin kesinliğini doğrulama,

- Gruplar arasında ihtiyaçlar hakkında fikir birliğinin sağlanması ve

- Uygulama üzerinde ihtiyaçların hayata geçirilmesi [40].

Yukarıda ki tüm çalışmalar dahilinde bir genelleme yapılması gerekirse "ihtiyaç belirleme süreci" üç ana aşama veya bileşenden oluşmaktadır ve bunlarda;

- Bilgi/ihtiyaçların toplanması,
- Modellenmesi/sunulması ve
- Doğrulanmasıdır.

İhtiyaç sınıflamanın ihtiyaç belirleme sürecinde durdurma kuralları gibi kullanılması

Birinci bölümde bilişim ihtiyaçlarının belirlenmesinde temel kavramları açıklarken, "YBS organizasyonların her kademesinde ihtiyaç duyulan bilgilere cevap verebilmek amacıyla, gerekli verileri toplayan, işleyen, saklayan ve doğru zamanda doğru yere ulaşmasını sağlayan böylece de yöneticilerin bu küreselleşen ortamda gerekli rekabet performansını sergilemelerini sağlamak amacıyla onları raporlarla destekleyen sistemler" olduğu belirtilmiştir. Bir organizasyon içerisinde bilişim sistemleri genel olarak fonksiyonel ve yönetimsel faaliyetlerden oluşmaktadır. Flynn' da bilişim sistemi içinde fonksiyonel ve yönetimsel faaliyetlerden bahsetmiştir [5]. Genel olarak ihtiyaçlar, sistemin gereklerini karşılamak amacıyla sistemin sergilemesi gereken özelliklerdir ve amaçlarına göre farklılık gösterirler. Bu farklılık ürün parametreleri ve süreç parametreleri yönünden ele alınarak bir sınıflandırma yapıldığında;

- Fonksiyonel ihtiyaçlar,
- Fonksiyonel olmayan ihtiyaçlar ve
- Yönetimsel ihtiyaçlar olarak sınıflandırılır [5].

Sommerville sınıflandırmasında, fonksiyonel ve fonksiyonel olmayan ihtiyaçlara ilave olarak alan ihtiyaçlarından bahsetmiştir [41]. Ayrıca bir ihtiyaç belirleme grubunun seçiminde, bilgi düzeyi, ve deneyimlerin önemli rol oynadığı bunların özellikle fonksiyonel ihtiyaçlar açısından önemli olduğu savunulmuştur [5].

Etkin bir bilişim sistemi için kullanıcı bilişim ihtiyaçlarının analizinin yapılması önerilir [19]. Bu nedenle ihtiyaç belirlemenin kullanıcı perspektifinden değerlendirilmesinde, ihtiyaçların fonksiyonel ve fonksiyonel olmayan ihtiyaçlar şeklinde ortaya çıktığı savunulmuştur [6].

Fonksiyonel ihtiyaçlar: Fonksiyonel ihtiyaçlar bazen yetenek olarak da adlandırılır. Fonksiyonel ihtiyaçlar, kullanıcıların amaçlarına ulaşma isteği ve yapmak istedikleri işlere yönelik bilişim ihtiyaçlarının yani; yeni ürün ve hizmetlere ait bilgi girişinin, işlemenin ve geliştirme biçiminin belirlenmesidir. Fonksiyonel ihtiyaçlar veya faaliyetler genellikle kullanıcının izlemesi gerekli olan yol olarak tanımlanır. Kısacası işleri yapabilmek için gerekli yönlendirmelerdir. Kullanıcı işini yaparken ürünle ilgili detaylandırılmış teknik bilgilere mümkün olabilecek en az çabayla erişmeyi amaçlar. Anlaşıldığı üzere fonksiyonel ihtiyaçlar kullanıcıların gerçekleştireceği faaliyetlerin özetlerini içeren işler yani onların karşılaştığı kısıtlar ve tercihlerdir.

Fonksiyonel olmayan ihtiyaçlar: Fonksiyonel olmayan ihtiyaçlar; kullanılacak hizmet ve ürüne ait çevrenin tanımı, kişisel tercihleri, kişilerin önünde ki engelleri, özel gereksinimleri, deneyimleri, kalite ihtiyaçlarını ve önceki bilgiler gibi kullanıcı karakteristiklerini tanımlayan ihtiyaçları içerir. Bu ihtiyaçlar daha da sınıflandırılacak olursa; performans, güvenlik, güvenirlik, bağdaşabilirlik ve daha bir çok ihtiyaçtan bahsedilebilir. Daha sonra gelebilecek fonksiyonel olmayan ihtiyaçlar ise maliyet kısıtlarından türetilenlerdir [5, 42].

İhtiyaçlar genellikle yukarıda ki gibi sınıflandırılmakla birlikte literatürde bir çok farklı sınıflandırma vardır. Fakat ihtiyaçların nasıl ifade edilmesi gerektiği üzerinde çok az fikir birliği vardır [43].

Sistem analizi çalışmaları, kullanıcılardan bilgi toplama ve sistem ihtiyaçlarını belirleme ile sonlanır. Burada bilgi toplamada ki amaç probleme çözüm üretmedir. Problem çözme veya karar verme durumlarında insanlar bilgi toplayarak problemi tam anlamıyla yapılandırabilmeyi veya problem hakkında uygun bir karar vermeyi arzular. Karar vermede önemli bir süreç olan bilgi toplama işi belli bir noktada durdurulmalıdır. Deneysel çalışma sonuçları, insanların bilgiye yönelik konularda fayda/maliyet analizini dengeli olarak yapamadıklarını göstermektedir [14].

Örneğin bir doktor, teşhise yönelik testleri yaparken belli bir noktada durmak ve karar vermek durumundadır. Ama teşhis kararını vermesine yetecek oranda bilgiye/teste gereksinimi vardır. Eğer gereğinden fazla test gerçekleştirmişse; pahalı ve gereksiz bilgidir. Fakat gerekenden az

test gerçekleştirmişse sağlıklı karar vermesine ve teşhiste bulunmasına yetecek oranda bilgi çıkarımında bulunmamış demektir ki buda istenmeyen bir durumdur. Bu nedenle şüphelendiği hastalıklara yönelik hangi testleri yapması gerektiğine ait bir bilgisi varsa veya elinde bir şablon varsa nerde duracağını biliyor demektir. İşte bu durum literatürde "durdurma kuralları" olarak ifade edilmekte ve ihtiyaçlara yönelik sınıflandırmalarında durdurma kuralları gibi kullanılacağı ve faydalı olacağı belirtilmektedir [15]. Çünkü, gereğinden fazla veya az bilgi çıkarma maliyetleri olumsuz yönde etkileyen faaliyetlerdir .

Gerçekleştirilen araştırmalarda ihtiyaçların sağlıklı olarak belirlenebilmesinde ihtiyaç sınıflarının faydalı olduğu savunulmakta ve bu kapsamda yapılan çalışmalarda, bir tasarım sürecinde analistin amacının yeterli miktarda bilgi (ihtiyaç) toplamak olduğu ve elde edilen bu bilginin (ihtiyacın) bir şekilde ölçülmesi gerektiği belirtilmektedir. Bu ölçümün gerçekleştirilmesinde belirlenen ihtiyaçları değerlendirilmek amacıyla hazırlanmış şablonlar veya ihtiyaç sınıflarından faydalanılmaktadır.

White ve Edwards karmaşık sistem ihtiyaçları ve bazı yaklaşımları inceleyerek bir sınıflandırma tanımlamış, İhtiyaç Mühendisliği (RE) yaklaşımının tüm çıkartılan bilgi/ihtiyaçları kontrolde faydalı bir şablon olduğunu ve aynı zamanda ihtiyaçların uygunluk, doğruluk (kesinlik), anlaşılırlık ve tamlığını desteklediğini belirtmişlerdir [44].

Browne ve Pitts, bu tür problemlerin aşılmasında durdurma kurallarının etkin rol oynadığı kanısına varmışlardır. Gerçekleştirdikleri çalışmada,

bir tasarım sürecinde analistin amacının yeterli miktarda bilgi toplamak olduğu ve bu bilginin ölçülmesi gerektiği fikrinden hareketle; ihtiyaçların bir sınıflandırmasını gerçekleştirmişlerdir. Bu sınıflandırma şeması bilişim sistemlerinin başarılı bir şekilde tasarlanması amacıyla kritik olduğuna inanılan varlıkların problem alanlarını içermektedir. Bu şema aynı zamanda durdurma kuralları gibi kullanılmış ve bu sayede eksik veya fazla bilgi çıkarımı engellenmiştir [14].

Gereğinden fazla ihtiyaç çıkarma maliyetleri çok büyük oranda artırmakta, gereksiz zaman ve kaynak tüketimine neden olmaktadır. Bunun yanında gerekenden az ihtiyaç çıkarma ise; arzu edilen sistemin amaç ve fonksiyonlarının yetersiz olmasına, tasarım problemlerine, tekrarlı tasarıma, uygulama zorluklarına ve dolayısıyla da sistem hatalarına neden olmaktadır. Gereğinde az ihtiyaç belirme geliştirme çalışmalarında özellikle sistemin testi sırasında ortaya çıktığından maliyetleri yüksek oranda artırmakta ve sistem geliştirme çalışmalarının başarısızlıkla sonuçlanmasına neden olmaktadır.

Durdurma kuralları Smith, Benson, ve Curley tarafından yargı ve çıkarsama temelli olarak iki kısımda incelenmiştir [45, 46].

Durdurma kuralları motivasyonel veya kavramsal olarak sınıflandırabilir. Motivasyonel durdurma kuralları dışsal faktörler tarafından örneğin; zaman, bütçe ve çevrede ki diğer durumlar tarafından yönlendirilir. İçsel faktörler veya kişinin düşünüş yönü

kavramsal durdurma kurallarını yönlendirir. Kavramsal durdurma kuralları yargı ve muhakeme temellidir.

İnsanların bilgiyi işleme kabiliyetlerinde ki kısıtlar göz önünde bulundurulduğunda bu bir sürpriz değildir. Yeterli miktarda kanıt varsa kişiler bu kuralcı durdurma kurallarına yönelmezler. Örneğin; araştırmalar insanların tam oluşmadan bilgi çıkarımını durdurduğunu göstermiştir. Buna bağlı olarak insanlar, toplam geri dönüşü veya karı göz önünde bulundurmaktadır [14, 47].

Bu olumsuz davranışlar bilgi çıkarımını daha olgunlaşmadan durdurmaya meyillidir. Bir çok durdurma kuralı genellikle ilave bilginin faydasını değerlendirmek amacıyla bilgi çıkarım maliyetlerinin ayarlanmasında karar vericiler tarafından kullanılır. Bununla beraber kişilerin çabalarının bilginin fayda ve maliyet ayarlamasında en uygundan daha az olduğuna yönelik kanıt vardır. Bilgi maliyetini tanımlamak zordur, bilginin kalitesinin ölçülmesi pek mümkün değildir.

Bilginin ekonomik değeri, ilave bilgi toplamanın karlılığını değerlendirmenin temelinde genellikle ekonomik karar teorisi kullanılmaktadır. Eğer ilave bilginin değeri onun çıkarım maliyetinden daha büyükse ilave bilgi toplanmalıdır. Benzer olarak marjinal araştırma teorisinde çıkarılan ilave bilgi eğer beklenen kazanç veya marjinal fayda bilginin marjinal maliyetinden daha büyükse ilave bilginin çıkarılması önerilir [14].

Bilişim sistemlerinin geliştirilmesi kapsamında Byrd ve diğerleri ihtiyaç sınıflandırmasını gerçekleştirmiş [19], daha sonra bu sınıflandırmayı Rogich geliştirmiştir [15]. Bu sınıflandırma şeması bilişim sistemlerinin başarılı bir şekilde tasarlanması amacıyla kritik olduğuna inanılan varlıkların problem alanlarını içermektedir [19]. Böylece bilişim sistemleri için ideal bir ihtiyaçlar kümesi her biri tanımlı sınıflardan önemli miktarda ihtiyaç içerebilecektir. Byrd ve diğerleri, Rogich ihtiyaç sınıflarını dört seviyede organize etmiştir. Bunlar; amaç, süreç, görev ve bilişim seviyeli ihtiyaçlardır.

Bu ihtiyaç kategorileri birçok diğer problem alanında ve herhangi bir sistem geliştirme çalışmasında kullanılabilmektedir [14, 15].

Bu ihtiyaç grupları sayesinde toplanan ihtiyaçlar kalite ve nicelik olarak etkin bir şekilde ölçülebilmektedir [15]. Çıkartılan ihtiyaçların bir analist tarafından ölçülmesi, ihtiyaçların derinliği ve genişliğinin belirlenmesi önerilmektedir. İhtiyaçların kalitesinin ölçülebilmesi amacıyla bir kodlama şemasına gereksinim olduğu savunulmaktadır [14, 15, 19].

İhtiyaç Belirleme Teknikleri ve Sınıflandırılması

Literatürde bilişim sistemlerinin geliştirilmesinde başarılı olabilmek ve ihtiyaç belirleme sürecini gerçekleştirebilmek amacıyla birçok teknik ve strateji geliştirilmiş [21, 48, 49] ihtiyaç analizi tekniklerinin farklılıkları ve sınıflandırılması verilmiştir [19]. Bu teknikler bazen modern/geleneksel oluşlarına bazen ihtiyaç belirleme süreci ve bazen de yaşanan problemlere göre sınıflandırılmıştır. Bu çalışma kapsamında

ise yetersiz/hatalı ihtiyaç belirlemenin üstesinden gelebilmek amacıyla önerilen teknikler incelenmiş ve çalışılma kronolojisine göre verilmiştir.

Literatürde yetersiz ihtiyaç belirlemenin üstesinden gelebilmek amacı ile önerilen teknikler

Taggart ve Tharp, "bilişim gereksinimlerini tanımlama"nın Yönetim Bilişim Sistemlerinin (MIS) başarılı bir şekilde gerçekleştirilmesinde çok önemli bir faktör olduğunu söylemiş ve çalışmalarında son yıllardaki çeşitli bilişim gereksinimlerini belirleme yaklaşımlarını içeren iki boyutlu bir inceleme gerçekleştirmişlerdir.

- Öncelikle çeşitli yaklaşımlar çerçevesinde (birkaç yaklaşım çerçevesinde) uygun, mevcut bilişim analizi metotlarını incelemiş,
- Bu metotların geliştirilmesine katkı sağlayacak araştırmaları inceleyerek öneri sunmuşlardır.

Çalışmalarında, bilişim gereksinimlerinin tanımlanması sürecinde ki problemlerin; yöneticiler, kullanıcılar ve analistlerden kaynaklandığını söyleyerek ihtiyaç analizi için çok amaçlı bir bakış açısı önermişlerdir [49].

Davis, bilişim ihtiyaçları belirleme metodu için strateji ve metot seçiminde aşamalar tanımlamış, belirsizliğe neden olarak; faydalanılan sistem, uygulama/bilişim sistemi, kullanıcılar ve analistleri göstermiştir. İhtiyaç belirleme için strateji seçimine yoğunlaşmış ve bilgisayar tabanlı bilişim sistemlerinin uygulama ve tasarımında bilişim

ihtiyaçlarını belirlemede dört strateji olduğundan bahsetmiş ve bunların;

- Soru sorma, (sorgulama)
- Mevcut bir bilişim sisteminden türetme,
- Kullanışlı bir sistemin özelliklerinden sentezleme ve
- Bir bilişim sisteminin gelişim süreci ile ilgili deneylerden keşfetme olduğunu iddia etmiştir.

Her bir strateji için bir çok metodoloji ve metotlar bulunmaktadır, fakat Davis çalışmasında en popüler stratejinin mevcut sistemin kullanıcılarının sorgulaması olduğunu, kullanılan metotların ise; kapalı sorular, açık sorular, beyin fırtınası, yönlendirilmiş beyin fırtınası, grup kararları olduğunu söylemiştir. İhtiyaç belirleme de hangi strateji ve metotların seçilmesi daha uygundur sorusuna cevap ararken de, geliştirilecek sistemde ki belirsizliğin derecesine göre bir seçim gerçekleştirmiş, belirsizliğin derecesi düşükse sorgulama stratejisini eğer yüksekse bilişim sistemi gelişim süreci ile ilgili deneylerden keşfetme stratejisini önermiştir [18].

Valusek ve Fryback, bilişim ihtiyaçları belirlemeyi sistemin kullanıcıları ve sistem geliştiriciler arasındaki engeller üzerine odaklanan bir süreç olarak tanımlamış ve bilişim sistemi geliştirmede mevcut araştırmaları temel alan kullanıcıya yönelik bir bilişim ihtiyacı belirleme aracı tasarlamışlardır [28].

Johnson ve Johnson, uzman sistemleri geliştirmede anahtar problemin uzman olan kişilerden bilginin çıkarılması saptamasını yapmışlar ve bu darboğazın aşılabilmesi amacıyla çeşitli metotlar kullanıldığını bu metotlardan en yaygın olarak kullanılanın da *"mülakat"* olduğunu söylemişlerdir. Çalışmalarında bir başka metottan faydalanmış ve bu metodu öğretme destekli *"teachback"* olarak adlandırmışlardır. Bu metot karşılıklı konuşma teorisi ve yarı yapılandırılmış görüşme metotlarına dayandırılmıştır. Fakat gerçekleştirdikleri örnek çalışmada öğretme destekli yapı tam manasıyla yapılandırılmış olmadığı için çok fazla mülakat deneyimi ve eğitime ihtiyaç duyulduğu bu nedenle de kısıtlayıcı yönleri olduğunu görmüşlerdir [50].

Yadav ve diğerleri; bilişim ihtiyaçlarının belirlenmesine yönelik analiz tekniklerini karşılaştırmak için bir yapı (çatı/model) geliştirerek, Tümleşik Tanım Metodu ve Veri Akış Diyagramı (DFD) modelleme teknikleri ile bir deneysel çalışma gerçekleştirmişlerdir. Çalışma sonucunda DFD'nin öğrenilmesi ve kullanılmasının daha kolay olduğunu, daha iyi bir sonuç ürettiğini fakat bunun çok açık olarak gösteremediklerini söylemişlerdir. Sonraki çalışmalar için bu tekniklerin daha iyi bir model üretip üretmeyeceğini belirleyen deneysel çalışmalara gereksinim olduğunu sonucuna varmışlardır [25].

Gutierrez, bilişim ihtiyaçlarını çıkartma sırasında analistler ve kullanıcılar arasındaki iletişimi geliştirmede tekniklere ve sistem geliştirmede deneysel bir yaklaşıma gereksinim olduğunu iddia etmektedir. Çalışmasında bilişim ihtiyaç analizi süreci uygulaması için

bazı teknikler tanımlayarak, deneysel teknik veya teknikler grubu seçimi için strateji formülasyonu gerçekleştirmiştir [51].

Holbrook, ihtiyaç çıkarımı için senaryoların nasıl kullanılacağını tanımlayan özel süreç modeli geliştirmiştir [52].

Goguen ve Linde çalışmalarında, bilgisayar tabanlı sistem ihtiyaçlarının çıkartılması amacıyla kullanılan bazı tekniklerin incelemesini ve değerlendirmesini gerçekleştirmişlerdir. Çalışma, içsel bakış (introspection), mülakat, anket, protokol analizi ve konuşma analizi tekniğini içermektedir. İhtiyaç toplama/çıkarmada anket temelli mülakatların etkileşim açısından kısıtlayıcı olduğundan bahsetmişlerdir. Bununla beraber mülakatı yapan ve mülakatı yapılan arasında anlamlı bir paylaşım geliştirmesi açısından oldukça başarılı olduğunu, açık uçlu mülakatların iletişim açısından daha az kısıta sahip olduğunu ve metotların bütünleşik kullanımının eksik bilgi/ihtiyaç problemini ortadan kaldırabildiğini söylemişlerdir [53].

Hughes ve diğerleri yazılım (sistem) geliştirmeye yönelik bir çok tekniğin, özel sistem tasarımcılarının, sistemin kullanıcıları ve sahiplerinden daha çok bakış açılarına odaklandıklarını ve anladıklarını savunmaktadır. Oysa bu her zaman geçerli değildir. Bu nedenle mevcut tekniklerin bir yapıya oturtulmasının ve geliştirdikleri sınıflandırma çatısının faydalı olduğunu savunmaktadırlar. Sonra ki çalışmalarda geliştirilen bu çatıya kullanılmakta olan diğer tekniklerinde ilave edilmesi ve bu çatıya göre değerlendirilmesi çalışması düşünülmektedir [54].

Sharp, bilgi çıkarımı ve ihtiyaç analizi süreçleri arasında birçok paralellikler olduğunu, her ikisinin de bilgisayar tabanlı sistemlerin kullanımı üzerine olduğunu söylemiş, ihtiyaç analizi ve bilgi çıkarımında kullanılan teknikler üzerine odaklanmıştır. Çalışmasında modelleme aktivitelerinde ihtiyaç duyulan bilgi çıkarımı ve ihtiyaç analizi tekniklerinin nesneye yönelik yaklaşımı benimseyenleri ele almıştır. Bilgi çıkarım amacının problem çözme olduğundan ve bu nedenle de ilgili kişilerden bilgi üretilmesi ve analiz edilmesi gerektiğinden bahsetmiştir. Bilgi toplamanın mülakat, tartışma ve gözleme dayalı olduğundan, kaynak olarak kullanım el kitapları, yönergeler, örnek olaylar ve insanlardan faydalanıldığından bahsedilmiş, bu nedenle çalışmada; mülakatlara, prototiplemeye, örnek olaylara ve senaryolara, değinilmiştir. Bu sayılan tekniklerin tümünün hem bilgi çıkarımı hem de ihtiyaç analizi için kullanımının uygun olduğu sonucuna varılmıştır [55].

Shlaer ve Mellor, nesneye yönelik analizin modern bir teknik olduğunu ve doğal insan düşüncesini sisteme aktarmaya çalıştığını söylemiştir [56, 57]. Nesneye yönelik yaklaşım geliştirilmeden önce teoride ve pratikte, ihtiyaç analiz metotlarında yapılandırılmamış analiz oldukça önemli bir yer tutmasına karşın nesneye yönelik analiz geliştirildikten sonra, nesneye yönelik tasarım ve nesneye yönelik programlama önemli bir yere sahip olduğunu dile getirmişlerdir [5, 49].

Tamai'ye göre yazılım süreçlerinin modellenmesi sadece yazılım araştırmalarında değil, aynı zamanda endüstride de önemli bir yere sahiptir. Çalışmada gerçek yazılım süreçlerinin anlaşılması

amaçlanmış, pratikte mevcut olan sistem planlama, ihtiyaç analizi ve tanımlama incelenmiştir. Çalışma yaklaşımı derinlemesine mülakat ve döküman incelemeyi içermektedir. Çalışmada Japon endüstrisinde pratikte uygulanmakta olan, sistem planlama ve ihtiyaç analizi/tanımlama aşamalarının yazılım süreçlerinde ki mevcut durumu incelenerek tanımlanmıştır. Çalışma sonucunda sistem geliştirmede mevcut olan düzenli ve düzenli olmayan yaklaşımlar arasında bir denge olduğu, düzenli yaklaşımların ihtiyaç tanımlamada uygun olmasına karşın kişilerin kabulünde pek uygun olmadığı kanısına varılmıştır [58].

Tamai mülakatın özellikle üst yönetimle gerçekleştirildiğinde oldukça uygun bir teknik olduğunu, beyin fırtınası tekniğinin problem analizi, çözüm bulma ve ihtiyaç çıkarma için oldukça uygun bir teknik olduğunu söylemiştir. Bu metot, metodu oluşturan Japon Kawakita Jiro'dan dolayı KJ metodu olarak da anılmaktadır [58].

Hevner ve Mills, ihtiyaç belirleme ve sistem ihtiyaçlarının çıkartılmasının; ihtiyaçların toplanması, modellenmesi, tanımlanması ve analizinden oluşan tekrarlı bir süreç olduğunu, sistem geliştirme aşamalarında çok önemli bir yere sahip olduğunu söylemiştir. Çalışmalarında ihtiyaç belirlemeyi yapılandırılmış-kutu metoduyla (box-structured method) gerçekleştiren bir yaklaşım sunmuşlardır. Yapılandırılmış-kutu ihtiyaç belirleme metodunun sonucu formal sistem tasarım teknikleri için temel sağlamaktadır [59].

Taksonomi; nesneleri sınıflandırma için esaslar ve kanunlardan hareketle sınıflandırma pratiği ve teorisidir. White ve Edwards'da

yaklaşımları inceleyerek karmaşık sistem ihtiyaçları için bir sınıflandırma tanımlamışlardır. Bu sayede fonksiyonel olmayan ihtiyaçlar ve operasyonel çevredeki büyüme, değişmeler göz önüne alınarak bir sınıflandırma gerçekleştirilmiş ve birçok alanda yazılım sistemleri için genel bir yapı sağlanmıştır. Bu yapı sistem kapsamını derinlemesine desteklemektedir. Yapı olarak ihtiyaç mühendisliği yaklaşımı kullanılmıştır (RE). Bu yaklaşım ihtiyaçları gruplandırmada bir şablon vazifesi görür. RE yaklaşımının tüm çıkartılan bilgi/ihtiyaçları kontrolde faydalı bir şablon olduğunu ve aynı zamanda ihtiyaçların uygunluk, doğruluk (kesinlik), anlaşılırlık ve tamlığını desteklediğini belirtmişlerdir. Taksonominin gerçekleştirilmesi amacıyla çalışma kapsamında farklı metotlar tartışılarak incelenmiştir [44].

Alexander çalışmasında, bilgisayar bilimi literatüründe ki örnekler aracılığı ile ihtiyaç mühendisliği geçmişine ışık tutarak insan merkezli bilimlerin (pisikoloji, sosyoloji, etnoloji…) problemin tanımlanmasına ve çözülmesine yardımcı olduğunu ve teknolojide ki birkaç ana eğilimin sistem geliştirme çalışmaları sırasında ihtiyaçların anlaşılmasında ki ilerlemeyi yönlendirdiğini söylemiştir. Bu eğilimlerin her biri geliştirme metotları üzerinde güçlü bir etki yaparak sistem ve ihtiyaç mühendisliği adı altında iki farklı disiplinin ortaya çıktığını söylemiş, ihtiyaçlar üzerine tarihsel bir bakış gerçekleştirmiştir [60].

White ve diğerleri bu çalışmada karmaşık sistem ihtiyaçlarını tanımlayarak fonksiyonel olmayan ihtiyaçlar ve operasyonel çevredeki büyüme, değişmeleri göz önüne alarak bir sınıflandırma

gerçekleştirilmiş ve birçok alanda kullanılabilecek bir yapı sağlamışlardır. Burada yapı olarak ihtiyaç mühendisliği yaklaşımı kullanılmıştır (RE). Bu yaklaşım ihtiyaçları gruplandırmada bir şablon vazifesi görmektedir [44].

Shaw ve Gaines, bilgi mühendisliği araştırmalarını inceleyerek, teoriler, metodolojiler ve araçlar sunmuşlardır. Bunlara paralel olarak ihtiyaç mühendisliği faaliyetlerini değerlendirerek bir analoji gerçekleştirmiş, bilgi tabanlı sistemlerin diğer ileri bilişim sistemlerinden farklılık gösterdiğini fakat ihtiyaç çıkarımı terimi çerçevesinde uyuştuğunu saptamışlardır [61].

Mc. Dermid ihtiyacı, "müşteriler, kullanıcıları tatmin edecek şekilde bir sistem geliştirmek için bilinmesi gereken her şey" şeklinde tanımlamış ve bilişim ihtiyaçlarını tanımlamada (Information Requirements Definition/IRD) yapı problemlerini incelemiştir. Çalışmasında bilişim ihtiyaçlarını tanımlamada yapı problemlerini incelemiştir. Öncelikle IRD içindeki geriye dönük çalışmalar ve seçilmiş problemler tartışılmış, özellikle yapısızlık problemi ve sunumun düzensizliği birkaç ihtiyaç çıkarım tekniği çerçevesinde tanımlanmıştır. Gerçekleştirilen araştırmalar sonucunda; organizasyonlardaki bilginin yönetilmesine bir gereksinim olduğu, sonuç olarak işletme kurallarının üç seviyesi olduğu ve her bir seviyedeki bilginin detaylı bir içeriğe sahip olduğu söylenebilir [33].

Lyytikainen, ihtiyaçları çıkarmak amacıyla çeşitli metotlar mevcut olmakla beraber eski mülakat tekniği öncelikli olduğunu, fakat yorucu

olması, bazen eksikliklere neden olması, dağınık ve kodlanamayan veriler üretmesi nedeniyle bazı tekniklerle desteklenmesi gerektiğini söylemiştir. Bu tür durumları aşabilmek amacıyla bir metot geliştirmiş ve buna RASKE metodolojisi adını vermiştir. Bu metodoloji bir kiliseye bilişim sistemi geliştirilmesinde kullanılmıştır. İleriki çalışmalarda bu metodun büyük bilişim sistemlerinin geliştirilmesinde ihtiyaç toplama için büyük bir takım projelerde kullanılmasına yönelik çalışmalar gerçekleştirileceği belirtilmiştir [62].

Ratcliffe, senaryo tekniğinin stratejik karar verme sürecinde; nasıl oluşturulacağı, planlanacağı üzerine çalışmış özellikle son 30 yıldır bu tekniğin önem kazandığını, çeşitli işletmelerde, devlet dairelerinde ve silahlı kuvvetlerle ilgili projelerde kullanıldığını açıklamıştır. Geleceğin her zaman için belirsizliklerle dolu olduğunu fakat uygun teknik kullanılarak ve doğru yaklaşımlar uygulanarak bu belirsizliklerin planlanarak yönetilebildiğini savunmuştur. Çalışması senaryo metodu ile yapı endüstrisinde stratejik karar verme için alternatif planların tanımlanması, geliştirilmesi ve tersinde senaryoların nasıl oluşturulacağı, planlanacağı üzerinedir. Çalışmada yapı endüstrisinin geleceği için senaryo yaklaşımıyla oluşturulan planlar, senaryoların geliştirilmesi ve çeşitlerini içermektedir [63].

Kumar ve Palvia çalışmalarında, global EIS'nin (Yönetici Bilişim Sistemi) yönetimi, kullanımı, geliştirilmesi ve tasarlanmasına yönelik çalışmalar gerçekleştirmişlerdir. Çalışmada 48 global organizasyondan elde edilen bilgiler rapor olarak düzenlenmiştir. Rapor kapsamında global bir EIS'nin bilişim ihtiyaçlarının belirlenmesi ve geçerliliğine

yönelik metodolojilerde bulunmaktadır [64]. Bu metodolojiler aşağıda Tablo 2.1'de verilmiştir.

Tablo 2.1. Genel bir Yönetici Bilişim Sistemlerinde bilişim ihtiyaçları belirlenmesi ve geçerliliği için metodolojiler

Metodolojiler	Kullanım sıklığı (%)
Yöneticileri sorgulama	81,3
Kritik başarı faktörleri	79,2
Prototipleme	77,1
Mevcut bir uygulamadan türetme	41,7
Sonuç-Anlam analizi	33,3

Literatürde yoğun olarak yöneticilerin sorgulanması, kritik başarı faktörleri ve prototipleme kullanılmaktadır. Literatürde bu metodolojiler konusunda hemfikir olan araştırmacılar vardır [16]. Literatürde karşılaştırma (bench marking), beyin fırtınası, katılımcı ihtiyaç belirleme (JRD), bilişim analizi ve iş süreci gibi metodolojilerde bulunmaktadır [64].

Wetherbe, bilişim sistemlerinin tasarlanması ve inşasında yöneticilerin bilişim gereksinimlerini belirlemenin çok zaman aldığını, hatalara sebep olduğunu ve birçok yöneticinin zamanlarının çoğunu; toplantılarda, telefon konuşmalarında veya okumada harcandığını, bu aktiviteler sırasında kullanışsız bilgilerin elenmesi gerektiğini söylemiştir. Çalışmada bilişim ihtiyaçları ve bu ihtiyaçları karşılamadaki hataların giderilmesine yönelik dört problemin çözümüne yönelik 4 çözüm sunmuştur. Çalışmasında, bilişim ihtiyaçlarının hızlı ve doğru olarak belirlenmesinin, yöneticilerin doğru karar verme gereksinimlerinin sağlanması ve sağlıklı sistemlerin

geliştirilmesinde anahtar rol oynadığını, başarısızlıkların insan ve ekonomik kaynakların gereksiz şekilde kullanılmasıyla ortaya çıktığını söylemiştir. Bu tür problemlere çözüm üretmek amacıyla sistem tasarımcılarının "çapraz fonksiyonel ve JAD'ı (a cross-functional ve JAD) kullanarak bilişim ihtiyaçları belirlemenin hızını ve doğruluğunu arttırabildiğini, yeni bir sistem için kavramsal ihtiyaçların yapılandırılmış mülakatlarla, detaylı ihtiyaçların prototipleme yardımıyla belirlenebildiğini söylemiştir [65].

Kazman ve diğerleri, yazılım geliştirmede senaryo kullanımıyla geliştiriciler ve üst düzey yöneticiler arasında iletişimin arttığını ve bu sayede analistlerle (sistem geliştiriciler) sistemin kullanıcıları arasında ki iletişim problemlerine çözüm sağlayan bir deneysel çalışma gerçekleştirmişlerdir [38].

Witharana ve Zahedi yazılım geliştirmede ihtiyaç analizinin çok önemli olduğunu, bununla beraber doğru kullanıcı ihtiyaçlarının belirlenmesinin kullanıcı ve geliştiriciler arasında ki iletişim problemleri nedeniyle oldukça zor olduğunu söylemişlerdir. Çalışmalarında, ihtiyaç analizi (RA) aşamalarını şekillendirmeyi ve her bir aşama için uygun metodolojileri tanımlamayı amaçlamıştır. Burada ki her bir aşama sürecinin geliştirilmesi ve kolaylaştırılmasında grup dinamiklerini kullanan grup karar destek (GDSS) sistemleri bulunmaktadır. Çalışma kapsamında bir grup karar destek sistemi prototipi geliştirilerek onun ihtiyaç analizi üzerinde ki etkisi test edilmiştir. Çalışmalarında, ihtiyaç analizinin kalitesini geliştirme ve kolaylaştırmada her bir aşama için grup karar destek sistemi temelli

uygun metodolojileri tanımlayarak ihtiyaç analizi aşamalarını şekillendirmişlerdir [66].

Ocker ve diğerleri çalışmalarında; yazılım geliştirmede grup çalışmasının dört farklı iletişim modeli için etkinlik karşılaştırılmasını deneysel olarak gerçekleştirmişledir. İletişim modelleri; yüz yüze, eş zamansız bilgisayar konferansı, eş zamanlı bilgisayar konferansı ve yüz yüze, eş zamansız bilgisayar konferansının birlikte kullanımıdır. Deneysel çalışma sonucunda bütünleşik metodun diğer üç metoda oranla daha etkin olduğu görülmüştür. Değerlendirme de başarı ölçütü olarak; yaratıcılık, kalite, çözüm tatmini ve süreç tatmini kullanılmıştır. Çalışmada durumlar kontrollü olarak seçilmiş kısacası rasgele alınmamıştır. Bu nedenle daha farklı deneysel çalışmalara gereksinim vardır [48].

Flynn ve Arce, işletme gereksinimleri ve bilişim teknolojisi (IT) sistemlerini destekleyen bir Bilgisayar Destekli Yazılım Mühendisliği (CASE tool) Araçları tanımlamışlardır. Bu araç, ihtiyaç belirleme ve bilişim teknolojisi planlama için KBF temellidir. Bu araç, ihtiyaç belirleme, IT planlamayı mantıklı ve sistematik olarak adreslemektedir. Bu yaklaşımın kullanımıyla; işletme ve teknik organizasyonel değişkenlerin içerikleri veritabanına kaydedilmekte ve bunlar daha sonraki işletme analizleri için temel alınmakta, bu bilgi hesaplamalarda ve raporlarda kullanılmaktadır [67].

Sommerville ve diğerleri, endüstriyel kullanım için çok perspektifli bir ihtiyaç mühendisliği yaklaşımı geliştirmişler ve buna Öngörüş

(PREview) adını vermişlerdir. Bu sayede sistem ihtiyaçlarını türetmiş ve organize etmişlerdir. Bu bakış açısının temelinde ihtiyaçların çıkartılması ve geçerliliği yatmaktadır. Bu amaçla sistem kullanıcılarına sorular yöneltilmiş ve alınan cevaplar ihtiyaçları türetmede kullanılmıştır. Bu yaklaşım sadece mevcut ihtiyaçlardan hareketle çalışabilmektedir [8].

Harker ve Eason, sistem geliştirme sürecinde paylaşımcıların ihtiyaçlarını üretmek amacıyla senaryolardan faydalanmışlar ve sistem geliştirme çalışmalarında senaryo tekniğinin kullanıcı ihtiyaçlarını belirlemede oldukça etkin ve uygun olduğunu göstermişlerdir [68].

Senaryoların yazılım ihtiyaçlarının çıkartılması, tanımlanması ve geçerliliğinde (doğrulanması) etkinliği kanıtlanmıştır. Bununla beraber pratikte senaryo tekniğini kullananlar bazı problemler yaşamaktadır. Bu problemler için bir sebep, senaryoların sayısının üstel olarak büyümesinden kaynaklanan senaryolar arası oluşabilecek ilişkilerin sayısıdır. Alspaugh ve diğerleri, bu ilişkiler formülüze edilebilirse; senaryoların oldukça kolay tanımlayıcı ve destekleyici olabileceğini söyleyerek çalışmalarında iki yaklaşım sunmuşlardır. Birinci yaklaşım, senaryolar arasında mevcut ortak hikayeciklerin tanımlanarak senaryo elemanlarının paylaştırılmasıdır. Bu sayede senaryoların tutarlılıkları test edilmiş olacaktır. İkinci yaklaşımda senaryolar arasında ki benzerlik ölçümü yaklaşımıdır. Bu iki yaklaşım senaryoların gelişimi ve yönetimi için etkin ve faydalı olduğu görülmüştür. Bir örnekle iki yaklaşımda çalışma içinde açıklanmıştır [69].

Sistem analizinde bilişim ihtiyaçlarını belirleme aşaması sistem geliştirme çalışmalarının başarısında oldukça önemli bir yere sahiptir. Pratikte ihtiyaçların çıkartılması bilgi toplama süreci üzerindeki kişilerin ve politikaların etkisiyle genellikle karmaşık ve problemlidir. Bu amaçla Taylor çalışmasında mülakat yeteneğinin geliştirilmesi amacıyla alternatif bir metot kullanmış, bu metot; role-play cases olarak adlandırılmıştır. Yetenek tabanlı eğitim araştırmalarında rol oynama olayları öğrencilere uygulanarak ilk sistem ihtiyaçlarının araştırılması ve geliştirilmesi sağlanmıştır [43].

Ko, yazılım geliştirme maliyetlerinin geniş bir yüzdesinin yazılım geliştirme sürecinin öncelikli fazına harcandığını ve bu fazın bilişim ihtiyaç analizi olarak adlandırıldığını söylemiştir. Çalışmasında, yapılandırılmış mülakat, grup oturumları ve senaryo-analiz tekniği kullanarak bir yazılım projesi gerçekleştirmiştir. Ayrıca etkin bir bilişim sistemi geliştirmek amacıyla, bilişim ihtiyacı toplama sürecinde hem üst yönetim hem orta kademe yöneticiler hem de son kullanıcıların bilgilerinden faydalanmıştır [7].

Byrd ve diğerleri bütünleşik çıkarım tekniğinin uygulanmasının tek bir teknik uygulamaya oranla daha faydalı olduğunu, bilişim ihtiyaçları toplama da, üst düzey yöneticiler, orta kademedeki yöneticiler ve son kullanıcıların bilgisi önderliğinde daha etkin bir bilişim sistemi geliştirmenin mümkün olduğunu savunmuşlardır [19].

Usher ve Kaber, FMS operasyonları gibi karmaşık süreç kontrollerine müdahale için bilişim ihtiyaç analizinden faydalanmış, analizde

kavramsal bir mühendislik araştırma metodolojisi kullanmışlardır. Bu metodoloji imalat alanında daha öncede kullanılmıştır.

Geleneksel üretim ve atölye tipi üretimden FMS'e geçildiğinde karar vermeye yönelik bilişim ihtiyaçları ve imalat amaçlarında değişiklikler olduğu gözlenmiştir. Amaca-Yönelik Görev Analizi (GTA/Goal-Directed Task Analysis) Endsley tarafından geliştirilmiş bilişim ihtiyaç değerlendirme metodolojisidir. Bu metot genel olarak [70];

- Karmaşık sistemlerin amaçlarını tanımlamayı, (dar boğazları ve problemli noktaları belirlemeyi),
- Alt amaçların elde edilmesini tanımlamayı,
- İş performansında karar vermede kritik soru oluşturmayı ve
- Operatör bilişim sistemlerini geliştirmede bu soruları cevaplamayı içerir.

Rogich ve Browne ihtiyaç belirlemenin, sistem geliştirmede çok önemli bir safha olduğunu fakat buna rağmen bu konuya yönelik çok az sayıda deneysel çalışma olduğunu söylemektedirler. Bu çalışmada birkaç soru sorma yaklaşımı geliştirilmiş ve test edilmiştir. Bu deneysel çalışmalar iş karekteristiği tekniğinin (Task Characteristics Technique) diğer iki tekniğe oranla daha yüksek miktarda ve detaylı ihtiyaç çıkardığını göstermişlerdir. Ayrıca bu tekniğin kavramsal zorlukların üstesinden gelmede analistlere ihtiyaç belirleme konusunda yardımcı olduğunu söylemişlerdir [38].

Coronado ve diğerleri, bilişim sistemlerinin çevik imalat sistemlerinin faydalarıyla da ilgilendiğini, imalat süreçleri içinde bilişim sistemlerinin büyüyen önemine rağmen literatürde çevikliği destekler yönde sistem ihtiyaç incelemelerine ait çok az örnek olduğunu söylemişlerdir. 14 imalat işletmesinde gerçekleştirilen çalışmanın sonuçları esnek bir yapının önemini ortaya çıkarmış ve bunu da işletme süreçlerinin desteklenmesi bileşen ve uygulamaların (ilave edilerek, modifiye edilerek ve ortadan kaldırılarak) güncelleştirilmesiyle sağlanabildiğini göstermiştir. Gerçekleştirdikleri çalışmada çevikliği destekleyen bilişim sistem ihtiyaçlarını tanımlamak için bir yapı tanımlamış ve bu yapının uygun olduğunu savunmuşlardır [71].

Moore ve Shipman, yazılım geliştirmenin kullanıcılardan, iş pratikleri ve tasarım seçenekleri hakkında bilgi toplamayı içerdiğini ve geleneksel olarak iki yolla gerçekleştiğini söylemiştir. Bunlar; mülakat ve katılımcı tasarımıdır (partcipatory design/PD). Bu tarz karşılıklı iletişim gerektiren metotlarda, yazılım mühendisleri ve kullanıcılar arasında iletişim problemleri ortaya çıktığından, analistler ve yazılım mühendisleri genellikle anket ve diğer kağıt temelli metotları kullanmayı tercih etmişler ve yeterli bilgi toplayamamışlardır. Bu çalışmada taslak ara yüzler aracılığı ile toplanan ihtiyaçlarla anket temelli olarak toplanan ihtiyaçları karşılaştırmış (Graphical Requirements Collector/GRC) fakat GRC'nin mevcut ihtiyaç toplama metotlarının yerini alamayacağını ancak destek bilgi sağlayacağını savunmuşlardır. GRC ve anket kombinasyonunun ihtiyaç çıkarmada

yazılım mühendisleriyle son kullanıcılar arasında ki iletişimi olumlu yönde etkileyeceğini savunmuşlardır [72].

Darke ve Jagielska sistemin tüm kullanıcılarının gereksinimlerini karşılamada ihtiyaç yetersizliklerinin, sistemin kabiliyeti üzerinde etkin rol oynadığını ve asıl önemli olanın o alana yönelik tüm uygulamaların (aktivitelerin) anlaşılarak ihtiyaç kümesinin sağlanması olduğunu söylemişlerdir. Çalışmalarında, bilgi keşfinden faydalanan bir yaklaşımla geçmiş verilerden hareketle sistem analistleri ve kullanıcıların uygulama alanını daha iyi anlamalarını sağlayarak ihtiyaç belirlemeyi geliştirmeyi amaçlamışlardır. Yapay sinir ağları ve kaba kümeleme tekniklerini karma bir şekilde kullanarak teknolojik bir çatı ve bilgi keşfi yardımıyla da ilgili alandaki uygulamaları detaylandıran bir süreç modeli tanımlamışlardır. Ayrıca çalışma bu yaklaşımın kullanılmasının pratikte etkin olduğunu gösteren örnek bir çalışmayı da içermektedir [73].

Literatürde çıkarımdan (requirements elicitation) paylaşımcıların gereksinimlerinin tümünün belirlenmesi olarak bahsedilmektedir. İhtiyaç çıkarma modellerinin çoğu özel metodolojiler veya tekniklere odaklanmıştır. Örneğin; Robertson'lar çalışmalarında geliştirdikleri ihtiyaç metodolojisi ihtiyaç çıkarım aktivitelerinin her bir aktivitesi için teknik öneren detaylı bir süreç modelini içermektedir [74].

Carroll ve Swatman, yapılandırılmış olayları kullanarak veri toplama problemlerinin nasıl aşılabileceği ve toplanan verilerle ihtiyaç mühendisliği sürecinin daha iyi anlaşılarak geliştirileceği üzerine bir

çalışma gerçekleştirmişlerdir. Burada asıl gerçekleştirilen gözlem metodunun yapılandırılmasıdır. İhtiyaç Mühendisliği (İM) sürecinin daha iyi anlaşılabilmesi amacıyla araştırmacılar tarafından kullanılan bütünleşik veri toplama metotları;

- Katılımcı gözlemi (Participant observation),
- Mülakat'tır.

Sonuç olarak, bu sayede İM süreci için çok daha detaylı veri toplanabilmektedir [75].

Salmeron ve diğerleri, İspanya'da ki her sektörden üç en büyük şirketi EIS geliştirme maliyetleri açısından incelemiş ve istatistiki veriler elde etmişlerdir. Çalışma verileri maliyet açısından değerlendirildiğinde, sistem geliştirme maliyetlerinin % 30'unu geliştirme/uygulama % 70'ini de bakım kapsamaktadır. Bakım maliyetlerinin bu kadar yüksek olması ihtiyaç belirleme sürecinde, ihtiyaçların yanlış ve yetersiz belirlenmesinden kaynaklanan maliyet artışlarıdır [76].

Browne ve Rogich, sistem geliştirmede kullanıcılardan ihtiyaç çıkartılmasının genellikle kavramsal, iletişimsel ve motivasyonel sebeplerle oldukça zor olduğunu bu tür problemler nedeniyle sistem geliştirme çalışmalarında başarısızlıklar yaşandığını belirtmişlerdir. Çalışmalarında sistem geliştirme çalışmalarının iyileştirilmesi için ihtiyaç çıkarımının kontrol altına alınarak geliştirilmesi gerektiğine değinmiş ve bu amaçla bir model geliştirmişlerdir. Model kapsamında, öncelikle sistemin kullanıcıları ve analistler arasında iyi bir iletişim

olması gereğini, ikinci olarak ihtiyaçların çıkartılmasında teori temelli yönlendirme tekniğinin geliştirilmesi ve üçüncü olarak da toplanan bilginin düzenlenmesi amacıyla ihtiyaç sınıflarının kullanılmasını önermişlerdir. Bu ihtiyaç sınıflarının daha sonra ki çalışmalarda ihtiyaç toplamada durdurma kuralları olarak kullanılabileceğini ve bunun analistlere rehberlik edeceğini savunmuşlardır [15].

Mannio ve Nikula, ihtiyaç mühendisliği yaklaşımında; mevcut prototipleme yöntemleri, prototipleri ve prototiplerle senaryoların birlikte kullanımlarını incelemişlerdir. Sonuç olarak prototiplerin paylaşımcılarla iletişimi artırdığını, senaryolarında yazılım sürecinde yardımcı olduğunu görmüşlerdir. Fakat burada sadece prototip ve senaryolar kullanıldığı diğer metotlar ihmal edildiği için bir eksiklik söz konusudur. Bu nedenle karşılaştırmalı bir çalışmaya gereksinim vardır [77].

Park ve diğerleri çalışmalarında; ihtiyaç analizini desteklemeyi amaçlamışlardır. Çalışmalarında muğlak, gereksiz ve uyuşmayan ihtiyaçları ihtiyaç cümleleri arasındaki benzerlikten hareketle ölçmeyi amaçlamışlar ve bu amaçla iki metot kullanmışlardır. Kullanılan bu metotlar yardımıyla arzu edilen sistemin; ihtiyaç kalitesini iyileştirmeyi gerçekleştirmişlerdir. Çalışmalarında ihtiyaçlarda oluşan potansiyel hataları belirleyerek ihtiyaç analizini destekleyen bir sistem önermişlerdir. Bu sistem; dokümanlar arasındaki bağımlılığı; bir ihtiyaç dokümanının kalitesini iyileştirerek (geliştirmeyi), cümleler arasındaki tutarsızlıkları düşürerek ve tamlığı arttırarak desteklemektedir. Bu fonksiyonları sağlamak için benzerlik ölçümü ve şema indeksleri

kullanılmıştır. Burada bütünleşik olarak kayan pencere yöntemi (sliding window metot) ve sözdizimsel ayrıştırıcı yöntem (syntactic parser metod) kullanılmıştır. İhtiyaç dokümanlarından kaynaklanan problemlerin tanımlanmasında analistlere destek sağlayan bir sistem geliştirilmiştir [78].

Lai'nin çalışması bilişim sistemlerinin geliştirilmesinde organizasyonal ihtiyaç analizinin etkinliğini artırmak amacıyla nesneye yönelik metodoloji ve sistem biliminin birlikte kullanımına yöneliktir. Bu iki metodoloji gerçek yaşam çalışmalarına uygulanarak formülüze edilmekte ve birlikte kullanımı önerilmektedir [79].
Lazar ve diğerleri, kullanıcı ihtiyaçlarını belirlemenin önemli olduğunu göz önüne alarak bir web tabanlı site geliştirmişlerdir. Çalışmada kullanıcı ihtiyaçlarını toplamak için farklı bölge ve yaşta ki kişilere yönelik farklı teknikler kullanılmıştır. Yakın mesafe ve bölgelerde yüz yüze mülakat yöntemi, uzak bölgeler için ise elektronik mail yöntemi ile kullanıcı ihtiyaçları toplanmaya çalışılmıştır. Makale de kullanıcı ihtiyaçlarını toplamada elektronik mail kullanımına ait incelemeler ve bir örnek çalışma gösterilmiştir [6].

Hickey ve Davis, doğası gereği yazılım geliştirmenin birçok süreçten oluştuğunu ve burada ki en büyük zorluğun ihtiyaçların çıkartılması süreci olduğunu söylemişlerdir. Çalışmalarında, bilginin kritik rolünü açıkça ortaya koyan ihtiyaç çıkarma süreçlerine ilişkin bir matematiksel model sunmuşlardır. İhtiyaç çıkarma sürecinde, uygun çıkarma tekniğinin seçimi bu model aracılığı ile gerçekleştirilmiştir. Model

tekniklerin seçiminde bilginin kritik rolünü açıkça ortaya koymaktadır [80].

Browne ve Pitts, tasarım problemlerinde analistler tarafından durdurma kurallarının kullanımı üzerine bir çalışma gerçekleştirmişlerdir. Çalışmada bilgi toplama sürecinde durdurma kurallarından faydalanılması durumunda az deneyimli analistlerinde çok deneyimli analistler kadar, çıkarttıkları bilginin kalitesi ve niceliği bakımından başarılı oldukları görülmüştür [14].

Eva, bilişim sistemi geliştirmede kullanıcı ihtiyaçlarını tanımlama ve karşılamada Hızlı Uygulama Geliştirme (HUG) tekniği (Rapid Aplication Development) ve İM yaklaşımından hareketle iki deneysel çalışma gerçekleştirmiştir. Fakat çalışmada HUG tekniğinin nasıl uygulanacağı ve ihtiyaçların çıkartılacağı konusunda daha geniş çaplı bir çalışmaya gereksinim olduğu bu nedenle de her iki disiplinin uygulamada etkileşimli olarak kullanılması amacıyla bir araştırma çalışmasına gereksinim duyulduğu saptanmıştır [81].

Alvarez çalışmasında bilgi çıkarımı ve ihtiyaç analizi sırasında mülakatların denenmesi için bir çatı tanımlamıştır. Araştırma bilgi ve ihtiyaç çıkarım tekniklerinin birebir aynı değillerse de çok benzer olduklarını göstermektedir. Her ikisinde de amaç müşterilerden (işlemcilerden), alan uzmanlarından veya son kullanıcılardan bilgi edinimine yöneliktir. Ayrıca çalışma bilgi/ihtiyaç çıkarım süreci sırasında analist ve işlemciler (kullanıcılar) arasında bir “iletişim problemi” olduğuna değinmekte, bu durumun analistler ve kullanıcılar

arasında farklı bir bakış açısı ve kavramsal sınırlar nedeniyle problemlere neden olduğunu göstermektedir. Bu tür problemlerin aşılabilmesi amacıyla mülakat, çıkarım teknikleri arasında favori olan ve tercih edilen bir tekniktir. Araştırmalar mülakatın veri çıkarım açısından oldukça zengin fakat zor bir metot olduğunu göstermiştir. Çalışmada mülakat sırasında oluşan boşlukların aşılabilmesi amacıyla hikayeler üretilmiş, bu sayede kullanıcılar ve analistler arasındaki iletişim olumlu yönde arttırılmıştır [20].

Kassel ve Malloy otomatik olarak ihtiyaçların çıkartılması ve özelliklendirilmesi üzerine başarılı olmuş tekniklerin kombinasyonundan oluşan, gerçek ve komple ihtiyaç çıkarmayı destekleyen bir prototip araç geliştirmişlerdir. Bu aracın, müşteri, kullanıcı, yazılım mühendisleri ve alan uzmanlarının beraberce ve bağımsız çalışmasına uygun olduğunu ve ihtiyaç çıkarmayı olumlu yönde etkilediğini söylemişlerdir.

- Yazılım mühendisleri ve analistlerin alanı anlamada zorlanması ve,
- Kullanıcılar, alan uzmanları, müşterilerin yazılım konusuna yabancı olması nedeniyle bu iki grup arasında oluşan iletişim problemine köprü olabilecek bir yaklaşım sunulmuştur.

Bununla beraber, XML (Extensible Styleseet Language) formatında alan bilgisi toplama ve sunma oldukça bezdirici olması nedeniyle bir sonra ki çalışmalarında GUI tabanlı bir yapıyla alan bilgisinin toplanması ve sunumunun kolaylaştırılması planlanmaktadır [82].

Zhu, ihtiyaç belirleme ve ihtiyaçları yapılandırmanın, sistem analizinde iki ana bileşen ve analiz safhasının ilk aşaması olduğunu söylemiştir. Bu aşamanın mümkün olan her kaynaktan bilgi toplamayla başladığını, bu amaçla ihtiyaçları toplamaya yardımcı olacak bazı geleneksel metotlar olduğunu bunların da; mülakat, anket, inceleme ve kullanıcıların doğrudan gözlemi olduğunu söylemiştir. Günümüzde JAD ve prototipleme gibi bazı modern ihtiyaç toplama metotları da ortaya çıkmıştır. JAD ve prototipleme önceki geleneksel metotlar temel alınarak geliştirilen modern ihtiyaç belirleme metodolojileridir [83].

Zhu çalışmasında, mülakat, anket, gözlem, doküman analizi, JAD ve prototipleme metodolojilerini ele almış ve bunları bilgi zenginliği, zaman gereksinimi, maliyet, gizlilik vb. yönlerden birbiriyle kıyaslamıştır. Kısaca çalışmanın amacı temel ihtiyaç belirleme ve ihtiyaç yapılandırma metotlarını kıyaslayarak bunların farklılık durumlarını belirlemek ve sistem ihtiyaç analizi için en iyi metotları tanıtmaktır [83].

Ani ve Edwards, İhtiyaç belirlemede sisteme ait dökümanların sürekli olarak incelenmesi, yeniden gözden geçirilmesi ve sistemin paylaşımcılarının ihtiyaçlarının anlaşılmasının sistem geliştirmede arzu edilen ve beklenen bir durum olduğunu söylemişlerdir. Paylaşımcılar alanla ilgili yetersiz bilgiye veya iyi yapılandırılmamış dokümanlara sahip olduklarında problemler ortaya çıktığından, çalışmada bu tür problemleri belirleyen bir nitel sistematik yaklaşım geliştirilmiştir. Bu yaklaşımı, İhtiyaç analizinde nitel sistematik yaklaşım olarak adlandırmışlardır. (QSARA/Qualitative Systematic Approach to

Requirements Analysis). Çalışmalarında, deneysel çalışmanın tasarımı, ayrılan kaynakların tanımlanması ve çalışma sırasında toplanan istatistiki veri analizi detaylı olarak tanımlanmaktadır. Bu amaçla 70 katılımcıdan toplanan kanıtlar QSARA'nın amaca yardımcı olduğu ve hipotezi desteklediğini göstermektedir. Sonuç olarak grafiksel bir sunum elde edilir. Bu sayede analistlerin alanı daha iyi anlaması ve algılaması sağlanmış olur [84].

Freeman, analistlerin sistem geliştirmede ihtiyaç çıkarma safhasında kavram haritası (concept-map) tekniği sayesinde daha sağlıklı ihtiyaçlar çıkartacağı hipotezi savunulmuş fakat çalışma sonucunda bu tezin çürütüldüğü veya bir başka deyişle ters bir sonuç verdiği görülmüştür [85].

Sandy, yeterince tamamlanmamış sistem hatalarının önemli organizasyonel ihtiyaçların giderilmemesinden kaynaklandığına dair çok sayıda yazılı kaynak bulunduğundan bahsetmiş. Bu hataların en önemli sebebinin organizasyonel kuralların analizinin açıkça gerçekleşmemesinden kaynaklandığını savunmuştur. Kuralların ihtiyaçların yakalanmasında ve modellenmesinde anahtar olduğunu, kuralların ihtiyaç mühendisliğinde "eksik bağlantılar" olduğunu söylemiştir. Kontrollü laboratuar ortamındaki deneyler sonucunda ihtiyaç mühendisliğinde kural tabanlı bir yaklaşımı doğruladığını, bu tür bir yaklaşım sayesinde ihtiyaç mühendisliğinin temeli olan ihtiyaç belirleme işi ve problemine de çözüm bulunacağını savunmuştur. Fakat gerçek kullanıcılarla ve gerçek yaşam durumlarında test edilmesi gerektiğini söylemiştir [86].

Tuunanen ve diğerleri çalışmalarında bir bilişim sisteminin dağınık şekildeki kullanıcılarına yani; müşteriler, tedarikçiler, ortaklar ve diğer tüm kullanıcılara yönelik ihtiyaçların belirlenmesi için yeni bir metodu örnek olay şeklinde vermişlerdir. Burada ihtiyaç mühendisliğinde son kullanıcılara yönelik yedi problem tanımlanmaktadır. Burada esnek bir yapılandırılmış mülakat, kavramsal modelleme (CSC), yorumlayıcı analizi ve bir sunum tekniği kullanılarak çalışma gerçekleştirilmiştir. Büyük bir bilişim sisteminin ihtiyaçları kullanıcılardan hareketle geliştirilmeye çalışılmıştır. Bu "Helsinki Sanomat" adlı Finlandiya'nın büyük bir gazetesidir. WARE diye adlandırılan bu metodoloji yöneticilere ve geliştiricilere kullanıcıların tercihlerini, önceliklerini ve muhakeme yapılarını anlamada oldukça faydalı olmuştur [87].

Browne ve Ramesh bilişim ihtiyaçlarını belirlerken;

- Ön belirleme,
- Doğrudan belirleme,
- Dolaylı belirleme ve
- Dışa vurum tekniklerini kullanmışlardır.

Bu teknikler içinde birçok farklı araç bulunmaktadır. Bu teknikleri kullanırken her bir teknik için; Doğrudan sorular, Ne-Eğer (What-if) analizi, Senaryolar, Şeytanın avukatlığı, Akış şemaları, Bilgi haritaları, Etki diyagramı, Karar haritası, Yakınlık diyagramı ve Not tahtası gibi araçlardan yararlanmışlardır [30].

Browne ve diğerleri bir tasarım sürecinde analistin amacının yeterli miktarda bilgi toplamak olduğunu ve bu bilginin ölçülmesi gerektiği fikrinden hareketle; ihtiyaçların bir sınıflandırması gerçekleştirmişlerdir. Bu sınıflandırma şeması bilişim sistemlerinin başarılı bir şekilde tasarlanması amacıyla kritik olduğuna inanılan varlıkların problem alanlarını içermektedir. Bu şema aynı zamanda durdurma kuralları gibi kullanılmış ve bu sayede eksik veya fazla bilgi çıkarımı engellenmiştir [29].

İhtiyaç mühendisliği metotlarında çoklu ortam uygulamalarına dayalı karşılaştırmalı teknoloji hızla artmaktadır. Bu çok açık olarak görülmekle beraber örneğin animasyon ve hikaye teknikleri yardımıyla paylaşımcıların geçerliliği geliştirdiği görülmektedir.

Gemino'nun çalışması ihtiyaçların doğrulanması, geçerliliği için paylaşımcılara bilgi sunumunda animasyon ve hikayelemenin faydalı olduğunu gösteren ilkler arasında yer almaktadır. Buna ilaveten çoklu öğrenmede kavramsal teorinin, ihtiyaç mühendisliği tekniklerinde tasarlama ve geliştirmede önemli bir araç olduğu savunulmaktadır. Ayrıca bu çalışmada animasyon hikayelemeden ayrılırsa ve tek başına kullanılırsa çok etkin olmadığı görülmüştür. Ama bunun kanıtlanabilmesi için daha çok çalışmaya gereksinim vardır. Bu çalışma sadece 58 kişi üzerinde denenmiştir. İhtiyaç mühendisliğinde ki zorluklardan birisi analiz sürecinde anlama ve iletişime yönelik konulardır. Bu amaçla daha etkin analiz tekniklerine ihtiyaç vardır. Bu çalışmanın sonuçları hikayeleme ve animasyonun bu konuda yardımcı olduğu yönündedir [88].

Huang'a göre doğru olmayan ihtiyaç tanımlamaları; büyük oranda bilişim sistem hatalarına sebep olmaktadır. Veri akış diyagramı ve nesne modeli gibi ihtiyaç tanımlama teknikleri, ihtiyaç tanımlamalarının doğruluğunu geliştirmede, analistlerin bilişim ihtiyaçlarını yakalamasında, anlamasında ve sunmasında yardımcıdır. Ayrıca analistlerin "kavramsal yetenekleri" ihtiyaç tanımlarını doğru olarak belirlemelerinde çok önemlidir. Kendi kendine –sorgulama "kavramsal yetenekler" geliştirmede okuma anlayışı ve problem çözme üzerine öğrenme araştırmaları alanında bir strateji olarak tanımlanmıştır. Bu amaçla çalışmada deneyim sahibi olmayan analistlerin kavramsal yeteneklerini geliştirmede kendi kendine sorgulama mekanizması önerilmiş ve ihtiyaçları doğru olarak tanımlamada başarılı sonuçlar elde edilmiştir [89].

Huang araştırmasında, deneyimsiz bilişim analistlerinin modelleme performansını geliştirmek için bilgisayar bütünleşik yazılım mühendisliği araçları (CASE tool) içine alan ve modelleme bilgisine odaklanmıştır. Bununla beraber bu araştırma ihtiyaç tanımlama için deneyimsizlikten kaynaklanan kavramsal limitlerin nasıl geliştirileceğini içermektedir. Araştırma da kendi kendine sorgulamanın etkinliği için bir tasarım stratejisi önerilmiştir. Bu sayede deneyimsiz anlistler daha doğru ihtiyaç tanımlamalarına rehberlik eden bir teknik tanımlamış olur [89].

Çıkarsamalar

İhtiyaç belirleme araştırma sonuçlarına göre; tüm problemlerin aşılabilmesi, sistem geliştirme çalışmalarının başarılı bir şekilde sonuçlanabilmesi, ihtiyaçlar kümesinin en uygun ve eksiksiz şekilde belirlenmesine bağlıdır. Bu nedenle birçok bilim adamı ve araştırmacı bu konuya yönelik araştırmalar gerçekleştirmiş, ihtiyaç belirlemeyle ilgili birçok teknik ve araç geliştirmiştir. Fakat ihtiyaç belirleme, ihtiyaç belirleme/geliştirme teknikleri, stratejileri, ihtiyaç sınıflandırma ile ilgili çok sayıda çalışma olmasına karşın ihtiyaç belirleme tekniklerinin karşılaştırılması ve ihtiyaç belirleme sürecinin modellenmesine yönelik araştırmalar çok kısıtlıdır. İhtiyaç belirleme konusunda yapılan araştırmalar, ihtiyaç belirleme tekniklerinin karşılaştırılmasında ve bu tekniklerin bütünleşik kullanımlarına ilişkin modellere ihtiyaç duyulduğunu ve bu alanda bir boşluk olduğunu belirlemektedirler.

Bilim adamları genellikle bilişim ihtiyaçlarının belirlenmesinde çoklu teknik kullanımın, ihtiyaçların çıkarılmasında daha uygun olduğunu savunmaktadırlar. Ancak çoklu tekniklerin nasıl kullanılacağına ilişkin evrensel bir model verilmemiştir. Organizasyonel süreçleri iyileştirmede; döngü zamanlarını indirgeme, tedarik zinciri optimizasyonu veya işletme süreçlerinin yeniden yapılandırılması önemli bir yer tutmaktadır. Bu çalışmaların gerçekleştirilebilmesi için de verilere ihtiyaç vardır. Bu verilerin veya bilgilerin de paylaşımcılardan; yani çalışanlar, müşteriler ve tedarik zinciri elemanlarından toplanması önerilmektedir.

Analiz süreci ve bilişim ihtiyaçlarının belirlenmesinde iki potansiyel problem mevcuttur: Birincisi; bilginin toplanması ve analitik süreçte oldukça önemli olan kişilerin davranışları, yetenekleri, yargılama ve çıkarsama (algılama, idrak) yetileridir. İkincisi ise; geleneksel ihtiyaç belirleme süreçlerinin genellikle hatalı süreç ihtiyaçları veya tam olmayan belirlemeler nedeniyle süreçlerin yetersizliğidir. Bu durumda genellikle hatalı ihtiyaç belirleme veya süreç iyileştirme çalışmalarına neden olmaktır.

Sonuç olarak, ihtiyaç belirleme alanında ihtiyaçları toplama/çıkarma sürecinde farklı metotlardan yararlanabilen, eksik ihtiyaç toplamadan kaynaklanabilecek problemleri aşabilecek ve birden fazla metodu yada tekniği birlikte kullanabilecek bir modele ihtiyaç duyulmaktadır. Üçüncü bölümde bu amaçla bu çalışma kapsamında geliştirilen bir model sunulmuştur.

BÖLÜM 3.

İHTİYAÇ BELİRLEME İÇİN BİR MODEL

Bu bölümde ikinci bölümde tartışılan literatür araştırması sonuçlarına göre ihtiyaç belirlemede ki problemleri aşabilmek için geliştirilen bir model önerilmiş ve modelin işleyişi tartışılmıştır.

Genel sistem geliştirme çalışmalarında ihtiyaçların belirlenmesi amacıyla, kullanılabilecek model üç aşamadan oluşmaktadır. Modelin birinci aşaması ihtiyaç belirleme sürecini, ikinci aşaması kural çıkarma ve teknik/ihtiyaç matrisi oluşturma sürecini ve üçüncü aşaması ise sunum tekniklerinin ürettiği ihtiyaç seviyelerini içermektedir. Model ve adımları aşağıda detaylı bir şekilde açıklanmıştır.

Birinci aşama olan, ihtiyaç belirleme sürecide kendi içinde dört alt aşamaya ayrılmaktadır. Bunlar; ihtiyaçların toplanması, toplanmış olan ihtiyaçların sunumu, alan uzmanlarınca sunulan ihtiyaçların doğrulanması ve sonuç olarak elde edilen ihtiyaçların sınıflandırılmasıdır. İhtiyaçların toplanması amacıyla;

- Anket,
- Gözlem,
- Mülakat,
- Yazılım inceleme,
- Literatür tarama ve
- Doküman analizi yöntemlerinden faydalanılmıştır.

İhtiyaçların sunulması amacıyla;

- Akış şeması,

- Çağrıştırıcı bilgi haritası,

- Karar haritası,

- Senaryo ve

- Benzerlik diyagramı tekniklerinden faydalanılmıştır.

Belirlenmiş olan tekniklerle sunulan ihtiyaçlar, sistemin alan uzmanlarınca doğrulandıktan sonra ise literatürde kullanılmakta olan bir şablon aracılığı ile dört farklı seviyede sınıflandırılmıştır. Bu seviyeler; amaç, süreç, görev ve bilişim seviyeli ihtiyaçları içermektedir. Bu ihtiyaç seviyelerinin, belirlenen ihtiyaçların yeterliliğinin kontrolünde şablon olarak kullanılması önerilmiş ve her ihtiyaç seviyesinin yeterli miktarda ihtiyaç içerdiği savunulmuştur [15]. Bu sayede belirlenen ihtiyaçların doğrulanması aşamasına katkı sağlanmıştır.

İkinci aşamada; ihtiyaç ve tekniklerin gruplanmasında, imalatta parça ve makinelerin gruplanmasında yaygın olarak kullanılan ROC algoritması çalışmaya uyarlanmış ve modelin evrenselleşmesi (genelleştirilmesi) amacıyla endüktif öğrenme, yapay zeka tekniği ve RULES-3 algoritması kullanılmıştır.

Üçüncü aşamada ise; sistem geliştirme çalışmalarında tüm ihtiyaçları kapsayan dört farklı ihtiyaç seviyesini ele alan bir ihtiyaç belirleme modeli geliştirilmiştir. Geliştirilen model Şekil 3. 1'de şematik olarak gösterilmiştir.

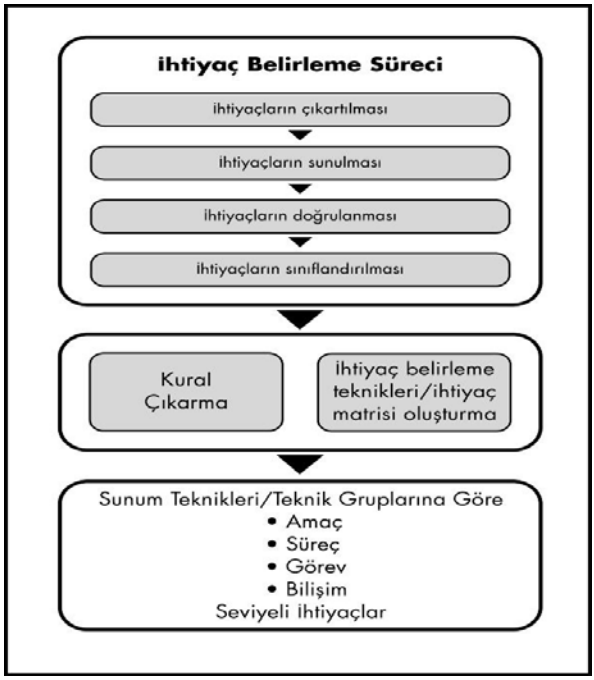
Önerilen Modelin Tanımlanması

İhtiyaç belirleme veya bilişim ihtiyaçları belirleme çalışmaları genellikle;

- Yeni bir sistem geliştirme sırasında (bilişim sistemi) veya
- Mevcut sistemin iyileştirilmesi sırasında gerçekleştirilir.

İhtiyaç belirleme, sistem geliştirme yaşam döngüsü metodunun ilk aşaması olan sistem analizinin bir alt aşmasıdır ve genellikle bilginin toplanması (ihtiyaçların toplanması), modellenmesi ve doğrulanması olarak üç aşamada özetlenmektedir. Ayrıca ihtiyaç sınıflandırma sayesinde ihtiyaçlar kümesinin kontrolü ve ihtiyaçların doğrulanması sağlanmaktadır. Bu sebeple modelin birinci aşaması dört basamaklı olarak geliştirilmiştir.

Gerçekleştirilen çalışmada; Öncelikli olarak "İhtiyaç Belirleme Süreci" ve sürecin aşamaları üzerine yoğunlaşılmış, bir sonraki aşamada ROC algoritması kullanılarak ihtiyaç grupları ve teknik grupları oluşturularak bu sayede ihtiyaç gruplarına göre hangi tekniklerin yada teknik kombinasyonlarının daha etkin olduğunun belirlenmesi gerçekleştirilmiştir. Bu çalışmaya eş zamanlı olarak da, yapay zeka tekniklerinden biri olan endüktif öğrenme yaklaşımı ve araç olarak da RULES-3algoritması yardımıyla ihtiyaç gruplarına göre teknik/teknik kombinasyonlarının etkinliği değerlendirilmiştir. Sistem ihtiyaçlarını modellemenin şematik gösterimi şekil 3.1'de sunulmuştur.



Şekil 3.1. Sistem ihtiyaçlarını modellemenin [SİM] şematik gösterimi.

Modelin Çalıştırılması

Bu bölümde SİM'in alt süreçleri olan; ihtiyaç belirleme, kural çıkarma, teknik/ihtiyaç matrisi oluşturma ve seviyelerine göre ihtiyaçları elde etme, kısaca önerilen modelin çalışması sırasıyla açıklanmıştır.

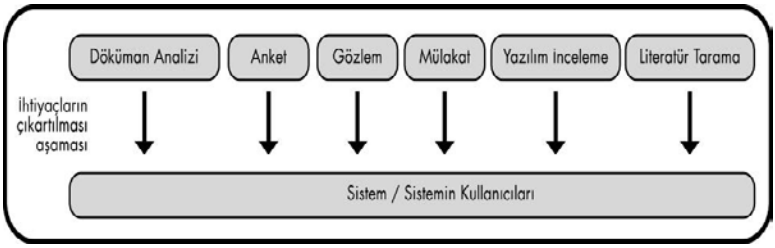
İhtiyaç belirleme süreci

İhtiyaçların toplanması ve sunulması amacıyla literatürde kullanılan oldukça fazla yöntem ve teknik bulunmaktadır. Modelin birinci aşaması olan ihtiyaç belirleme sürecinde kullanılan yöntem ve teknikler bu sürecin alt aşamaları kapsamında verilmiştir. İhtiyaçların

çıkartılmasında en yoğun olarak kullanılan teknik analistlerin amaçlanan sistem için sistemin kullanıcıları yani alanın uzmanlarıyla yaptığı görüşmeler/mülakatlardır. Bununla beraber yine aynı amaç için anket, gözlem, doküman taraması vb. yöntem/tekniklerde kullanılmaktadır. Ayrıca bu teknikler bilgi/ihtiyaç çıkartmada tek tek kullanılabildiği gibi bir arada da kullanılabilmektedir. Önerilen modelde birkaç teknik birlikte kullanılmıştır.

Bu bakış açısından hareketle model kapsamında; alan uzmanları/sistemin kullanıcılarıyla karşılıklı görüşmeler (mülakat), anket, gözlem, sisteme ait prosedürler/el kitapları/dokümanlar, piyasada alanla ilgili kullanılan yazılımlar ve literatüre ait bilgilerden faydalanma gibi birden çok tekniğe yer verilmiş ve ihtiyaç toplama/bilgi toplama süreci gerçekleştirilmiştir. Şekil 3. 2'de ihtiyaçların toplanması aşaması şematik olarak gösterilmiştir.

Fakat bu her zaman için bu tekniklerin tümünün bir arada kullanılması gerektiği anlamına gelmemelidir. Sistem analistleri geliştirilecek sistemin durumu ve tekniklerin özelliklerine göre tekniklerin tümünden veya bir kaçından faydalanabilirler.



Şekil 3.2. İhtiyaçların çıkartılması/toplanması aşaması

Geliştirilen modelin denenmesi sürecinde bir pilot uygulama gerçekleştirilmiş ve ihtiyaçların toplanması amacıyla bir anket geliştirilmiştir. Geliştirilen ankette hem içerikten/sistemden bağımsız sorular hem de içeriğe/sisteme bağımlı sorular kullanılmıştır. Sorular açık uçlu sorulardır. Ankette ki sorular Wetherbe'nin (1991) standartlaştırılmış sorularına dayanarak oluşturulmuştur. Oluşturulan sorularla çıkarılan bilgiler; organizasyonun amaçlarını, kullanıcıların amaçlarını, gerçekleştirilen işleri ve karar noktalarını tanımlar şekildedir. Günümüzde teknolojik alanda ki değişimler ve organizasyonların ihtiyaçlarının sürekli değişmesi, dinamik olması nedeniyle anketle bilgi toplama işinin belirli aralıklarla tekrarlı olarak gerçekleştirilmesi daha faydalı olacaktır. Ankette ayrıca bilgi toplanan kişilerin alanla ilgili yetkinliklerinin değerlendirilmesinde/anketin geçerliliğinin değerlendirilmesinde kullanılacak bazı sorularda bulunmaktadır.

Bilginin toplanması/ihtiyaçların çıkartılması aşamasında kullanılan, doğrudan yönlendirilmiş teknikler kapsamında ki içeriğe bağımlı ve bağımsız sorular, anket ve mülakatlarda ön yargılar, hatalı muhakeme ve hawthorne etkisinin üstesinden gelmek amacıyla kullanılmıştır [39].

Uygulamada vagon imalatı yapan bir işletmede yeniden yapılandırma amacıyla gerçekleştirilen bir proje çalışmasının verilerinden faydalanılmıştır. İşletmenin Personel Daire Başkanlığı kapsamında, eleman alımını sağlayan süreçlere ait veriler, çalışma konusu olarak belirlenen eleman temin sisteminin ihtiyaçlarının belirlenmesi amacıyla kullanılmıştır. Bu amaçla o dönemde Personel Daire Başkanı ve eleman

alım süreciyle ilgili çalışan kişilerle birebir olarak mülakatlar gerçekleştirilmiştir. Bir üst düzey, bir orta düzey ve dört işlevsel (işgören) seviyede ki çalışanla ortalama 60-90 dakika arasında ve ikişer defa olarak toplam 12 oturum gerçekleştirilmiştir. Oturumları gerçekleştiren deneyimli sistem analistleri mülakat sürecinde bazen ses kayıt cihazlarından faydalanmış bazen de not alarak çalışmışlardır. Gerçekleştirilen projede veri toplama yöntemlerinden biri olan anket yöntemi kullanılmamıştır.

Proje çalışması sırasında gözlem yöntemi aracılığı ile sistemdeki belge akışları, prosedürlerin işleyişi ve sistemde oluşan darboğazlar belirlenmiştir. Gözlem yönteminin uygulanması sürecinde gözlemci olarak sistem dışından, sistem analistleri gözlemci görevini üstlenmiştir. Gözlemciler Personel Daire Başkanlığı içerisinde çalışan kişilerin günlük bazda rutin olarak ve rutinin dışında gerçekleştirdikleri işlemleri gözlemlemişler, not almışlar ve değerlendirerek rapor hazırlamışlardır.

Belge taraması/doküman inceleme yönteminde, analistler Personel Daire Başkanlığı içinde mevcut olan; kullanım kitapları, prosedürler, iş akış şemaları, raporlar, formlar vb. belgelerden faydalanmışlardır. Bu sayede gerçekleşen işlemler ve sistemin fonksiyonları için hangi tür verilere gereksinim olduğu ve bunların ne şekilde hazırlanması gerektiği sağlıklı bir şekilde belirlenmiştir.

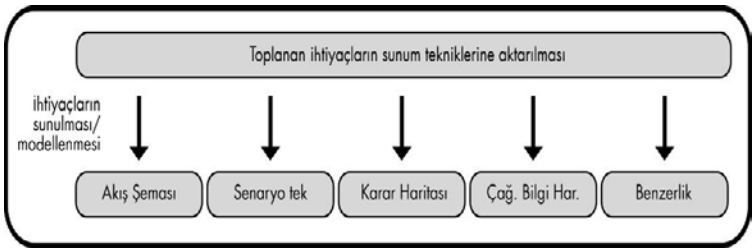
Tüm bu yöntemlere ilave olarak, bu alana uygun üretilmiş yazılımlar incelenmiş ve bunlardan elde edilen bilgilerde çalışmada kullanılmıştır.

Genel anlamda sistemin anlaşılması amacıyla ve analistlerin sisteme hakim olabilmelerini sağlayabilmek için sistemin fonksiyonlarına yönelik literatür taraması gerçekleştirilmiş alan ve fonksiyonları detaylı olarak incelenmiştir.

İhtiyaç belirleme sürecinin ikinci aşaması olan, ihtiyaçların sunulması aşamasında, tek bir teknikten ziyade birkaç teknikle ihtiyaç sunumunun daha doğru ve eksiksiz ihtiyaç belirlemede daha uygun olduğu için beş farklı teknikten faydalanılmıştır.

Bu teknikler [17, 29, 53];

- Akış şeması,
- Senaryo tekniği,
- Karar haritası,
- Çağrıştırıcı bilgi haritası ve
- Benzerlik diyagramı tekniğidir.



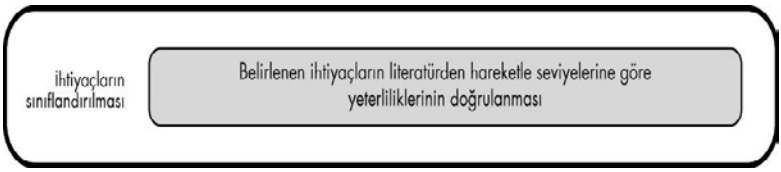
Şekil 3. 3 İhtiyaçların sunulması aşaması

Şekil 3.3'de ihtiyaçların sunulması aşaması şematik olarak gösterilmektedir. Geliştirilen modelde kullanılan teknikler, ihtiyaç

belirleme sürecinde ortaya çıkan problemlerin üstesinden gelmek amacıyla literatürden hareketle belirlenmiştir. Örneğin, Senaryo tekniği; hem bilgi çıkarımı hem de sunum aşamasında otomatikleşmiş davranışlardan kaynaklanan problemleri aşmak, Akış şeması; bilgi toplama, sunum ve doğrulama aşamalarında otomatikleşme, hatırlama, ihtiyaçların çeşitliliği/karmaşıklığı ve iletişim problemlerini aşmak, Çağrıştırıcı bilgi haritası; bilgi toplama, sunum ve doğrulama aşamalarında, ihtiyaçların çeşitliliği,/karmaşıklığı ve iletişim problemlerini aşmak, Karar haritası; bilgi toplama, sunum ve doğrulama aşamalarında, hatırlama ve iletişim problemlerini aşmak, Benzerlik diyagramı; bilgi toplama ve doğrulama aşamalarında, ihtiyaçların çeşitliliği/karmaşıklığı ve iletişim problemlerini aşmak amacıyla tercih edilmiştir [29].

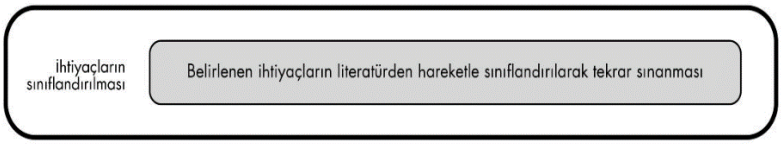
Bir sonra ki aşama olan ihtiyaçların doğrulanması/onaylanması aşamasında; alanın uzmanları olan kullanıcılar, teknikler aracılığı ile sunulan ihtiyaçların doğruluğu yani geçerliliği hakkında bir yargıya varacaklar veya sunum sırasında eksik kalan ihtiyaçlar varsa bunları tespit ederek ihtiyaçların yeniden düzenlenmesini isteyeceklerdir. Bu geri dönüşler oldukça faydalıdır. Bu sayede sistem geliştirmenin diğer aşamalarına geçmeden önce problem ve eksiklikler belirlenmiş olacaktır. Bu amaçla 6 sistem analisti ve 6 alan uzmanından faydalanılmıştır. Alan uzmanlarına sunum teknikleri hakkında bir bilgilendirme oturumu gerçekleştirilmiş ve daha sonra ortalama 60 dakika süren iki oturumla 6 alan uzmanı ve 6 deneyimli analistin

ihtiyaçları onaylaması aşaması gerçekleştirilmiştir. Aşağıda Şekil 3. 4'de ihtiyaçların doğrulanması aşaması şematik olarak görülmektedir.



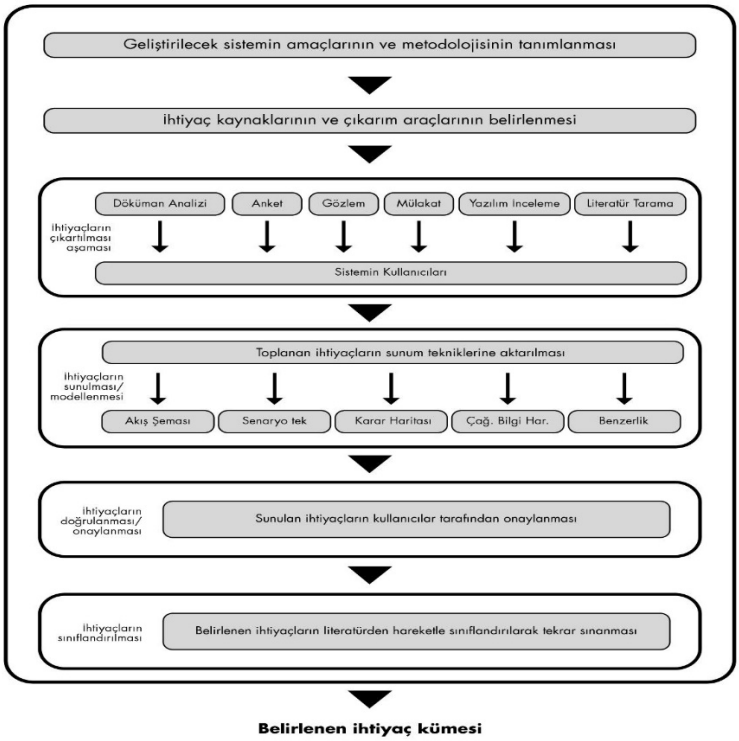
Şekil 3.4. İhtiyaçların doğrulanması aşaması

Belirlenen ihtiyaçlar, model gereğince amaç, süreç, görev ve bilişim seviyeli olarak sınıflanmış ve ihtiyaçlar kümesinin yeterliliği doğrulanmıştır. Bu sayede gözden kaçan ihtiyaçlar varsa yakalanarak eksiklikler giderilmiş ve ihtiyaçların doğruluk, eksiksizliği (tamlığı) kararı verilmiştir. Böylece "ihtiyaç belirleme süreci" tamamlanmış olmaktadır. Aşağıda Şekil 3. 5'de ihtiyaçların sınıflandırılması aşaması şematik olarak görülmektedir.



Şekil 3. 5 İhtiyaçların sınıflandırılması aşaması

Yukarıda aşama aşama olarak verilen modelin "İhtiyaç Belirleme Süreci" aşağıda Şekil 3. 6'da bütünleşik olarak görülmektedir.



Şekil 3.6. İhtiyaç belirleme süreci

ROC algoritmasının ihtiyaç belirlemede kullanılması

Hücresel imalat veya grup teknolojisi, yığın üretimde uzun imalat temin süreleri ve karmaşıklık problemlerini çözmek amacıyla bir üretim stratejisi olarak ortaya çıkmıştır. Hücresel imalatta ki asıl problem makine hücreleri ve ürün ailelerinin oluşturulmasıdır [90, 91, 92]. Grup Teknolojisi , parçaların üretim ve tasarımında ki benzerlik avantajlarından yararlanılarak parça ve tezgahların gruplandığı bir

üretim felsefesidir. Benzer parçalar parça aileleri biçiminde düzenlenmektedir. Bu sayede malzemelerin taşınması, maliyetlerin, hazırlık zamanlarının düşürülmesi sağlanarak imalat esnekliği, kalite ve iş tatmininin artması sağlanmaktadır. Buradaki anlayış, az sayıda oluşturulan aileyle çalışarak verimliliği artırmaktır. Bu amaçla üretim teçhizatı, makine grupları ya da hücrelere ayrılarak düzenlenmektedir.

Grup Teknolojisi uygulamasından elde edilecek başarı parça ailelerinin doğru olarak belirlenmesine bağlıdır [92, 93]. Bugüne kadar hücre oluşturma amaçlı bir çok araştırma yapılmış ve çok sayıda teknik geliştirilmiştir [90, 92, 94, 95]. Bunlardan bazıları;

- Ürün akış analizi,
- Sınıflandırma ve kodlama,
- Benzerlik katsayısı,
- Hücre içi akış analizi,
- Matematiksel araçları kullanan analitik metotlar; Çok değişkenli analiz, Grafik teorisi, Geometri, Matematik programlama ve Yapay zekadır [92].

Çalışmanın bu aşamasında belirlenmiş olan ihtiyaçların ard arda, kesintisiz olarak görülmesi ve benzer ihtiyaçları belirleyen tekniklerin tespiti amaçlanmıştır. Bu amaçla hücre oluşturma tekniklerinden ROC, ROC2, DCA ve benzerliklerden hareketle hesaplayarak gruplama sağlayan benzerlik katsayısı tekniği incelenmiştir.

Gerçekleştirilen literatür taraması kapsamında, bu teknikler içinde üretim kaynaklı olarak sınıflandırılan üç hücre oluşturma tekniği ve benzerlik katsayısı tekniğine değinilmiştir. Bunlar;

- Derece Sırasına Göre Kümeleme (ROC, Rank Order Clustering),
- Yeniden Düzenlenmiş Derece Sırasına Göre Kümeleme (ROC2, Rank Order Clustering2),
- Doğrudan Kümeleme Algoritması (DCA, Direct Clustering Algorithm) ve
- Benzerlik katsayısı tekniğidir.

Derece Sırasına Göre Kümeleme Algoritması (ROC)

Bu algoritma King tarafından parça ve tezgahların gruplandırılmasında kullanılmak üzere geliştirilmiştir. Bilgisayar uygulamalarında kolaylıkla kullanılabilecek basit, etkin ve analitik bir yöntemdir. Ayrıca hızlı yakınsama ve düşük hesaplama süresi özelliklerine sahiptir. Parça tezgah matrisindeki her satır ya da sütun iki tabanlı sayılar ile okunur. Prosedür her satır ya da sütun için bu iki tabanlı sayıları ondalık karşılıklarına çevirir. Daha sonra algoritma ardışık satır ya da sütunları alçak değerden yüksek değere tekrar düzenler. Bu işlem değerlerde hiçbir değişiklik olamayıncaya kadar devam eder [96].

Yeniden Düzenlenmiş Derece Sırasına Göre Kümeleme Algoritması2 (ROC2)

Bu algoritma King ve Nakornchai [97] tarafından ROC kümelendirmesindeki sınırlandırmaları ortadan kaldırmak için

geliştirilmiştir. Algoritma en sağ tarafı 1 olan sütunların satırlarını belirlemekle başlar. Bu satırlar birbirleri arasındaki satırlar korunarak sütunun en tepesine taşınır. İki tabanlı sayı kullanma zorunluluğu ortadan kaldırılmış, fakat derece sıralandırma aynen korunmuştur ve dolayısıyla sınırlandırmaları da aynen sürmektedir.

Direkt kümeleme algoritması (DCA)

Direk kümeleme algoritması, Chan ve Milner tarafından önerilen ve başlangıç parça/tezgah matrisinin pozitif değerleri solda olan satırları matrisin yukarısına ve pozitif değerleri üstte olan sütunları matrisin soluna çekerek düzenlemeye çalışan bir algoritmadır. Birkaç iterasyondan sonra, tüm pozitif değerler matrisin sol üst, sağ-alt köşegen ekseni üzerinde bloklar oluşturur. Fakat DCA' nın diyagonal eksende uygun bloklar oluşturmadığı gözlemlenmiştir [98].

Benzerlik katsayısı tekniği

Bu teknikte, başlangıç matrisindeki parça ve tezgahlar arasında benzerlik katsayısı hesaplanarak, parça aileleri ve makine hücreleri oluşturulur.İlk olarak, parça ya da makineler arasındaki benzerlikler hesaplanır. İkinci olarak ise, benzerlik seviyelerine bağlı olarak parçalar (tezgahlar) gruplandırılır. Bu amaçla litaratürde gerçekleştirilmiş bir çok çalışma ve algoritma mevcuttur [99, 100].

Gerçekleştirilen inceleme neticesinde, çalışmanın amacını görsel olarak da destekleyen, kullanımı oldukça basit olan ve birbirini diyagonal olarak takip etmesi istenen gruplama mantığının, ard arda kesintisiz

olarak ihtiyaçları görmeye olanak sağlaması nedeniyle ROC algoritması tercih edilmiştir. Çalışmada kullanılan ROC algoritması mantığı gereğince yazılmış olan program, visual basic 6.0'da kodlanmıştır.

ROC algoritması King tarafından parça ve tezgahların gruplandırılmasında kullanılmak üzere geliştirilmiş, bilgisayar uygulamalarında kolaylıkla kullanılabilecek basit, etkin ve analitik bir yöntemdir. Bu teknik; elemanları 0-1'den oluşan parça-makine matrisinin satır ve sütunlarının düzenlenmesiyle köşegen ekseni üzerinde kümeler oluşturmaya çalışır [97]. Matrisin satırlarında ki M1 ve M2 makineleri, P1 ve P2'de parçaları ifade etmektedir.

Matrisin hücre değeri $a_{ij} = 1$ ise; bu durumda a parçasının $i.$ makinesin de $j.$ işlemi gördüğü, $a_{ij} = 0$ ise a parçasının hiçbir işlem görmediği anlaşılmalıdır. Dizi esaslı tekniklerde parça aileleri ve makine hücreleri eş zamanlı olarak belirlenir. Bu sayede köşegen üzerinde oluşan bloklarda, hem makineler gruplanır hem de bu makine gruplarında işlenecek parça aileleri oluşturulur. Bu durum Şekil 3.7a ve Şekil 3.7b'de görülmektedir.

	P1	P2	P3	P4	P5
M1	1	1	1		
M2			1	1	1

Şekil 3.7a. Parça-makine matrisi

	P3	P2	P1	P5	P4
M1	1	1	1		
M2	1			1	1

Şekil 3.7b. Parça ailesi ve makine hücresi

Şekil 3.7a'da M1 ve M2 olmak üzere iki makine ve P1, P2, P3, P4 ve P5 olmak üzere beş farklı parça söz konusudur. Bu parça makine matrisinin düzenlenmesiyle Şekil 3.7b'de görülen iki grup bir diğer deyişle iki parça ailesi elde edilir. Burada ki gruplar ROC algoritması mantığı gereği birbirini diyagonal olarak izlemektedir. Bu nedenle makine sırasının değişim göstermemesine karşın, parçaların sırası değişerek bir gruplama gerçekleşmiş ve M2 makinesinde gerçekleşen P3 işlemi istisnai parça olarak belirlenmiştir.

Hücresel İmalat sistemlerinin tasarımında önemli olan bir diğer konu da oluşturulan hücrelerin çeşitli kriterlere göre değerlendirilmesidir [93, 94, 101]. GT literatüründe tezgah/makine hücresi ve parça ailesi oluşturma da başarı ölçütleri olarak temelde bazı esaslar göz önüne alınmaktadır. Bunlar; istisnai elaman sayısı, tezgah kullanım oranı, grup içerisindeki malzeme akışı gibidir. Gerçekleştirilen taramada hücre çözümlerinin değerlendirilmesinde üç veya beş ayrı başarı ölçütüne rastlanılmıştır. Fakat genel olarak 3 başarı ölçütü yardımıyla hücre oluşturma probleminin başarımı değerlendirilmektedir. Bu ölçütler;

- Grup verimliliği,

- Grup etkinliği ve

- Gruplama ölçüsüdür.

Yukarıda bahsedilen grup teknolojisi/hücresel imalat felsefesi ve ROC algoritması çalışma mantığının, sistem ihtiyaçlarının belirlemesinde uygulanabilirliği araştırılmıştır. Bu amaçla ROC algoritması uygulanırken, matrislerden faydalanıldığı ve matrisin satırlarına makineler, sütunlarına da parçalar yerleştirildiği gibi, bu modelde matrisin satırlarına ihtiyaç belirleme teknikleri, sütunlarına da belirlenmiş olan ihtiyaçlar yerleştirilerek, imalat dışında herhangi bir alanda uygulamasına rastlanılmamış olan ROC algoritmasıyla farklı bir amaç fakat benzer bir mantık yardımıyla bir analoji (benzetim) yapılmış, bu amaçla benzer parça/işlemleri ve tezgahları gruplama mantığı, benzer ihtiyaç ve tekniklerin gruplanmasında kullanılmıştır.

Aşağıda Şekil 3. 8a ve 3. 8b'de T1 ve T2'ler ihtiyaç belirlemede kullanılan teknikleri, İ1, İ2, İ3, İ4 ve İ5 ise ihtiyaçları göstermektedir. Modelde ROC algoritmasının uygulanması aşamasında matrisin sütunlarına ihtiyaç belirleme teknikleri tarafından belirlenen ihtiyaçlar, satırlarına da ihtiyaç belirleme teknikleri yerleştirilir. Teknikler; T1, T2, T3,….Tn, ihtiyaçlar ise; İ1, İ2, İ3,……İn şeklinde ifade edilir. Örneğin İ1 ihtiyacı T1 tekniği tarafından belirlenmişse "1" belirlenmemişse "0" yazılır. Bu şekilde tüm teknikler tarafından belirlenen ihtiyaçlar sırası ile girilir. ROC algoritması çalıştırılarak, algoritmanın çalışma mantığı gereği birbirine benzer ihtiyaçlar bir

araya gelerek birbirini diyagonal şekilde takip eden ihtiyaç aileleri ve teknik/teknik kombinasyonları oluşturulur.

	i1	i2	i3	i4	i5
T1	1	1	1		
T2			1	1	1

Şekil 3.8a. Teknik-ihtiyaç matrisi

	i3	i2	i1	i5	i4
T1	1	1	1		
T2	1			1	1

Şekil 3.8b. İhtiyaç ailesi ve Teknik/teknik kombinasyonu

Çalışmada kullanılan mantık, ROC algoritmasında parça ailelerinin birbirlerini diyagonal olarak takip ederek "işlemlerin ard arda kesintisiz olarak" birbirini izlemesidir. Modelde de parçalar yerine ihtiyaçlar ve tezgahlar yerine de ihtiyaç belirleme teknikleri kullanılmış ve ihtiyaçların ard arda birbirini izlemesi sağlanmıştır. Bu sayede "gerçek ihtiyaç kümesini" oluşturan tekniklerin bir arada görülmesi ve "asıl ihtiyaç kümesinin" hangi oranda sağlandığının anlaşılması mümkün olmuştur.

Grup teknolojisi mantığında bir parçanın birkaç tezgahta işlem görmesi gibi, bir ihtiyaçta birkaç teknik tarafından belirlenebilir veya sunulabilir. Çalışmada bu durum ihtiyaçların doğruluk ve geçerliliğinin oranı olarak değerlendirilmektedir.

Bu mantık sayesinde geliştirilen modelde ROC algoritması kullanılarak; hangi tür ihtiyaçları hangi tekniklerin belirlediği, hangi teknik/teknik kombinasyonlarının tüm ihtiyaçları belirlediği ve ihtiyaç

tekrarının ihtiyaçların doğrulanması olarak kabul edilmesi durumunda ihtiyaç geçerliliğinin kolayca görülmesi sağlanmıştır. Bu amaca ne kadar hizmet edildiğini değerlendirebilmek için başarı ölçütleri olan grup verimliliği, etkinliği ve gruplama ölçülerinden faydalanılmış fakat modelde ki amaca uygun bir sonuç elde edilemediği için belirlenen ihtiyaçların ve teknik/teknik kombinasyonlarının değerlendirilmesinde yeni geliştirilen başarı ölçütlerinden faydalanılmıştır. Yeni geliştirilen başarı ölçütlerine ait açıklamalar aşağıda verilmiştir.

Sistem ihtiyaçlarını belirleme için başarı ölçütleri

Model geliştirme çalışması kapsamında, teknikler ve ihtiyaçlara ROC algoritması uygulanmış ve elde edilen ihtiyaç ailelerinin değerlendirilmesi amacıyla "sistem ihtiyaçları belirleme için başarı ölçütleri" geliştirilmiştir. Bu amaçla geliştirilen üç başarı ölçütü; "Tekniklerin ihtiyaç belirleme etkinlik ölçüsü, Tekniklerin ihtiyaç doğrulama verimlilik ölçüsü ve Ortalama teknik verimliliği ölçüsü" ne ait açıklama ve notasyonlar aşağıda verilmiştir.

-Tekniklerin İhtiyaç Belirleme Etkinlik Ölçüsü

t_e = Tekniklerin İhtiyaç Belirleme Etkinlik Ölçüsü

$\dot{I}_t$ = İhtiyaçların toplamı (matris içindeki 1'lerin sayısı)

$\dot{I}_{kk}$ = Köşegen bloklarda kesişen ihtiyaçların (1'lerin) sayısı

$\dot{I}_{ok}$ = Köşegen olmayan bloklarda ki toplam ihtiyaçların (1'lerin) sayısı

i = İhtiyaç sayısı (teknik/teknik kombinasyonları tarafından sistem için belirlenmiş tüm ihtiyaçların sayısı)

t_{ei} = İhtiyaç sayısı olarak etkinlik

$$t_e = \frac{\dot{I}_t - (\dot{I}_{kk} + \dot{I}_{ok})}{i} \qquad 3.1$$

$$t_{ei} = t_e.i \qquad 3.2$$

-Tekniklerin İhtiyaç Doğrulama Verimlilik Ölçüsü (Tekrarlı İhtiyaçların Miktarı)

t_v = Tekniklerin İhtiyaç Doğrulama Verimlilik Ölçüsü

$\dot{I}_t$ = İhtiyaçların toplamı (matris içindeki 1'lerin sayısı)

t_{ei} = İhtiyaç sayısı olarak etkinlik

i = İhtiyaç sayısı (teknik/teknik kombinasyonları tarafından sistem için belirlenmiş tüm ihtiyaçların sayısı)

t_{vi} = İhtiyaç sayısı olarak verimlilik

$$t_v = \frac{\dot{I}_t - t_{ei}}{i} \qquad 3.3$$

$$t_{vi} = t_v.i \qquad 3.4$$

-Ortalama Teknik Verimliliği Ölçüsü

t_{ov} = Ortalama Teknik Verimliliği Ölçüsü

t_e = Tekniklerin İhtiyaç Belirleme Etkinlik Ölçüsü

t_v = Tekniklerin İhtiyaç Doğrulama Verimlilik Ölçüsü

i = İhtiyaç sayısı (teknik/teknik kombinasyonları tarafından sistem için belirlenmiş tüm ihtiyaçların sayısı)

t = Toplam teknik sayısı

$$t_{ov} = \frac{t_{ei}}{t.i} \qquad 3.5$$

Şekil 3.8b'ye ait "tekniklerin ihtiyaç belirleme etkinlik ölçüsü, ihtiyaç doğrulama verimlilik ölçüsü ve Ortalama Teknik Verimliliği Ölçüsü değerleri formüller yardımıyla aşağıda hesaplanmıştır.

t_e = 6 – (0 + 1) / 5 = 1.00 $\qquad$ t_{ei} = 5 * 1 = 5

Yukarıda hesaplanan t_e değerinin açıklaması yapılması gerekirse; Toplam ihtiyaç kümesinin elemanının 5 olduğu kabul edilmesi durumunda; Şekil 3.8b'de görülen T1 ve T2 tekniklerinin bu 5 ihtiyacı belirlediği görülmektedir. Bu da bizim tüm ihtiyaç kümemizi kapsamaktadır. Elde edilen t_e değeri de 1.00 ve t_{ei} değeri 5'dir. Kısacası bu iki teknik ihtiyaçlar kümesini oluşturan 5 ihtiyacın tümünü ifade etmekte ve t_{ei} değeri 5'e karşılık gelmektedir.

T1 ve T2 teknikleri toplam 5 ihtiyacın 3'ünü belirlemiştir. T1 tekniği, İ1, İ2 ve İ3 ihtiyaçlarını belirlerken, T2 tekniği de İ3, İ4 ve İ5 ihtiyaçlarını belirlemiştir. Her iki teknik birbirinden farklı 2'şer ihtiyaç belirlerken İ3 ihtiyacı her iki teknik aracılığı ile de belirlenmiştir. Bu durum $\mathbf{t_v}$ değeri ile ifade edilmektedir.

t_v = 6 – 5 / 5 = 0.20 $\qquad$ t_{vi} = 0.20 * 5 = 1

Eğer $\mathbf{t_v}$ değerini açıklamamız gerekirse; Bu bize teknikler aracılığı ile tekrarlı olarak sunulan/ifade edilen ihtiyaçları vermektedir.

(Tekniklerin belirlediği ve çakışan ihtiyaçlar.) Bu ihtiyaçların tekrarlı olarak sunulması aynı zamanda bunların önem seviyeleri hakkında bize fikir vermektedir. (İhtiyaç seviyelerinin ağırlıklandırılmasında kullanılabilir.)

t_{ov} = 5 / 2 * 5 = 0.50 %50

t_{ov} değeri ile ilgili bir açıklama yapmamız gerekirse; Her bir tekniğin ortalama olarak % de kaç oranında ihtiyaç belirlediğini açıklamaktadır.

Modelde kullanılan teknik/teknik kombinasyonları tarafından belirlenen ihtiyaçların değerlendirilmesi amacıyla, genellikle imalatta kullanılan ROC algoritması kullanılarak bir analoji gerçekleştirilmiştir.

Algoritmanın çalışma mantığı gereği birbirine benzer ihtiyaçlar ve teknik/teknik kombinasyonları gruplanarak ihtiyaç aileleri oluşturulmuştur. Şekil 3.8b'de görüldüğü gibi I. grupta ki ihtiyaç ailesi i3, i2 ve i1'den II. grupta ki ihtiyaç ailesi de i5 ve i4'dü içermektedir. Burada benzer ihtiyaçların oluşturduğu aileler farklı grupları meydana getirmekte ve ihtiyaçların karşılanma miktarlarını veya bir diğer adıyla etkinliklerinin değerlendirilmesinde yardımcı olmaktadır. Bu amaçla sistem ihtiyaçlarını belirlemeye yönelik geliştirilen başarı ölçütlerinden faydalanılmıştır.

Örnekleme sırasında iki teknik ve 5 ihtiyaçtan faydalanılmıştır. Bu durumun gözle görülmesi ve elle çalıştırılması oldukça kolaydır. Fakat çok sayıda teknik/teknik kombinasyonu ve ihtiyacın kullanılması durumunda bu algoritma oldukça faydalı olacaktır.

Önerilen modelde endüktif öğrenme yaklaşımı

Bu bölümde model kapsamında kullanılan eğitim setinin oluşturulması ve endüktif öğrenme tekniklerinden RULES-3 algoritması ile oluşturulan eğitim setinden kuralların çıkarılması açıklanmıştır.

Endüktif öğrenme algoritmasının çıktısı ya bir karar ağacı veya kurallar setidir. Karar ağacından kurallar kolaylıkla elde edilebilir. Aslında karar ağacının herbir dalı bir kural olarak değerlendirilir. Bir endüktif öğrenme algoritmasından istenen, mümkün olabilecek en genel kuralları elde etmesidir. RULES-3 algoritması elde edilen kuralları tek tek kontrol ederek en fazla örneği sınıflandıran kuralı, kısacası en genel kural seçer [102]. Bu çalışmada yukarıda bahsedilen avantaj ve uygulama kolaylığı nedeniyle RULES-3 algoritması tercih edilmiştir.

RULES-3 algoritması

Endüktif öğrenme teknikleri genel olarak iki bölümde incelenebilir.

- Karar ağacı üreten teknikler ve
- Doğrudan kural üreten teknikler .

Karar ağacı tabanlı teknikler alt sistemlere ayırma prensibiyle çalışırlar. Yani verilmiş olan örnekler setini alt setlere ayırır ve daha sonra bu alt setlerden hareketle karar ağacı oluştururlar. Bu tekniklerin çıktısı bir karar ağacıdır. CLS ve ID3 bu sınıf için önemli örnek algoritmalardır [102, 103, 104].

Doğrudan kural çıkaran algoritmalar ise adından da anlaşıldığı gibi karar ağacı oluşturmadan doğrudan kural oluşturan algoritmalardır. Bunlara örnek olarak da AQ ve RULES verilebilir. Her iki sınıfta ki algoritmalar uygulamalı yapay zekada önemli yer tutmaktadırlar. RULES algoritmasının 3 farklı versiyonu bulunmaktadır; RULES-1, RULES-2, ve RULES-3. RULES'un açılımı ise **RUL**e **E**xtraction **S**ystem şeklindedir. Kural Çıkarma Sistemi (Rule Extraction System).

RULES-1 de çıkarılacak kurallar üzerinde bir kontrol söz konusu değildir. Örneğin bir örnek için çıkarılabilecek kural sayısı belirlenemez. Algoritma daima çıkarılabilecek maksimum sayıdaki kuralı çıkarır. Bu bazen gerekli olmayabilir. RULES-3 ise, kullanıcının tanımlayacağı sayıda kural çıkarabilir. Bu, RULES-3'te örneklerin tek tek ele alınmasından kaynaklanmaktadır. Aşağıda RULES-3'ün temel adımlarında da görüldüğü gibi; Adım 2'de kullanıcı her örnek için çıkarılabilecek kural sayısını belirlemektedir. Algoritma, eğer bu sayıya ulaştı ise daha fazla kural çıkarmaz. Eğer bu sayıya ulaşılamıyorsa çıkarabildiği kadar kural çıkarır. Yani her örnek için en az bir, en çok belirlenen sayı kadar kural çıkarılacaktır. Bu işlem hem gereğinden fazla kural çıkarılmasını önleyecek hem de işlem zamanını kısaltacaktır. Özellikle karakteristik sayısının ve bunların alabileceği değerlerin çok olması halinde bu işlem çok yararlı olmaktadır.

RULES-3 algoritması elde edilebilen kuralları tek tek kontrol ederek en genel olanını seçer. Kısacası, eldeki örneği sınıflandırabilen kurallar çıkarıldıktan sonra bunların her birinin, verilen örnekler setinde, aynı

sınıfta geçen kaç örneği sınıflandırdığına bakılır. En fazla örneği sınıflandırabilen kural en genel kural kabul edilir ve seçilir.

RULES-3'ün temel adımları aşağıda görüldüğü gibidir [102];

Adım 1. Sayısal değer içeren karakteristikler için aralıkları tanımla

Adım 2. Çıkarılacak kurallar için minimum şart sayısını (N_{cmin}) belirle

Adım 3. Her örnek için çıkarılabilecek kural sayısını belirle

Adım 4. Sınıflandırılmamış bir örnek seç

Adım 5. $N_c = N_{cmin} - 1$

Adım 6. Eğer $N_c < N_a$ ise $N_c = N_c+1$

Adım 7. Eldeki örnekte kullanılan karakteristik değerleriyle bir *değerler dizisi* oluştur.

Adım 8. Değerler dizisinde yer alan elemanların N_c 'li kombinasyonlarını bul.

Adım 9. Eğer bu kombinasyonlardan en az bir tanesi kontrol setindeki tüm örneklerde sadece bir sınıfta geçiyorsa bunu kural haline getir, aksi halde Adım 6'ya git.

Adım 10. Çıkarılmış kurallar arasından en genel olanını seç.

Adım 11. Seçilmiş kural ile sınıflandırılabilen örnekleri ayıkla.

Adım 12. Şayet tüm örnekler sınıflandırılmışsa bitir, aksi halde Adım 4 'e git.

Burada N_c = ***şart sayısı*** ve N_a = ***karakteristik sayısı***'dır [102, 103]. Oluşturulan örnekler setinde sayısal değerler içeren karakteristikler varsa Adım1'de bunlar için aralıklar tanımlanır. Ancak bu şekilde sayısal değerli karakteristikler içeren örnekler seti içinde kurallar üretilebilir. Bu sürecin adımları aşağıda ki gibi gerçekleştirilir: İlk olarak, sayısal değerli karakteristikler için verilen örneklerde bulunan en küçük ve en büyük değerler tespit edilir. Daha sonra uzman

görüşünden veya literatürden hareketle bu değerler arasında aralıklar belirlenir ve her aralığa bir sözel etiket verilerek örnekler seti bu etiketlere göre oluşturulur. Bir kural en az bir, en fazla N_a yani toplam karakteristik sayısı kadar şart içerebilir. Adım2'de bir kuralın bulundurması istenen en küçük şart sayısı girilir. Adım3'te her örnek için çıkarılması istenen kural sayısı belirlenir. Adım4'te örnek setinden daha önce hiç kullanılmamış bir örnek rastgele seçilir. Adım5 ve Adım6'da seçilen örnek için şart sayısı düzenlenir. Adım7'de örnekte kullanılan N_a tane değer, bir diziye atanır. Adım8'de bu dizinin N_c'li kombinasyonları bulunur. Adım9'da bu kombinasyonların her birinin kural olup olmayacağı test edilir. Eğer bir kombinasyon örnekler setinde yalnızca bir sınıfta geçiyorsa bu örnek kural olarak belirlenir. Eğer hiçbir kural bulunamamışsa şart sayısı bir arttırılır. Bu işleme en az bir kural bulununcaya kadar tekrarlı olarak devam edilir. Adım10'da, eğer bir önceki adımda birden fazla kural bulunmuşsa bunların içerisinden oluşturulan örnekler setinden en fazla örneği sınıflandıran bir tanesi seçilir. Adım11'de seçilmiş kurallarla sınıflandırılabilen örnekler ayıklanır. Adım12'de bu kural kullanılarak kendisi için daha önce kural çıkarılmamış örnek kalmadıysa algoritma durur, eğer kaldıysa Adım4'e gidilerek işlemler tekrar edilir. En sonunda elde edilen tüm kurallar bir araya getirilerek bilgi tabanı oluşturulmuş olur. Bu bilgi tabanı eldeki örnekler setinin genelleştirilmiş halidir.

Algoritmanın adımlarından da görüldüğü gibi örnekler setinden kurallar çıkarmak tekrarlı bir işlemdir. Bu algoritmaya göre kodlanmış bir bilgisayar programı uzman sistemde çıkarım aracı olarak

kullanılabilir. Buradan çıkan kurallara göre öğrenmiş olan bilgisayar programı, kendini bir uzman gibi görerek benzer örnekler verildiğinde yeni örneklerin hangi kurala uyduğuna karar verebilir.

Eğitim setinin hazırlanması

Eğitim setinde kullanılması gereken veriler ve bu verilerin karekteristik bilgilerinden kural üretmenin en kolay yolu tablolardan faydalanmaktır. Tablodaki her bir satır bir nesneyi ifade eder ve tüm nesneler $U = \{x_1, x_2, x_3, ...x_n\}$ sınırlı uzay kümesi ile temsil edilir. Her bir örnek, şart ve karar olmak üzere eğer-ise (if-then) cümlelerinden oluşmaktadır. Şart (condition) cümlesinin karakteristikleri $k = \{k_1, k_2, k_3 .....k_m\}$ ile karar (decision) cümlesinin karakteristikleri, kısaca sınıfları $S = \{s_1, s_2, s_3 .....s_p\}$, eğitim setinin sınırlı karakteristik uzay kümesini oluşturmaktadır. Tablodaki d_{nm} notasyonu $n.$ örneğin m şart karakteristiği için sahip olduğu hücre değerini ve s_p notasyonu ise örneğin sınıf değerini ifade eder. Bu ifadelerden hareketle Tablo 3.1'de bir eğitim setinin genel yapısı görülmektedir.

Tablo 3.1. Eğitim setinin genel yapısı

	k_1	k_2	k_3	.....	k_m	S
x_1	d_{11}	d_{12}	d_{13}	.....	d_{1m}	s_1
x_2	d_{21}	d_{22}	d_{23}	.....	d_{2m}	s_2
.	.	.	.	.....	.	.
x_n	d_{n1}	d_{n2}	d_{n3}	.....	d_{nm}	s_p

Eğitim setinin hazırlanmasında, uygulama kapsamında Tablo 3.13'de eleman temin sistemi için geliştirilen eğitim setinin 5 sınıf aralığını da

ifade edebilen 5., 6., 9.,18. ve 23.örnekler kullanılarak Tablo 3.2'de görülen örnek bir eğitim seti oluşturulmuştur. Tablo 3.2'de ki 1. örnek, pilot uygulamada Tablo 3.13'de ki 5. örneğe, 2. örnek, 23. örneğe, 3. örnek, 6. örneğe, 4. örnek, 9. örneğe ve 5. örnek de 18. örneğe karşılık gelmektedir. Oluşturulan eğitim seti beş örnekli bir tablodur. Tablonun satırları örnekleri, sütunları da örneklerde göz önünde bulundurulan karakteristikleri ve hücrelerde karakteristiklerin almış olduğu değerleri ifade etmektedir. Tablo 3.2'de ki k_1 notasyonu; çalışmada ki "amaç seviyeli" ihtiyaçları, k_2 notasyonu "süreç seviyeli", k_3 "görev seviyeli" ve k_4 "bilişim seviyeli" ihtiyaçları ifade etmektedir.

Tablo 3.2. Eğitim seti örneği

Örnek	k_1	k_2	k_3	k_4	**Sınıf değeri**
1	0.07	0	0	1.00	**-**
2	0.93	1.00	0.86	1.00	**-**
3	0.86	0.40	0.57	0.0	**-**
4	0.57	0.40	0.43	1.00	**-**
5	0.93	0.40	0.57	1.00	**-**

Tablodaki sayısal değerler bir numaralı örnek için: "birinci teknik/teknik kombinasyonu ile (i, j) belirlenmiş ihtiyaçların birinci karakteristiği sağlama oranı 0.07; ikinci karakteristiğe göre 0.0, üçüncü karakteristiğe göre 0.0 ve dördüncü karekteristiğe göre 1.00" olduğunu ifade etmektedir. Eğitim setinde ki diğer örnekler de benzer anlamları ifade etmektedir.

Oluşturulan tabloda görüldüğü gibi eğitim setindeki sınıf değeri hücreleri boş bırakılmıştır. Sınıf değeri, (k_1, k_2, k_3, k_4) karakteristiklerinden hareketle hesaplanacaktır.

Bu karakteristiklere ait önem değerleri (ağırlıklar) bazen uzmanlar tarafından belirlenir bazen de geçmiş örnek çalışmalardan elde edilir. Bu çalışmada ki karakteristikler ve önem değerleri literatürde ki örnek bir çalışmadan faydalanılarak belirlenmiştir [15].

Çalışmanın ağırlıklandırılmasında kullanılan önem değerleri birinci karakteristik için (Amaç seviyeli ihtiyaçlar) $a_1 = 0{,}36$, ikinci karakteristik için (Süreç seviyeli ihtiyaçlar) $a_2 = 0{,}14$, üçüncü karakteristik için (Görev seviyeli ihtiyaçlar) $a_3 = 0{,}18$ ve dördüncü karakteristik ise (Bilişim seviyeli ihtiyaçlar) $a_4 = 0{,}32$ olarak alınmıştır. Tablo 3.2'nin son sütunu olan sınıf değeri, her karakteristik sayısal değerinin kendi önem değeri ile çarpılıp kümülatif olarak toplanmasından oluşmaktadır. Bu durumu ifade eden S_d değeri aşağıda görüldüğü şekilde formülüze edilmiştir.

$$S_k = \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} a_j \qquad 3.6$$

Bu formüle göre, birinci örneğin sınıf değeri aşağıdaki şekilde ifade edilerek hesaplanabilir.

Sınıf değeri= $(0.07 \ x \ a_1) + (0.0 \ x \ a_2) + (0.0 \ x \ a_3) + (1.0 \ x \ a_4)$

Sınıf değeri=

$$(0.07 \ x \ 0.36) + (0.0 \ x \ 0.14) + (0.0 \ x \ 0.18) + (1.0 \ x \ 0.32) = 0.35$$

Diğer örneklerin sınıf değerleri de aynı formülle hesaplanarak Tablo 3.3'deki eğitim seti elde edilmiş olur.

Tablo 3.3. Sınıf değerleri hesaplanmış sayısal değerli eğitim seti

Örnek	k_1	k_2	k_3	k_4	**Sınıf değeri**
1	0.07	0	0	1.00	0.35
2	0.93	1.00	0.86	1.00	0.95
3	0.86	0.40	0.57	0	0.47
4	0.57	0.40	0.43	1.00	0.66
5	0.93	0.40	0.57	1.00	0.81

RULES3 algoritmasının birinci adımını gerçekleştirebilmek amacıyla, sayısal değerli örnekler setinin dilsel tanım aralıklarına sahip olarak ifade edilmesi gerekmektedir. Aralıklar, karakteristik ve sınıf değerlerinin en büyük ve en küçük değerlerini kapsayacak biçimde tanımlanır. Aralık tanımlamalarına yönelik olarak endüktif öğrenme yaklaşımında genel olarak kabul görmüş bir yöntem söz konusu değildir. Bu nedenle en uygun yaklaşım olarak uzman görüşü kullanılabilir. Bu şekilde her aralık dilsel bir terim ile ifade edilerek örnekler setinin dilsel biçimi elde edilmiş olur. Elde edilmiş olan sayısal değerler, dilsel ifadelere dönüştürülerek problemlerin insan düşüncesine dayalı modellenmesinde kavramsal bir çerçeve elde edilmiş olur [105]. Modelde, şart karakteristik değerleri için "çok düşük", "düşük", "normal", "yüksek", "çok yüksek" ve sınıf değerleri için ise "hiç karşılanmamış", "az karşılanmış", "kısmi karşılanmış",

“çoğu karşılanmış”, “tümü karşılanmış” şeklinde dilsel ifadeler kullanılmıştır. Sayısal örnekler tablosunu dilsel olarak ifade edilmesini bir örnekle açıklamamız gerekirse: Tablo 3.3’deki örnekler setinden faydalanabiliriz Her karakteristik için yapılmış olan aralık tanımlamaları da Tablo 3.4’de verilmiştir.Tablo 3.4’de ki bu değerler uzman görüşünden hareketle oluşturulmuştur.

Tablo 3.4. Karakteristik değerler için tanım aralığı tablosu

k_1	k_2	k_3	k_4	**Dilsel İfadesi**
$0 \leq D_k \leq 0.20$	$0 \leq D_k \leq 0.30$	$0 \leq D_k \leq 0.20$	$0 \leq D_k \leq 0.30$	**ÇD:Çok düşük**
$0.20 < D_k \leq 0.40$	$0.30 < D_k \leq 0.55$	$0.20 < D_k \leq 0.30$	$0.30 < D_k \leq 0.50$	**D: Düşük**
$0.40 < D_k \leq 0.60$	$0.55 < D_k \leq 0.70$	$0.30 < D_k \leq 0.50$	$0.50 < D_k \leq 0.65$	**N: Normal**
$0.60 < D_k \leq 0.75$	$0.70 < D_k \leq 0.85$	$0.50 < D_k \leq 0.70$	$0.65 < D_k \leq 0.85$	**Y: Yüksek**
$0.75 < D_k \leq 1$	$0.85 < D_k \leq 1$	$0.70 < D_k \leq 1$	$0.85 < D_k \leq 1$	**ÇY:Çok yüksek**

Aynı yaklaşımla sınıf değerleri için de ayrı bir aralık tanımlaması yapılmış ve bu tanımlamalar da Tablo 3.5’de verilmiştir.

Tablo 3.5. Sınıf değeri için tanım aralığı tablosu

S_d	**Dilsel İfadesi**
$0 \leq D_k \leq 0.35$	**HK: Hiç karşılanmamış**
$0.35 < D_k \leq 0.50$	**AK: Az karşılanmış**
$0.50 < D_k \leq 0.70$	**KK: Kısmi karşılanmış**
$0.70 < D_k \leq 0.90$	**ÇK: Çoğu karşılanmış**
$0.90 < D_k \leq 1.0$	**TK: Tümü karşılanmış**

Bu tanım aralıkları baz alınarak sayısal örnek seti dilsel örnek setine dönüştürülür. Tablo 3.3'de ki sayısal değerli örnek setinin birinci örneğini ele alacak olursak; birinci örneğin, birinci karakteristiğinin (k_1) sayısal değeri *0.07* dir. Bu değer Tablo 3.4'deki karakteristik aralığı tablosunda $0 \leq D_k \leq 0.20$ aralığında bulunmaktadır ve dilsel ifade değeri de *çok düşük'tü*r. İkinci karakteristiğin (k_2) sayısal değeri 0.0; $0 \leq D_k \leq 0.30$ tanımlama aralığında bulunup dilsel ifade değeri *çok düşük*'tür. Üçüncü karakteristiğin (k_3) 0.0 sayısal değeri $0 \leq D_k \leq 0.20$ tanım aralığındaki dilsel ifade karşılığı *çok düşük* olmaktadır. Dördüncü karakteristiğin (k_4) sayısal değeri 1.0 ve $0.85 < D_k \leq 1$ tanım aralığında bulunup dilsel ifade değeri *çok yüksek*'tir.

Yukarıda ki yaklaşımdan hareketle tüm örnekler için sınıf değerlerinin dilsel ifadelerini bulmamız gerekirse; örneğin birinci örneğin Tablo 3.3'de görüldüğü gibi sınıf değeri *0.35*'dir. Bu değer Tablo 3.5'de ki sınıf tanım aralığı tablosunda $0 < D_k \leq 0.35$ sınıf aralığına karşılık gelmekte ve dilsel ifadesi *hiç karşılanmamış* olmaktadır. İkinci örneğin sınıf değeri *0.95*'tir. Bu değerde $0.90 < D_k \leq 1.00$ sınıf aralığına karşılık gelmekte ve dilsel ifadesi yine *tümü karşılanmış* olmaktadır. Üçüncü örneğin sınıf değeri *0.47*'dir. Bu değerde $0.35 < D_k \leq 0.50$ sınıf aralığına karşılık gelmekte ve dilsel ifadesi *az karşılamış* olmaktadır. Dördüncü örneğin sınıf değeri *0.66*'dır. Bu değerde $0.50 < D_k \leq 0.70$ sınıf aralığına karşılık gelmekte ve dilsel ifadesi *kısmi karşılanmış* olmaktadır. Beşinci örneğin sınıf değeri *0.81*'dir ve bu değerde $0.70 < D_k \leq 0.90$ sınıf

aralığına karşılık gelmekte ve dilsel ifadesi *çoğu karşılanmış* olarak bulunmaktadır. Tüm örneklerden hareketle Tablo 3.3'deki örnekler setinin dilsel şekli Tablo 3.6'da görüldüğü gibi elde edilmektedir.

Tablo 3.6. Eğitim setinin dilsel ifadesi

Örnek	k_1	k_2	k_3	k_4	**Sınıf Aralığı**
1	Çok Düşük	Çok Düşük	Çok Düşük	Çok Yüksek	Hiç Karşılanmamış
2	Çok Yüksek	Çok Yüksek	Çok Yüksek	Çok Yüksek	Tümü Karşılanmış
3	Çok Yüksek	Düşük	Yüksek	Çok Düşük	Az Karşılanmış
4	Normal	Düşük	Normal	Çok Yüksek	Kısmi Karşılanmış
5	Çok Yüksek	Düşük	Yüksek	Çok Yüksek	Çoğu Karşılanmış

Kural çıkarma

Bu bölümde hazırlanan eğitim seti ve RULES-3 endüktif öğrenme algoritması yardımı ile kurallar üretilmiştir. Bu kurallar seti aracılığı ile ihtiyaçların karşılanma sınıf değerlerini belirlemek için kullanılacak bilgi tabanı oluşturulmuştur.

Tablo 3.6'daki beş örnekten oluşan eğitim seti ele alınmıştır. Her bir örnek dört farklı karakteristiğin, çok düşük, düşük, normal, yüksek ve çok yüksek dilsel ifadelerine sahiptir. Sınıf değerleri ise hiç

karşılanmamış, az karşılanmış, kısmi karşılamış, çoğu karşılanmış ve tümü karşılanmış şeklindedir. RULES3 algoritması ile bu eğitim setinden kurallar çıkarma işlemi aşağıda adım adım açıklanmıştır.

Adım 1: Tablo 3.6'daki eğitim setinin oluşturulması sırasında aralık tanımları yapılmış ve sayısal değerli eğitim seti dilsel biçimde ifade edilmiştir. Bu nedenle artık burada tekrar aralık tanımı yapılmayacaktır. Aralık tanımı için Bölüm 3.2.3.2 Eğitim setinin hazırlanması'na bakınız.

Adım 2: Her kural için minimum şart sayısı, N_{cmin}=1 alınmıştır. Amaç aynı anda dört karakteristiği de göz önüne alan kurallar çıkarmaktır.

Adım 3: Her örnek için çıkarılacak kural sayısı minimum yani bir olarak belirlenmiştir.

Birinci iterasyon

Adım 4: Yukarıda Tablo 3. 6'da verilmiş olan ilk örneği seçelim. Bu örnek aşağıdaki gibidir:

k_1=ÇD; k_2=ÇD; k_3=ÇD; k_4=ÇY şartlarında, ihtiyaçların karşılanması=Hiç Karşılanmamış

Adım 5: N_c= N_{cmin}-1

N_c=1-1=0

Adım 6: Eğer Nc< N_a ise N_c=N_c+1 (Eğitim setinde N_a=4; k_1, k_2, k_3, k_4)

N_c=N_c+1; N_c=0+1=1

Adım 7: Eldeki örnekte kullanılan karakteristiklerin değerleriyle {ÇD, ÇD, ÇD, ÇY} değerler dizisi elde edilir.

Adım 8: N_c=1 olduğundan bu değerler tek tek ele alınacaktır.

Adım 9: Bu değerler tek tek ele alındığında k_1=ÇD değeri sadece bir sınıfta geçmektedir. Buradan elde edilen sonuç aşağıdaki gibidir.

Kural 1: *Eğer* k_1*=ÇD ise İhtiyaç Karşılama=Hiç Karşılanmamış*

Adım 10: Çıkarılan kuralı genel kural olarak seçeriz.

Adım 11: Bu kural ile sadece birinci örnek sınıflandırılabilir. Bu örnek bundan sonra ki kural çıkarma işlemlerinde göz önüne alınmayacaktır.

Adım 12: Henüz sınıflandırılmayan dört örnek kaldığından ikinci iterasyona geçilir.

İkinci iterasyon

Adım 4: İkinci örneği seçelim. Bu örnek aşağıdaki şekildedir:

k_1=ÇY; k_2=ÇY; k_3=ÇY; k_4=ÇY İhtiyaç Karşılama =Tümü Karşılanmış

Adım 5: N_c= N_{cmin}-1

N_c=1-1=0

Adım 6: Eğer Nc< N_a ise N_c=N_c+1 (Eğitim setinde N_a=4; k_1, k_2, k_3, k_4)

N_c=N_c+1; N_c=0+1=1

Adım 7: Eldeki örnekte kullanılan karakteristiklerin değerleriyle {ÇY, ÇY, ÇY, ÇY} değerler dizisi elde edilir.

Adım 8: N_c=1 olduğundan bu değerler tek tek ele alınacaktır.

Adım 9: Bu değerler tek tek ele alındığında k_3=ÇY değeri sadece bir sınıfta geçmektedir. Bunu kural haline getirdiğimizde aşağıdaki sonuç elde edilmiştir.

Kural 2: *Eğer* k_2 *=ÇY ise İhtiyaç Karşılama=Tümü Karşılanmış*

Adım 10: Çıkarılan kuralı genel kural olarak seçeriz.

Adım 11: Bu kural ile sadece ikinci örnek sınıflandırılabilir. Bu örnek bundan sonra

ki kural çıkarma işlemlerinde göz önüne alınmayacaktır.

Adım 12: Henüz sınıflandırılmayan üç örnek kaldığından üçüncü iterasyona geçilir.

Üçüncü iterasyon

Adım 4: Üçüncü örneği seçelim. Bu örnek aşağıdaki gibidir:

k_1=ÇY; k_2=D; k_3=Y; k_4=ÇD şartlarında İhtiyaç Karşılama =Az Karşılanmış

Adım 5: $N_c = N_{cmin}-1$

$N_c=1-1=0$

Adım 6: Eğer $N_c < N_a$ ise $N_c=N_c+1$ (Eğitim setinde $N_a=4$; k_1, k_2, k_3, k_4)

$N_c=N_c+1$; $N_c=0+1=1$

Adım 7: Eldeki örnekte kullanılan karakteristiklerin değerleriyle {ÇY, D, Y, ÇD} değerler dizisi elde edilir.

Adım 8: $N_c=1$ olduğundan bu değerler tek tek ele alınmalıdır.

Adım 9: Bu değerler tek tek ele alındığında k_4=ÇD değeri sadece bir sınıfta geçmektedir. Bunu kural haline getirdiğimizde aşağıda görülen sonuç elde edilmektedir.

Kural 3: *Eğer* k_4 *=ÇD ise İhtiyaç Karşılama=Az Karşılanmış*

Adım 10: Çıkarılan kuralı genel kural olarak seçilir.

Adım 11: Bu kural ile sadece üçüncü örnek sınıflandırılabilir. Bu örnek kural çıkarma işlemi için artık göz önüne alınmamalıdır.

Adım 12: Henüz sınıflandırılmayan iki örnek kaldığından dördüncü iterasyona geçilir.

Dördüncü iterasyon

Adım 4: Dördüncü örneği seçelim. Bu örnek aşağıda görüldüğü gibidir: k_1=N; k_2=D; k_3=N; k_4 =ÇY şartlarında İhtiyaç Karşılama =Kısmi Karşılanmış

Adım 5: N_c= N_{cmin}-1

N_c=1-1=0

Adım 6: Eğer Nc< N_a ise N_c=N_c+1 (Eğitim setinde N_a=4; k_1, k_2, k_3, k_4)

N_c=N_c+1; N_c=0+1=1

Adım 7: Eldeki örnekte kullanılan karakteristiklerin değerleriyle {N, D, N, ÇY} değerler dizisi elde edilir.

Adım 8: N_c=1 olduğundan bu değerler tek tek ele alınacaktır.

Adım 9: Bu değerler tek tek ele alındığında k_1=N değeri sadece bir sınıfta geçmektedir. Bunu kural haline getirdiğimizde aşağıda görülen sonuç elde edilmektedir.

Kural 4: *Eğer* k_1 *=N ise İhtiyaç Karşılama=Kısmi Karşılanmış*

Adım 10: Çıkarılan kuralı genel kural olarak seçeriz.

Adım 11: Bu kural ile sadece dördüncü örnek sınıflandırılabilir. Bu örneğe ait kural çıkarma işlemi bundan sonra göz önüne alınmayacaktır.

Adım 12: Henüz sınıflandırılmayan bir örnek kaldığından beşinci iterasyona geçilir.

Beşinci iterasyon

Adım 4: İkinci örneği seçelim. Bu örnek aşağıda görüldüğü gibidir:

k_1=ÇY; k_2=D; k_3=Y; k_4=ÇY şartlarında İhtiyaç Karşılama=Çoğu Karşılanmış

Adım 5: N_c= N_{cmin}-1

N_c=1-1=0

Adım 6: Eğer Nc< N_a ise N_c=N_c+1 (Eğitim setinde N_a=4; k_1, k_2, k_3, k_4)

N_c=N_c+1; N_c=0+1=1

Adım 7: Eldeki örnekte kullanılan karakteristiklerin değerleriyle {ÇY, D, Y, ÇY} değerler dizisi elde edilir.

8. Adım: N_c=1 olduğundan bu değerler tek tek ele alınacaktır.

Adım 9: Görüldüğü gibi bu değerlerin tümü tek tek alındığında birden çok sınıfta yer almaktadır. Bu durumda kural çıkarılamaz Adım 6' ya dönmek gerekir.

Adım 6: N_c=1+1=2

Adım 7: Değerler dizisi eskisi gibidir. {ÇY, D, Y, ÇY}

Adım 8: Bu değerlerin üçlü kombinasyonları aşağıda görüldüğü gibidir:

1. k_1=ÇY, k_2=D, k_3=Y

2. k_1=ÇY, k_2=D, k_4=ÇY

3. k_1=ÇY , k_3=Y, k_4=ÇY

4. k_2=D, k_3=Y, k_4=ÇY

Adım 9: Yukarıda ki 2, 3, 4 nolu kombinasyonlar sadece 5. örneği açıkladığından kural haline getirilebilir. Bu kurallar ise aşağıda ki gibidir.

Kural 5: *Eğer* k_1*=ÇY ve* k_2*=D ve* k_4*=ÇY ise İhtiyaç Karşılama=ÇK*

Kural 6: *Eğer* k_1*=ÇY ve* k_3*=Y ve* k_4*=ÇY ise İhtiyaç Karşılama=ÇK*

Kural 7: *Eğer* k_2*=D,* k_3*=Y,* k_4*=ÇY ise İhtiyaç Karşılama=ÇK*

Adım 10: Bu kurallardan üçü ilede sadece Tablo 3.6'da ki 5 nolu örnek sınıflandırılabilir. O halde bunlardan her hangi biri seçilebilir. Örneğin biz ilk kuralı seçelim.

Adım 11: Elde edilen kural ile sınıflandırılan 5 nolu örnek kural çıkarma için eğer ondan sonra başka örnekler olsaydı göz önüne alınmayacaktı.

Adım 12: Görüldüğü gibi tüm örnekler sınıflandırılmıştır ve bu nedenle iterasyona son verilebilir.

Bu kurallar bir araya getirildiğinde eğitim setinden elde edilen bilgi tabanı (kurallar seti) Tablo 3.7' de verilmiştir.

Tablo 3.7. RULES3 Algoritması ile çıkarılan kurallar seti

Kural 1	*Eğer* k_1 *=ÇD ise İhtiyaç Karşılama =Hiç Karşılanmamış*
Kural 2	*Eğer* k_2 *=ÇY ise İhtiyaç Karşılama =Tümü Karşılanmış*
Kural 3	*Eğer* k_4 *=ÇD ise İhtiyaç Karşılama=Az Karşılanmış*
Kural 4	*Eğer* k_1 *=N ise İhtiyaç Karşılama=Kısmi Karşılanmış*
Kural 5	*Eğer* k_1=ÇY ve k_2=D ve k_4=ÇY ise İhtiyaç Karşılama=*Çoğu Karşılanmış*

RULES3 algoritmasının çalışma mantığı gereği, öncelikle bir eğitim seti hazırlanmalıdır. Bir eğitim setinin elemanları; örneklere ait karekteristiklerin karşılanma miktarları, karekteristiklerin ağırlıkları ve sınıf değerleridir. Bu elemanlar daha sonra karşılaşılan ve çözüm üretilmesi gerekli durumların değerlendirilmesinde fayda sağlayacak kuralların üretilmesinde önemli bir yere sahiptir. Eğitim setinin girdileri bu elemanlar olarak kabul edilirse, çıktıları da kurallardır. Kurallar her bir örneği temsil yeteneğine sahiplik olarak değerlendirilebilir.

Geliştirilen Modelin Uygulaması

Gerçekleştirilen uygulamada bir eleman temin görevinin en uygun şekilde yerine getirilebilmesi ve sonuçta en uygun elemanın seçilmesi amacıyla doğru ve tam ihtiyaçlar kümesinin oluşturulması amaçlanmıştır. Daha öncede vurgulandığı gibi ihtiyaçların belirlenmesi çalışmalarında çok çeşitli metot, teknik ve araç kullanılmaktadır. Bu çalışmada ihtiyaçların sunumu aşamasında beş farklı teknikten faydalanılmıştır. Çalışmada kullanılan ihtiyaç sunum teknikleri; Akış

diyagramı, Senaryo tekniği, Karar haritası, Çağrıştırıcı bilgi haritası ve Benzerlik diyagramı tekniğidir. Amaç bu tekniklerden hareketle en doğru ihtiyaçlar kümesini oluşturmak ve bunların doğruluğunu performans kriterleri belirleyerek test etmektir. Bu doğrultuda bir firmanın belirli bir iş alanına yönelik eleman alacağı varsayılarak bu işe en uygun elemanın alınması (seçimi) amaçlanmıştır. Yukarıda sayılan beş sunum tekniği aracılığı ile 38 ihtiyaç belirlenmiş, bu ihtiyaçlar Browne ve Rogich'in geliştirdiği [15] ihtiyaç sınıflarından hareketle; amaç, süreç, görev ve bilişim seviyeli olarak sınıflanmıştır.

Bir sonra ki aşamada sunum tekniklerinin kombinasyonları oluşturulmuş, oluşturulan kombinasyonlarla belirlenen ihtiyaçlara ROC algoritması uygulanmıştır. Uygulamadan elde edilen sonuçlar, grup teknolojisi başarı ölçütleri aracılığı ile değerlendirilmiş, fakat elde edilen sonuçlar ve başarı ölçütleri çalışmanın amacına uygun olmadığından yeni başarı ölçütleri geliştirilmiştir. Bunlar; "tekniklerin ihtiyaç belirleme etkinlik ölçüsü, tekniklerin ihtiyaç doğrulama verimlilik ölçüsü ve tekniklerin ortalama verimlilik ölçüsü" dür.

Bu çalışmaya paralel olarak, endüktif öğrenme yaklaşımı ve RULES-3 algoritması aracılığı ile teknik ve teknik kombinasyonları için bir eğitim seti oluşturulmuş ve bu eğitim setinden 19 kural üretilmiştir.

İhtiyaçların çıkartılması

Önerilen modelle uygulama gerçekleştirebilmek için ihtiyaç duyulan veriler model çerçevesinde tanımlı yöntemler olan; alan uzmanları/sistemin kullanıcılarıyla karşılıklı görüşmeler; Personel

Daire Başkanlığı personeli ile mülakat, daire başkanlığına ait el kitaplarının (yönergeler, prosedürler) incelenmesi, literatür bilgileri ve ilave olarak piyasada bu konuya yönelik yazılımlar ışığında gerçekleştirilmiştir.

Bu metotların bir arada bütünleşik olarak kullanılması, bilgi çıkarımında bilgi eksiğine neden olmamak amacıyladır. Bu metotlar aracılığı ile elde edilen veriler düzenli bilgiler haline dönüştürülerek fonksiyonların gerçekleşmesine olanak sağlayan bilgi akışları elde edilmiştir. Yukarıda belirtilen yöntemlerle bilgi toplamada Personel Daire Başkanlığı'nda çalışan 10 sistem kullanıcısından faydalanılmıştır. Bunlar, işletmede yürürlükte olan sistemin kullanıcılarıdır. Bu kişilerin 1'i üst düzey, 2'si orta düzey ve 7'si de iş gören seviyesinde ki kişilerdir.

Üst düzey sistem kullanıcısı; Personel Dairesi Başkanı, orta düzey sistem kullanıcıları; Özlük İşleri Şube Müdürü ve Tahakkuk Şube Müdürü ve 7 iş gören seviyede ki kullanıcı ise bu servisin çalışanlarıdır.

Mülakat: Sistem ihtiyaçlarını belirleme de yardımcı olacak bilgilerin toplanmasında kullanılan yöntemlerden biri mülakattır. Mülakat yöntemi, sistem ihtiyaçlarının toplanmasında 3 üst ve orta düzey yöneticiyle ve 7 iş gören seviyede ki personelle gerçekleştirilmiştir. Mülakatlar da, 4 deneyimli sistem analistinden faydalanılmıştır. Analistler mevcut sistemin yöneticileriyle 45 dakika ile 60 dakika arasında süren ortalama 4'er oturum gerçekleştirmişlerdir. Sistem analistleri karşılıklı görüşmeler sırasında gerekli gördükleri bilgileri not

almışlardır. Bu notlardan elde edilen bilgiler rapor olarak hazırlanmış ve raporda ki bilgiler çalışmaya veri olarak dahil edilmiştir. Bu veriler/ihtiyaçlar bir sonra ki aşama olan "ihtiyaçların sunumu" aşamasında model kapsamında ki ihtiyaç sunum tekniklerine aktarılarak ifade edilmiştir.

Yazılım inceleme: Bu alana yönelik üretilmiş bir kaç yazılım incelenmiş, yazılımların modülleri, alt modülleri ve içeriklerinden elde edilen bilgiler ihtiyaçlar olarak düşünülerek çalışmada kullanılmıştır. Araştırmalar sonucunda, piyasada insan kaynakları sistemi modüllerinin içerdiği fonksiyonlar tespit edilmiş ve genellikle; kurumsal planlama, eleman seçme ve yerleştirme, sicil/bordro/eğitim yönetimi ve raporları içerdiği belirlenmiştir.

Literatür inceleme: Genel anlamda sistemin anlaşılması ve analistlerin sisteme hakim olabilmelerini sağlayabilmek için sistemin fonksiyonlarına yönelik literatür taraması gerçekleştirilmiş alan ve fonksiyonları detaylı olarak incelenmiştir.

Doküman analizi: Belge taraması/doküman inceleme de analistler Personel Daire Başkanlığı içerisinde ki kullanım el kitapları, prosedürler, iş akış şemaları, görev tanımları ve sorumluluklar, raporlar, formlar vb. belgelerden faydalanmışlardır. Bu amaçla sistemde incelemeler gerçekleştirilmiş ve elde edilen bilgiler değerlendirilmiştir. Bu sayede gerçekleşecek işlemler ve sistemin fonksiyonları için hangi tür verilere gereksinim olduğu ve bunların ne şekilde hazırlanması gerektiği sağlıklı bir şekilde anlaşılmıştır.

Tüm bu yöntemler aracılığı ile toplanmış olan bilgiler/ihtiyaçlar model kapsamında belirlenmiş olan sunum teknikleri yardımıyla ifade edilmiştir. Bundan sonra ki bölümde 5 farklı ihtiyaç sunum tekniği aracılığı ile toplanmış olan ihtiyaçların sunumu gerçekleştirilmiştir.

İhtiyaçların sunulması

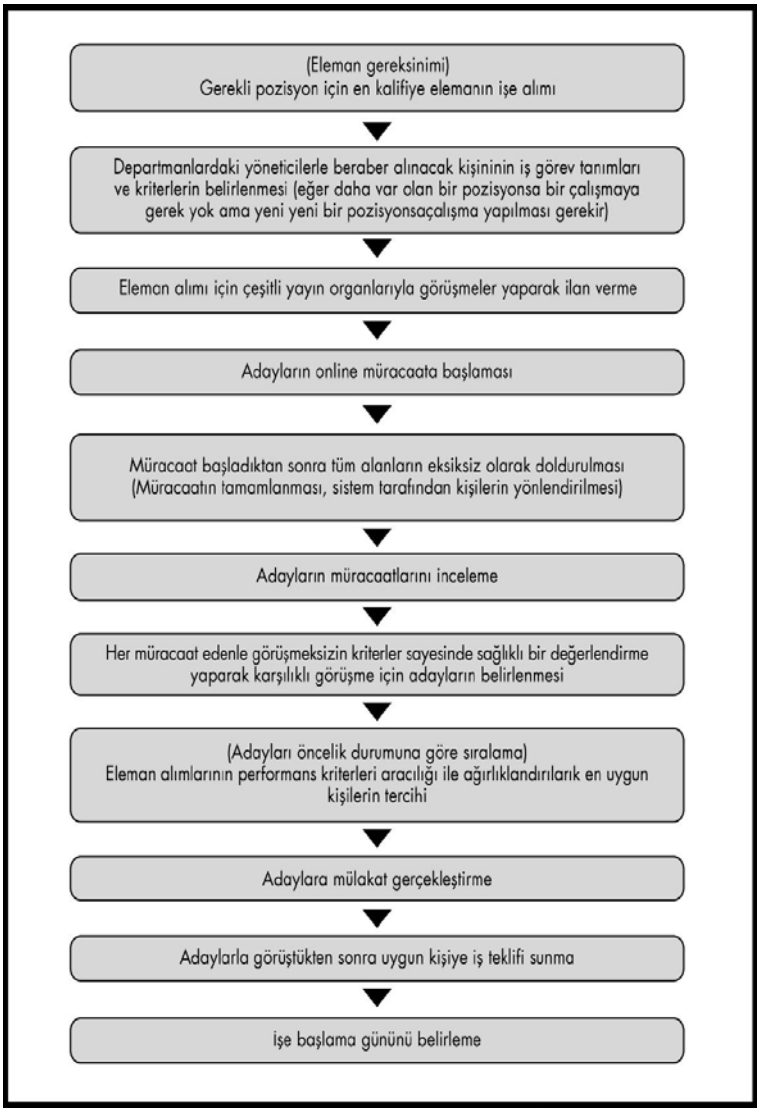
Yukarıda ki bölümde anlatıldığı şekilde toplanmış olan ihtiyaçlar model kapsamınca belirlenmiş olan 5 farklı teknik aracılığı ile sunulmuştur. Bunlar; Akış diyagramı, Çağrıştırıcı bilgi haritası, Karar haritası, Senaryo ve Benzerlik diyagramı tekniklerdir.

Akış diyagramı tekniği kullanılarak ihtiyaçların sunulması

İlk olarak akış diyagramı tekniği kullanılarak davranışsal boyuta yönelik olarak toplanmış ihtiyaçlar ifade edilmiştir. Akış diyagramı tekniği ile bir eleman temin sürecine ait aşama ve ihtiyaçlar Şekil 3.9'da gösterilmiştir.

Yukarıda akış diyagramı tekniği yardımı ile ifade edilen her aşama bir veya birden fazla ihtiyaç olarak kabul edilmiş ve toplam 12 ihtiyaç belirlenmiştir. Bu ihtiyaçlar sistem geliştirme çalışmaları sırasında bazen gerçekleşmesi gereken faaliyet bazen de modüllerde ki alanlar olarak değerlendirilir.

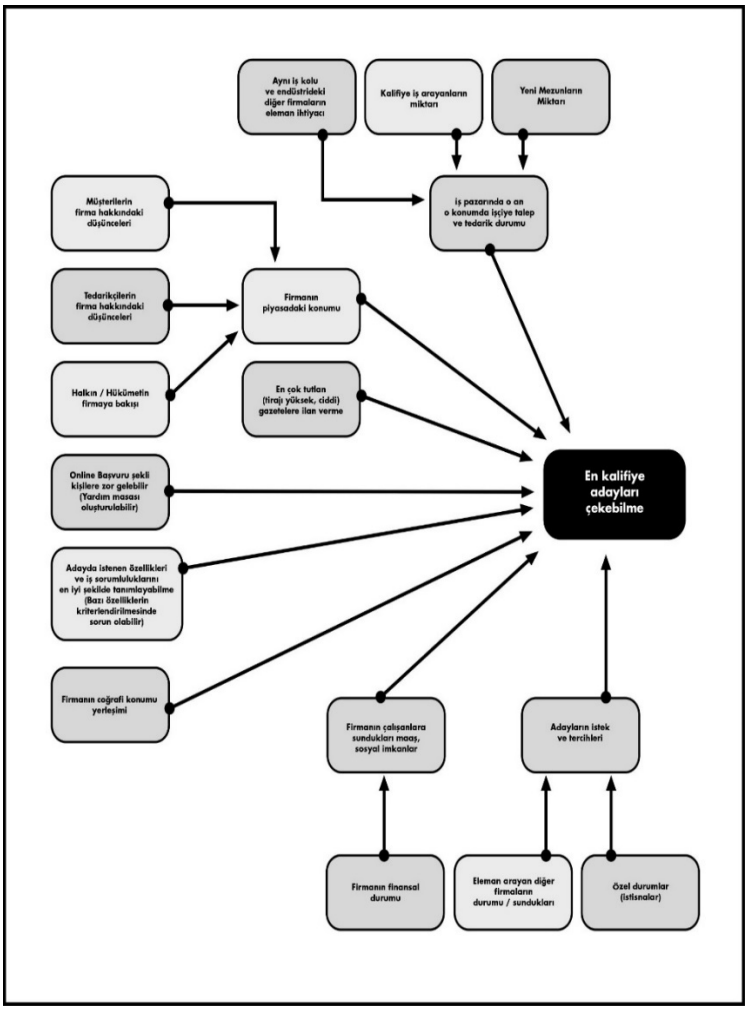
Amaç: Gerekli pozisyon için en kalifiye elemanın işe alınması



Şekil 3. 9 Eleman temin için hazırlanan akış şeması örneği

Çağrıştırıcı bilgi haritası tekniği kullanılarak ihtiyaçların sunulması

İkinci olarak çağrıştırıcı bilgi haritası tekniği kullanılarak, çeşitli yöntemlerle toplanmış olan ihtiyaçlar ifade edilmiştir. Çağrıştırıcı bilgi haritası tekniği ile bir eleman temin sürecine ait ihtiyaçlar Şekil 3.10'da gösterilmiştir.

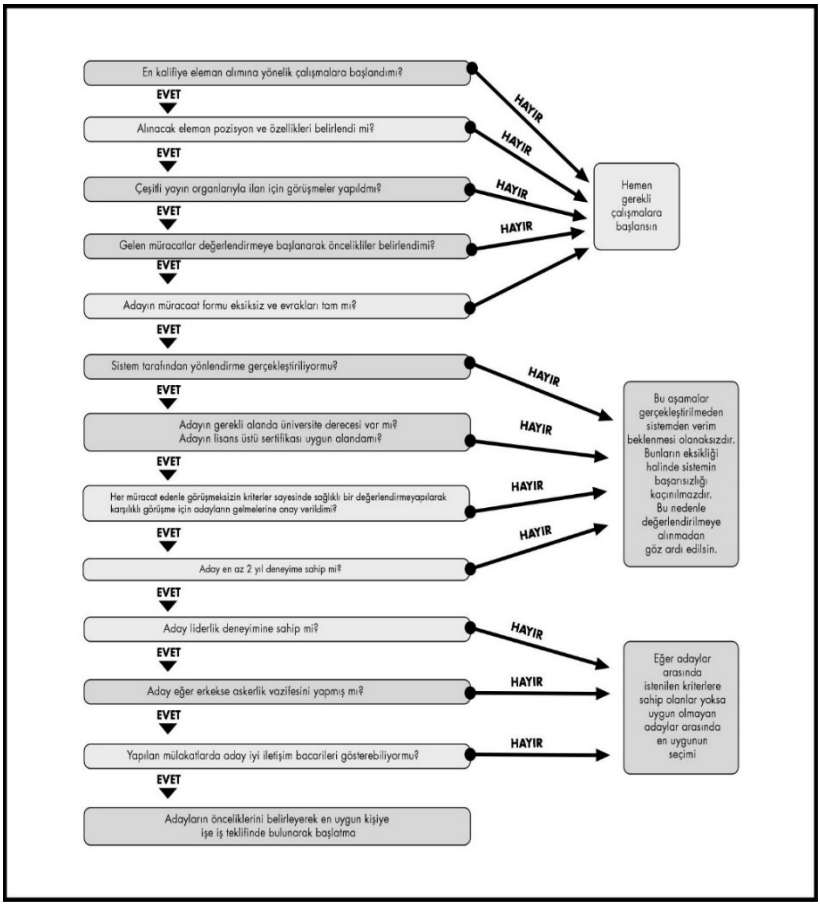


Şekil 3. 10 Eleman temin için hazırlanan çağrıştırıcı bilgi haritası

Çağrıştırıcı bilgi haritası tekniğinin birinci aşamasında ki süreçler ihtiyaç olarak kabul edilmiş ve bu teknik yardımı ile 9 ihtiyaç ifade edilmiştir. Ayrıca 9 tane de ikinci seviyede ihtiyaç belirlenmiştir. İkinci seviyede ki ihtiyaçlar çalışmada değerlendirilmeye alınmamış göz ardı edilmiştir.

Karar haritası tekniği kullanılarak ihtiyaçların sunulması

Eleman temin sistemi için karar haritası tekniği kullanılarak ifade edilen ihtiyaçlar Şekil 3.11'de gösterilmiştir.



Şekil 3. 11 Eleman temin için hazırlanan karar haritası

Karar haritası tekniğinin akış sürecinin her bir alt süreci/aşaması bir veya birden fazla ihtiyaç olarak değerlendirilmiş ve bu teknik yardımı ile eleman temin sürecine yönelik 16 ihtiyaç belirlenmiştir.

Senaryo tekniği kullanılarak ihtiyaçların sunulması

Toplanmış olan ihtiyaçlar eleman temin sürecini kapsayan bir senaryo geliştirilerek ifade edilmiştir. Geliştirilen bu senaryoda her durumun düzgün bir akış da gerçekleştiği düşünülmüştür. Kısacası her aşamada sorun yaşanmaksızın her faaliyetin olması gerektiği gibi gerçekleştiği kabul edilmiştir. Aşağıda ki paragrafta bir hikaye formatında yazılmış olan senaryo Tablo 3.8'de senaryo tekniğinin sahip olması gereken şekil gereğince düzenlenmiş ve oluşturulan tabloda ki her bir olay eleman temin sürecine ait ihtiyaç olarak değerlendirilmiştir.

Üretim planlama bölüm müdürü işten ayrılan planlama mühendisi yerine bir mühendis alınması isteğini fabrika müdürüne iletir. Fabrika müdürü insan kaynakları müdürünü arayarak konuyla ilişkin çalışmaları gerçekleştirmelerini ister. Bunun üzerine insan kaynakları müdürü gerekli elemana bir planlama mühendisinde olması gereken özellikleri de belirten bir ilan verilmesi için gerekenleri yapmasını ister. İnsan kaynaklarında ki bu kişi tüm çalışmaları gerçekleştirir. Aynı zamanda sistemin on-line başvuru hizmeti aracılığı ile başvurular gelmeye başlar. Bu sayede tüm başvuranlarla görüşmeye gerek kalmaksızın süreç devam edecektir. Tüm bu başvuruları derleyen kişi hazırladığı listeyi insan kaynakları müdürüne verir. Müdür listeyi gözden geçirdikten sonra bu kişilerden uygun olanların aranarak

randevu verilmesini ister ve listedeki kişilerle mülakat gerçekleştirir. Bunlar arasından en uygun kişiyi seçerek işe başlama işlemleri için gerekli servise gönderir.

Bu senaryo belli bir formata oturtulması ve diğer tekniklerle kıyaslanabilmesi için bir senaryo, bulunan özellikleri ön plana çıkaracak şekilde düzenlenmelidir. Öncelikle bir senaryoda yapılan literatür taraması sonucunda belirlendiği üzere belirli özelliklere sahip olmalıdır. Öncelikle bir senaryoda aktörler ve aktiviteler bulunmalıdır.

Senaryoda ki aktörler genellikle o sistemin kullanıcılarıdır. Bu aktör ve aktiviteler belirlenerek hikaye formatındaki bu senaryo bileşenlerine ayrılır;

Buradaki aktörler;

Üretim Planlama Müdürü
Fabrika Müdürü (Genel Müdür)
İnsan Kaynakları Müdürü
İnsan Kaynakları Elemanı
Ajans Görevlisi
İşe Başvuran Elemanlardır.

Bu aktörlerin eleman temin sistemi için geçerli olan sistemin paylaşımcıları olduğunu söyleyebiliriz.
Senaryonun İsmi: Eleman temin (her şeyin düzgün bir akışta gerçekleştiği)

Amaç: En kalifiye elemanın işe alınması için ihtiyaçların detaylandırılması

Geliştiren: Tijen ÖVER

Hedef: En kalifiye elemanın belirlenerek işe başlatılması

Tablo 3. 8 Eleman temin için hazırlanan senaryo

Olay No	Aktörler	Eylemler
1	**Ü. Planlama Müdürü**	Ayrılan mühendisin yerine uygun birinin alınması için genel müdür ile görüşme (Eleman gereksiniminin dile getirilmesi)
2	**Fabrika Müdürü/Genel Müdür**	Gerekli elemanın alımını gerçekleştirmek amacıyla insan kaynakları müdürünü çağırarak görevlendirme
3	**İnsan Kaynakları Müdürü**	Uygun personeli çağırarak iş sorumluluğu ve gerekli kriterlerin belirli olup olmadığı konusunda bilgi alama ve gerekli işlemleri yapmasını isteme
4	**İnsan Kaynakları Personeli**	Yayın gruplarıyla görüşerek eleman alımı için ilan verme
5	**Adaylar**	İlanı gören adayların firmanın web sayfası aracılığı ile sisteme girerek insan kaynakları departmanına CV' lerini ve gereken evrakları ulaştırması
6	**İnsan Kaynakları Personeli**	Gereken evrakları tam olarak ulaştıran ve belli başlı özellikleri taşıyan adayların listesini oluşturarak insan kaynakları müdürüne iletme

7	**İnsan Kaynakları Müdürü**	Bu türlü bir sistemle, her müracaat edenle görüşmeksizin kriterler sayesinde sağlıklı bir değerlendirme yaparak ve başvuruları şahsi değerlendirmeden çıkararak karşılıklı görüşme için adayların gelmelerine onay verme
8	**İnsan Kaynakları Personeli**	Adayları arayarak yer, saat ve gün vererek mülakat için haber verme
9	**Adaylar**	Söylenilen gün, saat ve yerde hazır bulunarak görüşmelerini gerçekleştirme
10	**İnsan Kaynakları Müdürü**	Yaptığı görüşmeler sonunda en uygun adayı belirleyerek, işe başlamayla ilgili işlemleri gerçekleştirmesi için kişiyi insan kaynakları personeline sevk etme

Senaryoda ki her bir olayın eleman temin süreci için bir veya birden fazla ihtiyaç olarak kabul edilmesinden hareketle, senaryo tekniği yardımı ile eleman temin süreci için 13 ihtiyaç belirlenmiştir.

Benzerlik diyagramı tekniği kullanılarak ihtiyaçların sunulması

Benzerlik diyagramı tekniği yardımı ile ihtiyaçların belirlenebilmesi için öncelikle mevcut sistemin kullanıcıları ve sistem geliştirme konusunda deneyimli kişilerden grup oluşturulmuştur. Grup oturumları aracılığı ile tartışmalar düzenlenmiş ve beyin fırtınası gerçekleştirilmiştir. Bu gerçekleşen oturumlar sırasında sisteme yönelik tartışma ve konuşmalar sistemli bir şekilde not alınmış ve grup içinde öne sürülen savlar Tablo 3.9'da verilmiştir. Tablo şeklinde verilen bu savlar benzerlik diyagramı kullanım mantığı gereğince düzenlenmiş,

savlardan hareketle gerçekleştirilmesi gereken sistemin kapsaması gereken alt sistemler belirlenmiştir.

Tablo 3. 9 Grup içinde geliştirilen savlar: Bir firmanın yöneticileri tarafından işçi temini sırasındaki görüşmeleri basamak basamak göstermektedir

Kişi A)	İşe alma kararlarımıza belirli bir yön sağlayacak bir sisteme ihtiyacımız var, hatta iş başvuruları bu sistem aracılığı ile iyi tasarlanmış bir arayüz sayesinde bile olabilir müracaat eden kişiler gerekli bilgileri bu şekilde bile ulaştırabilir.
Kişi B)	Ama bizim adımıza kararlar verecek bir sistem istemiyoruz.
Kişi A)	Haklısın ama biz kararlarımızı bir düzene oturtmak zorundayız.
Kişi C)	Ayrıca kullandığımız kriterlerin birbirleriyle tutarlı olması lazım
Kişi D)	Bu sistem bize kriterleri seçme konusunda yardımcı olabilir mi? Yani biz bu sisteme pozisyonun ne olduğunu söylersek, sistemin bize kriterleri söylemesi mümkün mü?
Kişi B)	Bunu bir uzman sistem olmadan yapmamız imkansız ve bizim bir uzman sisteme ihtiyacımız olduğunu düşünmüyorum.

Kişi E)	Bir yerde okudum; basit uzman sistemler çalışanların seçimin de yöneticilerden daha etkin kararlar verebiliyorlar, ayrıca yılın genellikle 2,3 ayında eleman arıyoruz ve başvuruların değerlendirilmesi zaman alıyor. Bu tür bir sistem büyük kolaylık sağlayabilir diye düşünüyorum.

Tablo 3. 9'un devamı: Grup içinde geliştirilen savlar: Bir firmanın yöneticileri tarafından işçi temini sırasındaki görüşmeleri basamak basamak göstermektedir

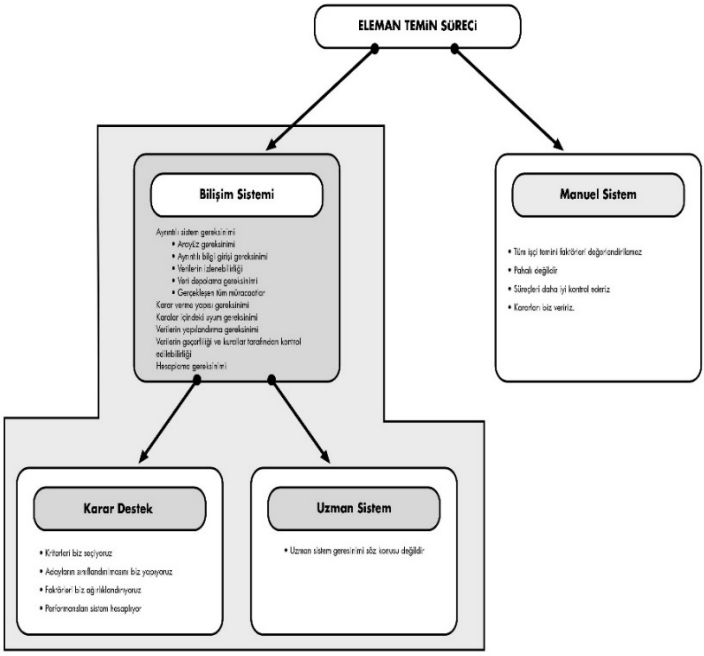
Kişi C)	Ben onu bunu bilmem, bu cins kararlar alabilen bilgisayarlara güvenmem.
Kişi A)	Ben biliyorum ki; uzman sistemlerin kriterlerin tutarlı olarak kullanılması açısından avantajları var, kriterleri tutarlı olarak uygulamayı adil kararlar vermek ve en iyi kişileri işe almak açısından kullanmalıyız.
Kişi C)	Sanıyorum , bizim ihtiyacımız olan kararları bir düzene oturtan ama son seçimi bize bırakan bir sisteme ihtiyacımız var, biliyorsun işte bu bir karar destek sistemi.
Kişi E)	Benim karar destek sistemlerine bir itirazım yok ama, bırakalım bilgisayarlar bilgi işlesin, o okuduğum makaleye göre yöneticiler kendi oluşturdukları kriterleri bile tutarlı olarak uygulayamıyorlarmış.
Kişi D)	Yani biz sisteme hangi kriterleri kullanmamız gerektiğini söyleyeceğiz, o da bize hangi adayları seçmemiz gerektiğini mi söyleyecek?
Kişi E)	Evet aynen öyle
Kişi B)	Ben böyle bir sisteme ihtiyacımız olduğuna inanmıyorum, hem pahalı hem de elemanların işe alış kararlarında işe yaramayacak, çünkü kişilerin bilgisayar tarafından değerlendirilemeyecek özellikleri olduğunu düşünüyorum.

Kişi F) Evet aynı karardayım, bir bilgisayar size işe başvuran kişinin görüntüsü, yüzündeki ifade yada işe olan ilgisini nasıl ölçebilir, ölçemez.

Kişi A) Kolay, bu gibi faktörleri bir puanlama sistemine oturtursun.

Kişi C) Ama bu gibi faktörlere fazla ağırlık vermemeliyiz, yoksa o an yüzünde aptal bir ifade vardı diye en kalifiye eleman elenmiş olabilir.

Öne sürülen savlar kapsamı alanına girdikleri sistem veya ifade ettiklerine göre gruplanmış ve gruplardaki ifadeler ihtiyaçlar olarak değerlendirilmiştir. Tüm bu ifade edilen sistem/alt sistemlerine ait yapı ve benzerlik diyagramı tekniği ile bir eleman temin sürecine ait ihtiyaçlar Şekil 3.12'de verilmiştir.



Şekil 3. 12. Eleman temin süreci için hazırlanan benzerlik diyagramı

Model kapsamında tanımlı olan; akış diyagramı, çağrıştırıcı bilgi haritası, karar haritası, senaryo ve benzerlik diyagramı tarafından sunulan ihtiyaçların oluşturduğu küme, sistemin ihtiyaç kümesi olarak kabul edilmiştir. İhtiyaç kümesinde toplam 38 ihtiyaç bulunmaktadır. İhtiyaçlar ve ihtiyaçların sunumunda kullanılan teknikler Tablo 3.10'da verilmiştir.

Tablo 3.10. Eleman temin sistemi için belirlenen ihtiyaçlar ve kullanılan teknikler [1/8 Nolu ihtiyaçlar]

		Teknikler				
İhti. No	**İhtiyaçlar**	**Akış diy.**	**Çağrıştırıcı Bil. Har.**	**Karar har.**	**Senaryo**	**Benzerlik diy.**
1	Eleman gereksinimi (Gerekli pozisyon için en kalifiye elemanın işe alınması)	*	*	*	*	
2	İş sorumluluğunun planlanması	*	*	*	*	
3	Yayın organlarına ilan verme	*	*	*	*	
4	Adayların müracaatı başlatması (Müracaatı yapanın ilk düşüncesi müracaat sürecini başlatmaktır. Müracaatı başlatmak ise müracaatı başlat ikonuna 2 kere tıklatmakla gerçekleşir)	*			*	
5	Müracaat başladıktan sonra ki işlemler (Başvuru işlemleri) (Kişi müracaat sürecini başlattığı zaman, tam müracaatı gerçekleştirsin veya gerçekleştirmesin tüm kriterlere ait bilgileri doldurması konusunda ona kılavuzluk eder)	*		*		
6	Kişileri yönlendirme gereksinimi	*		*		
7	Müracaatların incelenerek değerlendirilmesi	*		*	*	
8	Bu türlü bir sistemle her müracaat edenle görüşmeksizin kriterler sayesinde sağlıklı bir değerlendirme yaparak ve başvuruları şahsi değerlendirmeden çıkararak karşılıklı görüşme için adayların gelmelerine onay verme	*		*	*	

Tablo 3.10'un devamı. Eleman temin sistemi için belirlenen ihtiyaçlar ve kullanılan teknikler [8/15 Nolu ihtiyaçlar]

		Teknikler				
İhti. No	**İhtiyaçlar**	**Akış diy.**	**Çağrıştırıcı Bil. Har.**	**Karar har.**	**Senaryo**	**Benzerlik diy.**
8	Bu türlü bir sistemle her müracaat edenle görüşmeksizin kriterler sayesinde sağlıklı bir değerlendirme yaparak ve başvuruları şahsi değerlendirmeden çıkararak karşılıklı görüşme için adayların gelmelerine onay verme	*		*	*	
9	Mülakat	*		*	*	
10	Önceliklilerin tespit edilmesi (Eleman alımlarının performans kriterleri aracılığı ile ağırlıklandırılarak en uygun kişinin tercihi)	*			*	
11	İş teklifi	*		*	*	
12	İşe başlama gününün belirlenmesi	*		*	*	
13	Paylaşımcılar (Senaryoda ki aktörler) İşletme içindeki tüm departmanlar ve müracaatı yapan kişiler)				*	
14	İşçiye olan talep ve tedarik		*			
15	Firmanın piyasada ki konumu (O dönemde firmayla ilgi çıkan olumsuz bir haber kalifiye kişilerin müracaat etmemesi gibi bir durum yaratabilir)		*			

Tablo 3.10'un devamı Eleman temin sistemi için belirlenen ihtiyaçlar ve kullanılan teknikler [16/26] Nolu ihtiyaçlar]

		Teknikler				
İhti. No	**İhtiyaçlar**	**Akış diy.**	**Çağrıştırıcı Bil. Har.**	**Karar har.**	**Senaryo**	**Benzerlik diy.**
16	On-line başvuru şekli kişilere zor gelebilir (Yardım masası oluşturulabilir)		*			
17	Firmanın çalışanlara sundukları (maaş, sosyal imkanlar)		*			
18	Firmanın coğrafi konumu (Verilen ilanlar ve internet aracılığı ile on-line müracaatlar artabilir. Firmanın coğrafi konumu olumlu veya olumsuz etki yaratabilir)		*			
19	Adayların istek ve tercihleri		*			
20	Adayın müracaat formunun eksiksizliği			*	*	
21	Adayların eğitim durumları ve iş alanına uygunluğu			*		
22	İş deneyimleri			*		
23	Liderlik becerileri			*		
24	Askerlik			*		
25	İletişim becerileri			*		
26	Firmalar genellikle yılın 2,3 ayında eleman ararlar ve başvuruların değerlendirilmesi zaman alır. Bu tür bir sistem büyük kolaylık sağlayabilir					*

Tablo 3.10'un devamı Eleman temin sistemi için belirlenen ihtiyaçlar ve kullanılan teknikler [27/38] Nolu ihtiyaçlar]

		Teknikler				
İhti. No	**İhtiyaçlar**	**Akış diy.**	**Çağrıştırıcı Bil. Har.**	**Karar har.**	**Senaryo**	**Benzerlik diy.**
27	On-line başvuru için web sayfası gereksinimi				*	*
28	Karar verme yapısı gereksinimi					*
29	Kararlar arasında ki uyum gereksinimi					*
30	Karar destek sistemi gereksinimi					*
31	Veri depolama gereksinimi					*
32	Ayrıntılı bilgi girişi gereksinimi					*
33	Arayüz gereksinimi					*
34	Müracaatlar alfabetik olarak veya iş gruplarına göre kayıt altında olması					*
35	Verilerin izlenebilirliği					*
36	Verilerin yapılandırılma gereksinimi					*
37	Verinin geçerliliğinin kurallar tarafından kontrol edilebilirliği					*
38	Hesaplama gereksinimi					*
	Toplamda belirlenen ihtiyaç	**12/38**	**9/38**	**16/38**	**13/38**	**13/38**

Tablo 3.10; ihtiyaç numaralarını, belirlenen ihtiyaçları ve belirleyen teknikleri içermektedir. Örneğin 1 No'lu ihtiyaç; akış diyagramı, çağrıştırıcı bilgi haritası, karar haritası ve senaryo tekniği tarafından sunulmuştur. Teknikler altında ihtiyaçlara karşılık gelen yıldız (*) işaretleri bu durumu ifade etmektedir.

İhtiyaç kümesi oluşturulurken birkaç teknik tarafından sunulmuş olan herhangi bir ihtiyaç sadece bir defa sunulmuş gibi kabul edilmiştir. Bu şekilde değerlendirildiğinde eleman temin sisteminin ihtiyaçlar kümesinin 38 elemanı bulunmaktadır.

İhtiyaçların doğrulanması

İhtiyaç belirleme sürecinde belirlenmiş sunum teknikleriyle ifade edilen ihtiyaçlar, o alanın uzmanları/sistemin kullanıcıları veya o konu hakkında bilgi sahibi analistler tarafından incelenerek doğruluk ve geçerlikleri hakkında karar verilmelidir .

Beş sunum tekniği ile ifade edilen ihtiyaçlar geçerliliklerinin sağlanması amacıyla, 6 deneyimli sistem analisti tarafından incelenmiştir. Bu incelemeler oturumlar şeklinde gerçekleştirilmiştir. Bu oturumlar her bir sunum tekniği için iki kez olmak kaydıyla on kez tekrarlanmıştır. Her bir sunum tekniği için gerçekleşen oturumlar yaklaşık 45 ile 60 dakika civarında sürmüştür. Bu süreçte bazı çelişkiler ve tutarsızlıklar belirlenmiş, gerçek alan çalışması sırasında sorun teşkil etmemesi amacıyla not alınmıştır. Sonuç olarak, uzman analistler tarafından sistemin tüm gereksinimlerini karşılamaya yeterli olduğu

kararı verilmiştir. Bu doğrulamada analistler için alan ile ilgili yaptıkları araştırmalar ve deneyimleri önemli bir yere sahiptir.

Belirlenen ihtiyaçların sınıflandırılması

Bilişim sistemi geliştirme çalışmalarında ihtiyaç belirleme önemli bir yere sahiptir. Oluşturulan ihtiyaç kümesinde ki eksiklikler başarısızlıkla sonuçlanan sistem geliştirme çalışmalarına neden olmaktadır. Bu amaçla sistem tasarımı başlamadan önce ihtiyaçlar kümesinin ne denli tam ve eksiksiz olduğunun kararı verilmelidir. Bu sebeple ihtiyaçların bir yolla ölçülmesi gerekmekte ve ölçüm için çeşitli şablonlardan faydalanılmalıdır. Ancak bu sayede ihtiyaç kümesinin tamlığı konusunda karar vermek mümkün olabilmektedir.

Geliştirilen modelde de bu amaçla kullanılan bir şablondan faydalanılmış ve belirlenen ihtiyaçlar şablonda ki tanımlardan faydalanılarak sınıflandırılmıştır. Sınıflandırma dört farklı seviye içermektedir. İhtiyaçlar bu dört farklı seviyeye ve onların alt seviyelerine göre sınıflandırılarak tamlık ve eksiksizliği teyit edilmiştir.

Aşağıda Tablo 3. 11'de eleman temin sistemi için belirlenen ihtiyaçların dört ana seviyede ki dağılımı görülmektedir. Bu seviyeler; amaç, süreç, görev ve bilişim olarak verilmektedir. Sunum teknikleri aracılığı ile ifade edilen ihtiyaçlar, ihtiyaç numaraları baz alınarak seviyelerine göre verilmiştir. Bu dört seviyede ki ihtiyaçlarında alt seviyeleri mevcuttur.

Tablo 3.11. Eleman temin sistemi için belirlenen ihtiyaçların seviyelerine göre sınıflandırılması

İhtiyaç Seviyeleri	**İhtiyaç Numaraları**
Amaç Seviyeli İhtiyaçlar	1, 3, 7, 8, 10, 11, 12, 13, 14, 15, 17, 18, 19, 26
Süreç Seviyeli İhtiyaçlar	2, 9, 20, 21, 22
Görev seviyeli ihtiyaçlar	4, 5, 6, 16, 23, 24, 25
Bilişim İhtiyaçları	27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38

Eleman temin sistemi ihtiyaçlar kümesini oluşturan 38 ihtiyaç, genel ihtiyaç seviyeleri olan amaç, süreç, görev ve bilişim seviyeli olarak sınıflandırıldığında; bunların 14'ünün amaç, 5'inin süreç, 7'sinin görev ve 12'sinin bilişim seviyeli olduğu görülmüştür.

Tablo 3. 12'de amaç seviyeli, Tablo 3. 13'de süreç seviyeli, Tablo 3. 14'de görev ve Tablo 3. 15'de bilişim seviyeli ihtiyaçların alt seviyelere göre dağılımı verilmiştir.

Tablo 3.12. Eleman temin sistemi için belirlenen amaç seviyeli ihtiyaçlar

Amaç Seviyeli İhtiyaçlar	Belirlenen İhtiyaçlar
Amaç durumlu tanımlama	1 Gerekli pozisyon için en kalifiye elemanın işe alınması
Açık tanımlama	10 Önceliklerin tespit edilmesi (Eleman alımlarının performans kriterleri aracılığı ile ağırlıklandırılarak en uygun kişinin tercihi)
Bilgi tanımlama	26 Firmalar genellikle yılın 2, 3 ayında eleman ararlar ve başvuruların değerlendirilmesi zaman alır
En son değerler ve tercihler	12 İşe başlama gününün belirlenmesi 11 İş teklifi 19 Adayların istek ve tercihleri (Firmanın tercihte bulunduğu kişiden beklentileri kişinin isteklerinden daha fazla olmalıdır)
Düşünceler ve stratejiler	7 Müracaatların incelenerek değerlendirilmesi (Müracaatları kriterlendirerek daha sağlıklı bir tercih yapılabilir)
Nedensel teşhis / tanı	8 Bu türlü bir sistemle her müracaat edenle görüşmeksizin kriterler sayesinde sağlıklı bir değerlendirme yapılacak ve başvurular şahsi değerlendirmeden çıkacak
Bakış açısı	15 Firmanın piyasada ki konumu 17 Firmanın çalışanlara sundukları (O dönemde firmayla ilgi çıkan olumsuz bir haber kalifiye kişilerin müracaat etmemesi gibi bir durum yaratabilir)
Mevcut çevre desteği	3 Yayın organlarına ilan verme 18 Firmanın coğrafi konumu (yerleşimi) (Verilen ilanlar ve internet aracılığı ile on-line müracaatlar artabilir. Firmanın coğrafi konumu olumlu veya olumsuz etki yaratabilir)
Paylaşımcılar	13 (Senaryoda ki aktörler) İşletme içindeki tüm departmanlar ve müracaatı yapan kişiler

Amaç seviyeli ihtiyaçlarında kendi içinde 9 alt seviyesi bulunmaktadır. Amaç seviyeli 14 ihtiyacın alt seviyelere göre dağılımı gerçekleştirebilmek amacıyla alt seviyelerinin tanımlarından faydalanılmıştır. Amaç durumlu tanımlama, açık tanımlama, bilgi tanımlama, düşüncelere ve stratejiler, nedensel teşhis/tanı ve paylaşımcılara yönelik 1'rer ihtiyaç, bakış açısı ve mevcut çevre desteğine yönelik 2'şer ihtiyaç ve en son değerler/tercihlere yönelik ise 3 ihtiyaç belirlenmiştir.

Tablo 3. 13 Eleman temin sistemi için belirlenen süreç seviyeli ihtiyaçlar

Süreç Seviyeli İhtiyaçlar	
	Belirlenen İhtiyaçlar
Süreç Tanımlama	9 Mülakat (Müracaatı yapan kişi sisteme girdiği anda eleman temin sistemi çalışmaya başlar. Kişi mülakata çağrılana kadar bu süreç devam eder)
Süreç Bilgi Tanımları	21 Adayların eğitim durumları ve iş alanına uygunluğu 22 İş deneyimleri (Eğer müracaat eden kişinin özellikleri pozisyon için uygun değilse girdiği bilgiler dahilinde başka uygun bir iş varsa kişiye o iş sistem tarafından önerilebilir)
Zorluklar, Kısıtlar	20 Adayın müracaat formunun eksiksizliği (Eğer başvuru yapanın evrakları eksikse hiç değerlendirilmeye sokulmaz)
Roller ve Sorumluluklar	2 İş sorumluluğunun planlanması (Departmanlarda ki yöneticiler alınacak elemanın iş, görev tanımları ve kriterlerin oluşturulmasından sorumlu olacaklardır)

Süreç seviyeli ihtiyaçlar, kendi içinde dört alt seviye barındırmaktadır. Alt seviyeler olan; süreç tanımlama, zorluklar/kısıtlar ve roller/sorumluluklar 1'rer, süreç bilgi tanımları ise 2 ihtiyaç içermektedir.

Tablo 3. 14 Eleman temin sistemi için belirlenen görev seviyeli ihtiyaçlar

Görev Seviyeli İhtiyaçlar	Belirlenen İhtiyaçlar
Görev Tanımlama	4 Adayların müracaatı başlatması (Müracaatı yapanın ilk düşüncesi müracaat sürecini başlatmaktır. Müracaatı başlatmak ise müracaatı başlat ikonuna 2 kere tıklatmakla gerçekleşir)
Görev Bilgi Tanımları	5 Müracaat başladıktan sonra ki işlemler, Başvuru işlemleri (Kişi müracaat sürecini başlattığı zaman, tam müracaatı gerçekleştirsin veya gerçekleştirmesin tüm kriterlere ait bilgileri doldurması konusunda ona kılavuzluk eder)
Performans Kriteri	(Sisteme girebilmek için 3 kez deneme hakkı olmalıdır) 23 Liderlik becerileri 24 Askerlik 25 İletişim becerileri
Roller ve Sorumluluklar	16 Eğer müracaatı yapan kişiye sistemi kullanmak zor/zahmetli geliyorsa, yardım amacıyla bir yardım masası bulunmalıdır (Bireylere ve departmanlara yapılması gereken işler veya yapılası gereken işlerin seviyeleri bildirilmeli)
Doğrulama	6 Kişileri yönlendirme gereksinimi Müracaatı yapan kimseyi eğer eksik yolladığı evrak veya girilmemiş bilgi varsa sistem yönlendirmeli)

Görev seviyeli ihtiyaçlarda kendi içinde 5 alt seviyeye ayrılmış durumdadır. Bu tanımlardan hareketle sınıflanan ihtiyaçların 1'rer tanesi görev tanımlama, görev bilgi tanımları, roller/sorumluluklar ve doğrulamaya 3 tanesi de performans kriterlerine aittir.

Tablo 3.15. Eleman temin sistemi için belirlenen bilişim seviyeli ihtiyaçlar

Bilişim Seviyeli İhtiyaçlar	Belirlenen İhtiyaçlar
Bilginin Gösterilmesi	32 Ayrıntılı bilgi girişi gereksinimi (Müracaatı yapan firmaya ait bilgileri, firmada müracaatı yapan kişilere ait bilgileri görebilmelidir)
Ara yüz Tasarımı	27 Web sayfası gereksinimi 33 Ara yüz gereksinimi Müracaat eden kişileri karşılaştırabilmek için ekranda bunlara ait bilgileri bir arada görme isteği
Girdiler	31 Veri depolama gereksinimi (Gerçekleşen tüm müracaatlar veya diğer verilerin depolanması)
Depolanan Bilgi	34 Tüm müracaatlar alfabetik olarak veya iş gruplarına göre kayıt altında olmalı
Nesneler ve olaylar	35 Verilerin izlenebilirliği (Sistemden tüm müracaatlara ait bilgiler izlenebilmelidir)
Nesneler ve olaylar arasındaki ilişkiler	28 Karar verme yapısı gereksinimi 29 Kararlar arasında ki uyum gereksinimi 30 Karar destek sistemi gereksinimi (Planlama mühendisliği için gerçekleşen müracaatlar, Üretim planlama departmanının alında gözükmelidir)
Veri Özellikleri	36 Verilerin yapılandırılma gereksinimi (Planlama müh. İçin müracaat edecek kişi End. Müh. veya Sistem Müh.'den mezun olmalıdır)
Geçerlilik Kriteri	37 Yeni bir eleman gereksinimi söz konusu olduğunda buna ait görev tanımları ve sorumlulukları da sisteme girilmelidir
Hesaplamalar	38 Hesaplama gereksinimi(Sistem kriterleri puanlayarak değerlendirme yapacak kişilerin, bunları hesaplamasına izin vermelidir)

Bilişim seviyeli ihtiyaçlar 9 alt seviyede ele alınmıştır. Bilginin gösterilmesi, girdiler, depolanan bilgi, nesneler/olaylar, veri özellikleri geçerlilik kriteri ve hesaplamalar 1'rer ihtiyaç içerirken ara yüz tasarımı 2 ve nesneler/olaylar arasında ki ilişkiler ise 3 ihtiyaç içermektedir.

ROC algoritmasının eleman temin sistem ihtiyaçlarının belirlenmesinde kullanılması

Eleman temin sistemi için belirlenen ihtiyaçlar, ihtiyaç seviyelerine göre sınıflandırıldıktan sonra matrisel formda hazırlanmış ve ROC algoritması uygulanmıştır. ROC algoritması uygulanmasında ki amaç; algoritmanın çalışma mantığı gereği benzer ihtiyaçları bir araya toplayarak gruplar oluşturmak, bu gruplarda ki ihtiyaçlardan hareketle ihtiyaç aileleri oluşturarak her ihtiyaç ailesine uygun teknik kombinasyonunu belirlemektir.

ROC algoritmasının uygulanması aşamasında matrisin sütunlarına ihtiyaç belirleme teknikleri tarafından belirlenen ihtiyaçlar, satırlarına da ihtiyaç belirleme teknikleri yerleştirilir. Teknikler; M1, M2, M3,….Mn, ihtiyaçlar ise; P1, P2, P3,……Pn şeklinde ifade edilir. Örneğin P1 ihtiyacı M1 tekniği tarafından belirlenmişse "1" belirlenmemişse "0" yazılır. Bu şekilde tüm teknikler tarafından belirlenen ihtiyaçlar sırası ile girilir.

ROC algoritması çalıştırılarak, algoritmanın çalışma mantığı gereği birbirine benzer ihtiyaçlar bir araya gelerek birbirini diyagonal şekilde takip eden ihtiyaç grupları oluşturulur. ROC algoritması uygulandıktan sonra, bu algoritmayı değerlendirmede kullanılan performans

kriterlerine ait formüller bu matris çıktılarına uygulanarak, EK [3]'de tanımları ve formülleri verilen;

-Grup verimliliği,

-Grup etkinliği ve

-Gruplama ölçüleri hesaplanmış ve elde edilen sonuçlar değerlendirilerek yorumlanmıştır.

Grup teknolojisinde başarı ölçütleri olarak kullanılan bu notasyonların gerçekleştirilen çalışmaya uygun olup olmadığının değerlendirilmesi amacıyla, öncelikle teknik kombinasyonları 3'lü gruplar halinde ele alınmıştır. Aşağıda ki ROC çıktılarına ait tablolardan da bu durum görülebilmektedir. Her bir kombinasyon içinde üç teknik yer almaktadır. Bu 3'lü kombinasyon sonuçları, başarı ölçütleri yardımıyla değerlendirildikten sonra, elde edilen sonuçlara göre teknikler hem tek tek hem de 2'li, 4'lü ve 5'li kombinasyonlar ilave edilerek değerlendirilmiştir.

Beş tekniğin üçlü birleşimlerine ROC algoritması uygulama çalışması

Bu bölümde beş tekniğin üçlü olarak bir araya gelmesiyle oluşan 10 durum için sunulmuş olan ihtiyaçların programın ara yüzüne girildikten sonraki halleri ve algoritma çalıştırıldıktan sonraki hallerine ait görüntüler tablolaştırılarak sırasıyla verilmiştir. Ayrıca algoritma uygulandıktan sonra ihtiyaçların oluşturduğu gruplar ve bunların başarı ölçütleriyle değerlendirildikten sonra elde edilen sonuçlarda görülmektedir.

Tablo 3.16a'da T1-T2-T3 tekniklerinin birlikte kullanımıyla belirlenen ihtiyaçların ROC algoritması ekranına girilmiş ve Tablo 3.16b'de ROC algoritması uygulanmış durum tablolaştırılarak gösterilmiştir.

Tablo 3.16a T1-T2-T3 tekniklerinin birleşimine ROC algoritması uygulanmamış hal

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38
M1	1	1	1	1	1	1	1	1	1	1	1	1																										
M2	1	1	1											1	1	1	1	1	1																			
M3	1	1	1		1	1	1	1	1		1	1								1	1	1	1	1	1													

Tablo 3.16b T1-T2-T3 tekniklerinin birleşimine ROC algoritması uygulanmış hal

	P1	P2	P3	P5	P6	P7	P8	P9	P11	P12	P4	P10	P20	P21	P22	P23	P24	P25	P14	P15	P16	P17	P18	P19	P13	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38
M1	1	1	1	1	1	1	1	1	1	1	1	1																										
M3	1	1	1	1	1	1	1	1	1	1			1	1	1	1	1	1																				
M2																			1	1	1	1	1	1														

Grup verimliliği : 0.77

Grup etkinliği : 0.30

Gruplama ölçüsü : 0.54

Akış şeması : M1

Çağrıştırıcı bilgi haritası : M2

Karar haritası : M3

Tablo 3.17a'da T1-T2-T4 tekniklerinin birlikte kullanımıyla belirlenen ihtiyaçların ROC algoritması ekranına girilmiş halleri ve Tablo 3.17b'de ROC algoritması uygulanmış halleri tablolaştırılarak verilmiştir.

Tablo 3.17a T1-T2-T4 tekniklerinin birleşimine ROC algoritması uygulanmamış hal

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38
M1	1	1	1	1	1	1	1	1	1	1	1	1																										
M2	1	1	1											1	1	1	1	1	1																			
M4	1	1	1	1			1	1	1	1	1	1	1							1							1											

Tablo 3.17b T1-T2-T4 tekniklerinin birleşimine ROC algoritması uygulanmış hal

	P1	P2	P3	P4	P7	P8	P9	P10	P11	P12	P5	P6	P13	P20	P27	P16	P17	P18	P19	P14	P15	P22	P23	P24	P25	P26	P21	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38
M1	1	1	1	1	1	1	1	1	1	1	1	1																										
M4	1	1	1	1	1	1	1	1	1	1			1	1	1																							
M2	1	1	1													1	1	1	1	1	1																	

Grup verimliliği : 0.76

Grup etkinliği : 0.27

Gruplama ölçüsü : 0.48

Akış şeması : M1

Çağrıştırıcı bilgi haritası : M2

Senaryo tekniği : M4

Tablo 3.18a'da T1-T2-T5 tekniklerinin birlikte kullanımıyla belirlenen ihtiyaçların ROC algoritması ekranına girilmiş ve Tablo 3.18b'de ROC algoritması uygulanmış durum görülmektedir.

Tablo 3.18a T1-T2-T5 tekniklerinin birleşimine ROC algoritması uygulanmamış hal

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38
M1	1	1	1	1	1	1	1	1	1	1	1	1																										
M2	1	1	1											1	1	1	1	1	1																			
M5																										1	1	1	1	1	1	1	1	1	1	1	1	1

Tablo 3.18b T1-T2-T5 tekniklerinin birleşimine ROC algoritması uygulanmış hal

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P14	P15	P16	P17	P18	P19	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P25	P13	P20	P21	P22	P23	P24
M1	1	1	1	1	1	1	1	1	1	1	1	1																										
M2	1	1	1										1	1	1	1	1	1																				
M5																			1	1	1	1	1	1	1	1	1	1	1	1	1							

Grup verimliliği : 0.80

Grup etkinliği : 0.29

Gruplama ölçüsü : 0.60

Akış şeması : M1

Çağrıştırıcı bilgi haritası : M2

Benzerlik diyagramı : M5

Tablo 3.19a'da T1-T3-T4 tekniklerinin birlikte kullanımıyla belirlenen ihtiyaçların ROC algoritması ekranına girilmiş ve Tablo 3.19b'de ROC algoritması uygulanmış durum görülmektedir.

Tablo 3.19a T1-T3-T4 tekniklerinin birleşimine ROC algoritması uygulanmamış hal

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38
M1	1	1	1	1	1	1	1	1	1	1	1	1																										
M3	1	1	1											1	1	1	1	1	1																			
M4	1	1	1	1			1	1	1	1	1	1	1							1							1											

Tablo 3.19b T1-T3-T4 tekniklerinin birleşimine ROC algoritması uygulanmış hal

	P1	P2	P3	P7	P8	P9	P11	P12	P4	P10	P5	P6	P20	P13	P27	P22	P23	P24	P25	P21	P15	P16	P17	P18	P19	P26	P14	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38
M1	1	1	1	1	1	1	1	1	1	1	1	1																										
M4	1	1	1	1	1	1	1	1	1	1			1	1	1																							
M3	1	1	1	1	1	1	1	1			1	1	1			1	1	1	1	1																		

Grup verimliliği : 0.84

Grup etkinliği : 0.35

Gruplama ölçüsü : 0.68

Akış şeması : M1

Karar haritası : M2

Senaryo tekniği : M3

Benzer şekilde ; T1-T3-T5, T1-T4-T5, T2-T3-T4, T2-T3-T5, T2-T4-T5, T3-T4-T5 tekniklerinin birlikte kullanımıyla belirlenen ihtiyaçların ROC algoritması ekranına girilmiş ve sonuçlar değerlendirilmiştir.

Uygulama sonunda elde edilen; Grup verimliliği, Grup etkinliği ve Gruplama ölçüsü değerleri, derlenerek bir tablo oluşturulmuştur. Tablo 3. 20'de Eleman Temin Sistemi için Grup verimliliği, Grup etkinliği ve Gruplama ölçüsü değerleri verilmiştir.

Eleman temin sistemi için ROC algoritması sonuçları değerlendirilmiş, üç başarı ölçütünden elde edilen sonuçlar irdelenmiştir.

Başarı ölçütlerinden elde edilen değerlerin yüksek olması arzu edilen bir durumdur. Amaç tüm ihtiyaçları veya yapılan kabul gereğince asgari ihtiyacı içinde bulunduran teknik kombinasyonlarını belirlemek ve bunlara ait başarı ölçütlerini değerlendirmektir.

Oluşturulan Tablo 3.20'de, M1 yerine T1, M2 yerine T2, M3 yerine T3, M4 yerine T4 ve M5 yerine T5 ifadeleri kullanılmıştır.

Tablo 3. 20 Eleman Temin Sistemi için Grup verimliliği, Grup etkinliği ve Gruplama ölçüsü değerleri

Kombinasyon no	Grup verimliliği	Grup etkinliği	Gruplama ölçüsü
T1-T2- T3	0.77	0.30	0.54
T1- T2- T4	0.76	0.27	0.48
T1- T2- T5	0.80	0.29	0.60
T1- T3- T4	0.84	**0.35**	0.68
T1- T3- T5	0.86	**0.35**	0.73
T1- T4- T5	0.84	0.32	0.67
T2- T3- T4	0.77	0.30	0.51
T2- T3- T5	0.81	0.33	0.63
T2- T4- T5	0.79	0.29	0.59
T3- T4- T5	0.87	**0.35**	0.68

İlk başarı ölçütü grup etkinliği idi; Grup Etkinliği değerleri başarılarına göre sıralandığın da, en yüksek değer 0.35'dur. T1-T3-T4, T1-T3-T5 ve T3-T4-T5 kombinasyonları en yüksek grup etkinliğine sahip olan kombinasyonlardır. ROC algoritması çıktıları değerlendirildiğinde bu kombinasyonların en fazla ihtiyaç barındıran kombinasyonlar olmadığı görülmektedir. En fazla ihtiyacı barındıran T2-T3-T5 kombinasyonunun grup etkinliği değeri 0.33'dür ve 35 ihtiyaç belirlenmiştir. Bu değerde tabloda ki en yüksek değer değildir. Buradan elde edilen sonuçlar değerlendirildiğinde çalışmanın amacına uygun sonuçlar elde edilememiştir.

Eleman temin sistemi için, grup verimliliği ve gruplama ölçüsü değerlerine bakıldığında, yine amaç doğrultusunda bir sonuca ulaşılamamıştır. Kısaca, elde edilen sonuçlar ihtiyaçlar açısından değerlendirildiğinde mantıklı bir açıklama getirilememiştir. Bu nedenle çalışma kapsamında ihtiyaç belirlemeye yönelik başarı ölçütleri geliştirilmiştir.

Eleman temin sistemi için elde edilen ROC çıktılarının yeni başarı ölçütleri ile değerlendirilmesi

Çalışmanın bu aşamasında, elde edilen ROC çıktılarının yeni geliştirilen başarı ölçütleri aracılığı ile değerlendirilmesi için bir önce ki basamakta sadece 3'lü olarak kullanılan teknik kombinasyonlarının, 2'li, 3'lü, 4'lü ve 5'li grupları oluşturularak ROC algoritması uygulanmış, çıkan sonuçlar çalışmanın amacına uygun olarak geliştirilen yeni başarı ölçütleri aracılığı ile değerlendirilerek yorumlanmıştır.

Aşağıda verilen teknikler ile belirlenen ihtiyaçlar ROC algoritmasına göre değerlendirilirken oluşturulan kombinasyon grupları;

-Sadece teknikler: {M1},{M2}, {M3}, {M4} ve {M5},

-Tekniklerin ikili olarak gruplanması: {M1, M2}, {M1, M3}, {M1, M4}, {M1, M5}, {M2, M3}, {M2, M4}, {M2, M5}, {M3, M4}, {M3, M5} ve {M4, M5},

-Tekniklerin üçlü olarak gruplanması: {M1, M2, M3}, {M1, M2, M4}, {M1, M2, M5}, {M1, M3, M4 }, {M1, M3, M5}, {M1, M4, M5}, {M2, M3, M4}, {M2, M3, M5}, {M2, M4, M5} ve {M3, M4, M5},

-Tekniklerin dörtlü olarak gruplanması: {M1, M2, M3, M4}, {M1, M2, M3, M5}, {M1, M2, M4, M5}, {M1, M3, M4, M5}} ve {M2, M3, M4, M5} ve

-Tekniklerin beşli olarak gruplanması: {M1, M2, M3, M4, M5} şeklindedir.

-M1: (T1) Akış şeması,

-M2: (T2) Çağrıştırıcı bilgi haritası,

-M3: (T3) Karar haritası,

-M4: (T4) Senaryo ve

-M5: (T5) Benzeşme diyagramı tekniğini ifade etmektedir.

Elde edilen sonuçları ihtiyaçlar açısından değerlendirdiğimizde mantıklı bir açıklama getirilemediğinden, yeni başarı ölçütleri formüllerine gereksinim vardır. Bu aşamada bu konuya yönelik çalışılma gerçekleştirilmiş ve ihtiyaç belirlemeye yönelik yeni başarı ölçütleri geliştirilmiştir. Bunlar; Tekniklerin İhtiyaç Belirleme Etkinlik Ölçüsü, Tekniklerin İhtiyaç Doğrulama Verimlilik Ölçüsü ve Tekniklerin Ortalama Verimlilik Ölçüsü'dür. Yeni geliştirilen başarı

ölçütlerinden hareketle oluşturulan ihtiyaç aileleri ve teknik kombinasyonları hakkında yorumlar aşağıda verilmiştir.

Sadece tekniklere ROC algoritması uygulama çalışması

Modelde ihtiyaçların sunumu aşamasında beş farklı teknikten faydalanılmış ve bu teknikler T1, T2, T3, T4 ve T5 olarak adlandırılmıştı. Bu bölümde beş tekniğin her biriyle sunulmuş olan ihtiyaçların programın ara yüzüne girildikten sonraki halleri ve algoritma çalıştırıldıktan sonraki hallerine ait ekran çıktıları sırasıyla verilmiştir. Ayrıca algoritma uygulandıktan sonra ihtiyaçların oluşturduğu gruplar ve bunların başarı ölçütleriyle değerlendirildikten sonra elde edilen sonuçlarda görülmektedir.

Tablo 3.21a'da T1 tekniğinin belirlediği ihtiyaçlara ROC algoritması uygulanmamış ve Tablo 3.21b'de ROC algoritması uygulanmış durum görülmektedir.

Tablo 3.21a T1 tekniğine ROC algoritması uygulanmamış hal

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38
M1	1	1	1	1	1	1	1	1	1	1	1	1																										

Tablo 3.21b T1 tekniğine ROC algoritması uygulanmış hal

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38
M1	1	1	1	1	1	1	1	1	1	1	1	1																										

$t_e = 0.32$

$t_v = 0$

$t_{ov} = 0.32$

Benzer şekilde T2, T3, T4, T5, T1-T2, T1-T3, T1-T4, T1-T5, T2-T3, T2-T4, T2-T5, T3-T4, T3-T5 T4-T5, T1-T2-T3, T1-T2-T4, T1-T2-T5, T1-T3-T4, T1-T3-T5, T1-T4-T5, T2-T3-T4, T2-T3-T5, T2-T4-T5, T3-T4-T5, T1-T2-T3-T4, T1-T2-T3-T5, T1-T2-T4-T5, T1-T3-T4-T5, T2-T3-T4-T5 ve T1-T2-T3-T4-T5 tekniklerinin birlikte kullanımıyla belirlenen ihtiyaçların ROC algoritması ekranına girilmiş ve edilen sonuçlar, geliştirilen sistem ihtiyaçları başarı ölçütleri aracılığı ile değerlendirilmiştir. Sistem ihtiyaçları başarı ölçütleri;

- Tekniklerin ihtiyaç belirleme etkinlik ölçüsü (t_e),
- Tekniklerin ihtiyaç doğrulama verimlilik ölçüsü (t_v) ve
- Tekniklerin ortalama verimlilik ölçüsü'dür (t_{ov}).

Tablo 3.22, Tablo 3.23, Tablo 3.24 ve Tablo 3.25'in birinci sütunları teknikler ve teknik kombinasyonlarını, ikinci sütunlar t_e değerlerini, üçüncü sütunlar t_v değerlerini ve dördüncü sütunlar ise t_{ov} değerlerini göstermektedir.

Aşağıda Tablo 3. 22'de Eleman temin sistemi için sadece tekniklerin kullanılmasıyla elde edilen başarı ölçütleri değerleri görülmektedir. Burada model kapsamında geçerli olan 5 tekniğe ait t_e, t_v ve t_{ov} değerleri verilmiştir.

Tablo 3.22 Eleman temin sistemi için sadece teknikler için hesaplanan başarı ölçütleri

Teknik/Teknik Kombinasyonları	t_e	t_v	t_{ov}
T1	0.32	**0**	0.32
T2	0.24	**0**	0.24
T3	**0.42**	**0**	**0.42**
T4	0.34	**0**	0.34
T5	0.34	**0**	0.34

Tablo 3. 22 için en yüksek Tekniklerin ihtiyaç belirleme etkinlik ölçüsü (t_e) değerini alan Karar haritası tekniği (T3) 0.42 ile 16 ihtiyaç belirlenmiştir. En düşük t_e değeri olan 0.24'ü alan Çağrıştırıcı bilgi haritası tekniği (T2) ise 9 ihtiyaç belirlemiştir. Tekniklerin ihtiyaç doğrulama verimlilik ölçüsü (t_v) 5 teknik içinde "0" değerini almıştır. Çünkü teknikler tek tek ele alındığı için çakışan ihtiyaç söz konusu değildir. Tekniklerin ortalama verimlilik ölçüsü (t_{ov}) değerleri incelendiğinde, bu değerlerin Tekniklerin ihtiyaç belirleme etkinlik ölçüsü değerleri ile aynı olduğu görülmektedir. Bunun nedeni ise teknik sayısı 1 olduğu için teknik sayısına bölündüğünde değerlerin değişmemesinden kaynaklanmaktadır.

Tablo 3.23'da Eleman temin sistemi için tekniklerin ikili olarak bir arada kullanılmasıyla elde edilen başarı ölçütleri değerleri görülmektedir. Burada tekniklerin ikili olarak bir arada kullanımıyla elde edilen 10 farklı duruma ait t_e, t_v ve t_{ov} değerleri verilmiştir.

Tablo 3.23 Eleman temin sistemi için tekniklerin ikili kombinasyonu için hesaplanan başarı ölçütleri

Teknik/Teknik Kombinasyonları	t_e	t_v	t_{ov}
T1-T2	0.47	0.07	0.24
T1-T3	0.47	**0.26**	0.24
T1-T4	0.39	**0.26**	0.20
T1-T5	0.65	**0**	0.33
T2-T3	0.52	0.07	0.26
T2-T4	0.50	0.07	0.25
T2-T5	0.57	**0**	0.29
T3-T4	0.52	0.18	0.26
T3-T5	**0.76**	**0**	**0.38**
T4-T5	0.66	0.02	0.33

Tablo 3. 23 için, Tekniklerin ihtiyaç belirleme etkinlik ölçüsü (t_e) değerleri arasında en yüksek değeri alan Karar haritası ve Benzerlik diyagramı (T3-T5) tekniklerinin bir arada kullanımıyla oluşan kombinasyon 0.76 ile 29 ihtiyaç belirlenmiştir. En düşük t_e değeri olan 0.39'u alan Akış diyagramı ve Senaryo tekniği (T1-T4) bir arada kullanılarak 15 ihtiyaç belirlemiştir. Tekniklerin ihtiyaç doğrulama verimlilik ölçüsü (t_v) Akış diyagramı/Karar haritası tekniği ve Akış diyagramı/Senaryo tekniği için en yüksek 0.26 değerini alarak 10 tekrarlı ihtiyaç belirlemiştir. T1-T5, T2-T5 ve T3-T5 teknik kombinasyonları da hiç tekrarlı ihtiyaç içermediği için en düşük olan "0" değerini almıştır. Tekniklerin ortalama verimlilik ölçüsü (t_{ov}) değerleri incelendiğinde, en yüksek 0.38 değerini alan T3-T5

kombinasyonu için ortalama olarak her teknik tarafından ihtiyaçların %38'nin belirlendiği görülmüştür.

Tablo 3. 24'da Eleman temin sistemi için tekniklerin üçlü olarak bir arada kullanılmasıyla elde edilen başarı ölçütleri değerleri görülmektedir. Burada da tekniklerin üçlü olarak bir arada kullanımıyla elde edilen 10 farklı duruma ait t_e, t_v ve t_{ov} değerleri verilmiştir.

Tablo 3.24 Eleman temin sistemi için tekniklerin üçlü kombinasyonu için hesaplanan başarı ölçütleri

Teknik/Teknik Kombinasyonları	t_e	t_v	t_{ov}
T1-T2- T3	0.63	**0.34**	0.21
T1- T2- T4	0.55	**0.34**	0.18
T1- T2- T5	0.81	0.07	0.27
T1- T3- T4	0.52	0.55	0.18
T1- T3- T5	0.81	0.26	0.27
T1- T4- T5	0.68	0.28	0.23
T2- T3- T4	0.68	0.31	0.23
T2- T3- T5	**0.92**	0.07	**0.31**
T2- T4- T5	0.81	0.10	0.27
T3- T4- T5	0.84	0.26	0.28

Tablo 3. 24'da, Tekniklerin ihtiyaç belirleme etkinlik ölçüsü (t_e) değerleri arasında en yüksek değeri alan Çağrıştırıcı bilgi haritası, Karar haritası ve Benzerlik diyagramı tekniklerinin bir arada kullanımı ile oluşan (T2-T3-T5) kombinasyonu 0.92 ile 35 ihtiyaç belirlemiştir. Tekniklerin ihtiyaç doğrulama verimlilik ölçüsü (t_v) için Akış

diyagramı, Çağrıştırıcı bilgi haritası, Karar haritası diyagramı (T1-T2-T3) ve Akış diyagramı, Çağrıştırıcı bilgi haritası, Senaryo tekniklerinin (T1-T2-T4) bir arada kullanımıyla oluşan kombinasyonlar 0.34 ile 10 adet tekrarlı ihtiyaç belirlenmiştir. T2-T3-T5 teknik kombinasyonu ise Tekniklerin ortalama verimlilik ölçüsü (t_{ov}) değerleri için en yüksek 0.31 değerini alarak teknik başına ihtiyaçların %31'inin belirlendiği görülmüştür.

Aşağıda Tablo 3. 25'de Eleman temin sistemi için tekniklerin dörtlü ve beşli olarak bir arada kullanılmasıyla elde edilen başarı ölçütü değerleri görülmektedir. Burada tekniklerin dörtlü olarak bir arada kullanımıyla elde edilen 5 farklı duruma ve beşli olarak bir arada kullanımıyla elde edilen 1 duruma ait t_e, t_v ve t_{ov} değerleri verilmiştir.

Tablo 3.25 Eleman temin sistemi için tekniklerin dörtlü ve beşli kombinasyonu için hesaplanan başarı ölçütleri

Teknik/Teknik Kombinasyonları	t_e	t_v	t_{ov}
T1-T2- T3-T4	0.68	**0.63**	0.17
T1-T2- T3-T5	0.97	0.34	0.24
T1- T2-T4-T5	0.87	0.37	0.22
T1-T3-T4-T5	0.84	0.58	0.21
T2-T3- T4-T5	**1.00**	0.34	**0.25**
T1-T2- T3-T4-T5	**1.00**	**0.66**	**0.20**

Tablo 3. 25'de Tekniklerin ihtiyaç belirleme etkinlik ölçüsü (t_e) değerleri arasında en yüksek değeri alan Çağrıştırıcı bilgi haritası, Karar haritası, Senaryo ve Benzerlik diyagramı tekniklerinin bir arada

kullanımı ile oluşan (T2-T3-T4-T5) ve Akış diyagramı, Çağrıştırıcı bilgi haritası, Karar haritası, Senaryo ve Benzerlik diyagramı tekniklerinin bir arada kullanımı ile oluşan (T1-T2-T3-T4-T5) kombinasyonları 1.00 ile 38 ihtiyacın tamamını belirlemiştir. Tekniklerin ihtiyaç doğrulama verimlilik ölçüsü (t_v) için Akış diyagramı, Çağrıştırıcı bilgi haritası, Karar haritası diyagramı ve Senaryo tekniklerinin (T1-T2-T3-T4) dörtlü olarak bir arada kullanılması ve Akış diyagramı, Çağrıştırıcı bilgi haritası, Karar haritası, Senaryo ve Benzerlik diyagramı tekniklerinin (T1-T2-T3-T4-T5) tekniklerinin beşli olarak bir arada kullanımıyla 0.63 ile 13 ve 0.66 ile 13 adet tekrarlı ihtiyaç belirlenmiştir. T2-T3-T5-T5 teknik kombinasyonu için Tekniklerin ortalama verimlilik ölçüsü (t_{ov}) değeri en yüksek 0.25 değerini alarak teknik başına ihtiyaçların %25'ini, T1-T2-T3-T4-T5 teknik kombinasyonu için ise 0.20 ile teknik başına ihtiyaçların %20'sini,belirlendiği görülmüştür.

Sistem ihtiyaçlarının belirlemede endüktif öğrenme uygulaması

Bu bölümde modelin ikinci aşamasında, modelin örnek bağımlı olmaktan çıkarılarak genelleştirilmesi ve ihtiyaç sınıflarının dudurma kuralları gibi kullanılarak sistem ihtiyaçlarını belirlemede karşılaşılan problemlerin aşabilmesi amacıyla endüktif öğrenme yaklaşımı ve RULES-3 algoritmasından hareketle kurallar üretilerek çözüm sağlanmıştır.

Aşağıda ise endüktif öğrenme uygulama aşamaları gerçekleştirilmiştir;

- Örnekler belirlenmiş,
- Örneklerin karekteristikleri temsil değerleri hesaplanmış,
- Temsil değerlerinin dahil olduğu sınıf aralıkları belirlenmiş,
- Karekteristikler ağırlıklandırılmış,
- Örneklerin dahil olduğu sınıf değerleri belirlenmiş,
- Hesaplanarak belirlenen sayısal değerlere karşılık gelen dilsel değişkenler atanmış,
- Bu aşamaların sonucunda ise RULES-3 algoritmasının çalıştırılabilmesi için eğitim seti oluşturulmuştur.
- RULES-3 algoritması yardımıyla ise sistem ihtiyaçlarını belirleme de çözüm üretmesi planlanan kural çıkarması işlemi gerçekleştirilmiştir.

Tüm bu aşamalar gerçekleştirildikten sonra, Tablo 3.26'da görülen eğitim seti oluşturulmuştur. Eğitim setini oluşturan örnekler, sunum amacıyla kullanılan teknik/teknik kombinasyonlarını, karekteristiklerin altında ki sayısal değerler örneklere ait karekteristiklerin karşılanma miktarlarını, en son sütun olan sınıf aralığı sütunu ise örneklerin dahil olduğu sayısal değerleri ve dilsel ifadeleri vermektedir. Eğer bir eğitim setinin girdileri örnekler ve bu örneklere ait sayısal/dilsel ifadeler olarak kabul edilirse, çıktıları da kurallar olarak düşünülmelidir. Kurallar ise her bir örneği temsil yeteneğine sahiplik olarak değerlendirilebilir.

Tablo 3.26 Eleman temin süreci için hazırlanan eğitim seti

Örnekler Teknikler/ Kombinasyonlar	1. Karekteristik Amaç seviyeli (W_1: 0.36)		2. Karekteristik Süreç seviyeli (W_2: 0.14)		3. Karekteristik Görev seviyeli (W_3: 0.18)		4. Karekteristik Bilişim seviyeli (W_4: 0.32)		Sınıf Aralığı	
	Sayısal Değer	Dilsel İfade	Sayısal Değer	Dilsel İfade	Sayısal Değer	Dilsel İfade	Sayısal Değer	Dilsel İfade	Sayısal Değer	Dilsel İfade
T1	0,50	**N**	0,40	**D**	0,43	**N**	0	**CD**	0,31	**HK**
T2	0,50	**N**	0,20	**CD**	0,14	**CD**	0	**CD**	0,23	**HK**
T3	0,50	**N**	1,00	**CY**	0,71	**CY**	0	**CD**	0,45	**AK**
T4	0,57	**N**	0,40	**D**	0,14	**CD**	0	**CD**	0,29	**HK**
T5	0,07	**CD**	0	**CD**	0	**CD**	1,00	**CY**	0,35	**HK**
T1-T2	0,86	**CY**	0,40	**D**	0,57	**Y**	0	**CD**	0,47	**AK**
T1-T3	0,50	**N**	1,00	**CY**	0,86	**CY**	0	**CD**	0,47	**AK**
T1-T4	0,50	**N**	0,60	**N**	0,43	**N**	0,08	**CD**	0,37	**AK**
T1-T5	0,57	**N**	0,40	**D**	0,43	**N**	1,00	**CY**	0,66	**KK**
T2-T3	0,86	**CY**	1,00	**CY**	0,86	**CY**	0	**CD**	0,60	**AK**
T2-T4	0,93	**CY**	0,40	**D**	0,29	**D**	0	**CD**	0,44	**AK**
T2-T5	0,57	**N**	0,20	**CD**	0,14	**CD**	1,00	**CY**	0,58	**KK**
T3-T4	0,57	**N**	1,00	**CY**	0,86	**CY**	0	**CD**	0,50	**AK**
T3-T5	0,57	**N**	1,00	**CY**	0,71	**CY**	1,00	**CY**	0,79	**CK**
T4-T5	0,64	**Y**	0,40	**D**	0,14	**CD**	1,00	**CY**	0,63	**KK**
T1-T2- T3	0,86	**CY**	1,00	**CY**	1,00	**CY**	0	**CD**	0,63	**KK**
T1- T2- T4	0,93	**CY**	0,40	**D**	0,57	**Y**	0	**CD**	0,49	**AK**
T1- T2- T5	0,93	**CY**	0,40	**D**	0,57	**Y**	1,00	**CY**	0,81	**CK**
T1- T3- T4	0,57	**N**	1,00	**CY**	0,86	**CY**	1,00	**CY**	0,82	**CK**
T1- T3- T5	0,57	**N**	1,00	**CY**	0,86	**CY**	1,00	**CY**	0,82	**CK**
T1- T4- T5	0,57	**N**	0,60	**N**	0,43	**N**	1,00	**CY**	0,69	**KK**
T2- T3- T4	0,93	**CY**	1,00	**CY**	1,00	**CY**	0	**CD**	0,65	**KK**
T2- T3- T5	0,93	**CY**	1,00	**CY**	0,86	**CY**	1,00	**CY**	0,95	**TK**
T2- T4- T5	1,00	**CY**	0,40	**D**	0,29	**D**	1,00	**CY**	0,79	**CK**
T3- T4- T5	0,64	**Y**	1,00	**CY**	0,86	**CY**	1,00	**CY**	0,85	**CK**
T1-T2- T3-T4	0,93	**CY**	1,00	**CY**	1,00	**CY**	0	**CD**	0,65	**KK**
T1-T2- T3-T5	0,93	**CY**	1,00	**CY**	1,00	**CY**	1,00	**CY**	0,97	**TK**
T1-T2- T4-T5	1,00	**CY**	0,40	**D**	0,57	**Y**	1,00	**CY**	0,84	**CK**
T1-T3-T4-T5	0,64	**Y**	1,00	**CY**	1,00	**CY**	1,00	**CY**	0,87	**CK**
T2- T3-T4-T5	1,00	**CY**	1,00	**CY**	1,00	**CY**	1,00	**CY**	1,00	**TK**
T1-T2- T3-T4-T5	1,00	**CY**	1,00	**CY**	1,00	**CY**	1,00	**CY**	1,00	**TK**

Aşağıda Tablo 3.27'de, eleman temin sistemi için Tablo 3.26'da oluşturulan eğitim setinden hareketle RULES-3 tarafından üretilmiş olan 19 kural verilmiştir. Bu 19 kural yukarıda ki eğitim setinde verilmekte olan 31 örneğin tümünü karşılamaktadır. Bu oluşturulan eğitim seti ve buradan üretilen kurallar yardımıyla sonuç çıkarma temelli modelleme/endüktif öğrenme sayesinde özel bir örnekten hareketle genele doğru bir yaklaşım benimsenerek diğer sistemler içinde sağlıklı sonuçlar verebilecek bir model geliştirilmiş ve sonuç olarak modelin genelleştirilmesi sağlanmıştır.

Tablo 3.27 Eğitim setinden üretilen kurallar seti

Kural 1	Eğer; amac N ve surec D ve gorev N ve bilisim ÇD ise Sınıf HK
Kural 2	Eğer; amac N ve surec ÇD ve gorev ÇD ve bilisim ÇD ise Sınıf HK
Kural 3	Eğer; amac N ve surec D ve gorev ÇD ve bilisim ÇD ise Sınıf HK
Kural 4	Eğer; amac ÇD ve surec ÇD ve gorev ÇD ve bilisim ÇY ise Sınıf HK
Kural 5	Eğer; amac N ve surec N ve gorev N ve bilisim ÇD ise Sınıf AK
Kural 6	Eğer; amac N ve surec D ve gorev N ve bilisim ÇY ise Sınıf BK
Kural 7	Eğer; amac ÇY ve surec ÇD ve gorev ÇY ve bilisim ÇD ise Sınıf AK
Kural 8	Eğer; amac ÇY ve surec D ve gorev D ve bilisim ÇD ise Sınıf AK
Kural 9	Eğer; amac N ve surec ÇD ve gorev ÇD ve bilisim ÇY ise Sınıf BK
Kural 10	Eğer; amac N ve surec ÇY ve gorev ÇY ve bilisim ÇD ise Sınıf AK
Kural 11	Eğer; amac Y ve surec D ve gorev ÇD ve bilisim ÇY ise Sınıf BK
Kural 12	Eğer; amac ÇY ve surec D ve gorev Y ve bilisim ÇD ise Sınıf AK
Kural 13	Eğer; amac N ve surec ÇY ve gorev ÇY ve bilisim ÇY ise Sınıf ÇK
Kural 14	Eğer; amac N ve surec N ve gorev N ve bilisim ÇY ise Sınıf BK
Kural 15	Eğer; amac ÇY ve surec D ve gorev D ve bilisim ÇY ise Sınıf ÇK
Kural 16	Eğer; amac ÇY ve surec ÇY ve gorev ÇY ve bilisim ÇD ise Sınıf BK
Kural 17	Eğer; amac ÇY ve surec D ve gorev Y ve bilisim ÇY ise Sınıf ÇK
Kural 18	Eğer; amac Y ve surec ÇY ve gorev ÇY ve bilisim ÇY ise Sınıf ÇK
Kural 19	Eğer; amac ÇY ve surec ÇY ve gorev ÇY ve bilisim ÇY ise Sınıf TK

Üretilmiş olan her kural bir veya birkaç örneği temsil etmektedir. Bu durumun bir örnekle açıklanması gerekirse; “Kural 1, Eğer; amaç seviyeli ihtiyaçlar N ve süreç seviyeli ihtiyaçlar D ve görev seviyeli ihtiyaçlar N ve bilişim seviyeli ihtiyaçlar ÇD ise örneğin dahil olduğu Sınıf HK” Tablo 3.26’da ki 1 nolu örnek olan T1 tekniğinin sahip olduğu durumu ifade etmektedir.

Modelin üçüncü aşaması, üretilen kuralların bundan sonra geliştirilecek yeni sistemler için temel teşkil etmesidir. Kısaca açıklanması gerekirse bu kurallar sayesinde hangi teknik/teknik kombinasyonlarının hangi seviyedeki ihtiyaçları ne oranda karşıladığı veya karşılayacağı ile ilgili genel bir karar verme yapısı sağlanmış olmaktadır.

Bu bölümde oluşturulan eğitim seti, ROC algoritması çıktıları ve kurallardan hareketle uygulama değerlendirilmiştir. Örneğin; T1-T2-T3-T5 tekniklerinden faydalanılması durumunda (Akış diyagramı, Çağrıştırıcı bilgi haritası, Karar haritası ve Benzerlik diyagramı) ihtiyaçların tümünün karşılanabildiği görülmektedir. T2-T3-T5 teknikleri aracılığı ile de ihtiyaçların %95’i karşılanmakta fakat bu değer kabul gereği ihtiyaçların tümü karşılanmış değer grubu içinde yer aldığı için tümünün karşılandığı kabul edilmektedir.

Tablolar halinde verilen t_e, t_v ve t_{ov} değerleri inceldiğinde gerçekten en yüksek t_e değerinin aynı zamanda en fazla/tüm ihtiyacı karşılayan teknik/teknik kombinasyonuna karşılık geldiği. Örneğin en düşük t_e değerini alan T2 tekniği gerçekten 9 ihtiyaçla en az ihtiyaç belirleyen

teknik, en yüksek t_e değerini alan T3 tekniği de 16 ihtiyaçla en fazla ihtiyaç belirleyen teknik durumundadır.

İkili teknik kombinasyonlarında en düşük t_e değerini alan T1-T4 kombinasyonu 15 ihtiyaçla en az ihtiyaç belirleyen ikili kombinasyon, en yüksek t_e değerini alan T3-T5 kombinasyonu da 29 ihtiyaçla en fazla ihtiyaç belirleyen ikili kombinasyon durumundadır. Ayrıca bu çalışmada üçlü teknik kombinasyonlarından T2- T3- T5 aracılığı ile ihtiyaçların 35 tanesi belirlenmekte ve dolayısıyla bu grupta en yüksek t_e değeri olan 0.92'ye karşılık gelmektedir.

Dörtlü kombinasyonlarda T2-T3- T4-T5 en yüksek t_e değeri olan 1.0'la ihtiyaçların tamamını kısacası 38 ihtiyacın tümünü karşılamaktadır. t_v değerleri incelendiğinde ise; en yüksek t_v değerini alan T1-T3, T1-T3-T4, T1-T2-T3-T4 ve T1-T2-T3-T4-T5 kombinasyonlarının en fazla tekrarlı ihtiyaçları belirleyen kombinasyonlar olduğu görülmektedir. Tüm teknik ve teknik kombinasyonları için t_{ov} değerleri incelediğinde de en yüksek değeri alan T3, T3-T5, T2-T3-T5, T2-T3-T4-T5 ve T1-T2-T3-T4-T5 kombinasyonlarının en yüksek teknik başına ihtiyaçların ortalama belirlenme %'lerini belirleyen kombinasyonlar olduğu görülmüştür.

BÖLÜM 4.

MODELİN DEĞERLENDİRİLMESİ/SONUÇ

Modelde teknik/teknik kombinasyonlarının ihtiyaçları karşılanma miktarlarını değerlendirmede kullanılan başarı ölçütlerinden t_e ve eğitim setinin sınıf aralığı değerleri tablolar yardımı ile karşılaştırılarak elde edilen sonuçlar değerlendirilmiştir. Başarı ölçütlerinden t_e değerlerinin eğitim setinde ki sınıf aralığı değerleri ile karşılaştırmada kolaylık sağlaması ve aynı türden ifade edilebilmesi amacıyla t_e'ye ait sayısal değerler Tablo 3.5'de ki sınıf değerleri için belirlenmiş tanım aralığı tablosundan hareketle dilsel olarak ifade edilmiştir.

Tablo 4.28'de eleman temin sistemi için sadece tekniklerden faydalanılarak elde edilen t_e değerleri ve bunlara ait dilsel ifadeler verilmiştir.

Tablo 4.28 Eleman Temin sistemi için "Tekniklerin ihtiyaç belirleme etkinlik ölçüsü" değerleri

Örnekler **Teknikler/Teknik Kombinasyonları**	**Sınıf Aralığı**	
	t_e	**Dilsel İfade**
T1	0.32	**HK**
T2	0.24	**HK**
T3	0.42	**AK**
T4	0.34	**HK**
T5	0.34	**HK**

Tablo 4.29'da eleman temin sistemi için sadece tekniklerden faydalanılarak elde edilen sınıf aralığına ait sayısal değerler ve dilsel ifadeler verilmiştir.

Tablo 4.29 Eleman Temin sistemi için hazırlanan eğitim setinin sadece tekniklere ait Sınıf Aralığı değerleri

Örnekler **Teknikler/Teknik Kombinasyonları**	**Sınıf Aralığı**	
	Sayısal Değer	**Dilsel İfade**
T1	0,31	**HK**
T2	0,23	**HK**
T3	0,45	**AK**
T4	0,29	**HK**
T5	0,35	**HK**

Tablo 4.28 ve Tablo 4.29 dilsel ifadeler temel alınarak karşılaştırıldığında 5 tekniğinde aynı sınıf aralığı değerlerine sahip olduğu, ROC algoritması çıktısı olan t_e değerleri ve RULES-3 algoritması girdisi olan eğitim setine ait sınıf aralığı değerlerinin tümüyle aynı olduğu ve her iki algoritmanın bu konuda aynı sonuçları verdiği görülmektedir.

Tablo 4.30'da eleman temin sistemi için tekniklerin ikili kombinasyonlarından faydalanılarak elde edilen t_e değerleri ve bunlara ait dilsel ifadeler verilmiştir.

Tablo 4.30 Eleman Temin sistemi için tekniklerin ikili kombinasyonuna ait "Tekniklerin ihtiyaç belirleme etkinlik ölçüsü" değerleri

Örnekler **Teknikler/Teknik Kombinasyonları**	**Sınıf Aralığı**	
	t_e	**Dilsel İfade**
T1-T2	0.47	**AK**
T1-T3	0.47	**AK**
T1-T4	0.39	**AK**
T1-T5	0.65	**KK**
T2-T3	0.52	**KK**
T2-T4	0.50	**KK**
T2-T5	0.57	**KK**
T3-T4	0.52	**KK**
T3-T5	0.76	**ÇK**
T4-T5	0.66	**KK**

Tablo 4.31'de eleman temin sistemi için tekniklerin ikili kombinasyonlarından faydalanılarak elde edilen sınıf aralığına ait sayısal değerler ve dilsel ifadeler verilmiştir.

Tablo 4.31 Eleman Temin sisteminde tekniklerin ikili kombinasyonu için hazırlanan eğitim setine ait Sınıf Aralığı değerleri

Örnekler Teknikler/Teknik Kombinasyonları	Sınıf Aralığı	
	Sayısal Değer	**Dilsel İfade**
T1-T2	0,47	**AK**
T1-T3	0,47	**AK**
T1-T4	0,37	**AK**
T1-T5	0,66	**KK**
T2-T3	0,60	**KK**
T2-T4	0,44	**AK**
T2-T5	0,58	**KK**
T3-T4	0,50	**AK**
T3-T5	0,79	**CK**
T4-T5	0,63	**KK**

Tablo 4.30 ve Tablo 4.31 dilsel ifadeler temel alınarak karşılaştırıldığında T2-T4 ve T3-T4 kombinasyonları dışında kalan tüm ikili kombinasyonların aynı sınıf aralığı değerlerine sahip olduğu, ROC algoritması çıktısı olan t_e değerleri ve RULES-3 algoritması girdisi olan eğitim setine ait sınıf aralığı değerlerinin aynı olduğu ve her iki algoritmanın bu kombinasyonlar için aynı sonuçları verdiği görülmüştür.

Tablo 4.32'de eleman temin sistemi için tekniklerin üçlü kombinasyonlarından faydalanılarak elde edilen t_e değerleri ve bunlara ait dilsel ifadeler verilmiştir.

Tablo 4.32 Eleman Temin sistemi için tekniklerin üçlü kombinasyonuna ait "Tekniklerin ihtiyaç belirleme etkinlik ölçüsü" değerleri

Örnekler Teknikler/Teknik Kombinasyonları	Sınıf Aralığı	
	t_e	Dilsel İfade
T1-T2-T3	0.63	**KK**
T1-T2-T4	0.55	**KK**
T1-T2-T5	0.81	**ÇK**
T1-T3-T4	0.52	**KK**
T1-T3-T5	0.81	**ÇK**
T1-T4-T5	0.68	**KK**
T2-T3-T4	0.68	**KK**
T2-T3-T5	0.92	**TK**
T2-T4-T5	0.81	**ÇK**
T3-T4-T5	0.84	**ÇK**

Tablo 4.33'da eleman temin sistemi için tekniklerin üçlü kombinasyonlarından faydalanılarak elde edilen sınıf aralığına ait sayısal değerler ve dilsel ifadeler verilmiştir.

Tablo 4.33 Eleman Temin sisteminde tekniklerin üçlü kombinasyonu için hazırlanan eğitim setine ait Sınıf Aralığı değerleri

Örnekler Teknikler/Teknik Kombinasyonları	Sınıf Aralığı	
	Sayısal Değer	Dilsel İfade
T1-T2-T3	0,63	**KK**
T1-T2-T4	0,49	**AK**
T1-T2-T5	0,81	**CK**
T1-T3-T4	0,82	**CK**
T1-T3-T5	0,82	**CK**
T1-T4-T5	0,69	**KK**
T2-T3-T4	0,65	**KK**
T2-T3-T5	0,95	**TK**
T2-T4-T5	0,79	**CK**
T3-T4-T5	0,85	**CK**

Tablo 4.32 ve Tablo 4.33'de ki dilsel ifadeler kıyaslandığında T1-T2-T4 ve T1-T3-T4 kombinasyonları hariç tüm üçlü teknik kombinasyonlarına ait dilsel ifadelerin aynı sınıf aralığına karşılık geldiği görülmektedir. Bu durumdan hareketle, ROC algoritması çıktısı olan t_e değerleri ve RULES-3 algoritması girdisi olan eğitim setine ait sınıf aralığı değerlerinin aynı dilsel değişkenlere sahip olduğu ve her iki algoritmanın bu kombinasyonlar için aynı sonuçları verdiği söylenebilir.

Tablo 4.34'de eleman temin sistemi için tekniklerin dörtlü ve beşli kombinasyonlarından faydalanılarak elde edilen t_e değerleri ve bunlara ait dilsel ifadeler verilmiştir.

Tablo 4.34 Eleman Temin sistemi için tekniklerin dörtlü ve beşli kombinasyonlarına ait "Tekniklerin ihtiyaç belirleme etkinlik ölçüsü" değerleri

Örnekler **Teknikler/Teknik Kombinasyonları**	**Sınıf Aralığı**	
	t_e	**Dilsel İfade**
T1-T2-T3-T4	0.68	**KK**
T1-T2-T3-T5	0.97	**TK**
T1-T2-T4-T5	0.87	**ÇK**
T1-T3-T4-T5	0.84	**ÇK**
T2-T3-T4-T5	1.00	**TK**
T1-T2-T3-T4-T5	1.00	**TK**

Tablo 4.35'de eleman temin sistemi için tekniklerin dörtlü ve beşli kombinasyonlarından faydalanılarak elde edilen sınıf aralığına ait sayısal değerler ve dilsel ifadeler verilmiştir.

Tablo 4.35 Eleman Temin sisteminde tekniklerin dörtlü ve beşli kombinasyonları için hazırlanan eğitim setine ait Sınıf Aralığı değerleri

Örnekler **Teknikler/Teknik Kombinasyonları**	**Sınıf Aralığı**	
	Sayısal Değer	**Dilsel İfade**
T1-T2-T3-T4	0,65	**KK**
T1-T2-T3-T5	0,97	**TK**
T1-T2-T4-T5	0,84	**CK**
T1-T3-T4-T5	0,87	**CK**
T2-T3-T4-T5	1,00	**TK**
T1-T2-T3-T4-T5	1,00	**TK**

Tablo 4.34 ve Tablo 4.35'de ki tüm teknik kombinasyonları için dilsel ifadelerin tümü aynıdır. Bu nedenle, ROC algoritması çıktısı olan t_e değerleri ve RULES-3 algoritması girdisi olan eğitim setine ait sınıf aralığı değerlerinin tümüyle aynı olduğu ve her iki algoritmanın bu konuda aynı sonuçları verdiği söylenilebilir.

Tablo 4.36, Tablo 4.37, Tablo 4.38, Tablo 4.39 ve Tablo 4.40'da ROC algoritması çıktıları ve RULES-3 çıktıları sınıflar bazında karşılaştırmalı olarak görülebilmektedir.

Tablo 4.36'da HK sınıfı için teknik/teknik kombinasyonlarının Rules-3 ve ROC'a ait çıktılarının karşılaştırmalı olarak görülmesi sağlanmıştır.

Tablo 4.36 HK sınıfında teknik/teknik kombinasyonlarının Rules-3 ve ROC çıktıları için dağılımı

HK Sınıfı	
Rules-3	**ROC**
T1	T1
T2	T2
T4	T4
T5	T5

Tablo 4.36'da da görüldüğü üzere modelde tanımlı 4 tekniğin gerek ROC açısından gerekse RULES-3 açısından elde edilen değerler HK sınıfı için değerlendirildiğinde tümünün aynı sınıf aralığına düştüğü görülmüştür. Her iki grupta ki değerler farklı bir sınıf aralığı içinde yer almamıştır. Tabloda ki 4 teknik bazında her iki algoritmanın aynı sonuçları ürettiğini söylememiz mümkün olmaktadır.

Tablo 4.37'de AK sınıfı için teknik/teknik kombinasyonlarının Rules-3 ve ROC'a ait çıktılarının karşılaştırmalı olarak görülmesi sağlanmıştır.

Tablo 4.37 AK sınıfında teknik/teknik kombinasyonlarının Rules-3 ve ROC çıktıları için dağılımı

AK Sınıfı	
Rules-3	**ROC**
T3	T3
T1-T2	T1-T2
T1-T3	T1-T3
T1-T4	T1-T4
T2-T4	
T3-T4	
T1-T2-T4	

Tablo 4.37'de görüldüğü gibi AK sınıfına karşılık gelen RULES-3 ve ROC çıktıları değerlendirildiğinde RULES-3'de 7 teknik kombinasyonu yer alırken ROC'da 4 teknik kombinasyonu bulunmaktadır. Bunların 4'ü birbirine denk olan kombinasyonlarken 3'ü farklılık göstermektedir.

Tablo 4.38'de KK sınıfı için teknik/teknik kombinasyonlarının Rules-3 ve ROC'a ait çıktılarının karşılaştırmalı olarak görülmesi sağlanmıştır.

Tablo 4.38 KK sınıfında teknik/teknik kombinasyonlarının Rules-3 ve ROC çıktıları için dağılımı

KK Sınıfı	
Rules-3	**ROC**
T1-T5	T1-T5
T2-T3	T2-T3
T2-T5	T2-T5
T4-T5	T4-T5
T1-2-3	T1-T2-T3
T1-T4-T5	T1-T4-T5
T2-T3-T4	T2-T3-T4
T1-T2-T3-T4	T1-T2-T3-T4
	T2-T4
	T3-T4
	T1-T2-T4
	T1-T3-T4

Tablo 4.38'de görüldüğü gibi KK sınıfına karşılık gelen RULES-3 ve ROC çıktıları değerlendirildiğinde RULES-3'de 8 teknik kombinasyonu yer alırken ROC'da 12 teknik kombinasyonu bulunmaktadır. Bunların 8'i birbirine denk olan kombinasyonlarken 4'dü farklılık göstermektedir.

Tablo 4.39'de ÇK sınıfı için teknik/teknik kombinasyonlarının Rules-3 ve ROC'a ait çıktılarının karşılaştırmalı olarak görülmesi sağlanmıştır.

Tablo 4.39 ÇK sınıfında teknik/teknik kombinasyonlarının Rules-3 ve ROC çıktıları için dağılımı

ÇK Sınıfı	
Rules-3	**ROC**
T3-T5	T3-T5
T1-T2-T5	T1-T2-T5
T1-T3-T5	T1-T3-T5
T2-T4-T5	T2-T4-T5
T3-T4-T5	T3-T4-T5
T1-T2-T4-T5	T1-T2-T4-T5
T1-T3-T4-T5	T1-T3-T4-T5
T1-T3-T4	

Tablo 4.39'de görüldüğü gibi ÇK sınıfına karşılık gelen RULES-3 ve ROC çıktıları değerlendirildiğinde RULES-3'de 8 teknik kombinasyonu yer alırken ROC'da 7 teknik kombinasyonu bulunmaktadır. Bunların 8'i birbirine denk olan kombinasyonlarken 1'i farklılık göstermektedir.

Tablo 4.40'da TK sınıfı için teknik/teknik kombinasyonlarının Rules-3 ve ROC'a ait çıktılarının karşılaştırmalı olarak görülmesi sağlanmıştır.

Tablo 4.40 TK sınıfında teknik/teknik kombinasyonlarının Rules-3 ve ROC çıktıları için dağılımı

TK Sınıfı	
Rules-3	**ROC**
T2-T3-T5	T2-T3-T5
T1-T2-T3-T5	T1-T2-T3-T5
T2-T3-T4-T5	T2-T3-T4-T5
T1-T2-T3-T4-T5	T1-T2-T3-T4-T5

Tablo 4.40'da ÇK sınıfına karşılık gelen teknik/RULES-3 ve ROC için elde edilen değerler TK sınıfı için değerlendirildiğinde tümünün aynı sınıf aralığına düştüğü görülmüştür. Her iki grupta ki değerler farklı bir sınıf aralığı içinde yer almamıştır. Teknik bazında her iki algoritmanın aynı sonuçları ürettiğini söylememiz mümkün olmaktadır.

Tablo 4.36, Tablo 4.37, Tablo 4.38, Tablo 4.39 ve Tablo 4.40'da teknik/teknik kombinasyonlarının sahip olduğu sınıflar temel alarak değerlendirdiğimizde 31 örnek arasında 4 farklı durum gözlenmiştir. 31 örneği %100 olarak kabul ettiğimizde 4 farklı durum %13'e karşılık gelmektedir. Bu durumda her iki algoritmanın %87 oranında birbiriyle aynı sonuçları verdiğini söyleyebiliriz.

Ayrıca bu tablolar yardımıyla hangi teknik/teknik kombinasyonlarının ne tür sistem ihtiyaçlarına veya ihtiyaç sınıflarına ne derecede cevap verebildiği de kolay bir şekilde görülmektedir. Kısaca geliştirilen modelin çıktıları; Amaç, Süreç, Görev ve Bilişim seviyeli ihtiyaçlardır.

Sonuç olarak üretilen kurallara göre; amaç, süreç, görev ve bilişim seviyeli ihtiyaçların Çoğu karşılanmışsa bu Tümü Karşılanmışa karşılık gelmekte ve ihtiyaçları belirlemede T2-T3-T5 kombinasyonu oldukça etkin olmaktadır. Bu durum her iki algoritmanın kullanılması durumunda da aynı sonucu vermektedir. O zaman her iki algoritmanın aynı amaca hizmet ettikleri söylenebilir.

KAYNAKLAR

[1] LUCAS, H. C., BAROUDİ, J., "The role of information technology in organization design," Journal of Management Information Systems, Vol. 10, no. 4:, 1994, pp. 9-23.

[2] OZ, E., "MANAGEMENT INFORMATION SYSTEMS", Sixth Edition, The Pennsylvania State University, Great Valley, 2009, ISBN-13: 978-1-4239-0178-5, ISBN-10: 1-4239-0178-9

[3] JAMES A. S., Information Tehnology in Business, Principles, Practis and Opportunities, Prentice Hall, II edition, 1998.

[4] LAUDON K. C., LAUDON J. P., Management Information Systems Managing the Digital Firm, Thirteenth Edition, Pearson Education Limited 2014, http://dinus.ac.id/repository/docs/ajar/Kenneth_C.Laudon,Jane_P_.Laudon_-_Management_Information_Sysrem_13th_Edition_.pdf

[5] FLYNN, D.J., "Information Systems Requirements: Determination and Analysis", McGraw-Hill Companies, 1998

[6] LAZAR, J., HANST, E., BUCHWALTER, J., PREECE, J., "Collecting User Requirements in a Virtual Population, WebNet Journal: Internet Technologies, Applications and Issues, Vol. 2 No. 4, pp. 20-27, 2000

[7] KO, D.G., "Information Requirements Analysis and Multiple Knowledge Eticitation Techniques: Experience with the Pricing Scenario System", Proceedings of the 32nd Havai International Conference on System Sciences, 1999

[8] SOMMERVILLE, I., SAWYER, P., VILLER, S., "Viewpoints for Requirements Elicitation: a practical approach", Third International Conference on Requirements Engineering, IEEE Computer Society Press, pp. 74-81, 1998

[9] HANİSCH, J., THANASANKİT, T., CORBİTT, B., "Exploring the cultural and social impacts on the requirements engineering processes – highlighting some problems challenging virtual team relationships with clients", Journal of Systems and Information Technology, Vol. 5, No.2, pp. 1-19, 2001

[10] POLVI, M., "The Role Of Requirements Engineerıng Professionals In The Requirements Elicitation Process", Mini master's thesis in Information Systems, 10.5.2005

[11] SAGE, A.P., "System Engineering", Publisher: Wiley-IEEE, Publication, ISBN: 0471536393, Sate: July 28 1992

[12DARKE, P., SHANKS, G., "User viewpoint modelling: understanding and representing user viewpoints during requirements definition", *Information Systems Journal, Vol.* 7, pp. 213-239, 1997.

[13] LAZAR, J., HANST, E., BUCHWALTER, J., PREECE, J., "Collecting User Requirements in a Virtual Population, WebNet Journal: Internet Technologies, Applications and Issues, Vol. 2 No. 4, pp. 20-27, 2000.

[14] BROWNE, G.J., PITTS, M.G., "Stopping Rule Use During Information Search in Design Problems".

[15] BROWNE, G.J., ROGICH, M.B., "An empirical investigation of user requirements elicitation: comparing the effectiveness of prompting techniques", Journal of Management Information Systems, Vol. 17 No 4, pp. 223-249, Spring 2001.

[16] WATSON, H.J., FROLİCK, M.N., "Determining information requirements for an EIS", MIS, Quarterly, Vol. 17, No. 3, pp. 255-269, 1993.
[17] JANZ, B.D., FROLİCK, M.N., WETHERBE, J.C., "Human Perception: A Challenge to Organizational Process Optimization".
[18] DAVIS, G.B., "Strategies for information requirements determination", IBM Systems Journal, Vol. 21, No. 1, 1982.
[19] BYRD, T.A., COSSİCK, K.L., & ZMUD, R.W., "A synthesis of research on requirements analysis and knowledge acquisition techniques", Management Information Systems Quarterly, Vol. 16, No. 1, pp. 117-138, 1992.
[20] ALVAREZ, R., "Discourse Analysis of Requirements and Knowledge Elicitation Interviews", Proceedings of the 35th Hawaii International Conference on System Sciences-2002.
[21] LANG, M., DUGGAN, J., "A Tool to Support Collaborative Software Requirements Management", Requirements Engineering Vol. 6, pp. 161-172, 2001.
[22] VİLLER, S., I., "Social analysis in the requirements engineering process:from ethnography to method", Proceedings of the International Symposium on Requirements Engineering, pp. 6-13, IEEE Computer Society Pres, 1999.
[23] What Is the Cost of a Requirement Error? [article] By Joe Marasco - June 26, 2007. https://www.stickyminds.com/article/what-cost-requirement-error
[24] BERGERSEN B.M., "User satisfaction and influencing issues, https://www.cs.hioa.no/~frodes/rm/bard.pdf
[25] YADAV, S.B., BRAVOCO, A.T., CHATFIELD, A.T., RAJKUMAR, T.M., "Comparison of Analysis Techniques for Information Requirement Determination", Communications of the ACM, Vol. 31, No. 9, (1090-1097), September 1988, http://130.18.86.27/faculty/warkentin/papers/YadavEtAl1988_CACM329_AnalysisTechniques.pdf
[26] SHI, Y., SPECHT, P., STOLEN, J., WETERING, F.V., "A consensus ranking for information system requirements", Information Management & Computer Security Vol. 4 No. 1, pp. 10-18, 1996.
[27] LARSEN, T.J., NAUMANN, J.D., "An experimental comparison of abstract and concrete representations in systems analysis", Information and Management, Vol. 22, pp. 29-40, 1992.
[28] VALUSEK, J.., FRYBACK, D.G., "Information Requirements Determination: Obstacles Within, Among and Between Participants", ACM, 1985.
[29] BROWNE, G.J., RAMESH, V., "Improving information requirements determination: a cognitive perspective", Information & Management, Vol. 39, pp. 625-645, 2002 (2002) 1–21.
[30] BROWNE, G.J., RAMESH, V., PITTS, M.G., ROGICH, M.B., "Representing User Requirements: An Empirical Investigation of Formality in Modeling Tools" Elektronik adres: http://aisel.isworld.org/article_by_author .asp?Author_ID=209
[31] VİDGEN, R., "Stakeholders, soft systems and technology: separation and mediation in the analysis of information systems requirements", *Information Systems Journal, Vol.* 7, pp. 21-46, 1997.

[32] FLYNN, D. J., DAVARPANAH JAZİ, M., "Constructing user requirements: a social process for a social context", *Information Systems Journal, Vol.* 8, pp. 53-83, 1998.
[33] MCDERMID, D.C., "Structure and Information Requirements Definition, Working Paper Series", Number: 9906, September 1999.
[34] VALUSEK, J. R., AND FRYBACK, D. G., "Information requirements determination: obstacles within, among and between participants", Information Analysis: Selected Readings, pp. 139-51, Addison-Wesley, 1987.
[35] GREENSPAN, S., MYLOPOULOS, J., BORGİDA, A., "On Formal Requirements Modeling Languages: RML Revisited". *Proceedings of the 16th International Conference on Software Engineering,* pp. 135-147, Los Alamitos, IEEE Pres, 1994.
[36] ZMUD, R. W., ANTHONY, W. P., STAİR, R., "The use of mental imagery as a requirements technique", *Challenges and Strategies for Research in Systems Development,* pp. 337-50, 1992.
[37] VİTALARİ, N.P., "Structuring the Requirements Analysis Process for Information systems: A Proposition Viewpoint", *Challenges and Strategies for Research in Systems Development,* pp. 163-79, John Wiley, Chichester, England, 1992.
[38] ROGICH, M.B., BROWNE G. J., "Directed Questions for Structured Interviews in Requirements Determination", Elektroni adres: http://aisel.isworld.org /article_by_author.asp?Author_ID=2138
[39] KAZMAN, R., ABOWD, G., BASS, L., "Scenario Based Analysis of Software Architecture", IEEE Software, 13, 1996, pp. 47-55.
[40] NUSEIBEH, B., EASTERBROOK, S., "Requirements Engineering: a road map", ACM Press, Series-Proceeding-Article, 2000, pp. 35-46.
[41] SOMMERVILLE, I., Software Engineering, 6th edition, Addison-Wesley, 2001.
[42] User Centred Product Creation in Interactive Electronik Publishing (User Requirements Analysis), (http:www.vnet5.org/pub/approach/ura.html)
[43] TAYLOR, H., "Role-play Cases for Teaching Interviewing Skills in Information Systems Analysis", HERDSA Annual International Conference, Melbourne, 12-15 July 1999.
[44] WHITE, S., EDWARDS, M., "A Requirements Taxonomy for Specifying Complex Systems", IEEE, 1995.
[45] SMİTH, G.F., BENSON, P.G., CURLEY, S.P., "Belief, Knowledge, and Uncertainty: A . Cognitive Perspective on Subjective Probability", Organizational Behavior and Human Decision Processes, Vol. 48, pp. 291-321, 1991.
[46] CURLEY S.P. & BENSON P.G., "Judgment-Based And Reasonıng-Based Stoppıng Rules In Decısıon Makıng Under Uncertaınty", Working Paper,
http://ids.csom.umn.edu/Faculty/scurley/Home/curley/NicklesCurleyBenson1995.pdf
[47] KOGUT, C.A., "Consumer search behavior and sunk costs", Journal of Economic Behavior and. Organization, Vol. 14, pp. 381-392, 1990.

[48] OCKER, R., FJERMESTAD, J., HILTZ, S.R., TUROFF, M, "An Exploratory Comparison of Four Modes of Communication for Determining Requirements : Result on Creativitiy, Qualitiy and Satisfaction", IEEE, 1997.
[49] TAGGART, W.M., THARP, M.O., "A Survey of Information Requirements Analysis Techniques", Computing Surveys, Vol. 9, No 4, December 1977.
[50] JONSON, L., JONSON, E.N., "Knowledge Acquisition for Expert Systems", Prenum Pres by Alison L. Kidd, Review: Knowledge Elicitation Using, pp. 91-108, 1987.
[51] GUTIERREZ, O., "Experimental techniques for information requirements analysis", Information and Management, Vol. 16, No. 1, pp. 31-43, 1989.
[52] HOLBROOK, H., "A Scenario-Based Methodology for Conducting Requirements Elicitation," ACM SIGSOFT Software Engineering Notes, pp. 95-104, January 1990.
[53] GOGUEN, J.A., LINDE, C., "Techniques for Requirements Elicitation", In proceedings, Requirements Engineering, 93 edidded by Stephan Fickes and Anthony Finkelstein, IEEE Compiter Society, 1993, pages 152-164.
[54] HUGHES, K. J., RANKIN, R. M., SENNETT, C. T., "Taxonomy for Requirements Analysis", British Crown Copyright, Defence Research Agency, UK, 1994.
[55] SHARP, H., "Knowledge Acquisition or Requirements Analysis?", IEEE, 1994 (Institute of Electrical and Electronics Engineers).
[56] COAD, P., YOURDAN, E., "*Object-Oriented Analysis*", 2nd Edition. Englewood Cliffs, NJ.: Prentice-Hall, 1991.
[57] BOOCH, G., "Object-Oriented Analysis and Design With Applications", Second Edition, Benjamin/Cummings, Menlo Park, California, 1994.
[58] TAMAI, T., "Current practices in software processes for system planning and requirements analysis", Information and Software Technology, Vol.35, pp. 339-344, 1993.
[59] HEVNER, A.R., MILLES, H.D., "Box-structured requirements determination methods", Decision Support Systems Vol. 13 pp. 223-239, 1995, North-Holland.
[60] ALEXANDER, I., "A Historical Perspective on Requirements Engineering", 1997, Elektronik adres: http://easyweb.easynet.co.uk/~iany/consultancy/historical/historical_text.htm
[61] SHAW, M.L.G., GAINES, B.R., "Requirements Acquisition", Software Engineering Journal, pp. 149-165, May 1996.
[62] LYYTIAINEN, V., "Analysing Requirements for Content Management", Elektronik adres: http://www.cs.jyu.fi/~airi/papers/Lyytikainen-ICEIS-2003.pdf
[63] RATCLIFFE, J.S., "Scenario Building A Suitable Method for Strategic Construction Industry Planning?", https://www.researchgate.net/publication/265997244_Scenario_Building_A_Suitable_Method_for_Strategic_Construction_Industry_Planning
[64] KUMAR, A., PALVİA, P., "Developing Global Executive Information Systems", Elektronik adres: http://www.uncg.edu/bae/people/palvia/dss/GlobalEIS.Final.doc

[65] WETHERBE, J.C., "Determining Executive Information Requirements: Beter, faster and cheaper", Cicle Time Research, pp. 19
[66] Vitharana, Padmal and Zahedi, F. M. "Group Decision Support for Software Requirements Analysis," Proceedings of Association for Information Systems Conference, August 1997, pp. 443-443.
[67] FLYNN, D.J., ARCE, J.E., "A Case Tool to Support Critical Success Factors Analysis in IT Planning and Requiremets Determination", Information and Software Technology, Vol. 39, pp. 311-321, 1997.
[68] HARKER, S., EASON, K., "The Use of Scenarious for Organizational Requirements Generation", Proceedings of the 32nd Havai International Conference on System Sciences-1999.
[69] ALSPAUGH, T.A., ANTON, A.I., BARNES, T., MOTT, B.W., "An integrated scenario management strategy", In Fourth IEEE International Symposium on Requirements Engineering, pp. 142–149, June 1999.
[70] USHER, J.M., KABER, D.B., "Establishing Information Requirements for Supervisory Controllers in a Flexible Manufacturing Systems Using GTA", Human Factors and Ergonomics in Manufacturing, Vol. 10, No. 4, pp. 431-452, 2000.
[71] CORONADO, A.E., SARHADI, M., MILLAR, C., "Defining a Framework for Information Systems Requirements for Agile Manufacturing", International Journal of Production Economics 75, pp. 57-68, 2002.
[72] MOORE, J.M., SHIPMAN, F.M., "A Comparison of Questionnaire – Based and GUI – Based Requirements Gathering", IEEE, 2000.
[73] DARKE, P., JEGIELSKA, I., "Incorporating Knowledge Discovery Into The Process of Determining User's Information System Requirements", Proc. 10th Australasian Conference on Information Systems, 1999.
[74] ROBERTSON, S., ROBERTSON, J., "Mastering the Requirements Process", Addison-Wesley, 1999.
[75] CARROLL, J.M., SWATMAN P.A., "Building Understanding of Information Systems Practice: Research in the Field", Elektronik adres: http://www.deakin.edu.au/buslaw/infosys/docs/workingpapers/archive/Working_Papers_99/99_03_Carroll.pdf , 1999.
[76] SALMERON, J.L., LUNA, P., MARTINEZ, F.J., "Executive Information Systems in Major Companies: Spanish Case Study", Computer Standatrs & Interfaces, Vol. 23, pp. 195-207, 2001.
[77] MANNIO, M., NIKULA, U., "Requirements Elicitation Using a Combination of Prototypes and Scenarious", Elektronik adres: http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_WER01/mannio.pdf
[78] PARK, S., KIM, H., KO, Y., SEO, I., "Implementation of an efficient requirements-analysis supporting system using similarity measure techniques", Information and Software Technology, Vol. 42, pp. 429-438, 2000.
[79] LAI, L.S., "An Integration of Systems Science Methods and Object-Oriented Analysis for Determining Organizational Information Requirements", Systems Research and Behavioral Science Vol. 17, pp. 205-228, 2000.

[80] HICKEY, A.M., DAVIS, A.M., "Requirements Elicitation and Elicitation Technique Selection: A Model for Two Knowledge-Intensive Software Development Processes", Proceedings of the 36th Hawaii International Conference on System Sciences, 2003.
[81] EVA, M., "Requirements acquisition for rapid applications development", Information & Management 39, pp. 101-107, 2001.
[82] KASSEL, N.W., MALLOY, B.A., "An Approach to automate Requirements elicitain and Specification", Proceedings of the 7th IASTED International Conference Software Engineering and Applications November 3-5, 2003, Marina Del Rey, CA, USA.
[83] ZHU, Z., "Requirements Determination and Requirements Structuring" Elektronik adres: http://www.umsl.edu/~sauterv/analysis/6840_f03_papers/zhu/
[84] ANI, B.A., EDWARDS, K., "Am Emprical Study of a Qualitative Systematic Approach to Requirements Analysis (QSARA)", Proceedings of the 2004 International Symposium on Emprical Software Engineering, 2004.
[85] FREEMAN, L.A., "The Effects of Concept Maps on Requirements Elicitaion and System Models During Information Systems Development", Concept Maps: Theory, Methodology, Technology Proc. Of the First Int. Conference on Concept Mapping, Spain 2004.
[86] SANDY, G.A., "The Value of a Rules Base Approach to Requirements Engineering".
[87] TUUNANEN, T., PEFFERS, E., GENGLER, C.E., "Wide Audience Requirements Engineering (WARE): A Practical Method and Case Study", Helsinki School of Economics Working Papers, 2004, W-378.
[88] GEMINO, A., "Empirical comparisons of animation and narration in requirements validation", Requirements Engineering, Springer-Verlag London Limited 2003.
[89] HUANG, I.L., "A Design of Self-Questioning Mechanism for Information Requirement Specification", Allied Academies International Conference, Proceedings of the Academy of Information and Management Sciences, Vol. 9, No. 2, pp. 19-25, Las Vegas, 2005.
[90] GONCALVES, J.F., AND RESENDE, M.G.C., "An Evolutionary Algorithm for Manufacturing Cell Information", Computers & Industrial Engineering, 47, 247-273, 2004.
[91]. TSAI, C.C., AND LEE, C.Y., "Optimization of Manufacturing Cell Formation With a Multi-Functional Mathematical Programing Model", Int. Adv. Manuf. Technol., 30; 309-318, 2006.
[92] ESCOTO, R.P., MCDONNELL, L.R., SALORT, E.V., AND ESTEBAN, F.C.L., "Development of an Algorithm for the Application of Group Technology in the Design of Manufacturing Systems", Production Planning & control, Vol. 9, No. 3, 267-374, 1998.
[93] JOGLEKAR P. , CHUNG Q B. , TAVANA M., "Note On a Comparative Evaluation of Nine Well-Known Algorithms for Solving the Cell Formation Problem in Group Technology" , Journal Of Applied Mathematics & Decision Sciences, 5(3), 253- 268, 2001.

[94] ÖZMEHMET S., TUNALI S., “Hücre Oluşturma Amaçlı Bir Karar Destek Sistemi”, YAEM’2000.
[95] PARKIN, R.E., AND LI,M.L., “The Multi-Dimensional Aspects of a Group Technology Algorithm”, International Journal of Production Research, Vol. 35, No. 8, pp.2345-2358, 1997.
[96] KING, J.R., “Machine Component Grouping Formation in Group Technology”, International Journal of Management Science, Vol. 8(2), pp. 193-199, 1980.
[97] KING J. R. AND NAKORNCHAI V., “Machine-Component Group Formation in Group Technology : Review and Extension”, IJPR, 20 (2) : 117-133, 1982.
[98] CHAN, H. M., AND MILNER, D.A., “Direct Clustering Algorithm for Group Formation in Cellular Manufacture”, JMS, Vol. 1(1), pp. 65-74, 1982.
[99] MCAULEY, J., “Machine Grouping for Efficient Production”, Production Engineer, Vol. 51, No.2, pp. 53-57, 1972.
[100] GUPTA, T. AND SEIFODDINI, H., “Production Data Based Similarity Coefficient for Machine-Component Grouping Decisions in The Design of A Cellular Manufacturing System”, International Journal of Production Research, Vol. 28, No. 7, pp.1247-1269, 1990.
[101] BROWN, E.C., SUMICHRAST, R.T., “CF-GGA: a grouping genetic algorithm for the cell formation problem”, int. j. prod. res., vol. 39, no. 16, 2001.
[102] AKSOY, M. S., Yapay Zekada Öğrenme, Adapazarı, 1998.
[103] AKSOY, M. S., “Pruning Decision Trees Using Rules3 Inductive Learning Algorithm”, Mathematical and Computational Applications, Vol. 10, No. 1, pp. 113-120, 2005.
[104] AKSOY, M. S., “Dynamic System Modelling Using Rules3 Induction Algorithm”, Mathematical and Computational Applications, Vol. 10, No. 1, pp. 121-132, 2005.
[105] GILL, A., AND BOCTOR, C. R., “A Fuzzy Linguistic Approach To Data Quantification and Construction of Distance Measures for The Part Family Formation Problem”, Int. J. Prod. Res., Vol. 35, No. 9, pp. 2565-2578, 1997.

978-625-7954-15-0