

**eMine: Eye Tracking Teknolojisi Kullanarak Web  
Sayfalarının Dönüştürülmesi**

**Proje No: 109E251**

Yrd. Doç. Dr. Yeliz Yeşilada

ARALIK 2013

KKTC

## **ÖNSÖZ**

World Wide Web (web) günümüzde artık sadece bilgisayar üzerinden değil, her yerden erişilebilir hale gelmiştir. Yerine göre mobil kullanıcılar tarafından küçük bir el cihazı üzerinden ya da örneğin görme engelli kullanıcılarda olduğu gibi içeriğin görüntülenemediği durumlarda ekran okuyucu ile ses şeklinde erişilebilmektedir. Hâlbuki web sayfaları genelde görsel etkileşim amaçlı tasarlandıklarından dolayı, alternatif erişim yöntemleri ile sayfalara erişmek neredeyse imkânsızdır. Bu projedeki ana hedefimiz, göz tarama teknolojisinin özgün bir uygulamasını kullanarak kısıtlı ortamlarda kullanıcı deneyimini iyileştirmektir. Özet olarak göz tarama güzergâhlarını sayfanın kodu ile ilişkilendirip, daha sonra web sayfalarını farklı şekillere dönüştürüp daha kolay erişilebilmesini sağlamaktır.

Bu proje 1001 – TÜBİTAK – Bilimsel ve Teknolojik Araştırma Projelerini Destekleme (KKTC) programı çerçevesinde 109E251 proje kodu ile desteklenmiştir. TÜBİTAK’a sunduğu destekten dolayı müteşekkirimiz.

## İÇİNDEKİLER

|                                                                                       |      |
|---------------------------------------------------------------------------------------|------|
| 1. GİRİŞ .....                                                                        | 1    |
| 2. LİTERATÜR ÖZETİ.....                                                               | 5    |
| 2.1 Web Sayfalarını Görsel Bloklara Ayırma: Bölütleme.....                            | 6    |
| 2.2 Görsel Elemanlara Rol Atama .....                                                 | 9    |
| 2.3 Tarama Güzergâhlarını Analiz Etme Teknikleri .....                                | 11   |
| 2.4 Dönüştürme Teknikleri .....                                                       | 13   |
| 3. GEREÇ VE YÖNTEM .....                                                              | 17   |
| 3.1 Sayfadaki Görsel Elemanların Otomatik Keşfi .....                                 | 17   |
| 3.1.1 Görsel Blok Ayırıcı .....                                                       | 18   |
| 3.1.2 Kural Geliştirici .....                                                         | 21   |
| 3.1.3 Rol Atayıcı.....                                                                | 23   |
| 3.1.4 Teknik ve Kullanıcı Bazlı Deneyler ve Sonuçları.....                            | 23   |
| 3.2 eMine Algoritması'nın Geliştirilmesi ve Onaylanması .....                         | 28   |
| 3.2.1 Algoritma Analizleri .....                                                      | 29   |
| 3.2.2 eMine Algoritması'nın Geliştirilmesi ve İlk Testler .....                       | 31   |
| 3.2.3 eMine Algoritması'nın Uygulanması .....                                         | 32   |
| 3.2.4 eMine Algoritması'nın Deney ve Sonuçları .....                                  | 35   |
| 3.3 Dönüştürme Deneyleri ve Geçerliliği .....                                         | 40   |
| 3.3.1 Dönüştürme Teknikleri .....                                                     | 41   |
| 3.3.2 Dönüştürme Teknikleri Uygulamaları.....                                         | 43   |
| 3.3.3 Dönüştürme Teknikleri Deney ve Sonuçları.....                                   | 46   |
| 4. BULGULAR.....                                                                      | 52   |
| 5. TARTIŞMA.....                                                                      | 57   |
| 6. SONUÇ .....                                                                        | 59   |
| 7. KAYNAKLAR.....                                                                     | 60   |
| 8. EKLER .....                                                                        | i    |
| 8.1 Web Page Segmentation: A Review.....                                              | i    |
| 8.2 Heuristics for Visual Elements of Web Pages.....                                  | ii   |
| 8.3 Vision Based Page Segmentation: Extended and Improved Algorithm .....             | iii  |
| 8.4 Automatic Discovery of Visual Elements of Web Pages .....                         | iv   |
| 8.5 Evaluation of Automatic Discovery of Visual Elements of Web Pages.....            | v    |
| 8.6 eMINE Scanpath Analysis Algorithm .....                                           | vi   |
| 8.7 Correlation Between Scanpaths and Visual Elements of Web Pages .....              | vii  |
| 8.8 Web Page Transcoding Based on Eye Tracking .....                                  | viii |
| 8.9 Implementation and Evaluation of Web Page Transcoding Based on Eye Tracking ..... | ix   |

|      |                                                                              |      |
|------|------------------------------------------------------------------------------|------|
| 8.10 | Heuristic Role Detection of Visual Elements of Web Pages.....                | x    |
| 8.11 | Experiential Transcoding: An Eye Tracking Approach .....                     | xi   |
| 8.12 | Vision Based Page Segmentation Algorithm: Extended and Perceived Success.... | xii  |
| 8.13 | Analysis of Algorithms to Identify Patterns in Eye Tracking Scanpaths.....   | xiii |
| 8.14 | Web Page Segmentation – Approaches, Techniques and Algorithms .....          | xiv  |

## **TABLO LİSTELERİ**

|                                                                                      |    |
|--------------------------------------------------------------------------------------|----|
| Tablo 1. Görsel elemanların keşfi algoritması performans sonuçları.....              | 25 |
| Tablo 2. Rol belirleme algoritması performans sonuçları .....                        | 25 |
| Tablo 3. Görsel elemanların keşfi algoritmasının başarı sonuçları .....              | 27 |
| Tablo 4. Katılımcıların en iyi seviye tercihleri .....                               | 27 |
| Tablo 5. Rol atama algoritmasının başarı sonuçları .....                             | 28 |
| Tablo 6. Örnek güzergâhlar .....                                                     | 33 |
| Tablo 7. eMine Algoritması ile çıkarılan ortak güzergâhlar.....                      | 38 |
| Tablo 8. Verilen göreve ve kullanıcı sayısına göre çıkarılan ortak güzergâhlar ..... | 38 |
| Tablo 9. Kadın ve erkek katılımcılara göre çıkarılan ortak güzergâhlar .....         | 39 |
| Tablo 10. Aşinalığa göre çıkarılan ortak güzergâhlar .....                           | 39 |
| Tablo 11. Farklı seviyelerde bloklara ayrılan sayfaların ortak güzergâhı .....       | 40 |
| Tablo 12. Ortak güzergâh içinde bulunan elemanlara erişim süreleri.....              | 48 |
| Tablo 13. Görsel karşılaştırma sonuçları .....                                       | 49 |
| Tablo 14. Mobil simülasyon sonuçları.....                                            | 50 |

## ŞEKİL LİSTELERİ

|                                                                      |    |
|----------------------------------------------------------------------|----|
| Şekil 1. Sistem Mimarisi .....                                       | 17 |
| Şekil 2. eMine VIPS ACTF entegrasyonu .....                          | 20 |
| Şekil 3. ACTF ana sayfası ve otomatik yaratılan blok ağacı .....     | 20 |
| Şekil 4. Wordpress ana sayfası ve otomatik yaratılan blok ağacı..... | 21 |
| Şekil 5. eMine Ontolojisi'ndeki Header rolü tanımı.....              | 22 |
| Şekil 6. Header rolü için yaratılan JESS kuralı.....                 | 22 |
| Şekil 7. eMine ACTF rol atama entegrasyonu .....                     | 24 |
| Şekil 8. Görsel elemanlara ayrılmış örnek sayfa .....                | 30 |
| Şekil 9. eMine Algoritması'nın uygulaması .....                      | 32 |
| Şekil 10. eMine Ortak Güzergâh Çıkarma Algoritması.....              | 32 |
| Şekil 11. eMine Algoritması ACTF entegrasyon mimarisi .....          | 34 |
| Şekil 12. eMine Algoritması ACTF uygulaması .....                    | 35 |
| Şekil 13. Örnek JSON formatında ortak güzergâh .....                 | 35 |
| Şekil 14. Babylon sayfasındaki ortak güzergâh .....                  | 37 |
| Şekil 15. eMine dönüştürme mimarisi.....                             | 43 |
| Şekil 16. Blok-bazlı dönüştürme örneği .....                         | 44 |
| Şekil 17. Rol-bazlı dönüştürme örneği .....                          | 46 |

## ÖZET

World Wide Web (web) günümüzde artık sadece bilgisayar üzerinden değil, her yerden erişilebilir hale gelmiştir. Yerine göre mobil kullanıcılar tarafından küçük bir el cihazı üzerinden ya da örneğin görme engelli kullanıcılarda olduğu gibi içeriğin görüntülenemediği durumlarda ekran okuyucu<sup>1</sup> ile ses şeklinde erişilebilmektedir. Hâlbuki web sayfaları genelde görsel etkileşim amaçlı tasarlandıklarından dolayı, alternatif erişim yöntemleri ile sayfalara erişmek neredeyse imkânsızdır. Bu projedeki ana hedefimiz, göz izleme (eye tracking) teknolojisinin özgün bir uygulamasını kullanarak kısıtlı ortamlarda kullanıcı deneyimini iyileştirmektir. Projedeki **hedefimiz**, sayfanın kodu ile insanların tarama güzergâhını ilişkilendirerek sayfaları kısıtlı ortamlarda daha rahat erişilebilecek bir şekle dönüştürmektir. Bu temel hedefe ulaşabilmek için projenin dört **amacı** vardır: (1) Sayfanın kodunu kullanarak sayfadaki görsel elemanları otomatik olarak tespit etmek; (2) Tarama güzergâhlarının sayfadaki görsel elemanlarla ilişkilendirilebilmesi için özgün bir algoritma geliştirmek; (3) Yaratılacak algoritmaya bağlı olarak özgün dönüştürme teknikleri geliştirmek; (4) Kısıtlı ortamlarda geliştirilen dönüştürme tekniklerinin kullanıcının deneyimini iyileştirdiğini göstermek. Bu **sonuç** raporunda da sunduğumuz gibi projemiz önerilen hedefine varmış ve şu sonuçları elde etmiştir: (1) Sayfanın kodu kullanılarak sayfadaki görsel elemanlar otomatik olarak çıkarılabilmekte ve %80 gibi bir doğrulukla otomatik olarak rolleri atanabilmektedir; (2) Tarama güzergâhları sayfanın kodu ile otomatik olarak ilişkilendirebilmekte ve bu ilişki doğrultusunda kullanıcılar arasında ortak bir güzergâh çıkarabilmektedir; (3) Bu ortak güzergâh, görsel elemanlar ve otomatik olarak ortaya çıkarılmış roller farklı dönüştürme teknikleri ile başarı ile kullanılabilir; (4) Yapılan çalışmalarda bu tekniklerin kullanıcının deneyimini olumlu yönde etkileyeceği gösterilmektedir. Bu sonuçlar projenin başarı ile sonuçlandığını ve hedefine ulaştığını göstermektedir. Başarılı bir proje olarak da gelecekte birçok yönden geliştirilebileceğini göstermektedir: otomatik rol atama kısmı üzerinde farklı çalışmalar yapıp doğruluk oranı artırılmaya çalışılabilir, ortak tarama güzergâhı çıkarmak üzere farklı yaklaşımlarla değişik algoritmalar geliştirilebilir, bunları kullanarak da farklı dönüştürme teknikleri ve deneyleri yapılabilir.

---

1 Ekran okuma programları ekrandaki bilgiyi sesli olarak kullanıcıya iletmektedir.

## ABSTRACT

The World Wide Web (web) is ubiquitous - it can be accessed by a small device while the user is mobile or it can be accessed in audio if the user cannot see the content, for instance visually disabled users who use screen readers<sup>2</sup>. However, since web pages are mainly designed for visual interaction; it is almost impossible to access them in alternative forms. Our overarching goal is to improve the user experience in such constrained environments by using a novel application of eye tracking technology. Our **objective** is to use eye tracking data to generate an algorithm for identifying people's scanpaths and relate those scanpaths to visual elements of web pages. In order to achieve that, we have four **major** aims: (1) Identify visual elements of web pages automatically by using the underlying source code; (2) Develop a novel algorithm to identify scanpaths in terms of visual elements; (3) Develop novel transcoding techniques based on this algorithm; (4) Demonstrate that the proposed techniques improve the user experience in constrained environments. As it is presented in this final report, we have achieved our objective and reached the following **results**: (1) visual elements have been identified automatically and their roles been discovered with %80 accuracy; (2) a novel algorithm has been developed that successfully relates scanpaths with the underlying source code and identifies a common scanpath; (3) pages have been successfully transcoded by using the visual elements, their roles and the generated common scanpath; (4) these novel transcoding techniques have shown to improve the user experience. These results demonstrate that the project has successfully been completed and reached its objective. Future work of this project could be: investigating alternative methods to improve the accuracy of automatic role detection, developing alternative algorithms to identify common scanpaths and proposing different transcoding techniques.

---

<sup>2</sup> Screen reader is software that presents on screen data in audio.



## 1. GİRİŞ

World Wide Web (web) günümüzde artık sadece bilgisayar üzerinden değil, her yerden erişilebilir hale gelmiştir. Yerine göre mobil kullanıcılar tarafından küçük bir el cihazı üzerinden ya da görme engelli kullanıcılarda olduğu gibi, içeriğin görüntülenemediği durumlarda ekran okuyucu<sup>3</sup> ile ses şeklinde erişilebilmektedir. Hâlbuki web sayfaları genelde görsel etkileşim amaçlı tasarlandıklarından dolayı, alternatif erişim yöntemleri ile sayfalara erişmek neredeyse imkânsızdır (Vigo ve Harper, 2013a). Bu projedeki ana hedefimiz, göz izleme (eye tracking) teknolojisinin özgün bir uygulamasını kullanarak kısıtlı ortamlarda kullanıcı deneyimini iyileştirmektir.

Göz izleme çalışmaları 20 yılı aşkın bir süredir bilişsel süreçleri incelemek için yaygın olarak kullanılmaktadır (Rayner, 1998). Fakat, insanların göz hareketlerinin web kullanımı sırasında izlenmesi oldukça yeni bir alandır (Goldberg vd., 2002; Josephson ve Holmes, 2002, 2004). Okuma esnasında, göz genellikle yumuşak hareketler yapmaz, “sıçrama” (saccade) adı verilen sürekli ve hızlı hareketler yapar. Sıçramalar arasında gözün nispeten hareketsiz kaldığı sabitlemeler oluşur ve bu sabitlemeler kullanıcının dikkatini nereye verdiğini gösterir. Tarama güzergâhları (scanpath), görsel bir unsur sonucunda oluşan, sürekli tekrarlanan sıçrama ve sabitleme dizileridir. Daha önceki çalışmalarımızda, insanların web sayfalarında görsel elemanları nasıl kullandıklarını (Yeşilada vd., 2008b; Jay vd., 2006), dikkatlerini dinamik içeriğe nasıl verdiklerini (Jay ve Brown, 2008; Michailidou vd., 2008) ve basit ve karmaşık sayfaları nasıl algıladıklarını (Brown vd., 2009) anlamak için sıçrama ve sabitlemeler incelenmiştir. Belirtilen çalışmalar sonucunda oldukça fazla miktarda veri toplanmıştır. eMine adlı bu projedeki amacımız, toplanan bu verileri kullanarak insanların tarama güzergâhlarını oluşturacak ve bu güzergâhların sayfa içerisindeki görsel elemanlarla ilişkilendirilmesini sağlayacak bir algoritma çıkarmaktır. Güzergâh tespiti için “String-edit” algoritması kullanan bazı çalışmalar olmasına rağmen, bu çalışmalardan hiçbiri güzergâhlar, sayfadaki görsel elemanlar ve sayfanın temelini oluşturan kod arasındaki ilişkileri incelememektedir (Josephson ve Holmes, 2002; Takeuchi ve Habuchi, 2007). eMine Projesi’nin esas **hedefi** sayfanın kodu ile insanların tarama güzergâhını ilişkilendirerek sayfaları kısıtlı ortamlarda daha rahat erişilebilecek bir şekle dönüştürmektir. Bu temel hedefe ulaşabilmek için, projenin dört esas **amacı** aşağıdaki şekilde özetlenebilir:

1. Sayfanın kodunu kullanarak sayfadaki görsel elemanları otomatik olarak tespit etmek;
2. Tarama güzergâhlarının sayfadaki görsel elemanlarla ilişkilendirilmesi için özgün bir algoritma geliştirmek;

---

3 Ekran okuma programları ekrandaki bilgiyi sesli olarak kullanıcıya iletmektedir.

3. Yaratılacak algoritmaya bağılı olarak özgün dönüştürme teknikleri geliştirmek;
4. Kısıtlı ortamlarda geliştirilen dönüştürme tekniklerinin kullanıcının deneyimini iyileştirdiğini göstermek.

Bu hedeflere ulaşabilmek için proje aşağıdaki üç iş paketinden oluşmaktadır:

1. *Sayfadaki Görsel Elemanların Otomatik Keşfi*: Bu iş paketi, ilk hedefimiz olan sayfadaki görsel elemanların tespit edilmesinin otomatikleştirilmesi ile ilgilidir. Temel amaç, sayfadaki görsel elemanları rolleri ve sayfa kodu ile ilişkilendirip, bu işlemin geçerliliğini deneylerle onaylamaktır.
2. *eMine Algoritması'nın Geliştirilmesi ve Onaylanması*: Bu iş paketi, tarama güzergâhlarının sayfadaki görsel elemanlar bakımından tespit edilmesini sağlayacak olan algoritmanın geliştirilip test edilmesine adanmıştır. Dolayısı ile bu iş paketi ikinci hedefimiz ile ilgilidir.
3. *Dönüştürme Deneyleri ve Geçerliliği*: Bu son iş paketi, eMine Algoritması'nı kullanarak web sayfalarını dönüştürme teknikleri geliştirmeyi, teknik ve kullanıcı testleri ile geliştirilen tekniklerin geçerliliğini kanıtlamayı hedefler; dolayısı ile bu iş paketi üçüncü ve dördüncü hedeflerimize yoğunlaşır.

Projenin üç yıl içerisinde tamamlanması öngörülmüş ve öngörülen zamanda bu üç iş paketi tamamlanarak, proje başarı ile sonlandırılmıştır. Özet olarak, bu **sonuç** raporunda da sunduğumuz gibi, projemiz önerilen hedefine varmış ve şu sonuçları elde etmiştir: (1) Sayfanın kodu kullanılarak, sayfadaki görsel elemanlar otomatik olarak çıkarılabilmekte ve %80 gibi bir doğrulukla rolleri otomatik olarak atanabilmektedir; (2) Tarama güzergâhları sayfanın kodu ile otomatik olarak ilişkilendirebilmekte ve bu ilişki doğrultusunda kullanıcılar arasında ortak bir güzergâh çıkarılabilmektedir; (3) Bu ortak güzergâh, görsel elemanlar ve otomatik olarak ortaya çıkarılmış roller farklı dönüştürme teknikleriyle başarı ile kullanılabilir; (4) Yapılan çalışmalarda, bu tekniklerin kullanıcının deneyimini olumlu yönde etkileyeceği gösterilmektedir. Bu sonuçlar projenin başarı ile sonuçlandığını ve hedefine ulaştığını göstermektedir. Ayrıca, proje gelecekte birçok yönden geliştirilebilir: otomatik rol atama kısmı üzerinde farklı çalışmalar yapıp, doğruluk oranı arttırılmaya çalışılabilir, ortak tarama güzergâhı çıkarma üzerine farklı yaklaşımlarla değişik algoritmalar geliştirilebilir, bunları kullanarak da farklı dönüştürme teknikleri geliştirilebilir ve bu teknikler üzerinde deneyler yapılabilir.

Proje önerimizde de belirttiğimiz gibi, proje çıktılarımız düzenli olarak teknik rapor halinde yayınlanmıştır. Bu raporlar, yukarıdaki üç iş paketi altında yapılan çalışmaları detaylı bir şekilde anlatmaktadır ve şu şekilde özetlenebilir:

### 1. Sayfadaki Görsel Elemanların Otomatik Keşfi:

Teknik Rapor 0: Web Page Segmentation: A Review (bkz. 8.1. Bölüm)

Teknik Rapor 1: Heuristics for Visual Elements of Web Pages (bkz. 8.2. Bölüm)

Teknik Rapor 2: Vision Based Page Segmentation: Extended and Improved Algorithm (bkz. 8.3. Bölüm)

Teknik Rapor 3: Automatic Discovery of Visual Elements of Web Pages (bkz. 8.4. Bölüm)

Teknik Rapor 4: Evaluation of Automatic Discovery of Visual Elements of Web Pages (bkz. 8.5. Bölüm)

### 2. eMine Algoritması'nın Geliştirilmesi ve Onaylanması

Teknik Rapor 5: eMINE Scanpath Analysis Algorithm (bkz. 8.6. Bölüm)

Teknik Rapor 6: Correlation Between Scanpaths and Visual Elements of Web Pages (bkz. 8.7. Bölüm)

### 3. Dönüştürme Deneyleri ve Geçerliliği:

Teknik Rapor 7: Web Page Transcoding Based on Eye Tracking (bkz. 8.8. Bölüm)

Teknik Rapor 8: Implementation and Evaluation of Web Page Transcoding Based on Eye Tracking (bkz. 8.9. Bölüm)

Proje önerimizde de bahsettiğimiz gibi, proje çıktılarını kamu ile paylaşmak adına, proje için özel bir web sitesi tasarlanmıştır: <http://emine.ncc.metu.edu.tr/>. Burada düzenli olarak proje çıktıları, yapılan çalışmalar ve sonuçları yayınlanmıştır. Ayrıca, proje çıktılarımız tebliğ olarak hazırlanmış ve aşağıdaki uluslararası kongrelerde sunulmuştur:

1. Akpınar, E., Yeşilada, Y. 2013. "Heuristic role detection of visual elements of web pages". Web Engineering. Lecture Notes in Computer Science, vol. 7977, pp. 123-131. Editör: Daniel, F. ve Dolog, P., Li, Q. Springer. (bkz. 8.10. Bölüm)
2. Yeşilada, Y., Harper, S., Eraslan, Ş. 2013. "Experiential transcoding: an eye tracking approach". Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility. pp. 30:1-30:4. ACM, New York, NY, USA. (bkz. 8.11. Bölüm)
3. Akpınar, E., Yeşilada, Y. 2013. "Vision Based Page Segmentation Algorithm: Extended and Perceived Success". Current Trends in Web Engineering - ICWE 2013 Workshops and PhD Symposium. Lecture Notes in Computer Science, vol. 8295, pp. 238-252. Editör: Sheng Q. Z. ve Kjeldskov J. Springer. (bkz. 8.12. Bölüm)
4. Eraslan, Ş., Yeşilada, Y., Harper, S. 2013. "Analysis of algorithms to identify patterns in eye tracking scanpaths". Signal Processing and Communications Applications Conference (SIU), 2013 21st. pp. 1-4. (bkz. 8.13. Bölüm)

Projemizin sonuçlarının hakemli dergi yayını olarak yayınlanması planlanmıştır. Bu kapsamda aşağıdaki makale tamamlanıp, gönderilmiştir:

1. Yeşilada, Y. 2013. “Web Page Segmentation – Approaches, Techniques and Algorithms”, ACM Transactions on the Web. ACM. (bkz. 8.14. Bölüm)

Aşağıdakiler de teknik raporlarımızdan yola çıkılarak, tamamlanıp gönderilmesi planlanmaktadır:

2. Akpınar, E., Yeşilada, Y. 2013. “Automatic Discovery of Visual Elements of Web Pages”, Hazırlanıyor – ACM Transactions on the Web. ACM.
3. Eraslan, Ş., Yeşilada, Y., Harper, S. 2013. “Correlation Between Scanpaths and Visual Elements of Web Pages”, Hazırlanıyor – Behaviour and Information Technology. Taylor and Francis.
4. Akpınar, E., Yeşilada, Y. 2013. “Web Page Transcoding Based on Eye Tracking”, Hazırlanıyor – Interacting with Computers. Elsevier.

Bu sonuç raporunun geri kalan bölümleri, yapılan çalışmalar ile ilgili literatür özetini (bkz. 2. Bölüm), projede takip edilen yöntemleri, kullanılan gereçler ve teknikleri (bkz. 3. Bölüm), ve elde edilen bulguları (bkz. 4. Bölüm) sunup, detaylı tartışarak (bkz. 5. Bölüm), varılan genel sonuçları anlatmaktadır (bkz. 6. Bölüm).

## 2. LİTERATÜR ÖZETİ

Projemiz birçok disiplinin birlikte kullanıldığı bir projedir. Bu yüzden, projenin getirilerini tam olarak algılayabilmek için öncelikle web erişilebilirliği, mobil web ve göz izleme (eye tracking) konularında genel bilgi sahibi olunması gerekmektedir.

**Web Erişilebilirliği ve Mobil Web:** Web erişilebilirliği üzerine yapılan son çalışmalar hala birçok sayfanın en temel erişilebilirlik kurallarına bile uymadığını göstermektedir (European Union Policy Survey, 2005; Disability Rights Commission (DRC), 2004; Takagi vd., 2007; Coyne ve Nielsen, 2001; Vigo ve Harper, 2013a, 2013b). Web erişilebilirliği konusu ile ilgili yürütülen projelerin sayısı<sup>4</sup>, yasal konular<sup>5</sup> ve üretilen yönergelerin<sup>6</sup> sayısında göz ardı edilemeyecek bir artış olmasına rağmen, engelli kullanıcılar için web kullanımı hala zordur. Son zamanlarda yapılan bir çalışmada, Takagi vd. (2007) “görme engelli kullanıcıların web üzerinden alışveriş yaparken, görebilen kullanıcılardan 10 kat daha fazla zaman harcadığı” sonucuna varmıştır. Benzer şekilde, mobil web kullanıcılarının da, görme engelli kullanıcılar gibi uyum ve kullanım zorlukları yaşadıkları gitgide daha aşikâr olmuştur (Yeşilada vd., 2008a; Chuter ve Yeşilada, 2008). World Wide Web Consortium (W3C)’deki Mobile Web Initiative (MWI)<sup>7</sup>, sektördeki önemli kişilerle birlikte bu sorunu çözmeye çalışıyor olsa da, tasarımcıların yönergeleri takip etmedikleri herkes tarafından bilinmektedir (Regan, 2004; Harper ve Yeşilada, 2008a). Örneğin sayfalar, tasarımlarından dolayı genelde küçük bir ekrana sığmamakta ve bu da kullanıcının oldukça fazla miktarda yatay ve dikey fare kullanmasını gerektirmektedir. Ayrıca, günümüzde mobil veri aktarımı oldukça pahalıdır ve mevcut teknolojilerle oldukça yavaştır. Mobil web teknolojilerinde bu sorunların etkilerini hafifletebilecek gelişmeler herkes tarafından arzu edilmektedir.

**Göz izleme (Eye tracking):** İnsanların göz hareketlerini takip etmek 20 yılı aşkın bir süredir bilişsel işlemlerin araştırılmasında yaygın olarak kullanılmakta olsa da (Rayner, 1998), web kullanımında göz izleme oldukça yeni bir kavramdır (Goldberg vd., 2002; Josephson ve Holmes, 2002, 2004). Göz izlemenin en yaygın kullanımı da sayfaların standart tasarım ve düzeninin geliştirilip<sup>8</sup>, kullanılabilirliğinin ölçülmesidir (Russell, 2005). Araştırmalar ayrıca,

---

4 Web erişilebilirliği değerlendirme araçları, [w3.org/WAI/ER/tools/](http://w3.org/WAI/ER/tools/)

5 Web erişilebilirliği kuralları, [w3.org/WAI/Policy/](http://w3.org/WAI/Policy/)

6 WCAG 2.0, <http://www.w3.org/TR/WCAG20/>

7 MWI, <http://www.w3.org/Mobile/>

8 Eye-tracking araştırmaları, [www.useit.com/eyetracking/](http://www.useit.com/eyetracking/)

sayfa üzerindeki elemanların deęişik durumlarda ne kadar dikkat çektięi (ör. (Granka vd., 2004; Pan vd., 2004)), aranmakta olan bilgiye göre gözlerin nasıl hareket ettięi (Pirolli ve Cart, 1999) ve bir menü ararken önceki beklentilerin ve sayfanın karmaşıklık düzeyinin kullanıcıyı nasıl etkiledięi konularına da yoğunlaşmıştır (McCarthy vd., 2003). Daha önceki çalışmalarımızda sabit ölçüt (Takeuchi ve Habuchi, 2007) olarak kabul edilen sıçrama ve sabitlemeler incelenip, insanların web sayfalarındaki görsel elemanları nasıl kullandıkları (Yeşilada vd., 2008b; Jay vd., 2006), dinamik içeriklere dikkatlerini nasıl verdikleri (Jay ve Brown, 2008; Michailidou vd., 2008) ve basit ve karmaşık sayfaları nasıl algıladıkları (Brown vd., 2009) anlaşılmaya çalışılmıştır. Tarama güzergâhı analizi ise göz izleme verilerini incelemek için alternatif bir dinamik ölçüttür (Takeuchi ve Habuchi, 2007). Tarama güzergâhı ile ilgili bazı çalışmalar yapılmıştır; fakat bu çalışmalar web sayfalarını resim olarak kullanan “String-edit” algoritmasına yoğunlaşır (Josephson ve Holmes, 2002; Takeuchi ve Habuchi, 2007) ve bu çalışmalardan hiçbirisi tarama güzergâhı, sayfadaki görsel elemanlar ve sayfanın kodu arasındaki ilişkileri incelemez. Sayfanın kodu ile tarama güzergâhının ilişkilendirilmedięi koşullarda, göz izleme verileri ile yapılabilecekler çok sınırlıdır. Dolayısıyla bizim hedefimiz, göz izleme verilerini kullanarak tarama güzergâhlarını tespit etmek, bu güzergâhları sayfadaki görsel elemanlar ve sayfa kodu ile ilişkilendirmek ve sayfaların engelli ve mobil web kullanıcıları tarafından daha iyi algılanabilecek şekilde deęiştirilebilmesini sağlamaktır.

Bu proje kapsamında yaptığımız çalışmalar ile ilgili detaylı literatür taramasını aşağıdaki dört bölümde özetleyebiliriz. Bu tarama, web sayfalarını otomatik olarak görsel bloklara ayırma, bu blokların otomatik olarak rollerini keşfetme, göz izleme verilerindeki tarama güzergâhlarını analiz etme ve bu güzergâhları, görsel blokları ve görsel blokların rollerini kullanarak web sayfalarını dönüştürme tekniklerini içermektedir.

## **2.1 Web Sayfalarını Görsel Bloklara Ayırma: Bölütleme**

Web sayfalarının görsel bloklara ayrılması, temel olarak mobil web kullanıcıları (Yeşilada vd., 2004; Hwang vd., 2003; Whang vd., 2001) ve web sayfalarına sesli olarak erişen görme engelli kullanıcıların (Mahmud vd., 2007; Borodin vd., 2007; Yeşilada vd., 2004) erişebilirliğini arttırmak için web sayfalarının yeniden yapılandırılması amacıyla kullanılmaktadır. Bu konuda yapılan literatür taramaları, sayfaları bloklara ayırma işleminin bilgi erişimi (Cai vd., 2003; Cai vd., 2004b) ve görsel erişimi (Cai vd., 2004a) gibi pek çok alanda kullanıldığını göstermektedir. Bu bölümde, mobil web ve mobil erişilebilirlik konusunda yapılan çalışmalar kısaca anlatılmaktadır.

Mobil web pek çok erişilebilirlik sorunu içermektedir. Web sayfaları belirli ekran boyutları için tasarlanmıştır (Hattori vd., 2007; Yin ve Lee, 2004) ve büyük sayfalar için tasarlanmış olan web sayfalarının mobil cihaz ekran boyutları için küçültülmesi zor olabilmektedir (Millic-Frayling ve Sommerer, 2002). Bazı mobil cihazlar klavye ve fare gibi donanımlara sahip değildir (Xie vd., 2005), bazıları sınırlı işlemci gücüne sahiptir (Xie vd., 2005) ve çoklu ortam verileri bu tip cihazlarda problemlere sebep olmaktadır (Hwang vd., 2003; Whang vd., 2001). Ayrıca, mobil cihaz kullanıcılarının kullanım amaçları ve istekleri farklılaşmaktadır. Örneğin, mobil web kullanıcıları, fazla miktarda veri arasından aradıklarını bulmaya çalışmaktan ziyade, daha spesifik konularda net ve konu ile alakalı cevaplar almak istemektedir (Xie vd., 2005). Bir başka önemli problem de, web sayfası içerisindeki bloklar ile web sayfası elemanları arasındaki ilişkinin anlamsal olarak belirgin olmamasıdır. Bu da bir sayfanın diğer cihaz türlerinde veya alternatif platformlarda kolaylıkla görüntülenebilmesine engel olmaktadır (Xiang ve Shi, 2006; Yin ve Lee, 2004). Bloklara ayırma işlemi, mobil cihaz ve ekran okuyucu kullanıcılarının sayfaları daha iyi görüntüleyebilmesi açısından, blokların ilişkilendirilebilmesine olanak sağlamaktadır.

Literatürde bloklara işlemini web sayfalarının yeniden yapılandırılması için kullanan değişik yaklaşımlar mevcuttur. Yin ve Lee (2004) web sayfalarının önemli kısımlarını seçerek, mobil ortamda yalnızca bu kısımların gösterilmesini amaçlamaktadır. Benzer şekilde, Song vd. (2004) web sayfasındaki blokların önem derecelerini belirlemeyi amaçlamaktadır. Önem derecesi, web sayfasının hangi bölümlerinin öncelikli olarak kullanıcıya gösterileceği konusunda bilgi sağlamaktadır. Yin ve Lee (2005) ve Xie vd. (2005), sayfanın ana içeriğini belirleyerek kullanıcıya bu kısmı göstermektedir. Ahmadi ve Kong (2008) web sayfalarını içeriklerine göre alt sayfalara bölerek, sayfa içeriğini kullanıcıya özet şeklinde sunmayı amaçlamaktadır. Hwang vd. (2003) ve Whang vd. (2001), web sayfasında yer alan blok ve bileşenleri daha küçük alt sayfalar dizisine dönüştüren ve bağlantılar yardımıyla ilişkilendiren, dizinsel olarak bloklara ayırma (indexed segmentation) adlı bir yöntem öne sürmektedir. Hattori vd. (2007) web sayfalarını küçük alt sayfalara bölerek, bu alt sayfaları kullanıcıya bir "İçindekiler" bölümü ile sunmakta ve bağlantılarla her bir alt sayfaya erişim sağlamaktadır.

Blokları web sayfasında yakınlaşma – uzaklaşma gibi fonksiyonları gerçekleştirme amaçlı kullanan (Millic-Frayling ve Sommerer, 2002), gezinim ve içerik elemanlarını belirleyen ve ilişkili olanları seçerek sayfayı özetleyen (Ahmadi ve Kong, 2008), benzer şekilde içerik ile ilişkisi olmayan elemanları sayfadan çıkararak sayfayı sadeleştiren (Xiao vd., 2008a), gezinim çubuğunu belirleyerek sayfanın içerik sayfası olması durumunda bu kısmı kullanıcıya göstermeyen (Chen vd., 2001) çalışmalar da mevcuttur. Benzer şekilde, Chen vd.

(2001) sayfa içerisinde yalnızca sayfanın tasarımıyla ilgili bulunan ve içeriksel olarak anlam taşımayan bloklarla logo, reklam, iletişim bilgileri, telif bilgileri ve referanslar gibi özel bölümleri sayfadan çıkarmaktadır. Hwang vd. (2003) ve Whang vd. (2001) bölüm başlıklarını kullanarak farklı bölümleri ayırtıran bir dönüştürme yöntemi öne sürmektedir. Bu teknikte, sayfa yapısı değişmemekte, fakat bölüm başlıkları bağlantılara dönüştürülerek ilgili içeriğe yönlendirilmektedir. Xie vd. (2005), web sayfalarında bulunan farklı renklerdeki blokların önem derecelerini belirleyerek, kullanıcıya önemli blokları öne çıkaran minyatür bir yapı sunmaktadır.

Xiao vd. (2005) kullanıcıya sayfayı bütün olarak sunmakta ve kullanıcı sayfa üzerinde bölgelere tıklayarak ilgili içeriğe ulaşmaktadır. Benzer şekilde, Xiao vd. (2008b) sayfayı minyatür bir yapıda göstermekte ve kullanıcı fare imlecini bir bölüm üzerinde gezdirdiğinde, ilgili bölümün içeriğini asenkron şekilde çekerek kullanıcıya göstermektedir. Chen vd. (2005) ve Chen vd. (2003) de web sayfalarını minyatür bir yapıda göstermekte ve kullanıcılar alt sayfalara ulaşabilmektedir. Yang ve Shi (2009) ve Baluja (2006) minyatür yapı üzerinde bir blokla ilgili tüm bilgileri göstermekte, kullanıcı ilgili içeriğe ulaşabilmek için bir bloktan diğerine atlayabilmektedir.

Web erişilebilirliği göz önüne alındığında, görme engelli kullanıcıların da mobil web kullanıcılarına benzer sorunlar yaşadığı ve bu sorunların benzer yaklaşımlarla çözülmeye çalışıldığı görülmektedir (Yeşilada vd., 2008). Sayfa içeriğinin görsel olarak ayrıştırıldığı fakat bu ayrışımın sayfa kodlarına yansıtılmadığı durumlarda, ekran okuyucu gibi uygulamalar bu tip bilgilere ulaşamamaktadır. Bu sebeple, bu tip web sayfaları ekran okuyucular için erişilemez olmaktadır. Asakawa ve Takagi (2000) ve Takagi vd. (2002) web sayfalarının erişilebilirliğini arttıracak şekilde yeniden yapılandırmak için, sayfa içindeki görsel blokları belirlemeyi amaçlamaktadır. Bu çalışmada, blokların sayfa içindeki rollerini saptayabilmek için, blokları elle sınıflandıran bir yöntem öne sürülmüştür. Benzer bir yaklaşım, Yeşilada vd. (2004) tarafından Dante Projesi kapsamında öne sürülmüştür. Bu yaklaşımda, görsel bloklar bir ontoloji kullanılarak sınıflandırılmış ve web sayfaları görme engelli kullanıcılar için yeniden yapılandırılmıştır. Bir diğer benzer yaklaşım Lunn vd. (2011) tarafından öne sürülmüştür. Bu yaklaşımda ise, görsel elemanları bloklara ayırma işlemi sayfanın CSS kodu kullanılarak otomatize edilmiştir. Mahmud vd. (2007) ve Borodin vd. (2007), ekran okuyucularının sayfaları en başından başlayarak ve görsel elemanların tanımlandığı sırayı izleyerek okuduğunu belirtmiştir. Örneğin, sayfada ilk olarak sayfanın en üst bölümünde yer alan gezinim çubuğu, büyük başlıklar ve reklamlar okunmaktadır. Bu sebeple, sayfanın esas içeriğine ulaşmak için geçen süre, internet kullanımını zaman alan ve yorucu bir işlem haline



getirmektedir. Bu problemi çözmek amacıyla, Hearsay ve Csurf adlı işitsel tarayıcılar kullanılmaktadır. Bu tarayıcılar, kullanıcı bir bağlantıya tıkladığında, yönlendirilen sayfada yer alan ve bağlantıyla ilgili olan bölümü tespit etmekte ve kullanıcıyı doğrudan bu bölüme yönlendirmektedir. Bunun için de yönlendirilen sayfa bloklara ayırılmakta ve ilgili içerik tespit edilmektedir.

Bloklara ayırma yöntemleri yukarıdan aşağı ve aşağıdan yukarı yöntemler şeklinde iki ana başlık altında sınıflandırılabilir. Yukarıdan aşağı bloklara ayırma yöntemleri, sayfanın ana elemanı ile başlar ve sayfanın belirli özelliklerini kullanarak, döngüsel şekilde bu elemanı daha küçük parçalara ayırır (Baluja, 2006). Aşağıdan yukarıya bloklara ayırma yöntemlerinde ise, işlem sayfa içerisinde yer alan en küçük bölünemez birimlerle başlar (Baluja, 2006). Bloklara ayırma işlemini otomatikleştirmek için, buluşsal yöntemler (Ahmadi ve Kong, 2008), otomatik öğrenme (Baluja, 2006) ve kümeleme gibi farklı yöntemler de bulunmaktadır.

## **2.2 Görsel Elemanlara Rol Atama**

Görsel elemanları (blok) otomatik olarak tanıyan ve rollerini belirleyen farklı yaklaşımlar bulunmaktadır (Cai vd., 2003). Rol belirlemesi, farklı alanlarda değişik amaçlarla kullanılmaktadır. Bu bölümde, otomatik rol belirleme konusundaki mevcut çalışmalar özetlenecektir. Genel olarak, eMine Projesi kapsamında sunulan algoritmayla karşılaştırıldığında, mevcut yaklaşımlar içerdikleri rol çeşitliliği açısından zayıf kalmaktadır.

Web sayfalarını küçük ekranlı cihazlara adapte edebilmek için, çeşitli rol belirleme yöntemleri sunulmuştur. Yin ve Lee (2005), sayfa elemanlarını içerik, ilişkili bağlantılar, gezinim ve destek, reklam ve form şeklinde parçalara ayırmaktadır. Benzer şekilde, Ahmadi ve Kong (2008), çalışmalarında üst, ana içerik, sol ve sağ menüler, alt ve reklam gibi diğer rollere yer vermektedir. Bu çalışmada, bir görsel elemanın rolü, elemanın sayfa içerisindeki pozisyonuna göre belirlenmektedir. Fakat, bir elemanın pozisyonu sayfa tasarımına bağlı olarak değişebilmektedir. Bu sebeple, pozisyon rol belirlemede tek başına bir kriter olarak kullanılamaz. Chen vd. (2005) üst seviye blokları başlık, altlık, sol ve sağ kenar çubuğu ve sayfa gövdesi şeklinde sınıflandırmaktadır ve bu yapıyı anlamsal yapı şeklinde tanımlamaktadır. Başlık ve altlık bölümlerinin belirli bir yapıda olduğu ve başlığın sayfanın en üstünde, altlığın da sayfanın en altında yer aldığı varsayılmaktadır. Her ne kadar bu çalışmada temel üst seviye içerik blokları üzerinde yoğunlaşılsa da, çalışma üstte başlık, altta altlık, yanlarda kenar çubukları ve ortada sayfa içeriği yer alan belirli bir sayfa şablonunu esas almaktadır. Güncel sayfa tasarımları düşünüldüğünde, bu tip bir şablonun genellenmesi

oldukça zordur. Son olarak, Xiao vd. (2008b) benzer bir rol belirleme yaklaşımı sunmakta ve bağlantı bloklarını içerik bloklarından ayırma konusuna yoğunlaşmaktadır. Basit rol tanımlarının aksine, Chen vd. (2001) “Fonksiyon Bazlı Obje Modeli” adında oldukça kapsamlı bir model öne sürmektedir. Bu modele göre, bir web sayfası temel ve birleşik objelerden oluşur ve bu objeler, alt kategorilere bölünebilir. Öne sürülen modelin kapsamlı olmasına rağmen, aynı zamanda oldukça teorik bir yaklaşımdır ve son kullanıcının web sayfalarını nasıl anlayacağı ve kullanacağı göz önüne alınmamıştır.

Akıllı kullanıcı arayüzleri oluşturmak da web sayfalarının yapılarının iyi şekilde anlaşılmasını gerektirir. Bu yapının iyi bir şekilde anlaşılması, farklı cihazlarda ve sistemlerde daha akılcı gösterilebilmesine olanak sağlar. Xiang ve Shi (2006) bir web sayfasını temel görsel bloklar ve ayraçların bileşimi olarak tanımlar. Bu tanımda, görsel bloklar sayfa içinde yer alan ve daha küçük parçalara bölünemeyen görsel kısımlardır. Bu çalışmada yalnızca iki tip görsel elemana yer verilmektedir: metinsel olmayan bloklar (düğmeler, görseller, girdiler, vb.) ve metinsel bloklar (form elemanları haricinde bulunan metinsel içerik). Bu yaklaşım da kapsadığı rol çeşitliliği bakımından oldukça basit bir yaklaşımdır ve web sayfası elemanlarını yeterli şekilde anlatmamaktadır.

Bilgi erişimi ve veri madenciliği alanlarında da rol belirleme uygulamaları kullanılmaktadır. Kovacevic vd. (2002) web sayfalarının sınıflandırılmasında web sayfalarını bloklara ayırarak tutarlı objeler oluşturmayı amaçlamaktadır. Bu çalışmada temel olarak başlık, altlık, sol ve sağ menüler ve sayfanın merkezi belirlenmektedir. Bu bağlamda, bir grup rol tanımlanmıştır, fakat bu roller sayfanın görsel tasarımını belirli soyutlamalara dayandırmaktadır. Lin ve Ho (2002) iki çeşit blok tanımlamaktadır: bilgi içeren ve gereksiz bloklar. Bu tanımlar veri madenciliği sürecinde kullanılmaktadır. Bilgi içeren bloklar kullanıcı için sayfada anlamlı bir içeriğe sahip olan bloklardır. Gereksiz bloklar ise sayfa içinde belirli bir anlamı olmayan bloklardır. Şirket logoları, gezinim panelleri, reklam çubukları, vb. bu grup altında değerlendirilmiştir. Bu çalışmada esas amaç bilgi erişimini desteklemek olduğu için, rol tanımları basit düzeydedir. Benzer şekilde, Burget ve Rudolfova (2009) da veri madenciliği kapsamında rol belirleme uygulamaları kullanmıştır. Bu çalışmanın esas amacı, sayfanın esas içeriği, başlıklar, paragraflar ve yazar gibi bilgileri belirlemektir. Bu bilgiler veri madenciliği için kullanışlı olsa da, genel amaçlı bir rol tanımı için yetersizdir. Liu vd. (2003) veri madenciliği sürecini desteklemek için elemanların içerdiği bağlantı sayısına göre üç farklı rol tanımı yapmaktadır: gezinim çubuğu, gezinim listesi ve içerik blokları. Bağlantı içermeyen pek çok eleman bu sınıflandırmanın dışında kalmaktadır. Ayrıca, aynı bağlantı sayısına sahip iki farklı görsel eleman, sayfa içerisinde farklı amaçlarla kullanılabilir. Son olarak, Yi

vd. (2003), veri madenciliği için uygun olmayan blokları filtrelemeyi amaçlamaktadır. Her ne kadar, gezinim panelleri, telif bölümü, özel notlar gibi farklı tipte bloklar tanımlanmış olsa da, çalışmanın esas amacı görsel elemanların rollerinin belirlenmesi değildir.

Web erişilebilirliği, görsel elemanların rollerinin kullanışlı olduğu bir diğer alandır. Takagi vd. (2002), görme engelli kullanıcıların erişebilirliğini arttırmak için web sayfasında yer alan bölümleri tanımayı amaçlamıştır. Bu çalışmada uygun içerik, güncellenmiş dizin, genel dizin, rolsüz, başlık, altlık, reklam, silinecekler ve yerleşim tablosu rol tanımları yer almaktadır. Her ne kadar bu rol tanımları kapsamlı olarak görünse de, bazı roller eksiktir. Ayrıca bu roller, sistematik olarak ele alınmadığı, yalnızca dönüştürmeye yardımcı roller şeklinde tanımlandığı izlenimi vermektedir.

Sonuç olarak, web sayfalarında yer alan görsel elemanların rollerinin belirlenmesi, web sayfalarının mobil platforma adaptasyonu, akıllı web arayüzlerinin oluşturulması, bilgi erişiminin desteklenmesi, veri madenciliği ve görme engelli kullanıcıların erişebilirliğinin artırılması gibi pek çok alanda kullanılması bakımından önemlidir. Farklı çalışmalar farklı rol tanımlarına ve görsel eleman özelliklerine yer verse de, bu çalışmaların hiçbirisi rollerin derinlemesine anlaşılmasını sağlayamamaktadır. ARIA ve HTML5 gibi çeşitli standartlar mevcut çalışmalara oranla daha güçlü ve daha iyi rol listeleri içerse de, bu standartlar da rollerin otomatik tanımlanması konusunda eksik kalmaktadır.

### **2.3 Tarama Güzergâhlarını Analiz Etme Teknikleri**

Web sayfası üzerindeki tarama güzergâhları, çeşitli yöntemler ve algoritmalar kullanılarak analiz edilmiştir. Bunların birçoğu, bu tarama güzergâhlarını alfabetik bir dizin olarak kullanmaktadır. Tarama güzergâhlarını alfabetik bir dizin olarak oluşturmak için web sayfalarının ilgi alanlarının kullanılması gerekmektedir (Takeuchi ve Habuchi, 2007). Bu ilgi alanları, tablo düzeni (grid-layout) (Takeuchi ve Habuchi, 2007), web sayfasının kodu (Akpınar ve Yeşilada, 2012) veya görme odaklarının web sayfası üzerindeki dağılımı kullanılarak (Santella ve DeCarlo, 2004) oluşturulabilir.

Tarama güzergâhı üzerindeki çalışmalar genellikle “string-edit” algoritması üzerinde yoğunlaşır (Takeuchi ve Habuchi, 2007; Josephson ve Holmes, 2002). Bu algoritma, bir alfabetik dizini diğer bir alfabetik dizine çevirmek için gerekli olan en az sayıda ekleme, çıkarma ve değiştirme işlemini kullanarak, iki dizin arasındaki farklılığı hesaplamaktadır. Örneğin, “string-edit” algoritması “ABCD” ile “ABCE” alfabetik dizinlerindeki sadece son harfi

değiştirerek bunların birbirleri ile aynı olmasını sağladığından, aralarındaki farklılığı 1 (bir) olarak hesaplar. Bu farklılık kullanılarak, tarama güzergahları kategorilere ayrılabilir (West vd., 2006) ve/veya web sayfası üzerinde kullanıcı davranışları arasındaki farklılıklar incelenebilir (Josephson ve Holmes, 2002). Fakat, bu algoritmanın göz önünde bulundurulması gereken zayıf yönleri de vardır. İlk olarak, bu algoritma tüm ilgi alanları arasındaki değiştirmelerin değerini aynı olarak kabul etmesine rağmen, ilgi alanları farklı boyutlara sahip olabileceğinden ve ilgi alanları arasındaki mesafenin farklılık göstereceğinden dolayı tüm ilgi alanları arasındaki değiştirmelerin değerleri her zaman aynı olmayabilir. Değiştirme matrisleri (Substitution Matrix) ile bu ilgi alanları arasındaki değiştirme değerleri tutulabilir ve bu değerler kullanılarak farklılıklar hesaplanabilir (Takeuchi ve Habuchi, 2007). Yine bu matrisler kullanılarak, Needleman ve Wunsch iki alfabetik dizin arasındaki benzerliği hesaplamışlardır (Day, 2010).

Bir diğer zayıf özellik ise, "string-edit" algoritmasının görme odağı süresi gibi süre metriklerini göz ardı etmesidir. Ayrıca bu metrikler çeşitli şekillerde de yorumlanabilmektedir. Örneğin, görme odağının süresinin uzun olması algıda oluşan bir zorluğu düşündürebilmektedir (Poole ve Ball, 2005). Diğer yandan, ScanMatch yöntemi görme odağı süresini şu şekilde göz önünde bulundurmıştır: Bu yöntemde belli bir süre saptanarak ve bu süre kullanılarak tarama güzergâhının alfabetik dizin üzerinde bulunan ilgi alanı tekrarlandırılabilir (Cristino vd., 2010). Örneğin, saptanan süre 100 ms olursa ve ilgi alanı "A" üzerindeki görme odağı 200 ms ise, alfabetik dizinde bir adet "A" yerine iki adet "A" olması gerekir.

Yukarıda bahsedilen algoritma ve yöntemler aynı anda iki tane tarama güzergâhına uygulanabilmektedir. Oysa yaptığımız çalışmanın amacı en az iki tarama güzergâhı için ortak bir tarama güzergâhı belirlemektir. Geçiş Matrisi (Transition Matrix) olarak bilinen yöntem aynı anda ikiden fazla tarama güzergâhını kullanarak bir geçiş matrisi oluşturmaktadır (West vd., 2006). Bu matriste, her bir hücre satır ve kolon olasılıklarını içermektedir. Satır olasılıkları ile bir sonraki ilgi alanı; kolon olasılıkları ile de bir önceki ilgi alanı belirlenebilmektedir. Fakat, bu matris kullanılmaya başlandığı zaman iki önemli soruyu da beraberinde getirmektedir. Bunlar; "Başlangıç noktası ve bitiş noktasını simgeleyen ilgi alanları hangileridir?" ve "Hangi olasılıklar değerlendirilmelidir?".

Bu problemleri çözüme kavuşturmak amacıyla bazı diğer yöntemler göz önünde bulundurulabilir. Literatürde "Shortest Common Supersequence" adlı yöntemden söz edilmektedir (Raiha, 2010). Bu yöntem ile birden fazla kişiye ait ortak tarama güzergâhı belirlenebilmektedir. Maalesef bu yöntemin önemli zayıf yönleri de vardır. Örneğin, bu

yöntem ile “ABE”, “ADE” ve “AFE” için ortak güzergâh “ABDFE” olarak belirlenebilir. Bu belirlenen ortak güzergâh diğer güzergâhlara nazaran oldukça uzundur. Bununla birlikte oluşan ortak güzergâh tüm güzergâhlar tarafından desteklenmez. Mesela, “ABE”, “F”yi içermemektedir.

"T-Pattern" (Mast ve Burmeister, 2011) ve "eyePatterns Discover Patterns" (West vd., 2006) gibi bazı yöntemler, tarama güzergâhları içerisindeki alt güzergâhları bulmaya çalışmaktadır. "eyePatterns Discover Patterns" yöntemi tarama güzergâhları içerisinde fazladan bulunabilecek ilgi alanlarına tolerans göstermemektedir (West vd., 2006). Örneğin, “ABC”, “ABC” ve “ABCD”nin alt güzergâhı olarak bulunabilirken, “ABC” ve “ABXCD”nin alt güzergâhı olarak bulunamamaktadır. Bunun sebebi ise “X”tir. Bu örnek bu yöntemin indirgemeci yaklaşımını göstermektedir. "eyePatterns" adlı program alt güzergâh bulmak için iki güzergâhı sıralama yöntemini kullanır (West vd., 2006). Bir diğer yöntem olan "eSeeTrack" adlı yöntem tarama güzergâhlarını bir zaman çizgisinde gösterir ve görme odağı sürelerini kolayca görmemizi sağlar (Tsang vd., 2010). Bir diğer özelliği ise ilgi alanlarının sıklık derecesini göz önünde bulundurmasıdır. Bu şekilde, hangi ilgi alanından sonra hangi ilgi alanının gelebileceğini olasılıklarla göstermektedir.

"Multiple Sequence Alignment" isimli yöntem literatürde ortak güzergâh bulmak için önerilmiş olmasına rağmen, bu yöntem değerlendirilip geçerli hale getirilmemiştir (Hembrooke vd., 2006). "Dot-plots" adlı diğer bir algoritma ise ortak güzergâh bulabilmek için kullanılmıştır (Goldberg ve Helfman, 2010). Bu algoritma ikili güzergâhların ortak güzergâhını bularak hiyerarşik bir yapı oluşturur (Goldberg ve Helfman, 2010). Bu yapıda, tarama güzergâhları en alt seviyede bulunurken, ortak tarama güzergâhı ise en üst seviyededir. Bu algoritmanın da indirgemeci yaklaşımını ortadan kaldırmak için bazı istatistiksel yöntemler kullanılmıştır.

## **2.4 Dönüştürme Teknikleri**

Engelli ve mobil kullanıcılar için kötü tasarlanmış sayfaların erişilebilirliğini artırmanın bir yolu, bu sayfaları alternatif şekillere dönüştürmektir. Dönüştürme, sayfanın içeriğini değişik şekillerde sunarak, ses veya mobil cihazlar gibi alternatif ortamlara daha uygun hale getirmeyi amaçlamaktadır. Dönüştürme, bulgusal (heuristics) (Richards ve Hanson, 2004), kullanıcı tercihleri (Richards ve Hanson, 2004), harici açıklamalar (Takagi vd., 2002; Hori vd., 2000) ve vekil sunucu kullanımı (Büyükkökten vd., 2001, 2002) gibi değişik yollar kullanılarak gerçekleştirilebilir. Sayfayı dönüştürebilmek için atlama linki (skip link) (Takagi vd., 2002, 2007), içeriğin özetini yaratma (Büyükkökten vd., 2001, 2002, 2000; Yang ve Wang, 2003), sayfaların minyatür resimlerinin yaratılması (Chen vd., 2003), derecelendirme ve yeniden

düzenleme (Yin ve Lee, 2004), içeriği daha küçük parçalara bölme (Baluja, 2006; Hattori vd., 2007) ve alakasız içeriği ortadan kaldırma gibi değişik yöntemler kullanılabilir (Baudisch vd., 2004). Yapılan araştırmalar dönüştürme tekniklerinin kullanılmasının engelli ve mobil kullanıcıların web deneyimlerini iyileştirdiğini gösteriyor olsa da (Takagi vd., 2007, Yeşilada vd., 2007), etkili bir dönüştürme tekniği uygulanabilmesi için sayfanın içerik, yapı ve bağlamının (context) çok iyi anlaşılması gerektiği de açıktır (Yeşilada vd., 2008b). Maalesef bu çalışmaların birçoğu dönüştürme işlemini yaparken sayfanın nasıl kullanıldığını anlamaya çalışmamaktadır. Bize göre göz tarama çalışmaları bunu başarmak için çok önemli fırsatlar sunmaktadır. Aşağıda, dönüştürme teknikleri iki başlık altında detaylı şekilde incelenmektedir: farklı ekran boyutları için adaptasyon ve web erişilebilirliği ile ilgili yapılan çalışmalar:

**Farklı Ekran Boyutlarına Adaptasyon:** Son yıllarda web sayfalarının farklı ekran boyutlarına adaptasyonu konusunda pek çok çalışma yapılmıştır. Bu çalışmalarda öne sürülen yöntemlerden biri, sayfa içinde gereksiz veya içerik ile ilgisiz bölümlerin sayfadan çıkarılmasıdır. Xiao vd. (2008a) öncelikle sayfayı bloklara bölmekte ve içerik ile ilgisiz bölümleri sayfadan çıkarmaktadır. Benzer şekilde, Chen vd. (2001), bazı görsel elemanları, dekorasyon ve reklam gibi özel objeleri gruplayarak sayfa yapısından çıkarmaktadır. Fakat, bu tip objelerin sayfadan çıkarılması, geri kalan içeriğin ekrana uygun şekilde sığacağı anlamına gelmemektedir. Ayrıca, kullanıcılar bilinçli bir şekilde özellikle reklamlara ulaşmak isteyebilir, bu durumda sayfanın bazı bölümleri erişilemez hale gelecektir. Diğer yandan, bazı çalışmalar sayfanın yalnızca belirli bir bölümünü kullanıcıya göstermeyi amaçlamaktadır. Xie vd. (2005) ve Yin ve Lee (2005) öncelikle sayfa içinde en önemli bölümü belirlemekte ve bu bölümü kullanıcıya göstermektedir. Benzer şekilde, Yin ve Lee (2004) sayfanın yalnızca önemli bölümlerini göstermektedir. Fakat, önemsiz olarak tasnif edilen bölümler de bazı kullanıcılar için önemli olabilir. Bu durumda kullanıcı bu bölümlere erişememektedir. Bu çalışmaların aksine, Yin ve Lee (2005) ve Song vd. (2004) görsel öğeleri önem derecelerine göre sıralamakta ve sıralı erişim sağlamaktadır.

Bu konuda kullanılan bir başka yöntem de, sayfaların özetlenerek kullanıcının seçimine göre belirli bir kısmın gösterilmesidir. Ahmadi ve Kong (2008) ve Hattori vd. (2007) sayfayı alt sayfalara bölerek, kullanıcıya bir “İçindekiler” bölümü sunmaktadır. Chen vd. (2003, 2005) de sayfayı alt sayfalara bölmekte ve her bir alt sayfayı birbiriyle ilişkilendirmektedir. Benzer şekilde, Hwang vd. (2003) sayfaları alt sayfalara bloklara ayırarak alt sayfaları bağlantılar aracılığıyla sıralı bir şekilde sunmaktadır. Sayfa özetlemesi, farklı gösterim şekilleriyle de sağlanabilir. Xiao vd. (2005) sayfaları “Dokun ve görüntüle” adını verdikleri bir modelle

göstermektedir. Bu modelde, kullanıcının seçimine göre ilgili bölümün içeriği gösterilmektedir. Baluja (2006), Chen vd. (2003, 2005), Xiang vd. (2007), Xiao vd. (2008b) ve Yang ve Shi (2009), sayfayı küçük resimler kullanarak minyatür bir yapıya çevirmekte ve kullanıcının seçtiği küçük resimlere göre ilgili içeriği göstermektedir. Xie vd. (2005) bu yaklaşımı bir adım ileri götürerek, içerikleri önem derecelerine göre öne çıkarmaktadır. Bazı çalışmalar da, sayfayı özetlemekten ziyade, görsel öğeleri özetlemeyi amaçlamaktadır. Hwang vd. (2003) ve Whang vd. (2001) sayfa yapısını değiştirmemekte, fakat her bir başlığı ilgili içeriğe ulaşacak şekilde bağlantılarla değiştirmektedir. Başka bir yöntem de, her bir bloğun içeriğini yalnızca ilk cümleleri kalacak şekilde sayfadan çıkararak sayfayı özetlemektir (Ahmadi ve Kong, 2008).

Bazı sayfalar gömülü videolar gibi medya içeriği bulundurmaktadır. Sayfa tasarımının zayıf olduğu durumlarda, bu tip içerikler sayfaya sığmayabilir. Bu tip durumlarda ilgili içerik sayfaya sığacak şekilde küçültülebilir (Adzic vd., 2011). Başka bir yöntem de, video objelerinin içeriğini metinlere çevirerek kullanıcıya sunmaktır (Adzic vd., 2011). Fakat bu çalışmanın esas amacı, görsel elemanları orijinal şekliyle kullanıcıya göstermektir ve bu tip bir dönüştürme amaçlanmamaktadır.

**Web Erişilebilirliği:** Web erişilebilirliği konusunda literatürde önemli sayıda çalışma yapılmıştır. Asakawa ve Takagi (2000) ile Takagi vd. (2002) web sayfalarını ekran okuyucuları için daha erişilebilir hale getirmek amacıyla web sayfalarını bloklara ayırıp yeniden düzenlemeyi amaçlamışlardır. Her bir blok için, daha önceden tanımlanmış olan başlık veya alt bilgi gibi roller manuel olarak atanmıştır. Bu bloklar ve ek açıklama olarak eklenmiş roller kullanılarak, web sayfaları dönüştürülmüştür. Dönüştürme yaparken önemli olan bloklar web sayfasının üst kısımlarına yerleştirilirken, önemsiz olan bloklar ise web sayfasının alt kısmına konulmaktadır.

Web erişilebilirliğini daha fazla desteklemek amacıyla Yeşilada vd. (2007) zengin bir rol haznesi tanımlamışlardır. Buna benzer olarak Harper vd. (2006) ile Lunn vd. (2008a, 2008b) web sayfalarını dönüştürmek için yeni bir yöntem önermişlerdir. Bu yöntemle basit bir kelime haznesi ile CSS'e fazladan ek açıklama eklenerek web sayfasının dönüştürülmesi sağlanabilmektedir. Fakat, bu çalışmalar görsel elemanların rollerinin otomatik olarak belirlenmesi ile ilgili değil, görsel elemanlara manuel olarak açıklama eklenmesi ile ilgilidir.

Ekran okuyucuları gibi yardımcı uygulamalar, web sayfalarını “Belge Nesnesi Modeli” (Document Object Model - DOM) yapısına göre işlemektedir (Borodin vd., 2007; Mahmud vd.,

2007a, 2007b). Bundan dolayı, ekran okuyucular ile web sayfasına erişen kullanıcılar tarama yaparken oldukça uzun bir süreye ihtiyaç duymaktadırlar. Bu problemi çözmek amacıyla Borodin vd. (2007) ve Mahmud vd. (2007a, 2007b) web sayfasının elemanlarını otomatik olarak belirleyebilecek, önemsiz elemanları atlayarak, daha kısa sürede ana içeriğe ulaşmayı sağlayabilecek olan geometrik kümeleme algoritmasını geliştirmişlerdir.



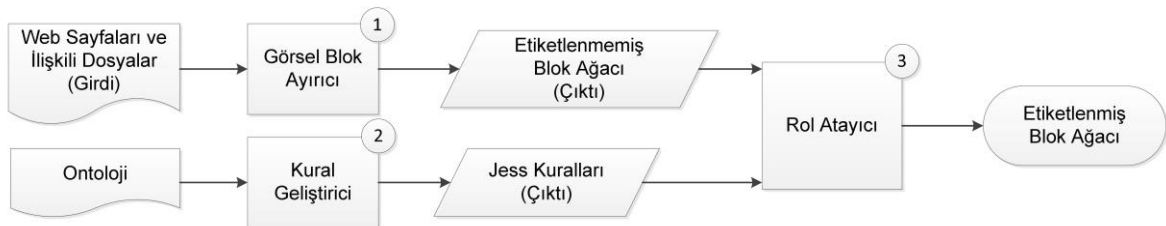
### 3. GEREÇ VE YÖNTEM

Proje önerimizde de sunduğumuz gibi, projemiz günlük olarak Dr. Yeliz Yeşilada tarafından yönetilmiş ve Manchester Üniversitesindeki WEL (Web Ergonomics Lab) laboratuvarı ile iş birliği içerisinde yürütülmüştür. Projemiz genel olarak önerilen iş-zaman çizelgesine göre takip edilmiştir. Bu iş-zaman çizelgesinde küçük kaymalar olsa da, büyük ölçüde öngörüldüğü şekilde tamamlanmıştır. Aşağıda projemizde takip edilen yöntemi ve kullanılan gereçleri projemizin üç iş paketi üzerinden özetliyoruz.

#### 3.1 Sayfadaki Görsel Elemanların Otomatik Keşfi

Bu iş paketinin esas hedefi sayfadaki görsel elemanların otomatik olarak tespit edilebilmesiydi. Bu hedefe ulaşabilmek için bu iş paketinde öngörüldüğü gibi temel olarak üç iş gerçekleştirildi: (1) Görsel elemanlarla ilgili bulgular toplandı, (2) web sayfaları otomatik olarak görsel bloklara ayrıldı ve (3) görsel bloklar ile rolleri otomatik olarak keşfedildi.

Web sayfalarını otomatik olarak tarayıp görsel elemanları çıkarabilmemiz için öncelikle görsel elemanlar hakkında bilgi toplamamız gerekmektedir. Bunun için çok detaylı bir literatür taraması yapılmıştır (bkz. 2.2. Bölüm). Yapılan bu literatür taraması, çoğu araştırmanın yalnızca belli başlı görsel elemanlar üzerinde yoğunlaşmış olduğunu ortaya çıkarmıştır. Fakat, bizim çalışmamız çok daha detaylı roller içerdiği için, literatür taraması beklenen ölçüde katkı sağlayamamıştır. Daha sonra, önceden geliştirilen çerçeve (Plessers vd., 2005; Yeşilada vd., 2004, Yeşilada vd., 2003) ile, birçok türde web sayfası ve kodları analiz edilip, kullanabileceğimiz görsel elemanlar ve rolleri ortaya çıkarılmıştır. Bunlar hakkında daha detaylı bilgi, yazılan teknik raporumuzda sunulmuştur (Yeşilada, 2011b).



Şekil 1. Sistem Mimarisi

Web sayfalarının otomatik olarak görsel bloklara ayrılabilmesi ve rollerinin otomatik keşfedilebilmesi için **Şekil 1**'deki üç parçadan oluşan sistem mimarisi oluşturuldu. Bu mimari daha sonra proje önerimizde de bahsettiğimiz gibi Eclipse Accessibility Tools Framework

(ACTF)<sup>9</sup> üzerinde uygulandı. Bu uygulama daha sonra deneysel platform olarak kullanıldı. Bu mimari üzerindeki temel parçaları şöyle anlatabiliriz.

### 3.1.1 Görsel Blok Ayırıcı

Web sayfalarını otomatik bloklara ayırmak için birçok algoritma geliştirilmiştir (Yeşilada, 2011a; Yeşilada, 2013). Bu algoritmalar farklı alanlarda farklı amaçlar için geliştirilmiştir. Bu algoritmaları sistematik olarak analiz edebilmemiz için bir çerçeve oluşturulmuştur. Bu çerçevede temel olarak şu sorulara cevap aranmıştır: Literatürde ne tür teknik yaklaşımlar, algoritmalar ve çözümler önerilmiştir? Sayfayı bloklara ayırma işlemi nerede gerçekleştirilmiştir? Otomatik bloklara ayırma işlemi için ne tür girdiler kullanılmıştır? Blokların rollerini otomatik olarak çıkarmak için bir çalışma yapılmış mıdır? Önerilen teknik, yaklaşım ve algoritmalar nasıl değerlendirilmiştir? Bu sistematik literatür taramasının sonuçları daha sonra teknik rapor olarak yazılmıştır (Yeşilada, 2011a) ve makale olarak düzenlenip, yayınlanması için gönderilmiştir (Yeşilada, 2013). Yaptığımız literatür taramasında ortaya çıkmıştır ki, en etkili algoritmalar sadece sayfanın kodunu değil, aynı zamanda görsel sunumunu da kullanan algoritmalarlardır. Literatür taramamızda yine Cai vd. (2003)'nin geliştirmiş olduğu VIPS (Vision Based Segmentation) algoritmasının, diğer algoritmalara göre çok daha etkili olduğu gözlemlenmiştir. Bu sebeple, bu projede VIPS algoritmasının temel alınmasına karar verilmiştir. VIPS algoritmasının ACTF entegrasyonu üzerinde çalışırken, bu algoritmanın aslında sadece HTML4<sup>10</sup>'ü göz önünde bulundurduğu ve birçok yönden yetersiz kaldığı ortaya çıkmıştır. Bu nedenle, ACTF ile entegrasyon çalışmalarını yaparken öncelikle bu algoritmayı iyileştirme ve geliştirme üzerinde durulmuştur. Bu konu üzerinde yaptığımız çalışmalar teknik rapor olarak (Akpınar ve Yeşilada, 2011) ve bir çalıştayda tebliğ olarak sunulmuştur (Akpınar ve Yeşilada, 2013).

Özet olarak VIPS algoritması üç parçadan oluşmaktadır: Blok çıkarma, blok ayırma ve genel sayfa yapısının oluşturulması. Bunlardan en önemli parça blok çıkarma kısmıdır. Blok çıkarma işlemi yapılırken HTML4 etiketleri, etiketlerin yapısı ve görsel özellikleri kullanılmaktadır. Hem VIPS'in kullandığı etiketlerde hem de görsel özelliklerde ciddi sorunların olduğu ortaya çıkmıştır (Akpınar ve Yeşilada, 2011; Akpınar ve Yeşilada, 2013). Bu sorunları aşmak amacıyla HTML5<sup>11</sup> için gerekli değişiklikler yapılmıştır. VIPS algoritmasının önerdiği ve kullandığı kurallar geliştirilip, yenileri eklenmiştir. Bu iyileştirme ve geliştirme çalışmaları tamamlandıktan sonra, VIPS algoritmasının ACTF ile entegrasyon çalışmaları tamamlanmıştır. Bu çalışmalar daha sonra açık kaynak kodlu Eclipse platformu

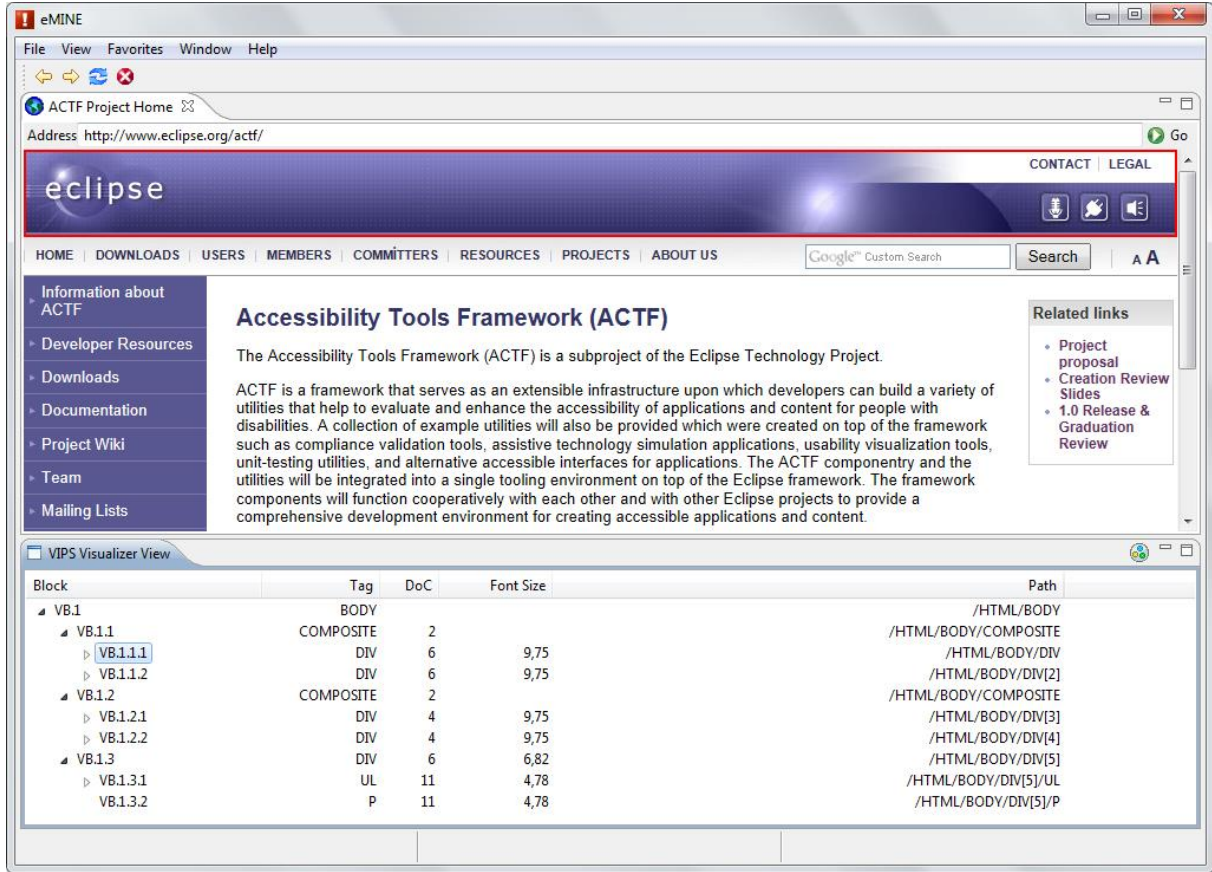
<sup>9</sup> ACTF, <http://www.eclipse.org/actf/>

<sup>10</sup> HTML4, <http://www.w3.org/TR/html4/>

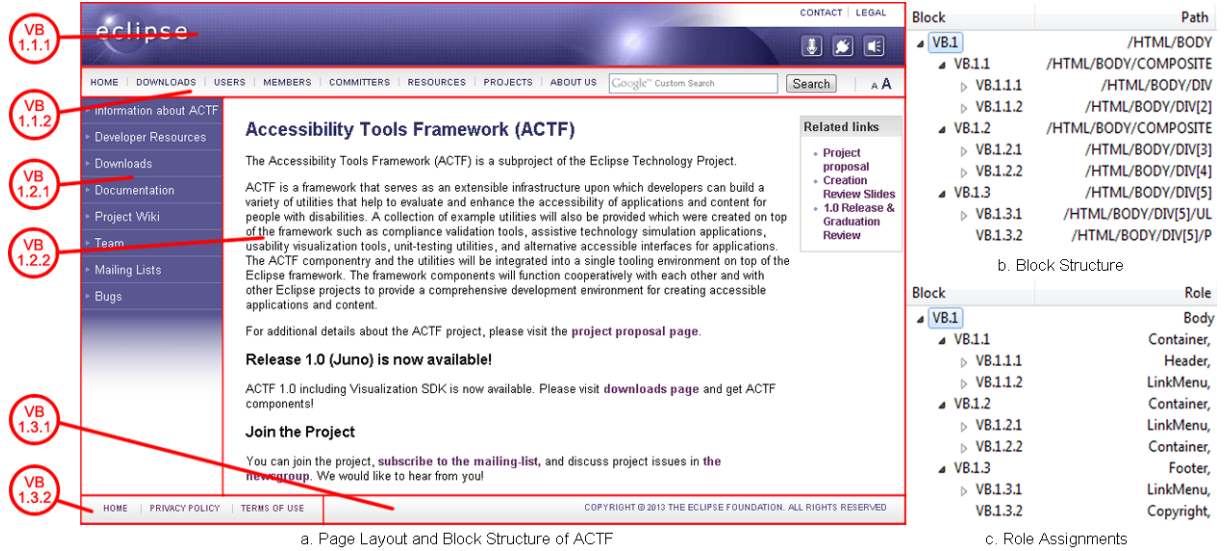
<sup>11</sup> HTML5, <http://www.w3.org/TR/html5/>

üzerinden yayınlanmıştır. Eclipse platformu çerçevesinde, ACTF yazılım paketlerini yükleyen kullanıcılar, eMine Projesi kapsamında geliştirilmiş olan bloklara ayırma algoritmasını da kullanabilmektedirler. Geliştirilen bloklara ayırma algoritması ile ilgili detaylara şuradan erişilebilir: <http://www.eclipse.org/actf/downloads/tools/eMine/build.php>.

Görsel blok ayırıcı kısmı bir web sayfasını ve ilişkili dosyaları girdi olarak alıp, etiketlenmemiş bir blok ağacı ortaya çıkarmaktadır. ACTF ve eMine Blok Ayırma Algoritması'nın entegrasyonu **Şekil 2**'de genel hatlarıyla gösterilmektedir. Geliştirilen algoritmayı yükledikten sonra, kullanıcı **Şekil 2**'de gösterilen yapıya sahip olur. Yaratılan program bir web tarayıcısı gibi çalışmaktadır. Kullanıcı herhangi bir web sayfasının adresini girdikten sonra o sayfa program tarafından otomatik olarak indirilip, bloklara bölünmektedir. **Şekil 2**'nin alt tarafında görüldüğü gibi, daha sonra sayfa içerisindeki tüm görsel blokların yapısını gösteren bir ağaç yapısı çıkarılmaktadır. Bu ağaç yapısı kullanılarak sayfa içindeki iç içe çıkarılan bloklara erişilebilir. Kullanıcı bu ağaç yapısındaki herhangi bir bloku seçerse, o zaman da sayfa içerisindeki o blok gösterilir. Örneğin **Şekil 2**'de, kullanıcı blok VB1.1.1'i seçtiği zaman, o blokun ekranda nasıl gösterildiği anlatılmaktadır. Ayrıca **Şekil 3** ve **Şekil 4**'te, otomatik olarak çıkarılan blok ağaçlarına daha detaylı örnekler verilmiştir.



Şekil 2. eMine VIPS ACTF entegrasyonu



Şekil 3. ACTF ana sayfası ve otomatik yaratılan blok ağacı



biçimsel tanımını vermektedir. Bu ontolojideki bilgiler daha sonra kurallar geliştirmek için kullanılmıştır. Bu kurallar standart kütüphaneleri olan ve çok yaygın kural uygulama yapısı olarak kullanılan, JESS Kural Dili kullanılarak yaratılmıştır (Friedman-Hill, 2008). Örneğin, **Şekil 6** sayfanın üstlüğü (Header) için yaratılan kuralı göstermektedir.

```
<owl:Class rdf:about="http://emine.ncc.metu.edu.tr/ontology/emine#Header">
  <rdfs:comment><![CDATA[Is typically printed at the top of a page and
    includes, for example, a company logo, the title of the page, list of
    links and sometimes a search engine.]]></rdfs:comment>
  <rdfs:label>Header</rdfs:label>
  <rdfs:subClassOf
    rdf:resource="http://emine.ncc.metu.edu.tr/ontology/emine#chunk"/>
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class
      rdf:about="http://emine.ncc.metu.edu.tr/ontology/emine#chunk"/>
    <owl:Restriction>
      <owl:onProperty rdf:resource=
        "http://emine.ncc.metu.edu.tr/ontology/emine#must_have_tag"/>
      <owl:hasValue
        rdf:resource="http://emine.ncc.metu.edu.tr/ontology/emine#Header"/>
    </owl:Restriction>
    <owl:Restriction>
      <owl:onProperty
        rdf:resource="http://emine.ncc.metu.edu.tr/ontology/emine#has_tag"/>
      <owl:hasValue
        rdf:resource="http://emine.ncc.metu.edu.tr/ontology/emine#Div"/>
    </owl:Restriction>
    <owl:Restriction>
      <owl:onProperty
        rdf:resource="http://emine.ncc.metu.edu.tr/ontology/emine#has_id"/>
      <owl:allValuesFrom>
        <owl:Class>
          <owl:oneOf rdf:parseType="Collection">
            <owl:Thing rdf:about="http://www.w3.org/2001/XMLSchema#header"/>
            <owl:Thing rdf:about="http://www.w3.org/2001/XMLSchema#hdr"/>
            <owl:Thing rdf:about="http://www.w3.org/2001/XMLSchema#hd"/>
            <owl:Thing rdf:about="http://www.w3.org/2001/XMLSchema#heading"/>
          </owl:oneOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
    ...
  </owl:intersectionOf>
</owl:Class>
```

**Şekil 5.** eMine Ontolojisi'ndeki Header rolü tanımı

```
(defrule Header00
  (block (must_have_tag $? /*.*header.*/ $?))
  => (bind ?*Header* (+ "8" ?*Header*)))
```

**Şekil 6.** Header rolü için yaratılan JESS kuralı

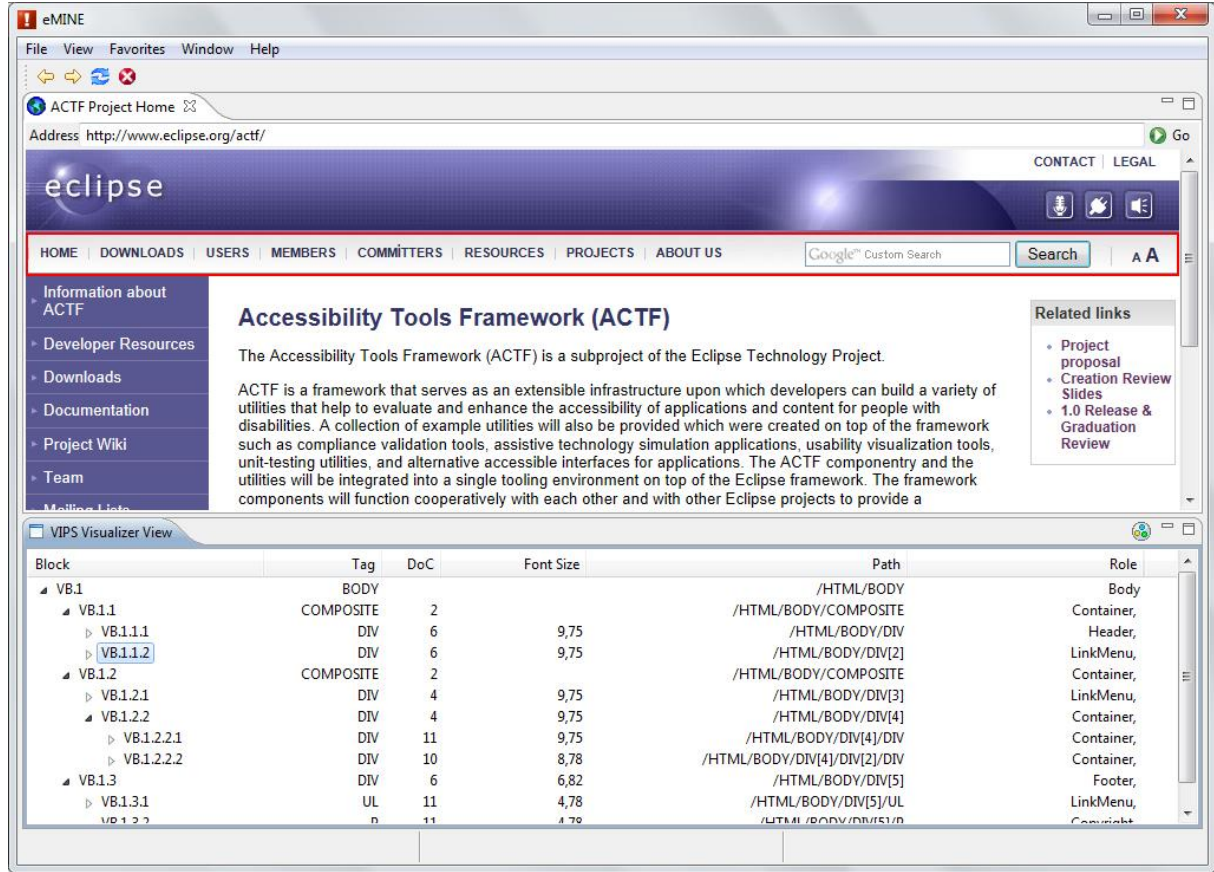
### 3.1.3 Rol Atayıcı

Bir önceki kısımda yaratılmış olan kurallar ve ilk kısımda yaratılmış olan blok ağacı, bu son kısımda girdi olarak kullanılmıştır. Genel olarak, otomatik şekilde yaratılan blok ağacındaki her blok bir birey olarak kabul edilir. Bu bireyler sistemde doğrular olarak kabul edilerek, bloklar üzerinde JESS kuralları uygulanmıştır. Bu kurallardan yola çıkarak, blokların birden fazla role sahip olması muhtemeldir. Bunlar arasından en yüksek olasılığı olan rol, verilen bloka atanmaktadır. Bu kısmın sonunda, ilk kısımda yaratılan etiketlenmemiş ağaç, etiketlenmiş olarak çıkmaktadır. **Şekil 7** bu kısmın ACTF üzerindeki entegrasyonunu göstermektedir. **Şekil 2** ile kıyaslandığında, **Şekil 7**'de görsel blok ağacında yer alan bloklara roller atanmış olduğu görülür. Aynı şekilde, **Şekil 3** hem etiketlenmiş, hem de etiketlenmemiş ağacı göstermektedir.

### 3.1.4 Teknik ve Kullanıcı Bazlı Deneyler ve Sonuçları

Bu çalışmayı test edip geçerliliğini denetlemek için teknik ve kullanıcı bazlı olmak üzere iki farklı deney düzenlenmiştir. Teknik deneyler sistemin teknik olarak performanslarını incelemiştir; kullanıcı deneyleri ise sistemin çıkardığı sonuçların doğruluğunu incelemiştir. Kullanıcılar ile yapılan deneyler, web ortamında tasarlanmış ve kullanıcıların web ortamında katılabileceği şekilde düzenlenmiştir. Bu deneyin tümünü proje ekibi tasarlayıp, yaratmıştır. Web üzerindeki bu deneye şu adresten ulaşılabilir: <http://emine.ncc.metu.edu.tr/eval/survey/>. Ayrıca yapılan deney ve sonuçları teknik rapor halinde de yayınlanmıştır (Akpınar ve Yeşilada, 2013b), tebliğ olarak da uluslararası bir kongrede sunulmuştur (Akpınar ve Yeşilada, 2013c).





Şekil 7. eMine ACTF rol atama entegrasyonu

Teknik deney ve sonuçlarını şöyle özetleyebiliriz:

**Amacı:** Görsel elemanların keşfi ve rol atamasının teknik olarak değerlendirilmesinin amacı, sistemin hafıza kullanımı ve cevap süresi gibi konularda performansını değerlendirmektir.

**Kullanılan Yöntem:** Performans testleri kapsamında 3 karmaşıklık grubuna ait 10'ar adet sayfa içeren toplam 30 adet web sayfası için teknik değerlendirme testleri uygulanmıştır. Bu testlerin sonuçları hafıza kullanımı ve geçen süre şeklinde her bir sayfa için hesaplanarak, karmaşıklık grupları için bu değerlerin ortalaması hesaplanmıştır.

**Sonuçlar:** Görsel elemanların keşfi ve rol ataması algoritmaları üzerinde yapılan performans testlerinin sonuçları **Tablo 1** ve **Tablo 2**'de verilmiştir. **Tablo 1** incelendiğinde, DOM yapısının çıkarıldığı bölümün diğerlerine oranla daha fazla sürede gerçekleştiği görülmektedir. Bunun sebebi de, web sayfalarının bu bölümde okunması ve programın kullanımına uygun hale getirilmesidir. Bu bölümde yapılacak geliştirmelerle sistem performansı artırılabilir. Blok sayısı sayfanın karmaşıklığı ile ilişkilidir. Yüksek karmaşıklık seviyesine sahip sayfaların blok sayıları, düşük karmaşıklık seviyesine sahip sayfalara göre



çok daha fazladır. Dolayısıyla, sayfaların karmaşıklık seviyesi, blok sayısı, görsel elemanların keşfi için geçen süre ve hafıza kullanımıyla doğru orantılıdır.

Tablo 1. Görsel elemanların keşfi algoritması performans sonuçları

|                                       | Karmaşıklık        | Yüksek      | Orta        | Düşük      |
|---------------------------------------|--------------------|-------------|-------------|------------|
| <b>DOM Yapısının Çıkarılması</b>      | <b>Hafıza Kul.</b> | 20659.53 KB | 16293.37 KB | 9370.56 KB |
|                                       | <b>Süre</b>        | 5069 ms     | 2120 ms     | 730 ms     |
| <b>Görsel Blokların Oluşturulması</b> | <b>Hafıza Kul.</b> | 21610.17 KB | 14351.68 KB | 9535.14 KB |
|                                       | <b>Süre</b>        | 30 ms       | 19 ms       | 4 ms       |
| <b>İçerik Yapısının Oluşturulması</b> | <b>Hafıza Kul.</b> | 21519.97 KB | 13711.88 KB | 9699.04 KB |
|                                       | <b>Süre</b>        | 93 ms       | 53 ms       | 21 ms      |
| <b>Toplam Eleman Sayısı</b>           |                    | 3,811       | 1,888       | 624        |
| <b>Maksimum Eleman Derinliği</b>      |                    | 22          | 21          | 16         |
| <b>Kontrol Edilen Eleman Sayısı</b>   |                    | 2,609       | 1,328       | 395        |
| <b>Geçersiz Eleman Sayısı</b>         |                    | 98          | 84          | 44         |
| <b>Görünmez Eleman Sayısı</b>         |                    | 69          | 35          | 26         |
| <b>Blok Sayısı</b>                    |                    | 569         | 237         | 65         |
| <b>Blok Derinliği</b>                 |                    | 13          | 11          | 7          |

Rol ataması için blok bazında geçen ortalama süre, her karmaşıklık seviyesi için hemen hemen yakın değerlere sahiptir. Dolayısıyla, tüm sayfa için geçen süre, blok sayısı ile doğru orantılıdır. Tüm karmaşıklık grupları için sayfa başına kullanılan hafıza miktarı da eşit şekilde gözlenmiştir. Bunun sebebi, rol ataması işleminin ortak kaynakları kullanarak gerçekleşmesidir. Bu yüzden, blok sayısı ile blok başına kullanılan hafıza miktarı ters orantılıdır.

Tablo 2. Rol belirleme algoritması performans sonuçları

| Karmaşıklık Grubu | Toplam Hafıza | Toplam Süre | Blok Başına Hafıza | Blok Başına Geçen Süre | Blok Sayısı |
|-------------------|---------------|-------------|--------------------|------------------------|-------------|
| <b>Düşük</b>      | 8369 KB       | 6576 ms     | 244.29 KB          | 102.29 ms              | 65          |
| <b>Orta</b>       | 7013 KB       | 23799 ms    | 36.44 KB           | 102.12 ms              | 237         |
| <b>Yüksek</b>     | 9165 KB       | 54837 ms    | 34.28 KB           | 101.95 ms              | 569         |
| <b>Genel</b>      | 8176 KB       | 29157 ms    | 100.2 KB           | 102.11 ms              | 298         |

Kullanıcı bazlı yapılan çalışmayı ve sonuçlarını genel olarak şöyle özetleyebiliriz:

*Amacı:* Genel olarak bu deneyin amacı yaptığımız çalışmanın, yani görsel elemanların otomatik keşfi ve rol atamasının, ne kadar iyi çalıştığını ortaya koymaktır.

*Kullanılan Yöntem:* Kullanıcılara deney sayfasına geldikleri zaman ilk olarak deney hakkında bilgi verilmiştir. Sonraki aşamada, kullanıcılardan cinsiyet, eğitim düzeyi gibi kişisel bilgilerin yer aldığı bir form doldurmaları istenmiştir. Daha sonra, kullanıcılara görsel elemanların keşfi algoritmasının otomatik olarak bloklara ayırmış olduğu sayfalar gösterilmiş ve onlardan sistemin bu sayfaları otomatik ayırma konusundaki başarısını değerlendirmeleri istenmiştir. Son kısımda da, kullanıcılardan otomatik olarak ayrılan görsel elemanlara rol atamaları istenmiştir. Bu atanan roller daha sonra geliştirilen sistemin otomatik olarak atadığı roller ile karşılaştırılmıştır.

*Kullanılan Sayfalar:* Bu deneyde toplam 30 farklı web sayfası kullanılmıştır. Bu 30 sayfa, Alexa Top 100<sup>14</sup> Listesi içinden, karmaşıklık yapılarına göre üç grup (karmaşık, orta ve basit) şeklinde seçilmiştir. Bu sayfaların karmaşıklık yapıları Michailidou (2010) tarafından önerilen teknik kullanılarak değerlendirilmiştir.

*Deney Yönetimi:* Bu deneyler web ortamında, yaklaşık olarak altı ay içerisinde tamamlanmıştır. Bu süre içerisinde 253 kişi deneye katılmış, bunlar arasından yalnızca 34 kişi en az üç sayfa için deneyleri tamamlamıştır. Katılımcılar elektronik ortamda davet edilmiştir: örneğin CHI duyurular, SIGWEB üye listesi gibi listeleri ve Bilgisayar Mühendisleri Odası üye listelerine davet gönderilmiştir.

*Katılımcılar:* Çalışmamıza toplam 34 katılımcı katılmıştır. Bunlardan 14'ü kadın, 20'si erkektir. Sekiz katılımcı 18-24 yaş arasında, 11 katılımcı 25-54 yaş arasında ve 2 kişi de 55 yaşından büyüktür. Tüm katılımcılar İnterneti günlük olarak kullandıklarını belirtmiştir. Kullanıcılar hakkındaki detaylı bilgilere teknik raporumuzdan erişilebilir (Akpınar ve Yeşilada, 2013b).

*Bilimsel Sorular:* Bu deney içerisinde genel olarak şu sorulara cevap aranmıştır: 1. Sayfaları ne kadar başarılı şekilde bloklara ayırabiliyoruz? İlk beş seviye içerisinde en çok tercih edilen bloklara ayırma seviyesi nedir? Kullanıcı profili ve sayfa karmaşıklığının sonuçlara etkisi nedir? Algoritmamız otomatik rol atamada ne kadar başarılıdır?

*Sonuçlar:* Görsel elemanların keşfi algoritmasının değerlendirme sonuçları **Tablo 3**'te verilmiştir. Sonuçlar incelendiğinde genel olarak seviye arttıkça sistemin başarısı da artmaktadır. 1. ve 2. seviyeler için sistem başarısı ortalamanın altında kalmıştır. 3. ve üzeri seviyeler için ise sistem başarısı ortalamanın üzerindedir.

---

<sup>14</sup>

<http://www.alex.com/>

Tablo 3. Görsel elemanların keşfi algoritmasının başarı sonuçları

|           | Düşük Karmaşıklık |          |                | Orta Karmaşıklık |          |                | Yüksek Karmaşıklık |          |                |
|-----------|-------------------|----------|----------------|------------------|----------|----------------|--------------------|----------|----------------|
|           | N                 | Ortalama | Standart Sapma | N                | Ortalama | Standart Sapma | N                  | Ortalama | Standart Sapma |
| 1. Seviye | 37                | 2.405    | 1.2793         | 66               | 2.349    | 1.2215         | 68                 | 2.25     | 1.2504         |
| 2. Seviye | 37                | 2.945    | 1.1772         | 66               | 2.833    | 1.158          | 68                 | 2.677    | 1.0851         |
| 3. Seviye | 37                | 3.513    | 1.044          | 66               | 3.303    | 0.9762         | 68                 | 3.177    | 0.9133         |
| 4. Seviye | 37                | 3.405    | 0.9267         | 66               | 3.606    | 1.1077         | 68                 | 3.471    | 1.0144         |
| 5. Seviye | 37                | 3.621    | 1.0633         | 66               | 3.576    | 1.2286         | 68                 | 3.706    | 1.1977         |

Katılımcıların sayfa yapısındaki ilk beş seviye arasından yaptıkları tercihlerin sonuçları **Tablo 4**'te verilmiştir. Buna göre, katılımcılar görsel elemanların keşfi için en iyi seviye olarak 5. seviyeyi, en kötü seviye olarak da 1. seviyeyi tercih etmiştir. 1. seviyede sadece sayfanın ana elemanları yer alırken, seviye arttıkça görsel eleman keşfi daha da detaylanmaktadır. Bu sonuçtan yola çıkarak, katılımcıların detaylı bloklara ayırma sonuçlarını görsel eleman keşfi için daha uygun buldukları söylenebilir.

Tablo 4. Katılımcıların en iyi seviye tercihleri

|           | En İyi Seviye |            |             |             | En Kötü Seviye |
|-----------|---------------|------------|-------------|-------------|----------------|
| 1. Seviye | 12 (7,0%)     | 14 (8,2%)  | 13 (7,6%)   | 18 (10,5%)  | 114 (66,7%)    |
| 2. Seviye | 23 (13,5%)    | 15 (8,8%)  | 16 (9,4%)   | 108 (63,2%) | 9 (5,3%)       |
| 3. Seviye | 22 (12,9%)    | 23 (13,5%) | 112 (65,5%) | 10 (5,8%)   | 4 (2,3%)       |
| 4. Seviye | 31 (18,1%)    | 92 (53,8%) | 19 (11,1%)  | 22 (12,9%)  | 7 (4,1%)       |
| 5. Seviye | 83 (48,5%)    | 27 (15,8%) | 11 (6,4%)   | 13 (7,6%)   | 37 (21,6%)     |

Sayfaların karmaşıklık değerlerinin sonuçlar üzerindeki etkilerini belirlemek için de bir dizi istatistiksel çalışma yapılmıştır. Bu çalışmalara göre, sayfaların karmaşıklık değeri görsel elemanların keşfi algoritmasının başarı değerlendirmesinde ve katılımcıların en iyi seviye tercihlerinde geçerli bir kriter değildir.

Benzer şekilde, katılımcıların kişisel özelliklerinin değerlendirme sonuçları üzerine etkileri araştırılmıştır. Yapılan testler, katılımcıların görsel eleman keşfini değerlendirmede verdikleri cevapların yaşları, eğitim seviyeleri, cinsiyetleri, tecrübeleri ve web tasarımı ile olan ilgileri ile ilişkili olduğu ortaya çıkmıştır. Daha yaşlı katılımcılar orta seviye için daha başarılı sonuçlar verirken, genç katılımcılar yüksek seviye için daha başarılı sonuçlar vermiştir. Benzer şekilde erkek katılımcılar orta ve düşük seviye için daha yüksek sonuçlar verirken, kadın katılımcılar yüksek seviyeler için daha yüksek sonuçlar vermiştir. Web tasarımı konusunda tecrübeli

katılımcılar, diğer katılımcılara göre daha düşük sonuçlar vermiştir. Web üzerine çalışmış olan katılımcılar, diğer katılımcılara göre daha düşük sonuçlar verirken, web üzerine eğitim almış katılımcılar bütün seviyelerde yakın sonuçlar vermiştir. Bu durum, bu katılımcıların bloklara ayırma işlemini seviye bazında değil, bir bütün olarak ele almış olmasıyla açıklanabilir. Katılımcıların en iyi seviye tercihleri de katılımcının kişisel özellikleriyle ilişkilidir. Katılımcıların eğitim seviyeleri, yaşları ve web tasarımıyla olan ilişkileri tercihlerini etkilemektedir. Katılımcıların cinsiyetleri ve web konusundaki tecrübeleri, en iyi seviye tercihlerini etkilememektedir.

Rol atama algoritmasının kullanıcı değerlendirmesi sonuçları **Tablo 5**'te verilmiştir. Bu değerlendirme sürecinde, ilk olarak sistem tarafından atanan rollerle katılımcı cevapları doğrudan karşılaştırılmıştır. Fakat katılımcıların verdiği cevaplar incelendiğinde bu cevapların doğruluğunun da değerlendirilmesi gerektiği sonucuna varılmıştır. Bu sebeple, sistemin atadığı roller ve bu rollerin katılımcı cevaplarıyla olan ilişkisi elle kontrol edilmiş ve bazı özel durumlarda katılımcı cevabıyla sistemin atadığı roller arasındaki küçük farklılıklar göz ardı edilmiştir. Örneğin, Header ve Title gibi katılımcının aradaki farkı ayırt edemeyebileceği roller için her iki cevap da doğru kabul edilmiştir. Buna göre kabul edilen sistem başarısı %80'dir. Katılımcılardan alınan geri bildirimler kullanılarak bilgi bankasını güncellemek ve daha iyi sonuçlar elde etmek de mümkündür.

Tablo 5. Rol atama algoritmasının başarı sonuçları

| Karmaşıklık Grubu | Sistem-Uzman Değerlendirmesi | Kabul Edilen Değerlendirme |
|-------------------|------------------------------|----------------------------|
| Düşük             | 79.82                        | 73.68                      |
| Orta              | 88.28                        | 79.77                      |
| Yüksek            | 88.47                        | 85.53                      |
| Genel             | 86.83                        | 80.82                      |

### 3.2 eMine Algoritması'nın Geliştirilmesi ve Onaylanması

Bu iş paketinin temel amacı sayfa içerisinde otomatik olarak çıkarılan blokların ve göz tarama çalışmalarında toplanan verilerin örtüştürülmesidir. Bu örtüştürme bir sayfayı daha kolay kullanılabilir şekle dönüştürmek için kullanılacaktır. Bu hedefe ulaşabilmek için, bu iş paketi iki görevden oluşmaktadır: (1) eMine Algoritması'nı geliştirmek ve (2) algoritmayı deneyler yaparak test etmek. Bu iş paketine ilk olarak literatür taraması ve elimizdeki verilerin düzenlenip test edilmesi ile başlanmıştır. Daha sonra eMine Algoritması'nın geliştirilmesi üzerinde çalışılmış ve son olarak da hem Manchester Üniversitesi hem de ODTÜ Kuzey

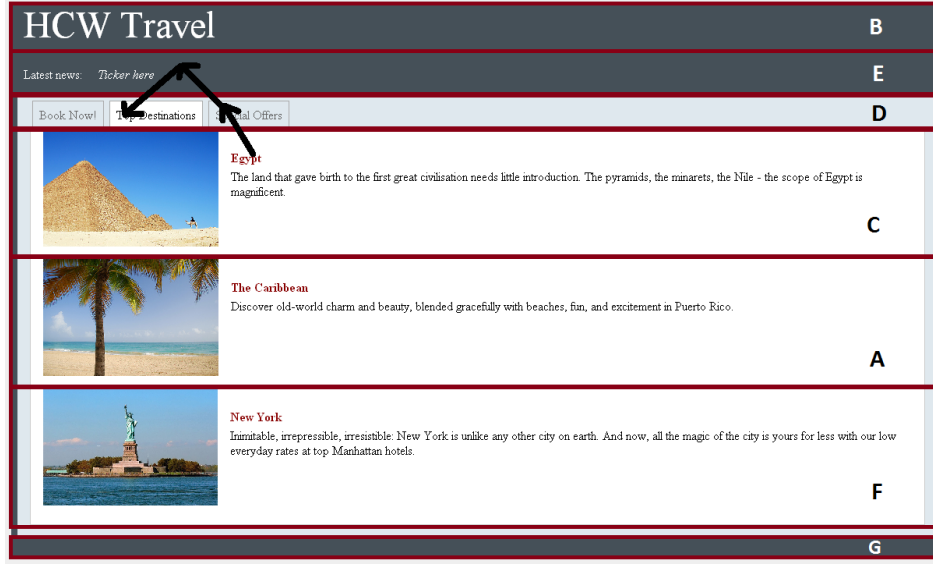
Kıbrıs Kampüsü'nde yapılan göz izleme çalışmaları ile onaylanması yapılmıştır. Bu paket içerisinde yapılan çalışmaları aşağıdaki üç başlık altında özetleyebiliriz.

### **3.2.1 Algoritma Analizleri**

İlk hedefimiz, algoritmanın nasıl geliştirilebileceği ile ilgili birtakım temel kurallar belirlemek olmuştur. Bu kurallar belirlenirken, şu ana kadar kullanılmış temel algoritmalar test edilmiş ve bu algoritmaların sınırlamaları göz önünde bulundurulmuştur. Yapılan analizlerin sonuçları hem teknik rapor (Eraslan vd., 2013a) hem de tebliğ olarak sunulmuştur (Eraslan vd., 2013b).

Başka bir çalışmada kullanılmak üzere yapılan bir göz izleme çalışmasının verileri (Brown vd., 2012), tarama güzergâhlarını incelemek için kullanılmış olan algoritmaları test etmek üzere kullanılmıştır. Bu çalışma için “HCW Travel” isminde bir web sayfası oluşturulmuş (**Şekil 8**) ve kullanıcılara bu web sayfasında yapmaları için çeşitli görevler verilmiştir. Testler için sadece ana sayfa kullanılmıştır. Ana sayfa için kullanıcıların yapması gereken görev ise web sayfasındaki “Latest News” bölümünü okumaları ve “Special Offers” linkine tıklamalarıdır.

Kullanılan bu sayfa, daha sonra ACTF üzerinde geliştirilen görsel elemanları otomatik bulma tekniği ile görsel elemanlarına bölünmüştür. Daha sonra, bu görsel elemanlar mevcut göz izleme verileri ile örtüştürölüp, her kullanıcı için bir göz izleme güzergâhı oluşturulmuştur. Örneğin bir kullanıcı için oluşturulan güzergâh şöyle olmuştur: “CACDADCBECD CB”. Bu tarama güzergâhları daha sonra hep birlikte kullanılarak, mevcut algoritmalar analiz edilmiştir.



Şekil 8. Görsel elemanlara ayrılmış örnek sayfa

Yapılan analiz sonuçları şöyle özetlenebilir:

- String-edit algoritması ve çeşitleri (örneğin, değiştirme matris kullanımı) iki tarama güzergâhı arasındaki farklılığı hesaplayabilmesine rağmen ortak bir güzergâh çıkarma açısından tek başına kullanılması uygun bir yöntem değildir (Levenshtein, 1966).
- Transition Matris yöntemi ile ilgi alanları arasındaki geçişleri olasılıklarla görmek mümkün olduğu halde, bu yöntem ortak bir güzergâh çıkarma açısından çok uygun bir yöntem değildir. Örneğin, yukarıda belirtildiği gibi bir ortak güzergâh çıkarılmaya çalışıldığı zaman, hangi ilgi alanının başlangıç noktası olması gerektiği bilinmemektedir. Ayrıca en yüksek olasılıklarla ortak güzergâh çıkarılmaya çalışıldığında, diğer olasılıklar yok sayılmaktadır. Bununla birlikte, başlangıç noktası bilinmediği gibi bitiş noktası da belirlenememektedir (West vd., 2006).
- “eyePatterns” uygulaması ortak tarama güzergâhlarını bulurken aralara giren ilgi alanlarına tolerans göstermemektedir. Örneğin, “DCDED” ortak güzergâhının **Tablo 6**’daki güzergâhlardan sadece 3 tanesinde (3., 7. ve 9.) olduğu gözlemlenmektedir. Aslında, 6. tarama güzergâhı ele alındığında (“(DCD)CAB(ED)CD”), bu güzergâhın içerisinde de “DCDED”nin bulunduğu görünmekte; fakat arada “CAB” olduğundan dolayı “DCDED” bulunamamaktadır. Bu sebeple, ortak tarama güzergâhı bulma açısından oldukça indirgemeci ve yüzeysel bir yaklaşımdır (Hembrooke vd., 2006).
- Dotplots algoritmasının da oldukça indirgemeci ve yüzeysel bir yaklaşım olduğu gözlemlenmiştir. Dotplots algoritması, kullanmış olduğu istatistiksel modeller sayesinde “eyePatterns” uygulamasının ortak tarama güzergâhı bulma özelliğine göre

daha toleranslıdır. Bu da ortak tarama güzergâhlarını bulurken istatistiksel modellemelerin kullanılabileceğini göstermektedir (Goldberg ve Helfman, 2010). eMine Algoritması bu kurallar çerçevesinde geliştirilmiştir.

### 3.2.2 eMine Algoritması'nın Geliştirilmesi ve İlk Testler

eMine Algoritması için öncelikle yukarıda bahsedilen kurallar göz önünde bulundurulmuştur. eMine Algoritması'nın geliştirilmesi, proje önerisinde de anlatıldığı gibi yinelemeli bir yöntem ile sürdürülmüştür. Algoritma şöyle özetlenebilir:

eMine Algoritması n kadar tarama güzergâhı almakta ve sonuç olarak bir tane ortak güzergâh oluşturmaktadır. Verilen güzergâhlarda yalnız bir adet tarama güzergâhı verildiği takdirde ( $n=1$ ), sadece bu güzergâhı vermektedir. Eğer birden fazla güzergâh varsa ( $n>1$ ), o zaman da verilen güzergâhlar arasında en çok benzeyen iki tane güzergâhı “Levenshtein Distance” yöntemini kullanarak ortaya çıkarmaktadır (Levenshtein, 1966; Rayner, 1998). Daha sonra, bu iki güzergâh mevcut listeden çıkarılıp, yerine ortak olan güzergâh koyulmaktadır. Ortak güzergâhı ise “Longest Common Subsequence (LCS)” yöntemi uygulanarak çıkarmaktadır (Chiang, 2009). Bu süreci tek bir güzergâh kalana kadar devam ettirmektedir. **Şekil 9**, diğer algoritmaları analiz etmek için kullandığımız güzergâh verilerine, bu algoritmanın nasıl uygulandığını göstermektedir. Algoritma 10 adet güzergâh ile başlatılmıştır. Bu 10 güzergâh arasından en çok benzeyen S2 ve S4 için LCS yöntemi uygulanıp ortak olan S24 çıkarılmıştır. Daha sonra, S2 ve S4 var olan güzergâh listesinden çıkarılıp yerine S24 eklenmiştir. Bu süreç tek bir güzergâh kalana kadar devam etmiştir. **Şekil 10** yaratılan bu algoritmayı detaylı olarak göstermektedir.





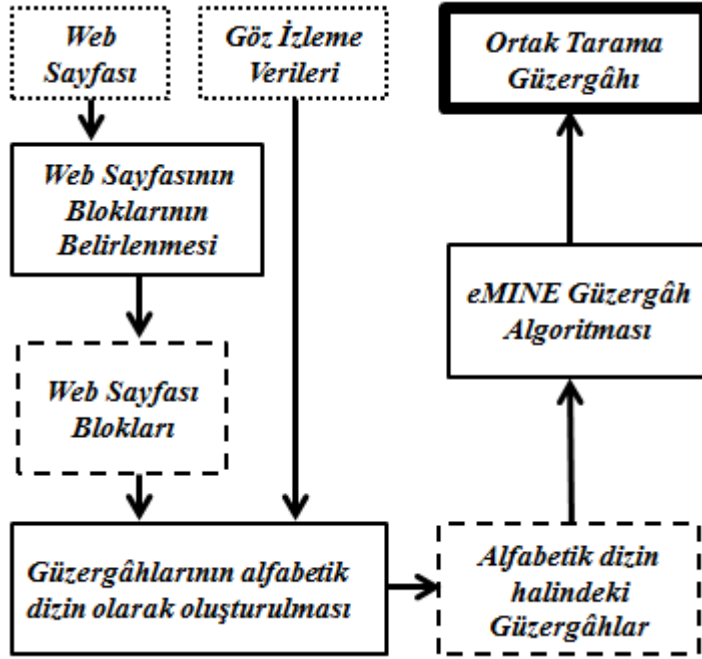
entegrasyon için **Şekil 11**'deki sistem mimarisi yaratılmıştır. Bu mimaride web sayfası ve o sayfa ile ilgili göz izleme verileri olmak üzere iki adet girdi vardır. Diğer fonksiyonel parçaları da şu şekilde özetlenebilir:

- *Web sayfasının bloklarının belirlenmesi*: Açık bir mimari olabilmesi için, bu kısımda farklı tekniklerin kullanılabilmesi sağlanılmıştır. Bu çalışmada, bir önceki iş paketinde geliştirilmiş olan bloklara ayırma algoritması kullanılarak, sayfa otomatik olarak bloklara ayrılmıştır (Akpınar ve Yeşilada, 2012; Akpınar ve Yeşilada, 2013a; Cai, 2003).
- *Güzergâhların alfabetik dizin olarak oluşturulması*: Gelen göz izleme verileri (özet olarak bu veriler kullanıcının baktığı ekran üzerindeki koordinatlar ve harcanan süreden oluşmaktadır) daha sonra sayfadaki bloklar ile örtüştürülmüş ve alfabetik dizin halinde otomatik olarak oluşturulmuştur. **Tablo 6** yaratılan örnek dizinleri göstermektedir.

Tablo 6. Örnek güzergâhlar

|     | Güzergâh                               |
|-----|----------------------------------------|
| S1  | ABCBDACACBEBDEBE                       |
| S2  | CAEBEBDEBEBEDEDCECEDECACDC             |
| S3  | CABFACDCDEDAEDFDAEDEDDEDEDACDCDEDC     |
| S4  | CACDCDCACFDEBECACDEBDEBEBEBEBECEDECECD |
| S5  | DACADEDDEDEDDEDC                       |
| S6  | DCDCABEDCD                             |
| S7  | BDCEBCDCDCDEDEDCD                      |
| S8  | CACDECECD                              |
| S9  | BCECDCDCDEDECDC                        |
| S10 | CACDADCBECDCB                          |

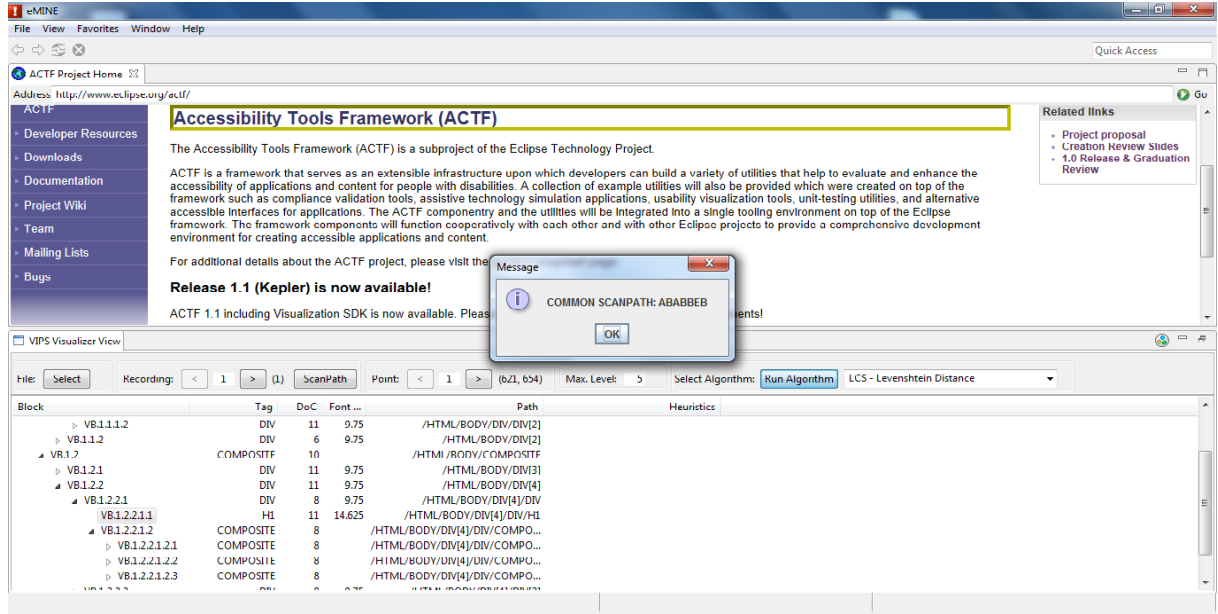
- *eMine Güzergâh Algoritması*: Alfabetik dizin şeklinde yaratılan güzergâhlar daha sonra eMine Algoritması'na girdi olarak kullanılmıştır. **Şekil 9** ve **Şekil 10** daha sonra bu algoritmanın, algoritma tarafından yaratılan ve **Tablo 6**'da verilen alfabetik dizinlere nasıl uygulandığını göstermektedir.



Şekil 11. eMine Algoritması ACTF entegrasyon mimarisi

Bu mimari daha sonra ACTF platformu üzerinde uygulanmıştır. Kullanıcı herhangi bir tarayıcıdaki gibi öncelikle web sayfasının adresini adres çubuğuna girip, o sayfayı yükleyebilmektedir. **Şekil 12'**de uygulama ve örnek sayfa gösterilmektedir. Sayfa yüklendikten sonra kullanıcı güzerğâh verilerini yükleyebilmekte ve sayfanın otomatik olarak blok ağacını çıkarabilmektedir. Yüklenen güzerğâh verileri, sistem üzerinde otomatik olarak blok ağacı ile örtüştürülmekte ve güzerğâhlar alfabetik dizin şeklinde yaratılmaktadır. Kullanıcı daha sonra eMine Algoritması'nı çalıştırarak ortak güzerğâh çıkarabilmektedir. Bu güzerğâhın daha sonra kullanılabilmesi için, kullanıcı bu ortak güzerğâhın JSON<sup>15</sup> dosya türünde çıktısını alabilmektedir. **Şekil 13** yaratılan ortak güzerğâhın JSON türünde nasıl çıkarılabileceğini göstermektedir.

<sup>15</sup> Javascript Object Notation, <http://www.json.org/>



Şekil 12. eMine Algoritması ACTF uygulaması

```

1 {
2   "main_path": ["VB.1.2.1.2",
3                 "VB.1.1.2",
4                 "VB.1.2.1.2",
5                 "VB.1.2.1.2",
6                 "VB.1.2.1.1.2.2",
7                 "VB.1.2.1.1.2.2"]
8 }

```

Şekil 13. Örnek JSON formatında ortak güzergâh

### 3.2.4 eMine Algoritması'nın Deney ve Sonuçları

Yapılan ön çalışmalarımız önerilen algoritmanın iyi sonuçlar verdiğini gösterse de, daha sonra önerilen algoritmayı test edebilmek için kullanıcı bazlı bir deney düzenlenmiştir. Bu deneyin detayları ve sonuçları da teknik rapor olarak yayınlanmıştır (Eraslan vd., 2013c). Yapılan bu deney genel olarak şu şekilde özetlenebilir:

**Amacı:** Tasarlanan bu deneyin esas amacı, önerilen algoritmanın geçerliliğini ve ölçeklenebilirliğini test etmektir. Ayrıca önerilen algoritmanın sonuçlarına kullanıcının cinsiyetinin, aşinalığının, sayfa karmaşıklığının ve sayfayı bloklara ayırma seviyesinin etkilerini ölçmektir.

*Bilimsel Sorular:* Yukarıdaki amaca ulaşabilmek için, bu deneyde araştırılan bilimsel sorular şu şekilde özetlenebilir: eMine Algoritması sayfa içerisindeki ortak güzergâhı web sayfasındaki görsel elemanlar bazında başarılı bir şekilde çıkarabilir mi? Çıkarılan ortak güzergâh kullanıcı sayısından nasıl etkilenir? Kullanıcı cinsiyeti ortak güzergâhı nasıl etkiler? Kullanıcının web sayfasına olan aşinalığı ortak güzergâhı nasıl etkiler? Bloklara ayırma seviyesi ortak güzergâhı nasıl etkiler?

*Deney Ekipmanı:* Kullanıcılar 17" monitörü olan TOBII T60 göz tarayıcısının karşısına oturtulmuştur. Ekran çözünürlüğü 120x1024 şeklinde kullanılmıştır. Web sayfaları HP EliteBook 8530p dizüstü bilgisayara yüklenmiş ve göz tarayıcı monitöründen kullanıcılara gösterilmiştir.

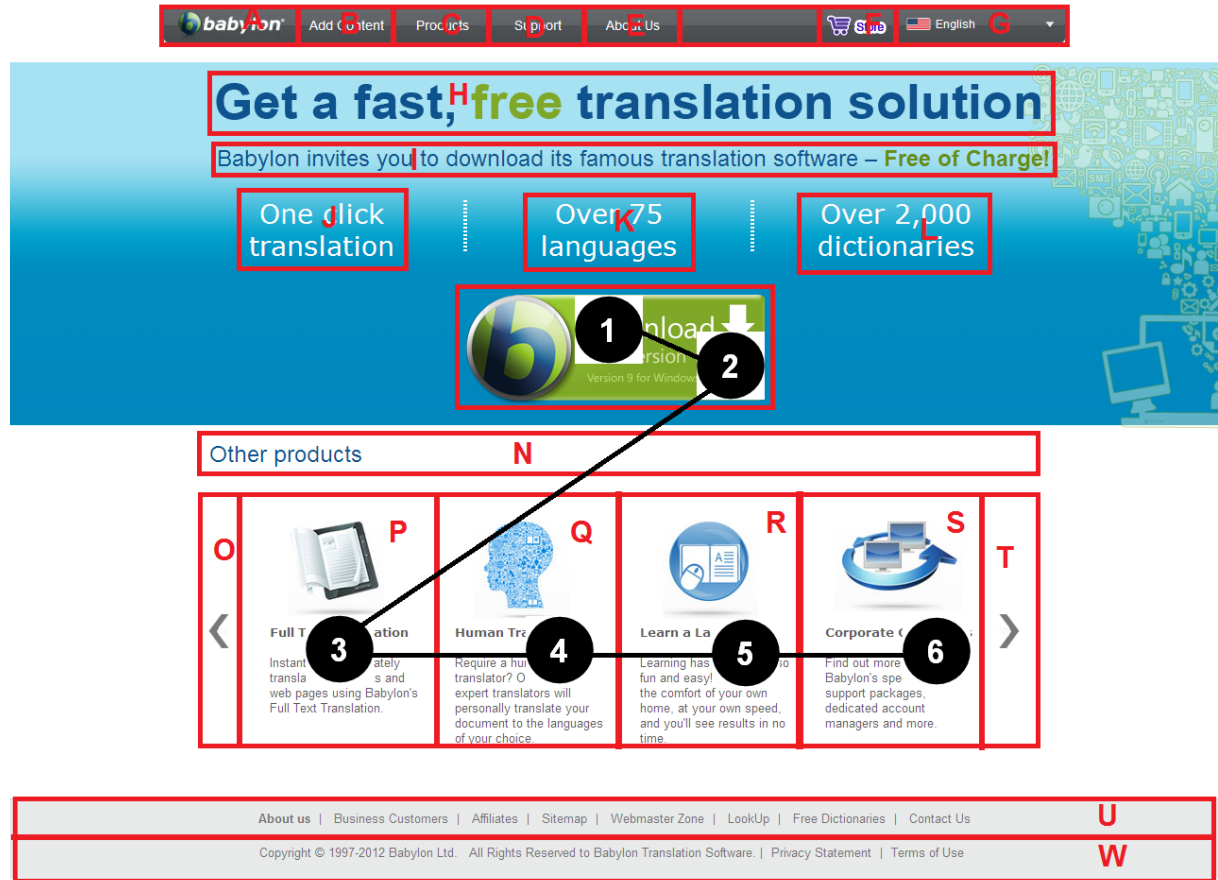
*Kullanılan Sayfalar:* Kullanılan web sayfaları daha önce yapılmış olan kullanıcı deneyinde (Akpınar ve Yeşilada, 2013b; Akpınar ve Yeşilada, 2013c) kullanılan sayfalar arasından seçilmiştir. Bunun amacı da, yapılan çalışmanın bütünlüğünü korumaktır. Önceden kullanılan sayfalar arasından altı tanesi, her bir karmaşıklık grubundan iki tane olacak şekilde rastgele seçilmiştir. Seçilen bu sayfalar ile ilgili detaylara ilgili teknik rapordan erişilebilir (Eraslan vd., 2013c).

*Kullanılan Yöntem:* Katılımcılara ilk olarak deney hakkında bilgi verilmiştir. Katılımcılardan kendileri hakkında cinsiyet ve eğitim düzeyi gibi bazı bilgiler doldurmaları istenmiştir. Daha sonra altı sayfanın her biri katılımcılara iki kez gösterilerek, bir tarama görevi, bir de arama görevi tamamlamaları istenmiştir. Deney süresince araştırma görevlisi tarafından katılımcının başarılı olup olmadığı not edilmiştir.

*Katılımcılar:* Deneyler hem ODTÜ Kuzey Kıbrıs Kampüsü'nde, hem de Manchester Üniversitesi'ndeki WEL Laboratuvarı'nda gerçekleştirilmiştir. Toplamda 20 erkek ve 20 kadın katılımcı deneylere katılmıştır. Tüm katılımcılar günlük olarak interneti kullandıklarını söylemişlerdir. Katılımcıların çoğu 18-34 yaş aralığı içerisinde, 18 katılımcı 18-24 yaş, 14 katılımcı 25-34 yaş, 6 katılımcı ise 35-54 yaş aralığındadır.

*Sonuçlar:* Yapılan deneyler, eMine Algoritması'nın genel olarak görsel elemanlar türünden başarılı bir şekilde ortak güzergâh çıkarabildiğini göstermektedir. Ortak güzergâhın kullanıcının cinsiyet ve web sayfasına olan aşinalığından etkilendiği görülmektedir. Detaylı sonuçlara ilgili teknik rapordan erişilebilir (Eraslan vd., 2013c). Bu sonuçlar genel olarak şu şekilde özetlenebilir:

*Geçerlilik* - “eMine Algoritması sayfa içerisindeki ortak güzergâhı web sayfasındaki görsel elemanlar bazında başarılı bir şekilde çıkarabilir mi?” Gösterilen altı sayfa içerisinde katılımcıların sayfaları tarama görevi yanında, sayfa içerisinde bazı bilgileri aramaları istenmiştir. Aradıkları bilgilere ve eMine Algoritması’nın çıkardığı ortak güzergâha bakıldığında, ortak güzergâhın sayfa içerisinde aranan bilgi ile örtüştüğü görülmektedir. Örneğin, Babylon sayfasında katılımcıların önce sayfa içerisinde Babylon’u bedava indirebilecekleri bağlantıyı, daha sonra da Babylon’un sunduğu diğer ürünleri bulmaları istenmiştir. **Şekil 14**’te eMine Algoritması’nın çıkardığı ortak güzergâh gösterilmektedir. Bu ortak güzergâha bakıldığında, katılımcılara verilen görev ile birebir örtüşmektedir: Katılımcılar önce Babylon’u bedava indirebilecekleri bağlantıya, daha sonra Babylon’un sunduğu diğer ürünlere bakmıştır. **Tablo 7** özet olarak deneyde kullanılan diğer sayfaları ve bu sayfalar üzerindeki ortak güzergâhları göstermektedir. Katılımcı sayılarındaki farklar, bazı katılımcıların tarama güzergâhlarında çıkan sorunlardan dolayı bu katılımcıların verilerinin kullanılamamasından kaynaklanmaktadır.



Şekil 14. Babylon sayfasındaki ortak güzergâh

Tablo 7. eMine Algoritması ile çıkarılan ortak güzergâhlar

| Sayfa   | Katılımcı Sayısı | Ortak güzergâh |
|---------|------------------|----------------|
| Apple   | 31               | E              |
| Babylon | 36               | MMPQRS         |
| AVG     | 36               | DFDF           |
| Yahoo   | 29               | E              |
| Godaddy | 34               | POO            |
| BBC     | 38               | M              |

*Ölçeklenebilirlik - Çıkarılan ortak güzergâh katılımcı sayısından nasıl etkilenir?* Çıkarılan ortak güzergâhın katılımcı sayısından nasıl etkilendiğini ortaya koymak için, güzergâhları 10, 20, 30, 30+ kişi olacak şekilde farklı sayılarda rastgele seçip gruplandırarak, eMine Algoritması'nın çıkardığı sonuçlar incelenmiştir. **Tablo 8'**de genel olarak katılımcı sayısına göre eMine Algoritması'nın çıkardığı ortak güzergâhın nasıl etkilendiğini gösterilmektedir. Genel olarak ortak güzergâhlar arasındaki benzerliklere bakıldığında, yaklaşık olarak %60 civarında benzedikleri görülmüştür. Bu da genel olarak katılımcı sayısının ortak güzergâha etkisinin orta düzeyde olduğunu göstermektedir. Bu düzey sayfalar ve göreve göre bazı farklılıklar göstermektedir.

Tablo 8. Verilen göreve ve kullanıcı sayısına göre çıkarılan ortak güzergâhlar

| Görev  | Sayfa   | Karmaşıklık | P=10    | P=20   | P=30   | P=30+  |
|--------|---------|-------------|---------|--------|--------|--------|
| Tarama | Apple   | Düşük       | CFFHF   | CFIF   | F      | CF     |
|        | Babylon | Düşük       | MMPP    | MM     | M      | M      |
|        | AVG     | Orta        | DDFPDD  | DFDF   | DD     | DD     |
|        | Yahoo   | Orta        | EDED    | EDDEE  | EED    | EED    |
|        | Godaddy | Yüksek      | PPRPPP  | PPRPP  | PPRP   | PPRP   |
|        | BBC     | Yüksek      | KLN     | KLN    | KLN    | NN     |
| Arama  | Apple   | Düşük       | FFEF    | EEA    | E      | E      |
|        | Babylon | Düşük       | MMPPQRS | MMPQRS | MMPQRS | MMPQRS |
|        | AVG     | Orta        | DFDFDFD | DFDF   | DFDF   | DFDF   |
|        | Yahoo   | Orta        | DEDE    | E      | E      |        |
|        | Godaddy | Yüksek      | POPOP   | POOO   | PO     | POO    |
|        | BBC     | Yüksek      | MNMNM   | NMNMNM | MNM    | M      |

*Cinsiyet etkisi - Katılımcı cinsiyeti ortak güzergâhı nasıl etkiler?* Katılımcıların cinsiyetlerinin çıkarılan ortak güzergâha etkisini incelemek için, tüm katılımcılar erkek ve kadın olarak gruplandırılmış ve bu gruplardan çıkan ortak güzergâhlar incelenmiştir. **Tablo 9**'da çıkarılan ortak güzergâhlar özetlenmektedir. Bu güzergâhlar arasındaki benzerliklere bakıldığında, tarama görevinde ortak güzergâhların %50.8, arama görevinde ise %66.1 benzedikleri görülmektedir. Çıkarılan ortak güzergâhların uzunlukları incelendiğinde ise, ortak güzergâh uzunluklarının da katılımcı cinsiyetinden dolayı farklılık gösterdiği ortaya çıkmıştır.

Tablo 9. Kadın ve erkek katılımcılara göre çıkarılan ortak güzergâhlar

| Görev  | Sayfa   | Karmaşıklık | Katılımcı Sayısı | Erkek    | Kadın  | Benzerlik |
|--------|---------|-------------|------------------|----------|--------|-----------|
| Tarama | Apple   | Düşük       | 18               | FFF      | FIHGF  | 40%       |
|        | Babylon | Düşük       | 18               | R        | RM     | 50%       |
|        | AVG     | Orta        | 18               | DFD      | FDFFF  | 40%       |
|        | Yahoo   | Orta        | 18               | DED      | EEDE   | 50%       |
|        | Godaddy | Yüksek      | 18               | PPP      | PPPP   | 75%       |
|        | BBC     | Yüksek      | 18               | LN       | KN     | 50%       |
| Arama  | Apple   | Düşük       | 15               | FFEF     | F      | 25%       |
|        | Babylon | Düşük       | 17               | MMRPPQRS | MMPQRS | 75%       |
|        | AVG     | Orta        | 18               | DFDF     | FDPDF  | 80%       |
|        | Yahoo   | Orta        | 14               | EDE      | E      | 66.7%     |
|        | Godaddy | Yüksek      | 17               | OPOO     | OPP    | 50%       |
|        | BBC     | Yüksek      | 18               | KMM      | KMM    | 100%      |

*Aşinalığın etkisi - Katılımcının web sayfasına olan aşinalığı ortak güzergâhı nasıl etkiler?* Katılımcılara deneye başlamadan önce deneyde kullanılan sayfaları ne kadar sıklıkla ziyaret ettikleri sorulmuştur. Bu bilgiler kullanılarak, katılımcıların sayfalara aşına olup olmadıklarına karar verilmiştir. Altı sayfa içerisinde üç tanesine yeterli sayıda katılımcı aşına olmadığı için bu sayfalar değerlendirmeye alınmamıştır. Diğer üç sayfa için (Yahoo, Apple ve BBC) çıkan sonuçlar **Tablo 10**'da gösterilmektedir. Genel olarak aşinalığın çok fazla olmasa da ortak güzergâhı etkilediği görülmektedir. Uzunluk olarak ortak güzergâhlarda fark olmasa da, katılımcıların sayfa ile aşına olmadıkları durumlarda farklı blokları ziyaret ettikleri görülmüştür.

Tablo 10. Aşinalığa göre çıkarılan ortak güzergâhlar

| Görev | Sayfa | Karmaşıklık | Katılımcı sayısı | Aşına olan | Aşına olmayan | Benzerlik |
|-------|-------|-------------|------------------|------------|---------------|-----------|
|-------|-------|-------------|------------------|------------|---------------|-----------|

|        |       |        |    |       |       |       |
|--------|-------|--------|----|-------|-------|-------|
| Tarama | Apple | Düşük  | 11 | FFJI  | FCFFE | 40%   |
|        | Yahoo | Orta   | 16 | EDED  | EEDE  | 50%   |
|        | BBC   | Yüksek | 15 | LKL   | KLN   | 33.3% |
| Arama  | Apple | Düşük  | 10 | FEFEE | EF    | 40%   |
|        | Yahoo | Orta   | 10 | E     | DEDE  | 25%   |
|        | BBC   | Yüksek | 16 | KMM   | KLLM  | 50%   |

*Blok seviyesi etkisi - Bloklara ayırma seviyesi ortak güzergâhı nasıl etkiler?* Web sayfaları öncelikle farklı seviyelerde bloklara ayrılmış ve daha sonra o seviyeler üzerinden ortak güzergâhlara bakılmıştır. **Tablo 11** üç farklı seviyede çıkan ortak güzergâhları göstermektedir. Bu tablodan da görüleceği gibi ve beklendiği üzere, bloklara ayırma seviyesi düştükçe, yani blokların daha küçük olduğu durumlarda, çıkan ortak güzergâhların uzunluğu artmaktadır.

Tablo 11. Farklı seviyelerde bloklara ayrılan sayfaların ortak güzergâhı

| Görev  | Sayfa   | Katılımcı Sayısı | Seviye 3                            | Seviye 4             | Seviye 5 |
|--------|---------|------------------|-------------------------------------|----------------------|----------|
| Tarama | Apple   | 36               | (C...E)F                            | FFFF                 | CF       |
|        | Babylon | 37               | (J...M)(P...T)                      | (P...S)              | M        |
|        | AVG     | 38               | (D...F)                             | D(EF)DD              | DD       |
|        | Yahoo   | 37               | (DE)                                | (DE)                 | EED      |
|        | Godaddy | 38               | (N...R)                             | POOO                 | PPRP     |
|        | BBC     | 37               | (K...M,O)N(K...M,O)N                | (MN)O(MN)            | NN       |
| Arama  | Apple   | 31               | (C...E)F                            | F                    | E        |
|        | Babylon | 36               | (J...M)(P...T)(J...M)(P...T)(P...T) | M(P...S)             | MMPQRS   |
|        | AVG     | 36               | (D...F)                             | D(EF)D(EF)           | DFDF     |
|        | Yahoo   | 29               | (DE)                                | DE                   | D        |
|        | Godaddy | 34               | (N...R)                             | P(NO)(NO)            | POO      |
|        | BBC     | 38               | (K...M,O)N(K...M,O)                 | (K...M,O)N(K...M,O)N | M        |

### 3.3 Dönüştürme Deneyleri ve Geçerliliği

Son iş paketinin ana hedefi, eMine Algoritması'nı kullanarak web sayfalarını dönüştürme teknikleri geliştirmek, teknik testler ve kullanıcı testleri ile bu tekniklerin geçerliliğini kanıtlamaktır. Özet olarak bu iş paketi iki görevden oluşmaktadır: (1) Dönüştürme teknikleri geliştirip uygulamak ve (2) Dönüştürme tekniklerini testler ile değerlendirmek. Bu iş paketinde



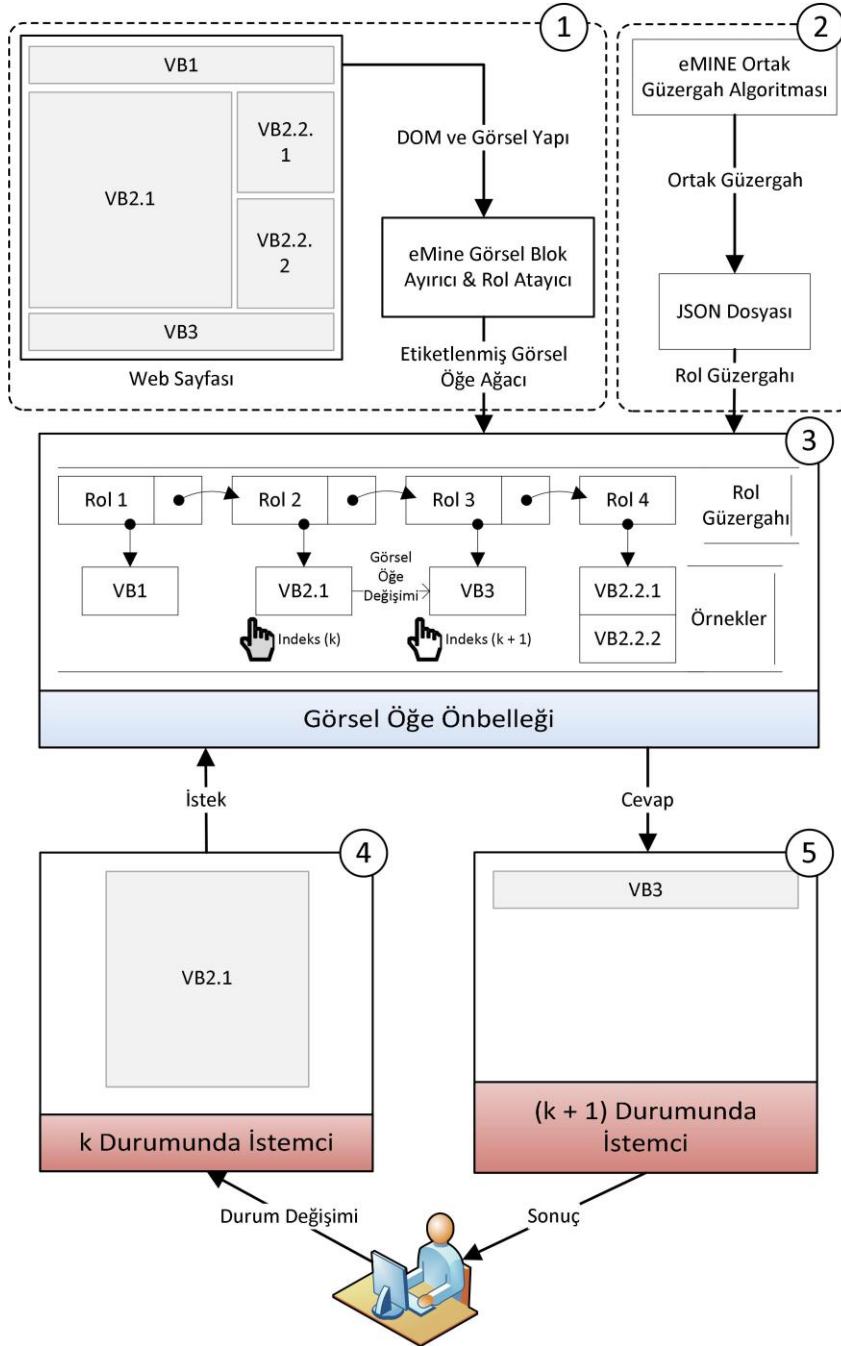
yaptığımız çalışmalar iki halinde teknik rapor halinde yayınlanmış olup (Akpınar ve Yeşilada, 2013c, 2013d), şu şekilde özetlenebilir:

### 3.3.1 Dönüştürme Teknikleri

Literatür özeti bölümünde de özetlendiği gibi, literatürde birçok teknik önerilmiş ve denenmiştir. Bu teknikler hem engelli kullanıcılar için web erişilebilirliğini iyileştirme, hem de mobil kullanıcıların erişimini iyileştirme amacındadır. Sayfayı dönüştürebilmek için atlama linki (skip link) (Takagi vd., 2002, 2007), içeriğin özetini yaratma (Büyükkökten vd., 2000, 2001, 2002; Yang ve Wang, 2003), sayfaların minyatür resimlerinin yapılması (Chen vd., 2003), derecelendirme ve yeniden düzenleme (Yin ve Lee, 2004), içeriği daha küçük parçalara bölme (Baluja, 2006; Hattori vd., 2007) ve alakasız içeriği ortadan kaldırma gibi değişik yöntemler önerilmiştir (Baudisch vd., 2004). Yapılan araştırmalar, dönüştürme tekniklerinin kullanılmasının, engelli ve mobil kullanıcıların web deneyimlerini iyileştirdiğini gösteriyor olsa da (Takagi vd., 2007; Yeşilada vd., 2007), etkili bir dönüştürme tekniği uygulanabilmesi için sayfanın içerik, yapı ve bağlamının (context) çok iyi anlaşılması gerektiği açıktır (Yeşilada vd., 2008b). Maalesef mevcut çalışmaların birçoğu dönüştürme işlemini yaparken sayfanın nasıl kullanıldığını anlamaya çalışmamaktadır. eMine Projesi'nde yapılan çalışma bu bağlamda bir ilktir. Yapılan çalışmada ilk olarak görsel bloklara ayrılmış web sayfası, rolleri ve ortak güzergâh bilgileri ile uygulanabilecek dönüştürme teknikleri araştırılmıştır. Bu araştırma sonucunda aşağıdaki beş tekniğin uygun olabileceği gözlemlenmiştir:

- *Seçici betimleme*: Yapılan çalışmalar sonucunda sayfa içerisindeki görsel elemanlar ve rolleri elde edilmiştir. Buradaki teknik, sayfa içerisindeki tüm görsel elemanları göstermek yerine, sadece belli görsel elemanları kullanıcıya sunmayı amaçlamaktadır.
- *Sayfa düzeninin değişmesi*: Yine sayfadaki görsel elemanlar, görsel elemanların rolleri ve ortak güzergâh kullanılarak sayfanın görsel düzeni değiştirilebilir.
- *Görsel elemanların sunumu*: Sayfa içerisinde elde edilen görsel elemanlar o sayfayı tasarlayan kişinin zorladığı şekilde değil, kullanıcının isteğine göre şekillendirilebilir.
- *Özet*: Literatürde birçok özetleme tekniği önerilmiştir, fakat bunların hiçbirisi görsel elemanların rollerine göre sayfayı özetlemeyi önermemiştir. Bu çalışmada, sayfa içerisindeki bloklar, blokların rolleri ve ortak güzergâh kullanılarak sayfanın özetinin çıkarılması amaçlanmıştır.
- *Sıralı ve doğrudan erişim*: Sayfa içerisinde çıkarılan ortak güzergâha doğrudan erişim sağlanabilir. Bu da diğer kullanıcıların deneyimlerini göz önünde bulundurarak sayfadaki belli görsel elemanlara doğrudan erişim sağlayacaktır.

Projenin bu son kısmında da, diğ er iř paketlerinde olduėu gibi deneysel platform olarak ACTF kullanılmıřtır. ACTF platformuna bu tip bir  alıřmayı entegre edebilmek i inde **řekil 15**'te verilen sistem mimarisi yaratılmıřtır. Bu da  nceden yaratılan     nemli par anın, otomatik bloklara ayırma, otomatik rol atama ve otomatik ortak g zerg h belirleme kısımlarının d n řt rme teknikleri i in kullanımını m mk n kılmıřtır. **řekil 15**'ten de g r leceėi gibi, d n řt rme teknikleri i in etiketlenmiř g rsel blok aėacı ve eMine Algoritması'nın yaratmıř olduėu ortak g zerg h olmak  zere iki farklı girdi kullanılmıřtır. Bu iki girdi, sayfanın kodu ve g rsel i eriėinin yanında  ok daha fazla bilgi sunmaktadır. Bunlar daha sonra d n řt rme tekniklerinde kullanılmıřtır.



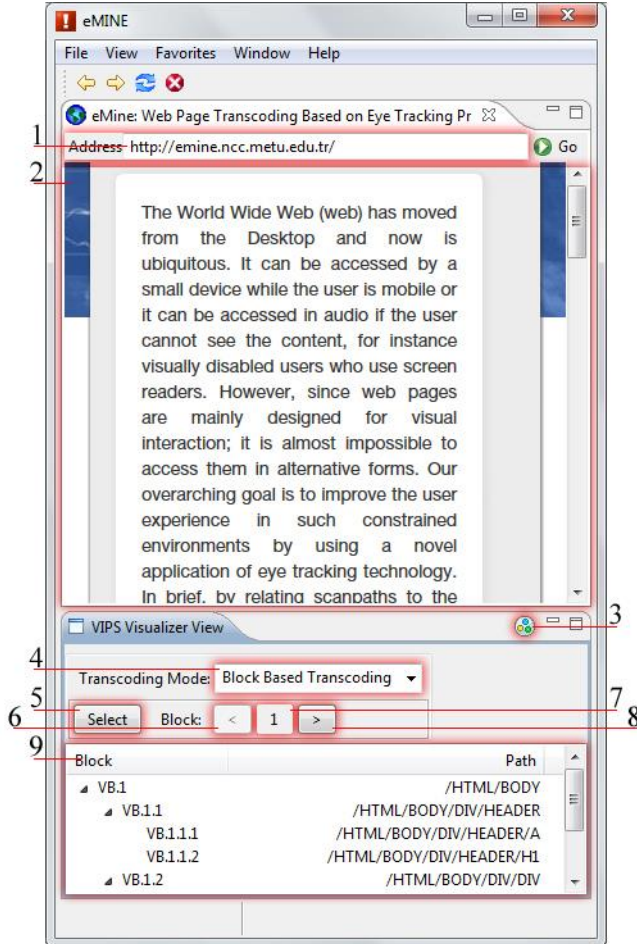
Şekil 15. eMine dönüştürme mimarisi

### 3.3.2 Dönüştürme Teknikleri Uygulamaları

Uygulanan tüm dönüştürme teknikleri yinelemeli bir yöntem kullanılarak geliştirilmiştir. Genel olarak iki tür dönüştürme tekniği geliştirilmiştir: (1) Blok bazlı dönüştürme ve (2) Rol bazlı dönüştürme.

**(1) Blok bazlı dönüştürme** modülü, bir web sayfasının ortak güzergâh kullanılarak dönüştürülmesine olanak sağlar. Ortak güzergâh, eMine göz izleme uygulaması kullanılarak oluşturulan bir JSON dosyası olarak sisteme yüklenir ve ortak güzergâh içinde yer alan blokların isimlerini liste şeklinde içerir. Bu blok isimleri sayfa içindeki bloklarla ilişkilendirilerek ilgili bloka erişim sağlanır.

Blok bazlı dönüştürme için geliştirilen uygulama, kullanıcıya bir web sitesine erişim, sayfa içeriğini görüntüleme, sisteme ortak güzergâh yükleme, sayfayı bloklara ayırarak dönüştürme, oluşturulan bloklar üzerinde ortak güzergâhı kullanarak ileri ve geri yönde hareket etme gibi özellikler sunmaktadır. **Şekil 16**'da bu uygulamanın bir örneği gösterilmektedir. Kullanıcı ortak güzergâh üzerinde bulunan bir bloktan diğerine geçmek için "Önceki" ve "Sonraki" butonlarını kullanmalıdır. Bu butonlar kullanıcının o anki dizinine göre görüntülenen bloku değiştirir. Ortak güzergâh üzerindeki ilk blokta önceki blok butonunun, son blokta ise sonraki blok butonunun kullanılmasına izin verilmez.

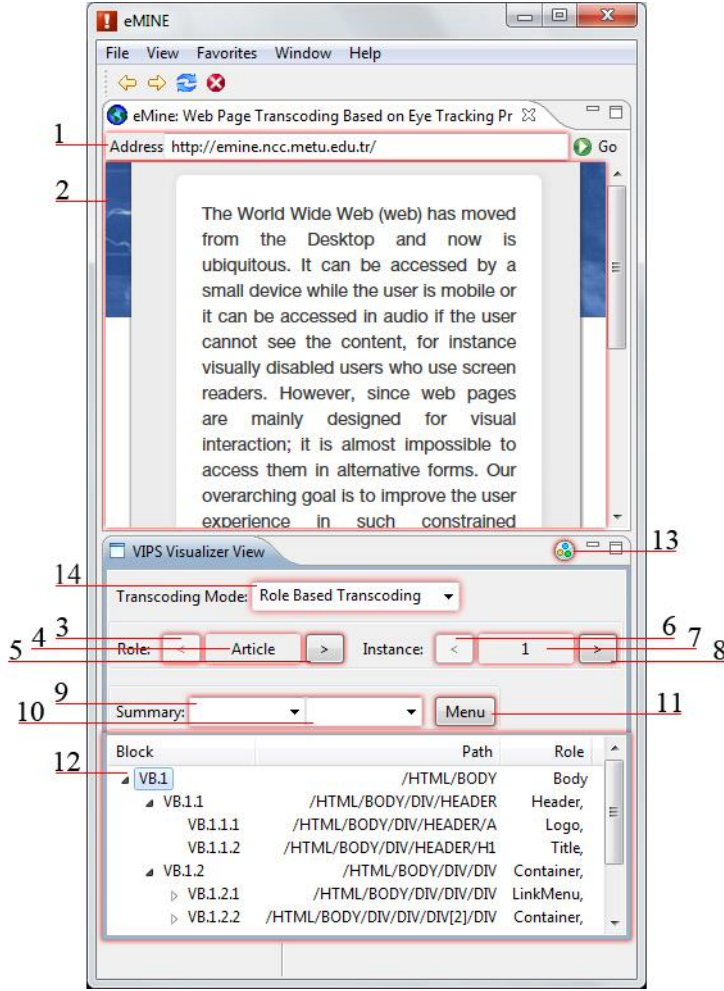


Şekil 16. Blok-bazlı dönüştürme örneği

**(2) Rol bazlı dönüştürme** modülünde ise sayfalar görsel elemanların sayfa içindeki rolleri dikkate alınarak dönüştürülür. Modül sayfa içindeki her bir görsel elemanın sayfa içindeki

rolünü belirler ve elemanları rollerine göre gruplar. Daha sonra, rol güzergâhı üzerinden bu elemanlara sıralı olarak veya direk erişim sağlanır. Rol güzergâhı blok bazlı dönüştürmenin aksine sisteme kullanıcı tarafından yüklenmez. Bunun yerine rol güzergâhı sabit bir JSON dosyası şeklinde kullanıcının yerel dosya sisteminde saklanır. Bu dosya her sayfa için ortak bir rol güzergâhı belirlemeyi amaçlamaktadır, fakat güncel tasarım eğilimleri ve web teknolojileri hızla değişmektedir. Dolayısıyla roller bir JSON dosyasında tutularak, bu dosyanın kolaylıkla güncellenebilmesine olanak sağlanmıştır. Ayrıca, farklı kullanım amaçları için farklı roller oluşturmak mümkündür. Mevcut sistemde buna da imkan verilmiştir.

Rol bazlı dönüştürme için geliştirilen uygulama, kullanıcıya bir web sitesine erişim, sayfa içeriğini görüntüleme, sayfayı bloklara ayırarak ve blokların rollerini belirleyerek dönüştürme, oluşturulan blokların rollerini kullanarak ileri ve geri yönde hareket etme, bir rolün altında gruplanmış olan elemanlar arasında ileri ve geri yönde hareket etme, sayfayı rol bazında özetleme, özet üzerinden bir role doğrudan erişim, bir rolü eleman bazında özetleme ve bu elemanlara doğrudan erişim gibi özellikler sunmaktadır. **Şekil 17**'de bu uygulamanın bir örneği gösterilmektedir.



Şekil 17. Rol-bazlı dönüştürme örneği

Kullanıcı rol güzergâhı üzerinde bulunan bir rolden diğerine geçmek için “Önceki Rol” ve “Sonraki Rol” butonlarını kullanmalıdır. Bu butonlar kullanıcının o anki rol dizinine göre görüntülenen rolü değiştirerek, rolün altında gruplanmış olan elemanlardan ilkinı gösterir. Ayrıca kullanıcı sayfa içerisinde yer alan bütün rolleri de listeleyebilir ve seçtiği role doğrudan erişebilir. Kullanıcı aynı role sahip farklı elemanlar arasında gezinmek isterse, “Önceki Eleman” ve “Sonraki Eleman” butonlarını kullanmalıdır. Ayrıca, güncel role sahip olan bütün elemanlar da listelenmekte ve kullanıcı bu elemanlara doğrudan erişebilmektedir. Son olarak, sayfa içinde yer alan bazı belirli rollere de doğrudan erişim sağlanmıştır. Örneğin, bir buton yardımıyla kullanıcı menü elemanlarına erişebilmektedir.

### 3.3.3 Dönüştürme Teknikleri Deney ve Sonuçları

Bu çalışmanın temel amacı, kullanıcıların web sayfalarına erişirken kullandıkları vakit ve kaynakların en aza indirilerek, erişilebilirliğin desteklenmesidir. Dolayısıyla, sistemin başarılı

olup olmadığı, web sayfalarına erişim üzerinde yapılan teknik ölçümler, kullanıcı bazlı simülasyon deneyleri sonucu ortaya çıkan kazanımın değerlendirilmesiyle bulunabilir.

Proje kapsamında yapılan ön değerlendirme çalışmasında, çeşitli uygulamalar kullanılarak sayfalara erişim süresi, erişim sırasında istemci uygulamanın CPU kullanımı, gönderilen toplam istek sayısı, indirilen toplam veri miktarı, kullanıcının bir sayfa içerisinde ortak güzergâhta yer alan elemanlara erişim süresi gibi ölçümler yapılmıştır. Bu ölçümler, orijinal sayfa yapısı ve dönüştürülmüş halleri için tekrarlanmış ve sonuçlar karşılaştırılmıştır. Bu deneyler için yine önceki iş paketlerinde yapılan deneylerde kullanılan sayfalar kullanılmıştır. Bu durum bütünlükçü bir deney yaklaşımı sunmuştur.

aDesigner<sup>16</sup> adlı kullanıcı simülasyon uygulaması kullanılarak yapılan ölçümler **Tablo 12**'de verilmiştir. Buna göre, düşük karmaşıklık seviyesindeki sayfalara erişim süreleri kabul edilebilir düzeydedir. Fakat, sayfanın karmaşıklığı arttıkça bu süre de artmakta ve 50 saniyenin üzerine çıkmaktadır. Bu da günlük İnternet kullanımında kullanıcı için zaman kaybına sebep olabilir. Dönüştürülmüş sayfalarda ise bu görsel elemanlara doğrudan ulaşılabilirdiği için, kullanıcı bu elemanlara ulaşmak için beklemek zorunda kalmaz. Sonuç olarak, sayfa içindeki görsel elemanlara erişim süresi, uygulanan dönüştürme yöntemiyle azaltılmıştır.

---

<sup>16</sup> aDesigner, <http://www.eclipse.org/actf/downloads/tools/aDesigner/>

Tablo 12. Ortak güzergâh içinde bulunan elemanlara erişim süreleri

| Sayfa   | Karmaşıklık | Görsel Eleman | Erişim Süresi |
|---------|-------------|---------------|---------------|
| Apple   | Düşük       | 1             | 9 sn.         |
|         |             | 2             | 5 sn.         |
|         |             | 3             | 10 sn.        |
| Babylon | Düşük       | 1             | 15 sn.        |
|         |             | 2             | 10 sn.        |
|         |             | 3             | 17 sn.        |
|         |             | 4             | 25 sn.        |
|         |             | 5             | 36 sn.        |
| Yahoo   | Orta        | 1             | 17 sn.        |
|         |             | 2             | 29 sn.        |
| Godaddy | Yüksek      | 1             | 53 sn.        |
|         |             | 2             | 51 sn.        |
|         |             | 3             | 66 sn.        |
| BBC     | Yüksek      | 1             | 17 sn.        |
|         |             | 2             | 69 sn.        |

Görsel karşılaştırma testleri, web sayfalarının yüklenme sürelerini, CPU kullanım süresini, toplam istek sayısını ve indirilen dosya büyüklüğünü karşılaştırmayı amaçlamaktadır. Testler, WebPageTest<sup>17</sup> uygulaması kullanılarak, orijinal sayfa ve dönüştürülmüş versiyonlar üzerinde tekrarlanmıştır. Ayrıca, ilk olarak CSS ve JavaScript dosyalarının, her dönüştürülmüş eleman için indirildiği farz edilmiş, daha sonra da bu dosyaların yalnızca birinci görsel eleman için indirildiği ve ön bellekte tutularak sonraki görsel elemanlar için indirilmediği farz edilmiştir. Bu testlerin sonuçları **Tablo 13**'te verilmiştir.

**Tablo 13**'teki değerler karşılaştırıldığında, yüklenme süresi, CPU kullanımı, istek sayısı ve indirilen dosya büyüklüğü gibi konularda, dönüştürme yapılmış olan sayfalarda daha verimli sonuçlar alındığı, CSS ve JavaScript dosyalarının ön bellekte tutularak tekrar indirilmediği varsayıldığında bu kazanımın artmış olduğu görülmektedir.

<sup>17</sup> Web Page Test, <http://www.webpagetest.org/>



Tablo 13. Görsel karşılaştırma sonuçları

| Sayfa   | Görsel Eleman | CSS ve JavaScript Dosyaları Dahil |                    |              |                      | CSS ve JavaScript Dosyaları Hariç |                    |              |                      |
|---------|---------------|-----------------------------------|--------------------|--------------|----------------------|-----------------------------------|--------------------|--------------|----------------------|
|         |               | Yüklenme Süresi (sn)              | CPU kullanımı (sn) | İstek Sayısı | Dosya Büyüklüğü (KB) | Yüklenme Süresi (sn)              | CPU kullanımı (sn) | İstek Sayısı | Dosya Büyüklüğü (KB) |
| Apple   | Orijinal      | 7.148                             | 1.216              | 39           | 445.603              | -                                 | -                  | -            | -                    |
|         | VE1           | 4.309                             | 0.686              | 19           | 131.109              | -                                 | -                  | -            | -                    |
|         | VE2           | 4.071                             | 1.014              | 18           | 131.414              | 2.131                             | 0.374              | 2            | 8.106                |
|         | VE3           | 4.158                             | 0.951              | 18           | 185.235              | 1.970                             | 0.234              | 2            | 61.894               |
| Babylon | Orijinal      | 4.890                             | 1.216              | 42           | 262.269              | -                                 | -                  | -            | -                    |
|         | VE1           | 3.433                             | 0.218              | 9            | 142.107              | -                                 | -                  | -            | -                    |
|         | VE2           | 2.391                             | 0.296              | 7            | 12.199               | 1.051                             | 0.702              | 2            | 3.993                |
|         | VE3           | 1.970                             | 0.218              | 7            | 14.596               | 0.652                             | 0.109              | 1            | 1.214                |
|         | VE4           | 1.976                             | 0.795              | 7            | 13.131               | 0.670                             | 0.436              | 1            | 1.249                |
|         | VE5           | 2.001                             | 0.717              | 7            | 13.158               | 0.653                             | 0.358              | 1            | 1.195                |
| AVG     | Orijinal      | 8.633                             | 1.029              | 77           | 427.53               | -                                 | -                  | -            | -                    |
|         | VE1           | 4.282                             | 1.263              | 23           | 232.277              | -                                 | -                  | -            | -                    |
|         | VE2           | 4.240                             | 1.310              | 24           | 248.006              | 3.153                             | 0.499              | 12           | 52.037               |
| Yahoo   | Orijinal      | 2.728                             | 1.918              | 50           | 474.294              | -                                 | -                  | -            | -                    |
|         | VE1           | 2.393                             | 0.499              | 23           | 209.916              | -                                 | -                  | -            | -                    |
|         | VE2           | 1.098                             | 0.733              | 16           | 105.822              | 1.102                             | 0.265              | 16           | 70.697               |
| Godaddy | Orijinal      | 6.262                             | 1.684              | 44           | 417.346              | -                                 | -                  | -            | -                    |
|         | VE1           | 3.898                             | 0.358              | 19           | 179.289              | -                                 | -                  | -            | -                    |
|         | VE2           | 3.027                             | 1.092              | 18           | 135.437              | 1.932                             | 0.234              | 3            | 4.768                |
|         | VE3           | 3.046                             | 0.655              | 18           | 138.089              | 1.329                             | 0.795              | 3            | 5.699                |
| BBC     | Orijinal      | 8.613                             | 2.059              | 70           | 584.507              | -                                 | -                  | -            | -                    |
|         | VE1           | 5.742                             | 0.686              | 25           | 187.706              | -                                 | -                  | -            | -                    |
|         | VE2           | 5.651                             | 1.294              | 31           | 246.801              | 2.231                             | 0.858              | 6            | 74.978               |

Son olarak, Akamai MobiTest<sup>18</sup> uygulaması kullanılarak dönüştürme işlemi mobil ortamda değerlendirilmiştir. Bu süreçte yapılan testler iPhone 5, iOS 6, AT&T ortamında üçer kez tekrarlanmıştır. Görsel karşılaştırmada olduğu gibi, testler öncelikle CSS ve JavaScript dosyalarının her eleman için ayrı ayrı indirildiği varsayılarak, daha sonra da bu dosyaların ön bellekte tutularak sadece ilk görsel eleman için indirildiği kabul edilerek tekrarlanmıştır. Bu testler mobil ortamda ortalama yüklenme süresi ve dosya büyüklüğü gibi hesaplamalar yapmaktadır.

<sup>18</sup> Mobitest, <http://mobitest.akamai.com>

Mobil ortam simülasyon test sonuçları **Tablo 14**'te verilmiştir. **Tablo 14**'teki değerler karşılaştırıldığında, yüklenme süresi ve indirilen dosya büyüklüğü gibi konularda, dönüştürme yapılmış olan sayfalarda daha verimli sonuçlar alındığı, CSS ve JavaScript dosyalarının ön bellekte tutularak tekrar indirilmediği varsayıldığında bu kazanımın artmış olduğu görülmektedir.

Tablo 14. Mobil simülasyon sonuçları

|         |               | CSS ve JavaScript Dosyaları Dahil |                          | CSS ve JavaScript Dosyaları Hariç |                          |
|---------|---------------|-----------------------------------|--------------------------|-----------------------------------|--------------------------|
| Sayfa   | Görsel Eleman | Ortalama Yüklenme Süresi          | Ortalama Sayfa Büyüklüğü | Ortalama Yüklenme Süresi          | Ortalama Sayfa Büyüklüğü |
| Apple   | Orijinal      | 4.85s                             | 474.55 KB                | -                                 | -                        |
|         | VE1           | 2.97s                             | 127.24 KB                | -                                 | -                        |
|         | VE2           | 3.11s                             | 127.55 KB                | 1.41s                             | 7.88 KB                  |
|         | VE3           | 3.75s                             | 180.11 KB                | 1.73s                             | 60.40 KB                 |
| Babylon | Orijinal      | 3.72s                             | 259.29 KB                | -                                 | -                        |
|         | VE1           | 2.58s                             | 138.18 KB                | -                                 | -                        |
|         | VE2           | 1.41s                             | 11.36 KB                 | 0.99s                             | 3.86 KB                  |
|         | VE3           | 1.58s                             | 13.70 KB                 | 0.70s                             | 1.17 KB                  |
|         | VE4           | 1.40s                             | 12.27 KB                 | 0.69s                             | 1.20 KB                  |
|         | VE5           | 1.48s                             | 12.30 KB                 | 0.69s                             | 1.15 KB                  |
| AVG     | Orijinal      | 7.20s                             | 401.41 KB                | -                                 | -                        |
|         | VE1           | 4.34s                             | 222.22 KB                | -                                 | -                        |
|         | VE2           | 3.81s                             | 237.95 KB                | 2.03s                             | 47.86 KB                 |
| Yahoo   | Orijinal      | 2.90s                             | 461.44 KB                | -                                 | -                        |
|         | VE1           | 2.53s                             | 178.31 KB                | -                                 | -                        |
|         | VE2           | 1.48s                             | 101.24 KB                | 1.15s                             | 69.30 KB                 |
| Godaddy | Orijinal      | 4.74s                             | 404.79 KB                | -                                 | -                        |
|         | VE1           | 3.22s                             | 171.95 KB                | -                                 | -                        |
|         | VE2           | 2.73s                             | 114.45 KB                | 1.66s                             | 4.19 KB                  |
|         | VE3           | 2.53s                             | 131.70 KB                | 1.32s                             | 5.09 KB                  |
| BBC     | Orijinal      | 4.97s                             | 574.68 KB                | -                                 | -                        |
|         | VE1           | 4.08s                             | 170.13 KB                | -                                 | -                        |
|         | VE2           | 3.29s                             | 234.83 KB                | 1.66s                             | 73.26 KB                 |

Değerlendirme çalışmasında kullanılan sayfalar, dünya çapında popüler sayfalar olup pek çok kullanıcı tarafından günlük olarak kullanılmaktadır. Fakat, bu sayfalarda yer alan ve kullanıcının dikkatini ilk aşamada çeken görsel elemanlara erişim süresi gayet yüksek olabilmektedir. Buna karşın, dönüştürme işlemi bu süreyi ortadan kaldırdığı için daha avantajlı bir yöntem sunmaktadır. Ayrıca, teknik karşılaştırmalar sonucu dönüştürme

işleminin indirme süresi, CPU kullanımı, istek sayısı ve indirilen dosya büyüklüğü gibi konularda da avantaj sağladığı görülmüştür.

#### 4. BULGULAR

Literatür özetinde de (bkz. 2. Bölüm) anlatıldığı gibi, web sayfalarına ekran okuyucu veya mobil cihazlar gibi kısıtlı ortamlardan erişmek hala oldukça zorlayıcıdır. Bununla mücadele etmenin bir yolu, web sayfalarını kısıtlı ortamlarda daha rahat okunabilecek şekilde dönüştürmektir. Fakat web sayfalarının düzgün bir şekilde dönüştürülebilmesi için, öncelikle insanların sayfaları nasıl okuduklarının iyice anlaşılması gerekmektedir. Literatür analizi bölümünde de izah edildiği gibi, göz izleme (eye tracking) bu amaç için çok uygun bir yöntemdir. Buna rağmen, göz izleme verilerinin sayfaları dönüştürme konusunda nasıl kullanılabileceği konusunda çok az araştırma yapılmıştır. eMine bu boşluğu, göz izleme verilerinin kısıtlı ortamlardaki kullanıcı deneyimini iyileştirmek için kullanmasını sağlayacak özgün bir algoritma yaratarak doldurmaktadır. Bu sonuç raporunun “Gereç ve Yöntem” bölümünde de aktarıldığı gibi (bkz. 3. Bölüm), proje kapsamında yapılan çalışmalar çok zengin bulgular içermektedir. Fakat burada bunları detaylandırmak yerine, proje önerisinde cevaplanması hedeflenen araştırma sorularının nasıl cevaplandırıldığı ortaya koyulup, daha sonra bu sorular çerçevesindeki genel bulgular tartışılmıştır. Proje önerisinde de sunulduğu gibi, projenin **özgünlüğü** bu soruların cevaplanmasında yatmaktadır.

**Bir sayfadaki görsel elemanlar nasıl otomatik olarak tanımlanır?** Daha önceki çalışmamız, insanların belli işlemleri yaparken sayfadaki görsel elemanları kullandığını göstermiş (Yeşilada vd., 2008b); bunun sonucunda da bu elemanları tespit etmekte kullanılabilecek bir çerçeve oluşturulmuştu. Bu çerçeve kullanıcılar (Yeşilada vd., 2003) ve göz izleme testleri (Yeşilada vd., 2008b) ile onaylanmış olmasına rağmen, bu işlem henüz otomatikleştirilememişti. Bu proje çerçevesinde öncelikle bu işlemin nasıl otomatikleştirebileceği ortaya konulmuştur. Bunun yapılabilmesi için elde edilen bulgular şöyle özetlenebilir: (1) görsel elemanlar hakkında bilgi toplama; (2) web sayfalarını görsel bloklara ayırma ve (3) bu görsel bloklara otomatik olarak roller atama. Bu bulgular, proje önerisinde sorulmuş olan detaylı araştırma sorularına cevap olarak şu şekilde özetlenebilir:

1. *Sayfalardaki görsel elemanlar nasıl otomatik olarak tanımlanır?* Bu proje çerçevesinde öncelikle görsel elemanlar hakkında detaylı bir çalışma yapılmış (Yeşilada, 2011a, 2011b, 2013), görsel elemanların genel özellikleri hakkında detaylı analizler yapılmış ve bu analizler sonucu ortaya çıkan bilgiler ontoloji adı verilen Semantik Web (Anlamsal Ağ) teknolojisi kullanılarak toparlanmıştır (Akpınar ve Yeşilada, 2013b, 2013c, 2013f). Oluşturulan ontoloji daha sonra görsel blokların rollerinin atanması kısmında kullanılmıştır. Bu ontoloji, eMine Projesi’nde otomatik rol

atama için kullanılmış olsa da, farklı projelerde farklı amaçlar için kullanılabilecek bir bulgu olarak ortaya çıkmıştır. eMine Ontolojisi proje sayfasında da erişime açıktır: <http://emine.ncc.metu.edu.tr/ontology/emine.owl>.

2. *Bu en iyi hangi yöntemle yapılabilir?* Yapılan çalışmada, otomatik olarak web sayfalarını görsel bloklara ayırma işleminin farklı araştırma alanlarında farklı amaçlar için çok yaygın bir şekilde araştırılmış olduğu ortaya çıkmıştır (Yeşilada, 2013). Bunların arasından yapılan araştırmada Cai vd. (2003)'nin geliştirmiş olduğu VIPS (Vision Based Segmentation) algoritmasının, diğer algoritmalarla göre çok daha etkili olduğu gözlemlenmiştir. Fakat araştırmamız VIPS algoritmasının ciddi sorunlarının olduğunu ve web sayfalarında kullanılan şu anki web teknolojileri için yetersiz kaldığını ortaya çıkarmıştır. Araştırmamızda daha sonra bu algoritmanın nasıl geliştirilebileceği ve güncellenebileceği ortaya konulmuş ve ACTF üzerinden açık kaynak olarak diğer araştırmacılar ile paylaşılmıştır (Akpınar ve Yeşilada, 2012, 2013a, 2013b).
3. *Sayfadaki görsel elemanlar ile sayfanın kodu arasındaki genellenebilir ilişki nedir?* Proje kapsamında, daha sonra görsel bloklara nasıl otomatik olarak rol atanabileceği gösterilmiştir. Bunun için Anlamsal Ağ (Semantik Web) temelli orijinal bir sistem mimarisi önerilmiş ve yaratılan ontoloji de kullanılarak bir sayfasının içerisinde bulunan görsel blokların nasıl çıkarılabileceği, bu görsel blokların rollerinin otomatik olarak nasıl atanabileceği ve bunların tümünün sayfa kodu ile nasıl ilişkilendirilebileceği gösterilmiştir. Buradaki en önemli bulgularımız, bunların otomasyonun nasıl yapıldığı konusundaki bilgi birikimi ve yapılabilmesi için önerilen sistem mimarisidir. Bu sistem mimarisi yalnız bu proje kapsamında değil, farklı amaçlar için de kullanılabilecek orijinal bir mimari olarak sunulmuştur (Akpınar ve Yeşilada, 2012b, 2013c, 2013f).

**Göz izleme (eye tracking) verilerini kullanarak, bir sayfada bulunan görsel elemanların tarama güzergâhını tespit etmek için kullanılabilecek en iyi algoritma nedir?** WEL laboratuvarında gerçekleştirilen çeşitli projeler<sup>19</sup> kayda değer miktarda göz izleme verisi ortaya çıkarmıştır (Jay vd., 2006, Yeşilada vd., 2008b, Brown vd., 2009, Jay ve Brown, 2008, Michailidou vd., 2008). Proje önerisinde sunulduğu gibi, bu projedeki esas hedef bu veriler ile “String-edit” algoritmasını kullanarak, verilen göz izleme verilerinin tarama güzergâhlarını tespit edebilecek yeni bir algoritma yaratmaktır. Bu bağlamda yapılan araştırmalarımızın bulguları şu şekilde özetlenebilir: Öncelikle proje kapsamında göz izleme verilerinin sayfanın

---

<sup>19</sup> WEL araştırmaları, <http://wel.cs.manchester.ac.uk>

kodu ile nasıl otomatik olarak ilişkilendirebileceği gösterilmiştir. Bu bulgu göz izleme teknolojisinin yalnız bu proje kapsamında değil, farklı projelerde de web sayfaları üzerinde kullanılabileceğini göstermektedir. Daha sonra projede “String-edit” algoritması çok daha detaylı incelenmiş, ve temelde bu algoritmayı kullanan ve ortak güzergâh çıkarılabilecek orijinal bir algoritma oraya çıkarılmıştır. Bu algoritma projede de önerildiği gibi, mevcut veriler kullanılarak geliştirilmiş ve yeni bir göz izleme deneyi ile de nasıl onaylanabileceği gösterilmiştir. Bu bulgular, proje önerisinde sorulan detaylı araştırma sorularına cevap olarak şu şekilde özetlenebilir:

1. *“String-edit algoritmasını kendi algoritmamıza temel teşkil edecek şekilde nasıl kullanabiliriz?* Çalışmalarımız, öncelikle bu konuda var olan algoritmaları elimizdeki veriler ile değerlendirerek başlamıştır. Yapılan bu çalışma bize “String-edit” algoritmasının tek başına bu amaç için kullanılamayacağını göstermiştir. Ayrıca yapılan bu ön çalışma, yaratılacak olan algoritma için kullanılabilecek tüm prensiplerin ortaya koyulmasına yardımcı olmuştur. Bu bulgular yalnız bizim algoritmamız kapsamında değil, bu alanda çalışma yapmak isteyen herkes tarafından kullanılabilecektir (Eraslan vd., 2013a, 2013b). Bu bulgular ışığında yine mevcut veriler ile “String-edit” algoritmasını temel alan bir algoritmanın nasıl geliştirileceği gösterilmiştir.
2. *Sayfadaki görsel elemanlar ve sayfa kodu baz alınarak tarama güzergâhı hakkında bir genelleme yapılabilir mi?* Bu konuda yapılan çalışmada, bir kullanıcının göz izleme verilerini otomatik olarak sayfa içerisindeki görsel bloklara nasıl örtüştürülebileceği gösterilmiştir. Buradaki bulgular, yine yalnız bizim projemiz için değil, bir tarama güzergâhını farklı şekilde değerlendirip kullanmak isteyen kişilerin de işine yarayacaktır. Bunun otomatik olarak nasıl yapılabileceği de, daha sonra deneysel platform üzerinde gösterilmiştir. Bu işlem temelde, algoritmamızın bir önceki adımda otomatik olarak çıkardığı görsel bloklar ve bu görsel blokların rolleri ile otomatik bir şekilde nasıl ilişkilendirilebileceğini göstermiştir (Eraslan vd., 2013c). Bu ilişkide daha sonra görsel elemanlar türünde ortak bir güzergâh çıkarılabileceği gösterilmiştir.
3. *Sayfada görsel elemanlar bakımından tarama güzergâhı yaratacak bu algoritmayı geliştirmenin ve test etmenin en iyi yolu nedir?* Önerilen orijinal eMine Algoritması temelde birden çok kişinin güzergâhını alıp, daha sonra “String-edit” algoritmasını temel alarak nasıl ortak bir güzergâh çıkarılabileceğini göstermiştir. Bu algoritmanın, mevcut algoritmalara kıyasla avantajlarının olduğu da, var olan veriler ile yapılan deneyler ile gösterilmiştir (Eraslan vd., 2013b). Bu tür bir algoritmanın nasıl test edilebileceği, daha sonra yaptığımız kullanıcı bazlı bir deney ile ortaya konulmuştur. Bu deneyde algoritmanın geçerliliğinin nasıl değerlendirilebileceği, nasıl ölçeklenebilir

olduğunun ölçülebileceği, katılımcı profilinin ortak güzergâha olabilecek etkisinin nasıl ölçülebileceği gösterilmiştir. Bu deneyin sonuçları, eMine Algoritması'nın geçerliliğini göstermesinin yanında, benzer algoritmaların da değerlendirmesi için kullanılacak bir yöntem ortaya koymuştur.

**Böyle bir algoritmayı kullanarak sayfaları nasıl dönüştürebiliriz? Mobil ve görme özürli kullanıcılara web sayfalarını nasıl etkin bir biçimde gösterebiliriz?** Dönüştürme, bir sayfayı kısıtlı ortamlarda daha rahat erişilebilecek şekilde yeniden düzenleme işlemidir. Daha önceki çalışmalarımız web sayfalarını dönüştürmenin kullanıcının web deneyimini iyileştirdiğini göstermiştir (Yeşilada vd., 2008b, 2007; Plessers vd., 2005); fakat proje önerisinde de web sayfalarını tarama güzergâhlarına göre nasıl dönüştürebileceğimizi bilmediğimiz ve daha önce bu alanda hiçbir çalışmanın yapılmadığını söylenmişti. Bu bağlamda bizim yaptığımız çalışma bir ilktir. Özetleyecek olursak, çalışmalarımız ilk olarak var olan dönüştürme tekniklerini ortaya koyarak başlamıştır. Daha sonra bu tekniklerin hangilerinin göz izleme verileri ile kullanılabileceği ortaya koyulmuştur. Bu alanda yapılan ilk çalışma olduğu için de, tekniklerin olabildiğince açık ve kullanıcı bazlı olmalarına dikkat edilmiştir. Daha sonra, bu tekniklerin nasıl uygulanabileceği deneysel ortamda gösterilmiş ve kullanıcının web sayfaları ile etkileşimini nasıl olumlu yönde etkileyebileceği gösterilmiştir. Bu kapsamda yaptığımız çalışmalar aslında proje önerisinde sorulan aşağıdaki sorulara yanıt aramak doğrultusundadır:

1. *Bu yüzden, özellikle eMine Algoritması'nı kullanarak sayfaları dönüştürmek için hangi teknikleri kullanabiliriz?* Literatürde pek çok dönüştürme tekniği önerilmiştir. Fakat yaptığımız çalışma ortaya çıkarmıştır ki, bu tekniklerin bazıları göz izleme verileri ile sayfalarda dönüştürme yapmak için kullanılamaz. Yapılan ön çalışmada genel olarak görsel bloklar ve bu görsel blokların rolleri baz alınarak uygulanabilecek tekniklerin çok daha iyi sonuç verebilecekleri gözlemlenmiştir (Yeşilada vd., 2013; Akpınar ve Yeşilada, 2013d). Daha sonra bu tekniklerin uygulanabilmesi için bir sistem mimarisi önerilmiş ve nasıl uygulanabileceği deneysel platform üzerinde gösterilmiştir. Bu platform üzerindeki uygulamalar, başka araştırmacılar tarafından da etkin bir şekilde kullanılabilecek ve deneylere devam edebilecek şekilde tasarlanmışlardır. Fakat en önemlisi, göz izleme verileri kullanılarak sayfaların nasıl dönüştürülebileceğini gösteren bir sistem mimarisi ortaya koymak olmuştur. Biz bu mimarinin belli bir deneysel platform üzerinde uygulamasını gözlemlemiş olsak da, aslında bu mimari çok daha farklı platformlarda da uygulanabilir bir mimaridir.
2. *Bu dönüştürme teknikleri ile kullanıcıların deneyimlerini iyileştirebildik mi? Hangi bağlamda önerilen teknikler etkili olmuştur?* Uyguladığımız dönüştürme tekniklerinin

nasıl test edilebileceđi ve kullanıcı üzerindeki etkilerinin nasıl gösterilebileceđi araştırılmıřtır. Buradaki bulgularımız, mevcut teknik alt yapılar ile çok etkin bir řekilde ortaya koyabileceđimizi göstermiřtir. Bu uygulamalar daha çok teknik kriterleri, kullanıcı bazlı kriterleri simülasyon ile ortaya koymaktadır. Bu uygulamalar, göz izleme verilerinden yola çıkarak, dönüřtürölen sayfaların kullanıcı deneyimi üzerindeki etkilerini başarılı bir řekilde ortaya koymuřtur. Yaptıđımız alıřmalar sadece bizim uyguladıđımız dönüřtürme tekniklerinin ve uygulamalarının başarısını deđil, bu alanda alıřma yaparak deđerlendirme yapmak isteyen kiřiler tarafından da kullanılabilir bir alt yapı ortaya koymuřtur. Özet olarak, yaptıđımız alıřma kapsamında, sayfalara eriřim süresi, eriřim sırasında istemci uygulamanın CPU kullanımı, gönderilen toplam istek sayısı, indirilen toplam veri miktarı, kullanıcının bir sayfa ierisinde ortak güzergâhta yer alan elemanlara eriřim süresi gibi ölçümler yapılmıřtır. Bu ölçümler, orijinal sayfa yapısı ve dönüřtürölmüř halleri iin tekrarlanmıř ve sonuçlar karřılařtırılmıřtır. Sonuçlar genel olarak dönüřtürme tekniklerinin kullanıcının deneyimini olumlu yönde etkileyeceđini göstermektedir.



## 5. TARTIŞMA

eMine Projesi'nin hedefi göz izleme verilerini kullanarak, insanların tarama güzergâhlarını tespit ederek bunları sayfadaki elemanlarla ilişkilendirecek bir algoritma yaratmak ve bu algoritmayı kullanarak kısıtlı ortamlardaki kullanıcı deneyimini iyileştirmektir. Bu hedefe ulaşmak için proje süresince gerçekleştirilen dört amaç vardır. Bu bölüm bu amaçları, projemizin kısıtlamalarını ve bu amaçlar doğrultusunda nasıl geliştirilebileceğini tartışmaktadır.

Projedeki ilk amacımız sayfanın kodunu kullanarak sayfadaki görsel elemanları teşhis etmektir. Daha önce gerçekleştirmiş olduğumuz çalışmalarımızda, sayfadaki görsel elemanları teşhis etmek için bir çerçeve geliştirmiştik. Bu projede ise hedefimiz bu işlemi otomatikleştirmektir (Yeşilada vd., 2003, 2007, 2008b). Bir önceki bölümde de aktardığımız gibi, başarı ile bu amaca ulaşılmıştır (bkz. 3.1. Bölüm). Yaptığımız çalışmada bir web sayfasını öncelikle farklı seviyelerde bloklara otomatik olarak ayırabileceğimiz gösterilmiştir. Yaptığımız kullanıcı bazlı deneylerde, kullanıcıların belli seviyeleri diğerlerine göre daha fazla tercih ettikleri görülmüştür. Fakat yaptığımız çalışmada, kullanıcılara belli görevler çerçevesinde hangi seviyeyi tercih ettikleri sorulmamıştır. Bu yüzden, farklı bir çalışma ile kullanıcıların belli görevler doğrultusunda hangi seviyeyi tercih edebilecekleri araştırılabilir. Bu tür bir araştırmanın sonuçları da dönüştürme tekniklerinde belli seviyelerin belli durumlarda tercih edebilme şansını ortaya koyacaktır. Bu amaç doğrultusunda yaptığımız diğer bir çalışma ise görsel blokların rollerinin otomatik olarak atanmasıdır. Bu doğrultudaki çalışmamız olasılık bazlı bir çalışmadır ve sonuçta %80'lik bir başarı ile görsel elemanların rollerinin otomatik atanabileceğini göstermektedir. Fakat, ileride bu yönde, özellikle farklı bulgusal (heuristic) veriler kullanılarak, rol atama kısmı üzerinde farklı çalışmalar yapıp doğruluk oranı artırılmaya çalışılabilir.

Projedeki ikinci amacımız, tarama güzergâhlarını sayfadaki görsel elemanlar bakımından teşhis edecek özgün bir algoritma geliştirmektir. Daha önceki çalışmalarda insanların tarama güzergâhlarını incelerken "String-edit" algoritması kullanılmıştır (Josephson ve Holmes, 2002; Takeuchi ve Habuchi, 2007). Bu algoritma web sayfalarında başarı ile uygulanmış olsa da (Josephson ve Holmes, 2002; Takeuchi ve Habuchi, 2007), yapılan çalışmalar sayfaların resimleri üzerinden olup, hiçbir tarama güzergâhı, sayfadaki görsel elemanlar ve sayfanın kodu arasındaki ilişkileri araştırmamıştır. Bizim hedefimiz "String-edit" algoritmasından yola çıkarak, tarama güzergâhlarını sayfadaki görsel elemanlar bakımından teşhis edecek yeni bir algoritma oluşturmaktır. Bir önceki bölümde ve "Gereç ve Yöntem"

bölümünde de (bkz. 3.2. Bölüm) detaylandırdığımız gibi bu amaç doğrultusunda “String-edit” algoritması temel alınarak orijinal bir algoritma geliştirilmiştir. Buradaki çalışmanın diğer çalışmalardan farkı, ortak güzergâhları web sayfalarının görsel elemanları ve kodları ile ilişkilendirerek çıkarıyor olmasıdır. Yaptığımız çalışmalar önerdiğimiz algoritmanın diğerlerine göre çok daha az indirgemeci olduğunu göstermiştir. Fakat bu doğrultuda çok daha fazla kullanıcı ile çalışmalar yapıp, bu konu çok daha derin araştırılabilir. Ayrıca ortak tarama güzergâhı çıkarmak üzere farklı yaklaşımlarla değişik algoritmalar da geliştirilebilir. Bizim önerdiğimiz çalışma, web sayfalarının bizim bloklara bölme çalışmamız kullanılarak bölünmesi ile başlamaktadır. Bu da, çıkan ortak güzergâhların, bizim bloklara bölme yaklaşımımız ile çok fazla ilişkilendirilmiş olduğu anlamına gelmektedir. Fakat farklı bir blok ayırma işlemi/algoritması ile de farklı bir çalışma yapıp, bizim bloklara ayırma yaklaşımımız ile karşılaştırılabilir. Son olarak, yaptığımız araştırmada göz tarama verileri üzerinde çok farklı yöntemlerle ön işleme yapılabileceği ortaya çıkmıştır. Bu araştırma kapsamında, bu tekniklerin çok küçük bir kısmını kullanılmıştır. Örneğin, belli sürenin altındaki odaklanmalar veriler içerisinden çıkarılmıştır. Fakat, bu konuda çok daha sistematik ve farklı yöntemler kullanarak veriler çok daha farklı yöntemlerle ön işlemden geçirilebilir. Hem bu yöntemler, hem de bu yöntemlerin ortak güzergâha etkileri daha detaylı araştırılabilir.

Projedeki üçüncü amacımız yeni algoritmamızın temel teşkil edeceği özgün bir dönüştürme tekniği geliştirmektir. Değişik araştırmalarda çeşitli dönüştürme teknikleri üzerinde çalışmalar yapılmış olmasına rağmen, bildiğimiz kadarı ile bu çalışmaların hiçbiri web sayfalarını dönüştürmek için göz izleme verileri kullanmamıştır. Bizim çalışmamız bir ilk olup, farklı tekniklerin, özgün bir sistem mimarisi ile nasıl uygulanabileceğini göstermiştir (bkz. 3.3. Bölüm). Fakat web sayfalarında kullanılan yeni teknolojilerin çok daha detaylı araştırılıp, bu teknolojilerin çok daha iyi anlaşılmasıyla farklı teknikler de önerilebilir. Bizim mimarimiz, zaten bu gibi durumlar göz önüne alınarak, açık bir mimari olarak tasarlanmıştır.

Projedeki dördüncü ve son amacımız da kısıtlı ortamlarda geliştirilen dönüştürme tekniklerinin kullanıcı deneyimini iyileştirdiğini göstermektir. Web sayfalarını teknik olarak dönüştürebiliyor olsak da (Takagi vd., 2007; Yeşilada vd., 2007), hala bu tekniklerin uygulamalarının kullanıcı deneyimini iyileştirdiğini göstermek zorundayız ve bu da en son amacımızdır. “Gereç ve Yöntem” bölümünde de (bkz. 3.3. Bölüm) detaylandırdığımız gibi bu amaç doğrultusunda birçok farklı yöntemle kullanıcı deneyimlerinin önerilen dönüştürme tekniklerinden nasıl etkileneceği gösterilmiştir. Yapılan araştırmalar tatminkar sonuçlar verse de, farklı kullanıcı bazlı araştırmalar ve deneyler uygulanıp, önerilen dönüştürme tekniklerinin etkileri farklı boyutlarda incelenebilir.

## 6. SONUÇ

Bu sonuç raporu TÜBİTAK tarafından desteklenen “eMine: Eye Tracking Teknolojisi kullanarak Web Sayfalarının Dönüştürülmesi” projesini sunmaktadır. Genel olarak proje kapsamında yapılan literatür taramasını, projede takip edilen yöntemleri, kullanılan gereç ve teknikleri, elde edilen bulguları ve sonuçları sunup, tartışmaktadır.

eMine Projesi’nin hedefi göz izleme teknolojisinin özgün bir uygulamasını kullanarak kısıtlı ortamlarda kullanıcı deneyimini iyileştirmektir. Özet olarak amacı, web sayfalarının kodu ile insanların tarama güzergâhını ilişkilendirerek, sayfaları kısıtlı ortamlarda daha rahat erişilebilecek bir şekle dönüştürmektir. Bu sonuç raporunun da anlattığı gibi, eMine Projesi önerilen hedefine varmış ve şu sonuçları elde etmiştir: (1) Sayfanın kodu kullanılarak sayfadaki görsel elemanlar otomatik olarak çıkarılabilmekte ve %80 gibi bir doğrulukla rolleri otomatik olarak atanabilmektedir; (2) Tarama güzergâhları sayfanın kodu ile otomatik olarak ilişkilendirebilmekte ve bu ilişki doğrultusunda kullanıcılar arasında ortak bir güzergâh çıkarılabilmektedir; (3) Bu ortak güzergâh, görsel elemanlar ve otomatik olarak ortaya çıkarılmış roller farklı dönüştürme teknikleri ile başarı ile kullanılabilmektedir; (4) Yapılan çalışmalarda bu tekniklerin kullanıcının deneyimini olumlu yönde etkileyeceği gösterilmektedir. Bu sonuçlar projenin başarı ile sonuçlandığını ve hedefine ulaştığını göstermektedir. Başarılı bir proje olarak da gelecekte birçok yönden geliştirilebilir: otomatik rol atama kısmı üzerinde farklı çalışmalar yapıp doğruluk oranı arttırılmaya çalışılabilir, ortak tarama güzergâhı çıkarmak üzere farklı yaklaşımlarla değişik algoritmalar geliştirilebilir, bunları kullanarak da farklı dönüştürme teknikleri ve deneyleri yapılabilir.

## 7. KAYNAKLAR

Adzic V., Kalva H., Furht B. 2011. "A survey of multimedia content adaptation for mobile devices", *Multimedia Tools and Applications* 51, 379–396.

Ahmadi, H., Kong, J. 2008. "Efficient web browsing on small screens". *Proceedings of the working conference on Advanced visual interfaces*, 23–30, AVI '08. ACM.

Akpınar, E., Yeşilada, Y. 2012. "Vision Based Page Segmentation: Extended and Improved Algorithm". *Teknik Repor* - <http://wel-eprints.cs.manchester.ac.uk/154/>. Middle East Technical University Northern Cyprus Campus.

Akpınar, E., Yeşilada, Y. 2013a. "Vision Based Page Segmentation Algorithm: Extended and Perceived Success". *Current Trends in Web Engineering - ICWE 2013 Workshops and PhD Symposium. Lecture Notes in Computer Science*, vol. 8295, pp. 238-252. Editör: Sheng Q. Z. ve Kjeldskov J. Springer.

Akpınar, E., Yeşilada, Y. 2013b. "Evaluation of Automatic Discovery of Visual Elements of Web Pages". *Teknik Repor* - <http://wel-eprints.cs.manchester.ac.uk/162/>. Middle East Technical University Northern Cyprus Campus.

Akpınar, E., Yeşilada, Y. 2013c. "Heuristic role detection of visual elements of web pages". *Web Engineering. Lecture Notes in Computer Science*, vol. 7977, pp. 123-131. Editör: Daniel, F. ve Dolog, P., Li, Q. Springer.

Akpınar, E., Yeşilada, Y. 2013d. "Web Page Transcoding Based on Eye Tracking". *Teknik rapor* - <http://wel-eprints.cs.manchester.ac.uk/165/>. Middle East Technical University Northern Cyprus Campus.

Akpınar, E., Yeşilada, Y. 2013e. "Implementation and Evaluation of Web Page Transcoding Based on Eye Tracking". *Teknik rapor* - <http://wel-eprints.cs.manchester.ac.uk/166/>. Middle East Technical University Northern Cyprus Campus.

Akpınar, E., Yeşilada, Y. 2013f. "Automatic Discovery of Visual Elements of Web Pages". *Teknik Repor* - <http://wel-eprints.cs.manchester.ac.uk/161/>. Middle East Technical University Northern Cyprus Campus.

Asakawa, C., Takagi, H. 2000. "Annotation-based transcoding for nonvisual web access". ASSETS' 00, 172–179. ACM.

Asakawa, C., Takagi, H. 2008. "Transcoding". Web Accessibility: A Foundation for Research. Editör: Harper, S., Yeşilada, Y. Human-Computer Interaction Series, pp. 231–260. Springer.

Baudisch, P., Xie, X., Wang, C., Ma, W. 2004. "Collapse-to-zoom: Viewing web pages on small screen devices by interactively removing irrelevant content". In 17th Annual ACM Symposium on User Interface Software and Technology (UIST 2004). ACM.

Baluja, S. 2006. "Browsing on small screens: Recasting web-page segmentation into an efficient machine learning framework". Proceedings of the 15th International Conference on World Wide Web, pages 33–42. ACM.

Borodin, Y., Mahmud, J., Ramakrishnan, IV., Stent, A. 2007. "The hearsay non-visual web browser". Proceedings of the 2007 international cross-disciplinary conference on Web accessibility (W4A), W4A '07, pp 128–129. ACM.

Brown, B., Jay, C., Harper, S. 2012. "Tailored presentation of dynamic web content for audio browsers". International Journal of Human-Computer Studies, 70(3):179 – 196.

Burget, R., Rudolfova, I. 2009. "Web page element classification based on visual features". Proceedings of the 2009 First Asian Conference on Intelligent Information and Database Systems, 67 –72.

Büyükkökten, O., Garcia-Molina, H., Paepcke, A., Winograd, T. 2000. "Power browser: Efficient web browsing for PDAs". Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pages 430–437. ACM.

Büyükkökten, O., Garcia-Molina, H., Paepcke, A. 2001. "Accordion summarization for end-game browsing on PDAs and cellular phones". Proceedings of the SIGCHI conference on Human factors in computing systems, pages 213–220. ACM.

Büyükkökten, O., Garcia-Molina, H., Paepcke, A., Winograd, T. 2002. "Efficient web browsing on handheld devices using page and form summarisation". ACM Transactions on Information Systems, 20(1):82–115. ACM.

Cai, D., He, X., Li, Z., Ma, W.Y., Wen, J.R. 2004a. "Hierarchical clustering of www image search results using visual, textual and link information". Proceedings of the 12th annual ACM international conference on Multimedia, 952–959, MULTIMEDIA '04. ACM.

Cai, D., He, X., Wen, J.R., Ma, W.Y. 2004b. "Block-level link analysis". Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, 440–447, SIGIR '04. ACM.

Cai, D., Yu, S., Wen, J., Ma, W. 2003. "VIPS: a vision based page segmentation algorithm". Teknik Rapor MSR-TR-2003-79, Microsoft Research.

Chen, Y., Ma, W., Zhang, H. 2003. "Detecting web page structure for adaptive viewing on small form factor devices". In Proceedings of the Twelfth International World Wide Web Conference. ACM.

Chen, Y., Ma, W., Zhang, H. 2003. "Detecting web page structure for adaptive viewing on small form factor devices". Proceedings of the Twelfth International World Wide Web Conference. ACM.

Chen, J., Zhou, B., Shi, J., Zhang, H., Wu, Q. 2001. "Function-based object towards website adaptation". Proceedings of the Tenth International World Wide Web Conference. ACM.

Chen, Y., Xie, X., Ma, W.Y., Zhang, H.J. 2005. "Adapting web pages for small-screen devices". IEEE Internet Computing 9, 50–56.

Chiang, C. 2009. "A genetic algorithm for the longest common subsequence of multiple sequences". Master's thesis, National Sun Yat-sen University.

Chuter, A., Yeşilada, Y. 2008. "Relationship between Mobile Web Best Practices (MWBP) and Web Content Accessibility Guidelines (WCAG)". <http://www.w3.org/TR/mwbp-wcag/>. W3C. Son erişim tarihi: 05 Aralık 2013.

Coyne, K., Nielsen, J. 2001. "Beyond ALT text: Making the web easy to use for users with disabilities". Nielson Norman Group. <http://www.nngroup.com/reports/usability-guidelines-accessible-web-design/>. Son erişim tarihi: 05 Aralık 2013.

Cristino, F., Mathot, S., Theeuwes, J., Gilchrist, I.D. 2010. "Scanmatch: a novel method for comparing fixation sequences". *Behav. Res. Methods* 42(3), 692–700.

Day, R.F. 2010. "Examining the validity of the needleman&wunsch algorithm in identifying decision strategy with eye-movement data". *Decision Support Systems* 49(4), 396–403.

Disability Rights Commission (DRC). 2004. "The web: Access and inclusion for disabled people".

Eraslan, Ş., Yeşilada, Y., Harper, S. 2013a. "eMINE Scanpath Analysis Algorithm". Teknik Rapor - <http://wel-eprints.cs.manchester.ac.uk/163>. Middle East Technical University Northern Cyprus Campus ve University of Manchester.

Eraslan, Ş., Yeşilada, Y., Harper, S. 2013b. "Analysis of algorithms to identify patterns in eye-tracking scanpaths". *Signal Processing and Communications Applications Conference (SIU)*, 2013 21st. pp. 1-4.

Eraslan, Ş., Yeşilada, Y., Harper, S. 2013c. "Correlation Between Scanpaths and Visual Elements of Web Pages". Teknik Rapor - <http://wel-eprints.cs.manchester.ac.uk/164>. Middle East Technical University Northern Cyprus Campus ve University of Manchester.

Eraslan, Ş., Yeşilada, Y., Harper, S. 2013d. "Understanding Eye Tracking Data for Re-engineering Web Pages". *Current Trends in Web Engineering - ICWE 2013 Workshops and PhD Symposium. Lecture Notes in Computer Science*, vol. 8295, pp. 345-349. Editör: Sheng Q. Z. ve Kjeldskov J. Springer.

European Union Policy Survey. "eAccessibility of Public Sector Services in the European Union". <http://m.epractice.eu/en/library/281630>. Son erişim tarihi: 05 Aralık 2013.

Friedman-Hill, E. 2008. "JESS: the rule engine for the Java platform". <http://herzberg.ca.sandia.gov/>, Son erişim tarihi: 04 Aralık 2013.

Goldberg, J., Helfman, J. 2010. "Scanpath clustering and aggregation". *Proceedings of the 2010 Symposium on Eye-Tracking Research and Applications, ETRA '10*, pages 227–234. ACM.

Goldberg, J., Stimson, M., Lewenstein, M., Scott, N., Wichansky, A. 2002. "Eye tracking in web search tasks: design implications". ETRA '02: Proceedings of the 2002 symposium on Eye tracking research & applications, pages 51–58. ACM.

Granka, L., Joachims, T., Gay, G. 2004. "Eye-tracking analysis of user behavior in www Search". SIGIR '04, pages 478–479. ACM.

Harper, S., Bechhofer S., Lunn. 2006. "Taming the inaccessible web". Proceedings of the 24th annual ACM international conference on Design of communication. SIGDOC '06, pp 64–69. ACM.

Hattori, G., Hoashi, K., Matsumoto, K., Sugaya, F. 2007. "Robust web page segmentation for mobile terminal using content-distances and page layout information". Proceedings of the 16th international conference on World Wide Web, pages 361–370. ACM.

Hembrooke, H., Feusner, M., Gay, G. 2006. "Averaging scan patterns and what they can tell us". Proceedings of the 2006 symposium on Eye tracking research & applications, ETRA '06, pages 41–41. ACM.

Hori, M., Kondoh, G., Ono, K. 2000. "Annotation-based web content transcoding". Proceedings of the Ninth International World Wide Web Conference, pages 197–211. ACM.

Hwang, Y., Kim, J., Seo, E. 2003. "Structure-aware web transcoding for mobile devices". IEEE Internet Computing 7(5), 14–21.

Jay, C., Brown, A. 2008. "User review document: Results of initial sighted user investigations". Teknik rapor - <http://hew-eprints.cs.man.ac.uk/49/>. The University of Manchester, 2008.

Jay, C., Stevens, R., Glencross, M., Chalmers, A. 2006. "How people use presentation to search for a link: expanding the understanding of accessibility on the web". 2006. Proceedings of the 2006 international cross-disciplinary workshop on Web accessibility (W4A), pages 113–120. ACM.



Josephson, S., Holmes, M.E. 2002. "Visual attention to repeated internet images: testing the scanpath theory on the world wide web". ETRA 2002, pp. 43–49. ACM.

Josephson, S., Holmes, M. 2004. "Age differences in visual search for information on web Pages". ETRA '04: Proceedings of the 2004 symposium on Eye tracking research & applications, pages 62–62. ACM.

Kovacevic, M., Diligenti, M., Gori, M., Milutinovic, V. 2002. "Recognition of common areas in a web page using visual information: a possible application in a page classification". Proceedings 2002 IEEE International Conference on Data Mining, Washington, DC, USA, IEEE Computer Society, 250–257.

Levenshtein, V. 1966. "Binary codes capable of correcting deletions, insertions, and reversals". Soviet Physics Doklady, 10(8):707–710.

Lunn, D., Bechhofer, S., Harper, S. 2008a. "The sadie transcoding platform". W4A '08: Proceedings of the 2008 international cross-disciplinary conference on Web accessibility (W4A) pp 128–129. ACM.

Lunn, D., Bechhofer, S., Harper, S. 2008b. "A user evaluation of the sadie transcoder". Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility, Assets '08, pp 137–144. ACM.

Mast, M., Burmeister, M. 2011. "Exposing repetitive scanning in eye movement sequences with t-pattern detection". IADIS IHCI 2011, Rome, Italy, pp. 137–145.

McCarthy, J., Sasse, M., Riegelsberger, J. 2003. "Could I Have the Menu Please? An Eyetracking Study of Design Conventions". HCI2003, pages 401–414.

Michailidou, E. 2010. "ViGRAM: Visual Complexity Rankings and Accessibility Metrics". PhD thesis.

Mahmud, J., Borodin, Y., Das, D., Ramakrishnan, IV. 2007a. "Combating information overload in non-visual web access using context". Proceedings of the 12th international conference on Intelligent user interfaces, IUI '07, pp 341–344. ACM.

Mahmud, J., Borodin, Y., Ramakrishnan, IV. 2007b. "Csurf: a context-driven non-visual webbrowser". Proceedings of the 16th international conference onWorldWideWeb. WWW '07, pp 31–40. ACM.

Lin, S.H., Ho, J.M. 2002. "Discovering informative content blocks from web documents". Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, 588–593. ACM.

Liu, B., Chin, C.W., Ng, H.T. 2003. "Mining topic-specific concepts and definitions on the web". Proceedings of the 12th international conference onWorldWideWeb, ACM, 251–260.

Lunn, D., Harper, S., Bechhofer, S. 2011. "Identifying behavioral strategies of visually impaired users to improve access to web content". ACM Trans. Access. Comput. 3(4), 13:1–13:35.

Mahmud, J.U., Borodin, Y., Ramakrishnan, I.V. 2007. "Csurf: a context-driven non-visual webbrowser". Proceedings of the 16th international conference on World Wide Web, 31–40. WWW '07. ACM.

Milic-Frayling, N., Sommerer, R. 2002. "Smartview: Flexible viewing of web page contents", In: Poster Proceedings of the Eleventh International World Wide Web Conference. ACM.

Plessers, P., Casteleyn, S., Yeşilada, Y., Troyer, O., Stevens, R., Harper, H., Goble, C. 2005. "Accessibility: A web engineering approach". Proceedings of the 14th international conference on World Wide Web, WWW 2005, pages 353–362. ACM.

Pan, B., Hembrooke, H., Gay, G., Granka, L., Feusner, M., Newman, K. 2004. "The determinants of web page viewing behavior: an eye-tracking study". ETRA '04, pages 147–154. ACM.

Pirolli, P., Cart, S. 1999. "Information foraging". Psychological Review, 106(4).

Poole, A., Ball, L.J. 2005. "Eye tracking in human-computer interaction and usability research: Current status and future". Editor: Ghaoui, C. Prospects. Encyclopedia of Human-Computer Interaction. Idea Group, Inc., Pennsylvania.

Raiha, K.-J. 2010. "Some applications of string algorithms in human-computer interaction". In: Elomaa, T., Mannila, H., Orponen, P. (eds.) Ukkonen Festschrift 2010. LNCS, vol. 6060, pp. 196–209. Springer.

Rayner, K. 1998. "Eye movements in reading and information processing: 20 years of research". *Psychological bulletin*, pages 372–422.

Regan, B. 2004. "Accessibility and design: A failure of the imagination". Proceedings of the 2004 International Cross-Mark Russell. Using eye-tracking data to understand first impressions of a website. *Usability News* 7.1, 7(1), 2005.

Richards, J., Hanson, V. 2004. "Web accessibility: A broader view". Proceedings of the Fourteenth International World Wide Web Conference, pages 72–79. ACM.

Song, R., Liu, H., Wen, J.R., Ma, W.Y. 2004. "Learning block importance models for web pages". Proceedings of the 13th international conference on World Wide Web, 203–211. WWW '04. ACM.

Takagi, H., Asakawa, C., Fukuda, K., Maeda, K. 2002. "Site-wide annotation: Reconstructing existing pages to be accessible". *ASSETS'02*, pages 81–88. ACM.

Takagi, H., Saito, S., Fukuda, K., Asakawa, C. 2007. "Analysis of navigability of web applications for improving blind usability". *ACM Transactions on Computer-Human Interaction*, 14(3):13. ACM.

Takeuchi, H., Habuchi, Y. 2007. "A quantitative method for analyzing scan path data obtained by eye tracker". In: *CIDM 2007*, March 1-April 5, pp. 283–286.

Tsang, H.Y., Tory, M., Swindells, C. 2010. "eseetrack: Visualizing sequential fixation patterns". *IEEE Transactions on Visualization and Computer Graphics* 16(6), 953–962.

Santella, A., DeCarlo, D. 2004. "Robust clustering of eye movement recordings for quantification of visual interest". *ETRA 2004*, pp. 27–34. ACM.

Vigo, M., Harper, S. 2013a. "Coping Tactics Employed by Visually Disabled Users on the Web". *International Journal of Human-Computer Studies* 71(11), 1013-1025. Elsevier.

Vigo, M., Harper, S. 2013b. "Evaluating Accessibility-in-Use". 10th International Cross-Disciplinary Conference on Web Accessibility, W4A 2013, article 7. ACM.

Whang, Y., Jung, C., Kim, J., Chung, S. 2001. "Webalchemist: A web transcoding system for mobile web access in handheld devices". Optoelectronic and Wireless Data Management, Processing, Storage, and Retrieval, 102–109.

West, J., Haake, A., Rozanski, E., Karn, K. 2006. "eye Patterns: Software for Identifying Patterns and Similarities Across Fixation Sequences". Eye-Tracking Research and Applications (ETRA) conference, pages 149–154, San Diego, California.

Xiang, P., Yang, X., Shi, Y. 2007. "Web page segmentation based on gestalt theory". Multimedia and Expo 2007 IEEE International Conference (ICME).

Xiang, P., Shi, Y. 2006. "Recovering semantic relations from web pages based on visual cues". Proceedings of the 11th international conference on Intelligent user interfaces, 342–344, IUI '06. ACM.

Xiao, X., Luo, Q., Hong, D., Fu, H. 2005. "Slicing\*-tree based web page transformation for small displays". Proceedings of the 14th ACM international conference on Information and knowledge management, 303–304. CIKM '05. ACM.

Xiao, Y., Tao, Y., Li, Q. 2008a. "Web page adaptation for mobile device". Wireless Communications, Networking and Mobile Computing.

Xiao, Y., Tao, Y., Li, W. 2008b. "A dynamic web page adaptation for mobile device based on web2.0". Proceedings of the 2008 Advanced Software Engineering and Its Applications, 119–122. IEEE Computer Society.

Xie, X., Miao, G., Song, R., Wen, J.R., Ma, W.Y. 2005. "Efficient browsing of web search results on mobile devices based on block importance model". Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications, 17–26, IEEE Computer Society.

Yang, X., Shi, Y. 2009. "Enhanced gestalt theory guided web page segmentation for mobile browsing"ç Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 03, 46–49, WI-IAT '09. IEEE Computer Society.

Yang, C., Wang, L. 2003. "Fractal summarization for mobile devices to access large documents on the web". Proceedings of the Twelfth International World Wide Web Conference. ACM.

Yeşilada, Y. 2011a. "Web Page Segmentation: A Review". Teknik Rapor - <http://wleprints.cs.manchester.ac.uk/148/>. University of Manchester and Middle East Technical University Northern Cyprus Campus.

Yeşilada, Y. 2011b. "Heuristics for Visual Elements of Web Pages". Teknik Repor - <http://wleprints.cs.manchester.ac.uk/149/>. University of Manchester and Middle East Technical University Northern Cyprus Campus.

Yeşilada, Y. 2013. "Web Page Segmentation – Approaches, Techniques and Algorithms", Submitted to ACM Transactions on the Web.

Yeşilada, Y., Chuter, A., Henry, S. 2008a. "Shared Web Experiences: Barriers Common to Mobile Device Users and People with Disabilities". <http://www.w3.org/WAI/mobile/experiences>. W3C. Son erişim taihi: 05 Aralık 2013.

Yeşilada, Y., Harper, S., Eraslan, Ş. 2013. "Experiential transcoding: an eyetracking approach". Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility. pp. 30:1-30:4. ACM, New York, NY, USA.

Yeşilada, Y., Stevens, R., Goble, C. 2003. "A foundation for tool based mobility support for visually impaired web users". Proceedings of the Twelfth International World Wide Web Conference, WWW2003, pages 422–430. ACM.

Yeşilada, Y., Harper, S., Goble, C., Stevens, R. 2004. "Screen readers cannot see (ontology based semantic annotation for visually impaired web travellers)". Proceedings of the International Conference on Web Engineering (ICWE), pages 445–458. Springer.

Yeşilada, Y., Stevens, R., Harper, S., Goble, C. 2007. "Evaluating DANTE: Semantic transcoding for visually disabled users". *ACM Transactions on Computur-Human Interaction*, 14(3):14. ACM.

Yeşilada, Y., Jay, C., Stevens, R., Harper, S. 2008b. "Validating the use and role of visual elements of web pages in navigation with an eye-tracking study". In *17th International World Wide Web Conference*, Beijing, China. ACM.

Yi, L., Liu, B., Li, X. 2003. "Eliminating noisy information in web pages for data mining". *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 296–305. ACM.

Yin, X., Lee, W. 2004. "Using link analysis to improve layout on mobile devices". *Proceedings of the Thirteenth International World Wide Web Conference*, pages 338–344. ACM.

Yin, X., Lee, W.S. 2005. "Understanding the function of web elements for mobile content delivery using random walk models". *Special interest tracks and posters of the 14th international conference on World Wide Web*, 1150–1151, WWW '05. ACM.

## **8. EKLER**

### **8.1 Web Page Segmentation: A Review**

## **8.2 Heuristics for Visual Elements of Web Pages**



### **8.3 Vision Based Page Segmentation: Extended and Improved Algorithm**

## **8.4 Automatic Discovery of Visual Elements of Web Pages**

## **8.5 Evaluation of Automatic Discovery of Visual Elements of Web Pages**

## **8.6 eMINE Scanpath Analysis Algorithm**

## **8.7 Correlation Between Scanpaths and Visual Elements of Web Pages**

## **8.8 Web Page Transcoding Based on Eye Tracking**

## **8.9 Implementation and Evaluation of Web Page Transcoding Based on Eye Tracking**

## **8.10 Heuristic Role Detection of Visual Elements of Web Pages**



## **8.11 Experiential Transcoding: An Eye Tracking Approach**

## **8.12 Vision Based Page Segmentation Algorithm: Extended and Perceived Success**

### **8.13 Analysis of Algorithms to Identify Patterns in Eye Tracking Scanpaths**

## **8.14 Web Page Segmentation – Approaches, Techniques and Algorithms**