

T.C.  
FIRAT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

169292

**WEB TABANLI PROGRAMLAMA DİLLERİNİN  
DEĞERLENDİRİLMESİ VE PERFORMANS ANALİZİ**

Ferhat BAĞÇACI

Tez Yöneticisi  
Prof. Dr. Asaf VAROL

YÜKSEK LİSANS TEZİ  
ELEKTRONİK ve BİLGİSAYAR EĞİTİMİ ANABİLİM DALI

ELAZIĞ, 2005

**FIRAT UNIVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**WEB TABANLI PROGRAMLAMA DİLLERİNİN DEĞERLENDİRİLMESİ VE**  
**PERFORMANS ANALİZİ**

**Ferhat BAĞÇACI**

**YÜKSEKLİSANS TEZİ**

**ELEKTRONİK ve BİLGİSAYAR EĞİTİMİ**

**ANABİLİM DALI**

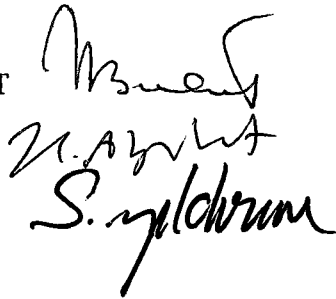
Bu tez, 26.08.2005 tarihinde aşağıda belirtilen jüri tarafından oybirliği /oyçokluğu ile başarılı / başarısız olarak değerlendirilmiştir.

Danışman: Prof. Dr. Asaf VAROL

Üye: Doç.Dr. Muammer GÖKBULUT

Üye: Doç.Dr. Z.Hakan AKPOLAT

Üye: Yrd.Doç.Dr. Selçuk YILDIRIM



Bu tezin kabulü, Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 07/10/2005 tarih ve 2005-29/1 sayılı kararıyla onaylanmıştır.

## TEŞEKKÜR

Bu tez çalışması süresince benden yardımlarını esirgemeyen danışman hocam Sayın Prof. Dr. Asaf VAROL'a, maddi, manevi her türlü desteği sağlayan çok sevdiğim ve değer verdiğim aileme teşekkürü bir borç bilirim.

Ferhat BAĞÇACI



## İÇİNDEKİLER

Sayfa

### TEŞEKKÜR

### İÇİNDEKİLER .....I

### ŞEKİLLER LİSTESİ .....V

### TABLolar LİSTESİ .....VII

### KISALTMALAR LİSTESİ .....VIII

### ÖZET .....IX

### ABSTRACT .....X

## 1. GİRİŞ ..... 1

### 1.1. WEB TABANLI YAZILIMLARIN ÖZELLİKLERİ ..... 5

#### 1.1.1. Web Tabanlı Yazılımlara Genel Bakış ..... 5

#### 1.1.2. Web Tabanlı Yazılımların Çalışma Prensipleri ..... 5

#### 1.1.3. Web Tabanlı Yazılımların Avantajları ..... 5

##### 1.1.3.1. Düşük İlk Maliyet ve Bakım Maliyeti ..... 6

##### 1.1.3.2. Erişilebilirlik ..... 7

##### 1.1.3.3. Bakım ve Kurulum Kolaylığı ..... 7

#### 1.1.4. Sunucu Tabanlı Uygulama Geliştirme ..... 7

## 2. WEB TABANLI YAZILIMLAR ..... 9

### 2.1. ASP ..... 9

#### 2.1.1. ASP ve IIS ilişkisi ..... 9

#### 2.1.2. ASP Sayfalarının İşlenmesi ..... 10

#### 2.1.3 ASP'nin Dili ..... 11

#### 2.1.4 ASP'nin Unsurları ..... 12

#### 2.1.5. ASP İçin Gereklilikler ..... 13

#### 2.1.6. ASP'nin Kullanım Alanları ve Avantajları ..... 13

#### 2.1.7. ASP İle Bileşen Kullanma ..... 14

#### 2.1.8. ASP'de Oturum Kavramı ..... 14

### 2.2. PHP (Personel Home Page) ..... 15



|                                                                                             |    |
|---------------------------------------------------------------------------------------------|----|
| 2.2.1. PHP Dilinin Yapısı.....                                                              | 16 |
| 2.2.2. PHP'nin Tercih Nedenleri.....                                                        | 17 |
| 2.2.2.1. PHP'nin Çalıştırılabileceği Ortamlar .....                                         | 18 |
| 2.2.2.2. Yüksek Performans .....                                                            | 18 |
| 2.2.2.3. PHP'in Büyük Ölçekli Veri Tabanı Uygulamalarındaki Konumu .....                    | 19 |
| 2.2.2.4. İnternet Standartlarına Uyumluluk.....                                             | 20 |
| 2.2.2.5. PHP'in Gelişmiş Özellikleri.....                                                   | 20 |
| 2.2.2.6. PHP'in Ekonomikliği .....                                                          | 20 |
| 2.2.3. PHP ile Uygulama Geliştirme .....                                                    | 21 |
| 2.2.3.1. MySQL ile Veri Tabanı Uygulamaları.....                                            | 21 |
| 2.2.3.2. LDAP Kullanımı .....                                                               | 22 |
| 2.2.3.3. Oturum Yönetimi.....                                                               | 23 |
| 2.2.4. PHP'de Güvenlik.....                                                                 | 24 |
| 2.2.5. PHP ile Veritabanı.....                                                              | 25 |
| 2.2.5.1. PHP-MySQL İlişkisi .....                                                           | 26 |
| 2.3. CGI (Comman Gateway Interface) .....                                                   | 27 |
| 2.3.1. CGI İle Hazırlanan Kullanıcı Arayüzlerinin Başlıca Avantajları ve Dezavantajları. 28 |    |
| 2.3.1.1. Avantajları.....                                                                   | 28 |
| 2.3.1.2. Dezavantajları.....                                                                | 29 |
| 2.3.2. CGI'ın Önemi ve İşlevi .....                                                         | 29 |
| 2.3.3. CGI Programcıları İçin Bir Programlama Dili Seçme İşleminin Gerçekleştirilmesi. 32   |    |
| 2.3.4. CGI Programcılarının Kullanılma Sebepleri .....                                      | 33 |
| 2.3.5. CGI Programlamanın Özellikleri.....                                                  | 33 |
| 2.3.6. Ortam Değişkenleri .....                                                             | 35 |
| 2.3.7. Standart CGI Girdisi.....                                                            | 37 |
| 2.3.8. CGI Başlıkları.....                                                                  | 37 |
| 2.3.8.1. CGI Başlıklarında Kullanılan Content-Type'lar .....                                | 37 |
| 2.3.8.2. CGI Başlıklarında Kullanılan Location'lar .....                                    | 38 |
| 2.3.8.3. CGI Başlıklarında Kullanılan Status'lar.....                                       | 38 |
| 2.3.9. CGI'in Alternatifleri .....                                                          | 39 |
| 2.3.10. CGI'in Çalıştırıldığı Web Sunucular.....                                            | 41 |

|                                                           |           |
|-----------------------------------------------------------|-----------|
| 2.4. ASP, PHP, CGI Kurulumu ve Çalıştırılması.....        | 43        |
| 2.4.1. ASP'nin Çalıştırılması.....                        | 43        |
| 2.4.1.1. PWS Kurulumu .....                               | 43        |
| 2.4.1.2. IIS Kurulumu.....                                | 47        |
| 2.4.2. PHP'nin Çalıştırılması.....                        | 48        |
| 2.4.3. CGI'nin Çalıştırılması .....                       | 53        |
| <b>3. WEB TABANLI YAZILIMLARIN KARŞILAŞTIRILMASI.....</b> | <b>54</b> |
| 3.1. ASP, PHP, CGI'ın Karşılaştırılması.....              | 54        |
| 3.1.1. ASP.NET ve Avantajları .....                       | 54        |
| 3.1.1.1. Daha Hızlı Web Uygulamaları .....                | 54        |
| 3.1.1.2. Güçlü Veritabanı Fonksiyonları .....             | 54        |
| 3.1.1.3. Hafıza Sızıntısı ve Çökme Koruması .....         | 54        |
| 3.1.1.4. Çoklu Dil Desteği.....                           | 55        |
| 3.1.1.5. .Net'in Yapısı.....                              | 55        |
| 3.1.1.6. HTML'in ASP'deki Durumu .....                    | 55        |
| 3.1.2. ASP.NET'in Dezavantajları.....                     | 55        |
| 3.1.2.1. Windows Zorunluluğu.....                         | 55        |
| 3.1.2.2. Güvenlik .....                                   | 55        |
| 3.1.2.3. Lisans Gereksinimi.....                          | 55        |
| 3.1.2.4. Eklenti Sorunu.....                              | 56        |
| 3.1.3. PHP (Personal Home Page).....                      | 56        |
| 3.1.3.1. Çöp Toplayıcısı .....                            | 58        |
| 3.1.3.2. PHP'nin Avantajları .....                        | 58        |
| 3.1.3.3. PHP'nin Dezavantajları.....                      | 60        |
| 3.1.3.4. Performans.....                                  | 60        |
| 3.1.4. ASP/PHP Karşılaştırılması.....                     | 61        |
| 3.1.5. CGI (Common Gateway Interface) .....               | 61        |
| 3.1.5.1. CGI'da Güvenlik .....                            | 61        |

|                                                                                 |            |
|---------------------------------------------------------------------------------|------------|
| <b>4. UYGULAMALAR.....</b>                                                      | <b>63</b>  |
| 4.1. Uygulama-1 (0-10.000.000 Arası Sayan Döngü).....                           | 63         |
| 4.2. Uygulama-2 (20.000 Adet JPG Resmi Yükleme) .....                           | 72         |
| 4.3. Uygulama-3 (Avi yükleme).....                                              | 82         |
| 4.4. Uygulama-4 (10.000.000 Defa 100 Faktöriyel Hesaplama).....                 | 89         |
| 4.5. Uygulama-5 (8.96 MB Büyüklüğündeki Yeni.txt Dosyasından Bilgi Okutma)..... | 99         |
| <br><b>5. SONUÇ VE DEĞERLENDİRMELER. ....</b>                                   | <b>107</b> |
| <b>KAYNAKÇA .....</b>                                                           | <b>109</b> |
| <b>ÖZGEÇMİŞ .....</b>                                                           | <b>111</b> |



## ŞEKİLLER LİSTESİ

|                                                                                    |    |
|------------------------------------------------------------------------------------|----|
| Şekil 2.1. ASP'nin Temel İşleyişi .....                                            | 11 |
| Şekil 2.2. İstemci Sunucu Çalışması .....                                          | 34 |
| Şekil 2.3. Sunucuda Taleplerin Yorumlanması .....                                  | 35 |
| Şekil 2.4. PWS Kurulumu .....                                                      | 44 |
| Şekil 2.5. PWS Kurulumu .....                                                      | 45 |
| Şekil 2.6. PWS Kurulumu .....                                                      | 45 |
| Şekil 2.7. Kurulum Sonrası Ayarların Yapılması.....                                | 46 |
| Şekil 2.8. Kurulum Sonrası Ayarların Yapılması.....                                | 46 |
| Şekil 2.9. IIS Kurulumu .....                                                      | 47 |
| Şekil 2.10. FoxServ 3.1 Kurulumu.....                                              | 48 |
| Şekil 2.11. FoxServ 3.1 Kurulumu.....                                              | 49 |
| Şekil 2.12. FoxServ 3.1 Kurulumu .....                                             | 49 |
| Şekil 2.13. FoxServ 3.1 Kurulumu.....                                              | 50 |
| Şekil 2.14. FoxServ 3.1 Kurulumu.....                                              | 50 |
| Şekil 2.15. FoxServ 3.1 Kurulumu.....                                              | 51 |
| Şekil 2.16. FoxServ 3.1 Kurulumu.....                                              | 51 |
| Şekil 2.17. FoxServ 3.1 Kurulumu.....                                              | 51 |
| Şekil 2.18. FoxServ 3.1 Kurulumu.....                                              | 52 |
| Şekil 2.19. FoxServ 3.1 Kurulumu.....                                              | 52 |
| Şekil 2.20. FoxServ 3.1 Kurulumu.....                                              | 52 |
| Şekil 2.21. FoxServ 3.1 Kurulumu.....                                              | 53 |
| Şekil 4.1. Test1.asp sayfa görüntüsü ve Windows Görev Yöneticisi.....              | 65 |
| Şekil 4.2. CPU Kullanımı.....                                                      | 65 |
| Şekil 4.3. CPU Kullanımı(%100).....                                                | 65 |
| Şekil 4.4. Test1.php Sayfa Görüntüsü ve Windows Görev Yöneticisi .....             | 68 |
| Şekil 4.5. CPU Kullanımı.....                                                      | 68 |
| Şekil 4.6. CPU Kullanımı(%100).....                                                | 68 |
| Şekil 4.7. Test1.cgi Sayfa Görüntüsü ve Windows Görev Yöneticisi.....              | 71 |
| Şekil 4.8. CPU Kullanımı.....                                                      | 71 |
| Şekil 4.9. CPU Kullanımı(%100).....                                                | 71 |
| Şekil 4.10. Test2.asp Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi.....  | 75 |
| Şekil 4.11. CPU Kullanımı.....                                                     | 75 |
| Şekil 4.12. CPU Kullanımı(%100).....                                               | 75 |
| Şekil 4.13. Test2.php Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi ..... | 78 |

|                                                                                    |     |
|------------------------------------------------------------------------------------|-----|
| Şekil 4.14. CPU Kullanımı.....                                                     | 78  |
| Şekil 4.15. CPU Kullanımı(%100).....                                               | 78  |
| Şekil 4.16. Test2.cgi Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi ..... | 81  |
| Şekil 4.17. CPU Kullanımı.....                                                     | 81  |
| Şekil 4.18. CPU Kullanımı(%100).....                                               | 81  |
| Şekil 4.19. Test3.asp Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi.....  | 84  |
| Şekil 4.20. CPU Kullanımı.....                                                     | 84  |
| Şekil 4.21. Test3.php Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi ..... | 86  |
| Şekil 4.22. CPU Kullanımı.....                                                     | 86  |
| Şekil 4.23. Test3.cgi Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi ..... | 88  |
| Şekil 4.24. CPU Kullanımı.....                                                     | 88  |
| Şekil 4.25. Test4.asp Sayfa Görünümü ve Windows Görev Yöneticisi.....              | 92  |
| Şekil 4.26. CPU Kullanımı.....                                                     | 92  |
| Şekil 4.27. CPU Kullanımı(%100).....                                               | 92  |
| Şekil 4.28. Test4.php Sayfa Görünümü ve Windows Görev Yöneticisi .....             | 95  |
| Şekil 4.29. CPU Kullanımı.....                                                     | 95  |
| Şekil 4.30. CPU Kullanımı(%100).....                                               | 95  |
| Şekil 4.31. Test4.cgi Sayfa Görünümü ve Windows Görev Yöneticisi .....             | 98  |
| Şekil 4.32. CPU Kullanımı.....                                                     | 98  |
| Şekil 4.33. CPU Kullanımı(%100).....                                               | 98  |
| Şekil 4.34 Okuma.asp Sayfa Görünümü ve Görev Yöneticisi .....                      | 101 |
| Şekil 4.35. CPU Kullanımı.....                                                     | 101 |
| Şekil 4.36. CPU Kullanımı(%100).....                                               | 101 |
| Şekil 4.37. Okuma.php Sayfa Görünümü ve Görev Yöneticisi .....                     | 103 |
| Şekil 4.38. CPU Kullanımı.....                                                     | 103 |
| Şekil 4.39. CPU Kullanımı(%100).....                                               | 103 |
| Şekil 4.40. Okuma.cgi Sayfa Görünümü ve Görev Yöneticisi .....                     | 105 |
| Şekil 4.41. CPU Kullanımı.....                                                     | 105 |
| Şekil 4.42. CPU Kullanımı(%100).....                                               | 105 |

## TABLÖLER LİSTESİ

|                                                                                                            |     |
|------------------------------------------------------------------------------------------------------------|-----|
| <b>Tablo 2.1.</b> PHP3.x ile PHP4.x Performans Analizi .....                                               | 18  |
| <b>Tablo 2.2.</b> ASP-PHP Karşılaştırması.....                                                             | 19  |
| <b>Tablo 2.3.</b> CGI Ortam Değişkenleri.....                                                              | 36  |
| <b>Tablo 2.4.</b> Sık Kullanılan MIME Tipleri .....                                                        | 38  |
| <b>Tablo 2.5.</b> CGI Başlık Statüleri.....                                                                | 39  |
| <b>Tablo 4.1.</b> ASP, PHP, CGI Sayfalarının Test Sonuçları(Döngü) .....                                   | 72  |
| <b>Tablo 4.2.</b> ASP, PHP, CGI Sayfalarının Test Sonuçları(JPG Resmi Yükleme).....                        | 82  |
| <b>Tablo 4.3.</b> ASP, PHP, CGI Sayfalarının Test Sonuçları(Avi Yükleme).....                              | 89  |
| <b>Tablo 4.4.</b> ASP, PHP, CGI Sayfalarının Test Sonuçları(Faktöriyel Hesaplama) .....                    | 99  |
| <b>Tablo 4.5.</b> ASP, PHP, CGI Sayfalarının Test Sonuçları(Dosya Okuma) .....                             | 106 |
| <b>Tablo 5.1.</b> ASP, PHP ve CGI Yazılımlarının Farklı İşlem Çeşitleri Karşısındaki İşletme Süreleri..... | 108 |



## KISALTMALAR LİSTESİ

|        |                                                                                                   |
|--------|---------------------------------------------------------------------------------------------------|
| TCP/IP | :İnternet İletişim Protokolü                                                                      |
| ASP    | :Active Server Pages (Aktif Sunucu Sayfaları)                                                     |
| IIS    | :Internet Information Service (İnternet Bilgi Servisi)                                            |
| ISAPI  | :Internet Server Application Programming Interface (İnternet Sunucu Uygulama Programları Arayüzü) |
| HTTP   | :Hyper Text Transfer Protocol (Zengin Metin Taşıma Protokolü)                                     |
| CGI    | :Comman Gateway Interface (Yaygın Geçiş Kapıları Arayüzü)                                         |
| PHP    | :Personel Home Page (Kişisel Web Sayfaları)                                                       |
| PWS    | :Personel Web server (Kişisel Web Sunucuları)                                                     |
| JSP    | :Java Server Pages (Java Sunucu Sayfaları)                                                        |



## ÖZET

### WEB TABANLI PROGRAMLAMA DİLLERİNİN DEĞERLENDİRİLMESİ VE PERFORMANS ANALİZİ

Ferhat BAĞÇACI

Fırat Üniversitesi  
Fen Bilimleri Enstitüsü  
Elektronik-Bilgisayar Eğitimi Anabilim Dalı

2005, Sayfa:112

Bu tez çalışmasında, web tabanlı programlama dilleri konusunda yapılan araştırmalar ve yöntemler incelenmiş ve incelemeler sonucunda genelde web tabanlı programlama için ASP, PHP, ve CGI dillerinin kullanıldığı görülmüştür. Ayrıca bu programlama dillerinin ortak özellikleri olan web tabanlı yazılımlara genel bakış, avantajları ve dezavantajları, çalışma prensipleri, kullanım alanları, bakım ve kurulum kolaylıkları vs. yönünden bir değerlendirilmesi verilmiştir. Bunun yanı sıra, ASP, PHP ve CGI web programlama dillerinin her biri ile ayrı ayrı hazırlanan 5 farklı uygulamanın sonuçları birbirleri ile karşılaştırılarak, hız açısından hangi programlama dilinin daha hızlı olduğu saptanmıştır.

**Anahtar Kelimeler:** Web tasarımı, Web programlama, Performans analizi



## **ABSTRACT**

### **THE EVALUATION OF WEB BASED PROGRAMMING LANGUAGES AND PERFORMANCE ANALYSIS**

**Ferhat BAĞÇACI**

**Firat University  
Graduate School of Natural and Applied Sciences  
Department of Electronic and Computer Science**

**2005, Page:112**

In this thesis, we examine the methods and the researches about the web based programming languages and it is seen that ASP, PHP and CGI are used mostly for web based programming. Also a general evaluation is given for the examined web based programming languages according to their common features, advantages, disadvantages, working principles, application areas, maintenance, set up conveniences. Moreover, it is compared the web based programming languages according to their working performances with five different application samples which are prepared by us using ASP, PHP and CGI.

**Keywords:** Web design, Web programming, Performance analysis

## 1. GİRİŞ

İletişim ve bilişim alanındaki gelişmeler, günlük yaşantımızı yakından ilgilendirmektedir. Geçen yüzyılın ikinci yarısında belirgin olan teknolojik gelişmeler, çeşitli otonom cihazlar olarak insanoğluna hizmet vermektedir. Son yıllarda otomasyon sistemlerindeki muhteşem gelişmeler sayesinde, insan gücüne dayalı birçok işin teknoloji ürünleri ile kolayca yapılabildiği bilinmektedir.

Bilgi ve iletişim teknolojileri birbirleriyle yarışarcasına gelişmeyi sürdürürken, çağımızda bu iki kavram birbirinden ayrılmaz hale gelmiştir. Bilginin iletişimi ve paylaşımı olan internet; 21 yüzyıla damgasını vuracak bir bilgi işlem teknolojisidir. Günümüzde artık "Sanayi toplumu/devrimi" kavramları, yerini bilgi toplumuna (3. Sanayi devrimi diye de adlandırılmaktadır) bırakmıştır[1].

Günümüzde İnternet'in kullanımı çok hızlı bir şekilde gelişme göstermektedir. Uzaktan eğitimle doğrudan ilişki içerisinde olan bu aracı kullanmaya başlayanlar artık kolay kolay terk edememektedirler[2].

Bütün bu gelişmeler içerisinde yazılım teknolojisi de günden güne gelişmektedir. Son 10 yıl içinde yazılım teknolojisi, günlük yaşantıyı daha fazla ilgilendirmeye başlamıştır. Özellikle cep telefonları, otomobiller, otomobil bilgisayarları, ev kontrol sistemleri, dijital yayınlar ve en önemlisi internet, gelişen yazılım teknolojilerinin yoğun kullanıldığı alanlardır.

21. yüzyılın başlarından itibaren yazılım alanında çok önemli gelişmeler kaydedilmektedir. Büyük yazılım şirketleri, mükemmel programlarını çok kısa sürede geliştirebilmekte ve toplumun hizmetine sunabilmektedir. Başlı başına bir sektör haline gelen yazılım sahasında Hindistan gibi gelişmekte olan ülkelerin de önemli söz sahibi oldukları görülmektedir.

Bu bölümde web tabanlı yazılımların genel özellikleri üzerinde durulmuştur. Bölüm 1'de ise web tabanlı yazılımlara genel bir bakış, çalışma prensipleri, avantajları, bakım ve kurulum kolaylıkları yönünden bir değerlendirilmesi verilmiştir. Bölüm 2'de ASP, PHP, CGI web programlama dillerinin yapısı, tercih nedenleri, gereklilikler, güvenlik, veritabanı, ekonomikliği, bu dillerin bilgisayara kurulumu ve çalıştırılması vs. gibi web programlama dillerinin özelliklerinden daha ayrıntılı olarak bahsedilmiştir. Bölüm 3'de ASP, PHP, CGI dillerinin birbiri ile karşılaştırılması yapılarak avantajları ve dezavantajları verilmiştir. Bölüm 4'de ise ASP, PHP ve CGI dillerinin her biri ayrı ayrı kullanılarak oluşturulan 5 uygulamanın (toplam 15 örnek) sonuçları birbirleri ile karşılaştırılarak hız açısından hangi dilin daha üstün olduğu ortaya konulmaya çalışılmıştır. Bölüm 5'de ise kıyaslama yapılırken kullanılan farklı

uygulamaların söz konusu yazılımların aynı işlevi ne kadar sürede yapmış olduğunun test sonuçları bir tablo halinde verilmiştir.

Web tabanlı programlama dilleri alanında çok büyük gelişmeler yaşanmakta ve bazen hangi yazılım daha mükemmel olduğu konusunda tereddütler dahi geçirilebilmektedir.

Web üzerinde bilgi, temel olarak statik ve dinamik olmak üzere iki farklı şekilde görüntülenir. Statik bilgi içeren sayfalar bir defa yazılıp sunulmasına rağmen, dinamik sayfalar temel olarak istemci istekte bulununca oluşturulur. Bu şekilde dinamik sayfalar oluşturmak için web programlama dilleri kullanılır[3].

Web programcılığı, artık standart haline gelen HTML (Hiper Metin Anlamlandırma Dili) ile başlamış ve CGI, JavaScript, ASP, PHP gibi dillerle devam etmiştir. Bu dillerin bazıları sunucu tarafında bazıları ise istemci tarafında çalışmaktadır. İlk olarak HTML'i ele alalım; HTML dosyalarının içeriği bağlanılan sunucu tarafından istemci bilgisayarında gösterilir. Yani tüm görüntüleme işlemini istemci bilgisayar yapı. HTML'den sonra çıkan JavaScript teknolojisi de buna yakındır. Yani dosyalar bağlanılan bilgisayar tarafından istemci bilgisayarına yollar ve istemci bilgisayarında düzenlenip görüntülenir. JavaScript içeren sayfalar diğer sayfalara göre daha geç yüklenmektedir. Çünkü tüm JavaScript kodları istemci bilgisayarında çalıştırılır ve tabi bu da performans kaybına yol açar. Sunucu tarafında çalışan programlama teknolojilerinde ise tüm kodlar sunucu tarafında çalıştırılır ve istemciye sadece salt HTML kodları gönderilir. Buradan anlaşılacağı üzere sunucu tarafı programlamada sayfa içerisinde bulunan tüm kodlar sunucu üzerinden çalıştırılır. Bu yönüyle sunucu bilgisayarının performansı nispetinde sistem o denli hızlı işler. İstemci bilgisayar ise sadece sunucudan gelen salt HTML sayfalarını görüntüler.

Sunucu tarafı programlamanın avantajları göz önüne alındığında elektronik ticaret için en kullanışlı platform olma avantajı vardır. Çünkü güvenlik bu tür uygulamalarda çok önemlidir ve sunucu tarafı programlama bu güvenliği sağlamaktadır. Bu yönüyle sunucu tarafı programlamanın en önemli avantajı güvenlidir. Örneğin JavaScript kodları, kaynağı görüntülediğinde görülebilmektedir. Eğer güvenlik açısından kullanıcıların görmemesi istenilen bir program çalıştırılıyorsa sunucu tarafı programlama yapılmalıdır. Bu tip programlamanın ilk örneği CGI ile gerçekleşmiştir. PERL dili kullanılarak üretilen bu kodlar pek çok web tasarımcı tarafından rahatça kullanılamamıştır. Çünkü CGI çalıştırmak için hosting (Web sayfalarının yayınlandığı ev sahibi bilgisayar) yapılan yerde bu desteğin verilmesi gerekmektedir. Fakat asıl en büyük sorun, bu kodları geliştirirken hiçbir şekilde denenememesi olmuştur. Çalıştığından emin olmak için programın sunucuya yüklenmesi gerekmektedir[4].

Sunucu taraflı programlama ile veritabanına bağlanıp, gelen verilerden faydalanarak sayfalar yapılabilir. Bunun için SQL, Access veya Oracle kullanılabilir.

Kullanıcı herhangi bir tarayıcı vasıtası ile ilgili siteye bağlandığında, ulaşmak istediği dosyayı, istekte bulunan ve bu hizmeti sunan web sunucu kullanıcıya gönderir. Ancak bu durum .html ve .htm uzantılı dosyalar için böyle iken, sunucu taraflı programlama ile hazırlanan bir dosyaya istekte bulunduğu web sunucu istenilen bu dosyayı önce yorumlar ve bu işlemin sonucunda üretilen standart HTML kodları kullanıcıya gönderilir.

Sonuç olarak kullanıcıya iletilecek HTML dosyası, kullanıcı istekte bulunana kadar oluşturulmaz. Kullanılan tarayıcı tipi, yazılan sunucu taraflı programlama dosyaları için önemli değildir. Çünkü yorumlanan sunucu taraflı programlama dosyaları standart HTML çıktı üretirler.

ASP, PHP, CGI gibi diller sunucu taraflı betik dilleri, JavaScript, VBScript, JavaScript gibi diller ise istemci taraflı betik dilleri olarak bilinir.

Web sayfalarında sıkça rastlanılan; ziyaretçi defteri, anket, forum gibi uygulamalar, sunucu tabanlı programlama dilleri kullanılarak oluşturulmuştur. Tüm alışveriş sitesi, arama motorları, müzayede ve benzeri siteler de yine bu türde uygulamalar kullanılarak yapılmıştır. Sunucu tabanlı programlama dilinin adı ne olursa olsun temelde akıllı web siteleri yapmak için yaratılmışlardır[5].

PHP (Personel Home Page-Hypertext Preprocessor) ve ASP (Active Server Pages)'de sunucu tabanlı betik dilleridir. PHP ve ASP'nin temel ortak özellikleri HTML içine yazılmalarıdır. Sayfa uzantısı .php veya .asp olan bir dosya sunucudan istendiğinde, sunucu bu sayfayı PHP veya ASP yorumlayıcısına yönlendirir. Yorumlayıcı PHP veya ASP kodlarını yorumlar, yapılması gerekenleri yapar ve ziyaretçinin tarayıcısına sonucu gönderir.

*Özet olarak, bir sunucu taraflı programlama ile hazırlanan sayfa şunlardan oluşur:*

- Bu sayfaların içinde kod vardır.
- Bu sayfaların kodları web sunucusu tarafından icra edilir.
- Bu sayfa kullanıcının tarayıcısına salt HTML kodu olarak gelir.

Web, internet üzerinde yer alan grafiksel bir servistir. Hazırlanan içeriklerin (grafik, yazı, ses, animasyon, görüntü vb.) dünya üzerinde yer alan tüm kullanıcılara iletilmesi ve kullanıcıların bu dosyalara ulaşabilmesi için internete bağlı bir bilgisayar üzerinde çalışan uygulama yazılımı web sunucu olarak tanımlanabilir. Web sunucu programları kuruldukları işletim sistemine göre farklılık gösterebilir. Sözgelimi web sunucu olarak Unix işletim sistemini

kullanan bilgisayarlar Apache (Apachenin NT’de çalışan versiyonları da vardır) kullanırken, Windows işletim sistemini kullanan bilgisayarlar (sunucu) ise, web sunucu programı olarak IIS (Internet Information Server) kullanabilir.

Bu çalışma, yaygın kullanılan web tabanlı programlama dilleri hakkında bazı önemli bilgilerin sunulmasını, bu dillerin kıyaslanmasını ve web tabanlı programlarla yapılmış uygulamaları içermektedir. Böylece gerçek gözleme dayanılarak yapılan yorumlar ile hangi dilin kullanılmasının daha avantajlı olabileceğine açıklık getirilmiştir.



## **1.1. WEB TABANLI YAZILIMLARIN ÖZELLİKLERİ**

### **1.1.1. Web Tabanlı Yazılımlara Genel Bakış**

İnternet çağında yaşadığımız bu dönemde, her yeni çıkan donanım mutlaka internete bağlanabilir şekilde üretilmektedir. İnternete bağlanabilen her donanımda ise bir web tarayıcısı bulunmaktadır. Web tabanlı herhangi bir yazılım, HTML standartlarına uyduğu sürece herhangi bir web tarayıcısı tarafından görüntülenebilir. Bu özellik web yazılımlarının en küçük el bilgisayarından, en büyük MainFrame'lere kadar her tür donanımda kullanılabilmesini sağlamaktadır. İşte bu yüzden, web tabanlı yazılımlar son yılların en popüler yazılımları olmuşlardır.

Sunucu tarafında ise değişik seçenekler sunulmuştur. Örneğin, günümüzün en popüler web suncusu olan Apache geliştirilmeye Unix (TM) sistemlerde başlanmış, ardından Windows(TM) sistemlere uygun hale getirilmiştir[3].

### **1.1.2. Web Tabanlı Yazılımların Çalışma Prensipleri**

Birçok kişi tarafından paylaşılan bilgi ("birçok kişi"lere istemci denilir) tek bir bilgisayarda bir veri tabanı halinde depolanır. Bu veri tabanı içindeki bilgileri görüntüleme, değiştirme, silme gibi işlemler bu "tek" bilgisayarda gerçekleştirilir ("tek bilgisayar" a da sunucu denilir). İstemci, web tarayıcısını kullanarak sadece hangi bilgiyi istediğini ya da hangi işlemin yapılması gerektiğini belirtir. İstemcilerden gelen tüm işlemler sunucu tarafından gerçekleştirilerek, sonuçları istemcilere ulaştırılır.

Sunucu bilgisayar ile istemci bilgisayarı arasındaki veri iletişimi TCP/IP (İnternet İletişim Protokolü) üzerinden HTTP protokolü ile yapılır. Güvenliğin sağlanması için SSL altyapısından faydalanılır. Bu protokoller internette sürekli olarak kullanılan standart ve güvenilir protokoller olduğu için yazılımcıların; istemci bilgisayar ile sunucu bilgisayar nasıl haberleşir, haberleşme sırasında verilerin başkaları tarafından okunması nasıl engellenir gibi sorulara çözüm üretmesine gerek kalmamaktadır. Bu da maliyetlerin düşmesi anlamına gelmektedir[6].

### **1.1.3. Web Tabanlı Yazılımların Avantajları**

İnternet, dünyamızda çok geniş bir kullanım alanı bulmuş ve çok hızlı bir şekilde yayılma göstermiştir. Bu yayılma sonucunda, insanların istedikleri her konudaki bilgiye internet üzerinden çok kolay bir şekilde ulaşması sağlanmıştır. Günümüzde bir çok özel teşebbüs ve kamu kurumları, faaliyet gösterdikleri alanların bir çoğunu internet üzerinden gerçekleştirmektedir. İnternet, kullanıcılarına bir çok kolaylıklar sağlaması yanında, zamandan ve işgücünden çok önemli tasarruflar sağlamaktadır. İşte kullanıcıların zaman ve işgücünden

tasarruf sağlayabilmeleri için bazı araçlara gereksinim duyulmaktadır. İnternet üzerindeki bu araçlar ise web tabanlı yazılımlardır. İnternetin hızlı bir şekilde yayılması, web tabanlı yazılımlara olan talebi arttırmıştır. Artan bu taleplerin karşılanması için bu konu ile uğraşan çeşitli kurum ve kuruluşlar yazılımlar üreterek bu pazarda yer almaya çalışmışlardır. İnternet veya web tabanlı yazılımlar diğer kişisel yazılımlara göre çok fazla avantajlarının olması nedeniyle tercih edilmektedir. Bu avantajlardan bazıları,

- Kurulumunun ve kullanımının kolay olması,
- Ekstra masrafların yapılmaması,
- Güncelleme ve program değişikliklerinin tek noktada yapılması nedeniyle daha az işgücü kullanılması,
- Çok geniş kitlelere ulaşma imkanı,
- Bakım masraflarının oldukça az olması,
- Yönetiminin ve kontrolünün çok kolay olması,
- İnternet bağlantısı olan her yere ulaşma imkanı sağlaması,
- Düşük ilk maliyet ve bakım maliyeti,
- Erişilebilirlik,
- Bakım ve kurulum kolaylığı[7].

Şimdi bunlardan birkaçına göz atalım;

#### 1.1.3.1. Düşük İlk Maliyet ve Bakım Maliyeti

- İstemci için ayrı bir yazılım ya da modül hazırlanması gerekmemektedir.
- Veri iletiminin güvenliği için ayrı bir protokol hazırlanmasına gerek kalmamaktadır.
- İstemci ile sunucu bilgisayarların birbirleri ile haberleşmesi için ayrı bir protokol hazırlanmasına ya da soket işlemlerine gerek kalmamaktadır.
- Sistemin kurulması sırasında istemci bilgisayarın yerel ağa (veya internet'e) tanıtılması dışında ayrıca bir program kurulumuna ya da ayar yapılmasına gerek bulunmamaktadır.
- Tüm karmaşık işlemler sunucu tarafından yapıldığı için istemci bilgisayarın donanım olarak çok güçlü dolayısıyla da yüksek maliyetli olmasına gerek kalmamaktadır.
- Asıl yazılım sadece sunucuda bulunduğundan; güncellemeler, bakım ve yenilemeler yapıldığında, istemcilerde herhangi bir değişiklik yapılmasına gerek kalmamakta ve bu da bakım maliyetini düşürmektedir[6].

#### 1.1.3.2. Erişilebilirlik

- İstenildiği takdirde (gerekli güvenlik önlemleri alınarak) internet üzerinden erişime açılabilir ve dünyanın her yerinden erişilebilir. Farklı coğrafik konumlarda bayileri, şubeleri ya da temsilcilikleri olan kuruluşlar için idealdir.
- İşletim sisteminden bağımsız olarak erişilebilir. Erişim için gerekli tek şart uygun bir web tarayıcısıdır (Windows, Linux, Mac, Palm, Cep Telefonu)[6].

#### 1.1.3.3. Bakım ve Kurulum Kolaylığı

- Yerel ağ yapısı hazır olan kuruluşlar için kurulumu gereken tek şey sunucu bilgisayarıdır.
- Yazılım sadece sunucuda bulunduğundan güncellemeler ve bakım yapıldığında istemcilerde herhangi bir değişiklik yapılmasına gerek duyulmamaktadır.
- İnternet erişimi olduğu takdirde yazılım ile ilgili bakım, güncelleme, yenileme gibi işlemler uzaktan yapılabilmektedir. Örneğin Elazığ'daki bir sistemin güncellemeleri Ankara'dan yapılabilmektedir.
- Değişiklik ve güncellemeler yapılırken, (çoğu zaman) sistemin durdurulmasına gerek kalmamaktadır[6].

#### **1.1.4. Sunucu Tabanlı Uygulama Geliştirme**

HyperText Transfer Protocol (HTTP)'nin ortaya çıkışı ile internet gerçek kimliğine kavuşmuştur. Bu sayede web sunucusu ile konuşan bir web browserin sunucudan aldığı zengin içerikli bilgi uç kullanıcılara resim, ses, video vs. gibi zengin içerikte ve kullanıcı dostu bir ara yüzle ulaştırılabilir hale gelmiştir.

Bu zengin içeriğin değişmeyen sayfalar olarak tutulması kullanıcılar için çok şey ifade etmediği için onlara içeriği otomatik değişebilen, hatta veri tabanı erişimi sunan siteler oluşturmak için çok çeşitli teknolojiler geliştirilmiştir.

Web'de dinamik sayfalar oluşturmanın temel olarak iki yolu vardır:

- Sunucu tarafında çalışan (server-sided) uygulamalar kullanmak.
- İstemci tarafında çalışan (client-sided) uygulamalar kullanmak.

İstemci tarafında çalışan uygulama olarak Java Appletleri veya Netscape'in JavaScript'i veya Microsoft'un VBScript'i gibi script dilleri kullanılabilir.



#### Avantajları:

- *Sunucuyu Meşgul Etmemesi:* Veri girişi kontrolleri (data validation), menüler, genişleyebilir listeler gibi istemci tarafında yapılabilecek işler için sunucu meşgul edilmemiş olur.
- *Bant Genişliğini Etkili Kullanma:* Yukarıdaki avantaj bant genişliğini de etkilemiş olmaktadır.

Sunucu tarafında çalışan (server-sided) uygulamalar kullanmak ise şu noktalarda istemci tarafı uygulamalara üstünlük sağlamaktadır.

- Web tarayıcılarında scriptler için standart bulunmamaktadır. Bu nedenle bir tarayıcıda çalışan bir script diğerinde çalışmayabilir. Tarayıcıların java appletlerini yorumlamada kullandıkları java sınıflarının versiyonu, kullanılan appletleri çalıştıramayabilir. Kullanıcıdan yeni sınıfları yüklemesi (download etmesi) gerekebilir.
- Sunucu tarafta çalışması zorunlu bazı uygulamalar olabilir (veri tabanı erişimi, işletim sistemi komutları veya bazı araçlar ancak sunucu üzerinde çalıştırılabilirler).
- Bant genişliğini kullanım açısından işlemlerin önemli bir bölümünün sunucu tarafında yapılması gerekebilir. (Bir veri tabanı sunucusuna evden sunucuya ait bir istemci yazılımla bağlanıldığı takdirde pek çok ara komutun iletilmesi söz konusu olabilir. Sunucu tabanlı uygulama ise sadece sorguyu alacak ve sonucu gönderecektir. Bu noktada HTTP protokolünün bindireceği yük de hızı azaltabilir. Kurulacak denge önemlidir.)
- Uygulamaların güncel tutulması ve bunu yaparken de istemciler üzerinde değişiklik yapılmayıp sadece sunucu üzerinde (tek bir merkezden) gereken değişikliği yapmak tercih edilebilir.

Günümüzde sunucu tarafı uygulama geliştirme yönünde genel bir eğilim bulunmaktadır. Uygulamaların tek bir merkezden sunumu ve hatta kullanıcılara program değil ağ (network) üzerinden hizmet satma giderek daha çok önem kazanmaktadır.

Ancak web tarayıcılarının neredeyse işletim sistemleri kadar çoğaldığı günümüzde hem istemci hem de sunucu taraflarda yukarıdaki faktörleri göz önüne alarak dengeyi koruyacak şekilde uygulamalar geliştirmek gerekir[4].

## 2. WEB TABANLI YAZILIMLAR

### 2.1. ASP (*Active Server Pages*)

ASP (Active Server Pages) türkçesi *Aktif Sunucu Sayfaları* dinamik web sayfaları üretmek için microsoft firması tarafından geliştirilmiş bir internet teknolojisidir. ASP birçok dille yazılabilir (VBScript , JScript , PerlScript , PHPScript).

ASP aktif sayfalar üretmek için geliştirilmiştir. Klasik HTML (Hyper Text Markup Language) web sayfası yapımında oldukça işlevsiz kaldığı noktalar olabilmektedir. Örnek vermek gerekirse ziyaretçilerin siteye yazı ekleyebileceği bir ziyaretçi defteri, site webmasterına mail yollamanızı sağlayacak bir form yapımı, daha ileri gidersek bir forum veya aktif portal yapımı gibi örneklerde, HTML işlevsiz kalmaktadır. Bu gibi uygulamaları yapmak için ASP oldukça iyi bir çözümdür.

ASP tek başına bir programlama dili değildir. Yani ASP belgeleri bir program gibi yazılıp, ardından derlenip çalıştırılmazlar. ASP belgeleri “.asp” dosyası olarak yapacağı işleme göre HTML kodlarından ve VBScript gibi scripting (komut dosyası yazmak) komutlarından oluşur. ASP sayfalarının yaratılmasında herhangi bir dile bağlı olmamak için script (komut dosyası yazmak) dilleri kullanılmıştır. Ancak programcıların HTML, VBScript ve JavaScript script yazma dillerinin temellerini bilmesi gerekir[5].

ASP, Javascript gibi, VB Script gibi komutları düz yazı dosyası olarak kaydedilen ve çalışabilmesi için yorumlayıcıya ihtiyacı olan bir dildir. ASP sayfaları herhangi bir text editöründe yazılarak uzantısı “.asp” olarak kaydedilip oluşturulabilir.

ASP server(sunucu) tarafında çalışan bir teknolojidir. Uzantısı ASP olan sayfa sunucu bilgisayarda işlendikten sonra istemci bilgisayara sadece HTML formatında bir sonuç sayfası gönderilir. Yani istemci bilgisayar ASP kodlarını görememektedir.

#### 2.1.1. ASP ve IIS İlişkisi

ASP Microsoft’un en yeni web sunucu teknolojilerinden birisidir ve web uygulamaları geliştirmeyi oldukça kolaylaştırmaktadır. ASP, JScript ve VBScript gibi scripting dillerine benzemekte, fakat server tarafından çalışmaktadır. ASP’nin VBScript, Jscript, Perl veya diğer scripting dilleri desteklemesi nedeniyle web geliştiricileri daha başka scripting dillerini öğrenmek zorunda kalmamaktadır. Böylece ASP web geliştiricilerinin var olan bilgilerinden yararlanabilmelerini de sağlamaktadır. ASP, interaktif web uygulamalarını geliştirmeyi kolaylaştıran bir teknolojidir.

ASP Microsoft’un temel internet stratejisi Active Platform’un önemli bir parçasıdır. Aslında ASP web server üzerine yüklenmiş bir komponenttir. Bu komponent uzantısı ASP olan dosyaları işlemekte ve bu ASP dosyalarının istemlerine cevap vermektedir. Fakat bu durum

ASP'nin ISAPI (Internet Server Application Programming Interface) veya IDC (Internet Database Connector) gibi render teknolojisi olduğu manasına gelmektedir. Daha doğrusu, ASP, IDC ve ISAPI uygulamaları için tamamlayıcı bir teknolojidir. Örneğin; çoğu zaman sorgulamaya gerek kalmayan basit bir data kayıt uygulaması ortaya konulmak isteniyorsa, bu durumda basitçe IDC teknolojisi kullanılmalıdır. Fakat data sorgusu, transmisyonu (bazen replikasyonu) ve kompleks hesaplamalar içeren uygulamalar ortaya konulacaksa, tabii ki ASP daha uygun bir teknoloji olmaktadır. Görüleceği gibi, geliştiricinin her bir teknoloji için farklı bir ortam kullanması gerekse bile, ASP diğer web teknolojilerini tamamlayan bir teknoloji olarak karşımıza çıkmaktadır[6].

ASP sadece kendisi için yazılmış olan DLL'i kullanır (asp.dll). Bu dosya standart olarak web sunucu da yer almaktadır (sadece IIS 1.0 sonrası) (Winnt\System32\inetrv dizininde yer almaktadır). Bu DLL sadece ASP dosyalarını (genellikle .ASP uzantılıdır) okuyup içerisindeki script komutlarını işlemek ve sonuçlarını HTML ve metin içeriği ile birlikte Web gözeticiğine yollamak görevini üstlenir.

### **2.1.2. ASP Sayfalarının İşlenmesi**

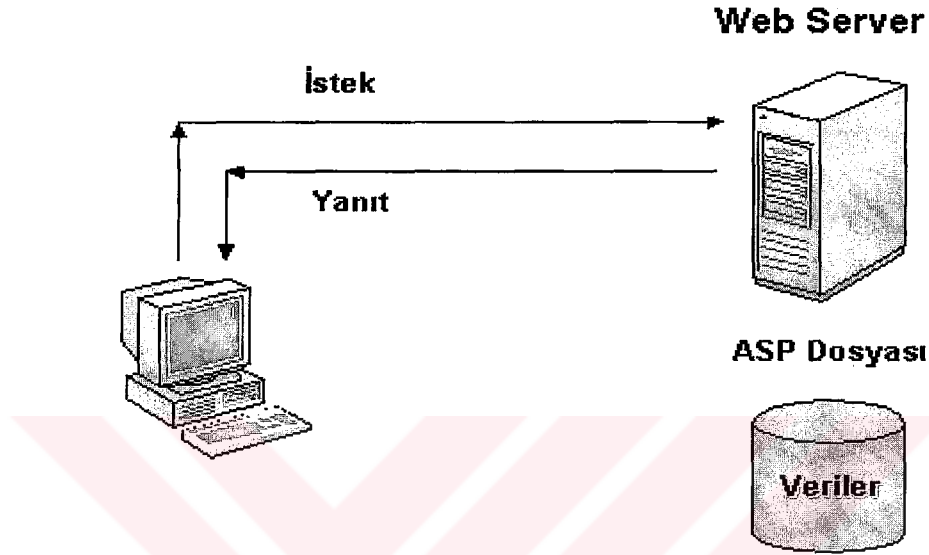
ASP arabiriminin işleyişi oldukça basittir. Sisteme ASP eklentisi yüklendiği zaman Internet Information Server'ın sahip olduğu scripting özellikleri işlemeye başlar. İstemci tarafı sunucudan normal bir HTML sayfasını istediğinde, IIS bu dosyayı sabit diskten çağırarak istemci tarafındaki Web tarayıcıya yollar. İstemci tarafından yapılan istek bir ASP sayfasıysa IIS bunu sayfanın uzantısından anlayarak ASP eklentisini devreye sokar. ASP eklentisi sayfa içindeki scriptleri yine IIS'a çalıştırdıktan sonra oluşan HTML sayfasını istemci tarafındaki Web tarayıcısına yollar, bu esnada kullanıcının sunucuda gerçekleşen işlemlerden haberi olmamaktadır. ASP sisteminin dezavantajı Web sunucusuna daha fazla yük getirmesidir. Normal bir HTML sayfasının sabit diskten alınarak sunulması sunucuya fazla yük getirmezken, ASP sayfalarının işlenmesi esnasında sunucunun bellek ve işlemci kaynaklarından daha fazla yararlanacaktır. Yukarıdaki işlemler anlatılırken ASP'nin getirdiği en büyük avantajlardan biri olan veritabanı kullanımına hiç değinilmemiştir, ASP, sunucu üzerinde yüklü ODBC veritabanlarını kullanabilir, herhangi bir şirkete ait veritabanlarını intranete veya internete açabilir[7].

*ASP dosyalarının işleyişi:*

1. Kullanıcı ASP dosyasının adresini yazar. Örneğin: <http://www.bizimsirket.com/giris.asp> ya da <http://localhost/giris.asp>. Böylece istemci sunucudan istekte bulunur.
2. Tarayıcı Internet Information Server(IIS) sunucusuna ASP dosyası için bir istek gönderir.
3. Web sunucusu (IIS), istenilen ASP dosyasını disk ya da bellekten alır.

*Bu aşamada dosyanın HTML kısmı ve ASP kod kısmı ayrılır.*

4. Web sunucusu dosyayı özel bir program olan ASP.DLL'e gönderir.
5. ASP dosyası içinde komutlar işlenir. Bu işlemin sonucu bir HTML dosyasıdır.
6. HTML dosyası tarayıcıya geri gönderilir.
7. HTML dosyası kullanıcının tarayıcısı tarafından işlenir ve sonuç kullanıcıya gösterilir[5].



**Şekil 2.1.** ASP'nin Temel İşleyişi.

ASP uzantılı dosyalar, asp.dll yardımı ile derlenmektedir. Bu derlenmede, birinci basamak olarak ASP dosyası içerisinde server taraflı kod olup olmadığı denetlenir. Eğer dosya içerisinde sunucu taraflı işletilecek bir kod bulunmaz ise IIS tarafından direkt istemciye gönderilir. Bu Windows2000'de yeni bir özellik olarak eklenmiştir. Bu sayede .asp uzantılı dosyaları kullanmaya olanak tanımaktadır (içerisinde sunucu taraflı çalışacak kod bulunmayan dosyalara bile asp uzantısı verilebilmektedir).

Eğer DLL içinde server taraflı çalışacak bir kod olan dosya ile karşılaşırsa; satır satır bu dosyayı işleterek script blokları içerisindeki komutları işletir ve çıktısını yine aynı yere gelecek şekilde bir çıkış dosyasına kaydeder. Bu dosyada istemci bilgisayarına gönderilerek işlem tamamlanır.

### **2.1.3 ASP'nin Dili**

ASP, bir teknolojidir. Kendi başına bir yazım kuralı yoktur. ASP tekniğini kullanabilmek için, ASP sayfasının talep edilmesi halinde ziyaretçiye gönderilmeden önce

ASP.DLL'ye teslim edilmesi bu teknolojinin kullanılabilmesi için hemen hemen tek şarttır. Bu, dosya uzantısı .asp yapılarak sağlanır.

ASP.DLL ise, dünyada mevcut bütün Script dilleri ile verilecek komutları kabul edebilir. Sadece ASP.DLL'ye sayfadaki kodların hangi dilde olduğunu söylemek gerekir. Bu durum, ASP sayfasının birinci satırında yapılır. Örneğin ASP'ye VBScript dilini kullanmasını belirtmek için bu satır şöyle yazılır:

`<% @Language=VBScript %>`

ASP sayfalarında genellikle VBScript, JavaScript ve JScript kullanılır. Ancak Perl dilinden türetilen PerlScript, PHP'den türetilen PHPScript de giderek ilgi çeken ASP dilleri arasına girmektedir[8].

#### **2.1.4 ASP'nin Unsurları**

ASP kodları nesnelere yöneliktir, onları kullanma ve onlardan bir sonuç alma veya onlara bir sonuç aktarma amacına yöneliktir. ASP'nin nesneleri altı grupta toplanır:

*Application/Uygulama:* Bir ASP sitesi, gerçekte bir uygulama programı olarak görülür. Bu, HTML/CGI geleneğine aşina tasarımcı için yeni bir kavramdır. ASP'yi icat edenler; bir ziyaretçi ASP sayfasından girerek, bir sitede gezinmeye başladığında, onu bir programı işleten bilgisayar kullanıcısı olarak görürler. Böylece, site, her ziyaretçinin karşısına çıktığında "bir program çalışmış" gibi sayılır. Bu yaklaşımın Web tasarımcısı olarak kazandırdığı imkanlar ise şöyledir:

*Session/Oturum:* Bir ziyaretçi siteye geldiğinde, hangi sayfayı talep ederse etsin, bu bağlantı ASP açısından bir oturum sayılır. Her oturumun belirli bir süre devam eden özellikleri, değişkenleri ve değerleri vardır. Site tasarımında oturum özelliklerinden geniş ölçüde yararlanılır.

*Request/Talep:* Browser'dan Server'a ulaşan bütün bilgiler, Request (Talep) nesnesinin öğeleridir. Bu nesneyi kullanarak, istemciden gelen her türlü HTTP bilgisi kullanılır.

*Response/Karşılık:* Server'dan ziyaretçinin bilgisayarına gönderilen bütün bilgiler, çerezler (cookie) ve başlıklar (Header) Response (Karşılık) nesnesinin öğeleridir. Bu nesneyi kullanarak ziyaretçiye göndermek istenilen haberler gönderilir.

*Server/Sunucu:* ASP, Web Server programını bir nesne olarak ele alır ve onun sağladığı imkanların kullanılmasını sağlar.

*ObjectContext/Nesne Bağlamı:* Microsoft'un Transaction Server (MTS) programının sunduğu hizmetlere erişimi sağlar. MTS, ASP sayfaları içinden, uygulama programlarından

yararlanılmasını sağlar. ASP dilindeki uzmanlık ileri düzeylere ulaştığı zaman MTS veObjectContext nesnesinden yararlanılabilir[8].

#### **2.1.5. ASP İçin Gereklilikler**

ASP sayfalarının çalışabilmesi için sayfaların bulunduğu bilgisayarın bir Web sunucusu haline getirilmesi gerekir. Bunu için Microsoft'un iki ürünü vardır: Birincisi Windows 95 ve 98 için ayrı sürümleri olan PWS (Personal Web Server) yani Kişisel Web Sunucusu, diğeri Windows NT ve Windows 2000 için geliştirilmiş IIS (Internet Information Services) yani internet bilgi servisidir. Bu iki ürünün arasında bazı farklılıklar vardır. PWS en fazla 5-10 ziyaretçinin aynı anda bağlanabileceği küçük çaplı uygulamalarda kullanılır. Eğer çok fazla sayıda kişiye hızlı hizmet vermek isteniyorsa, o zaman IIS bu iş için daha uygundur.

ASP uygulamaları yazmak için bir editör gerekir. Bu piyasada bulunan herhangi bir editör olabilir. Web tasarımı için özel olarak hazırlanmış editörler bu iş için daha uygundur. Sadece Notepad kullanarak bir ASP sayfası hazırlanabilir. Özel olarak hazırlanmış editörlerden Microsoft FrontPage, Allaire HomeSite ya da Cofffe Cup HTML Editor kullanılabilir. Bu özel editörler kullanılırsa kullanımı pratik olan sürükle-bırak nesneleri kullanabilir ve yazılan kodları ayırt edebilmek için renklendirme özelliklerinden yararlanılabilir.

#### **2.1.6. ASP'nin Kullanım Alanları ve Avantajları**

- ASP dinamik Web sayfaları oluşturmak için kullanılır. Web sayfalarını canlandıracak bir tekniktir. Web sayfalarına çok farklı işlemlerin yapılması yeteneğini kazandırır.
- ASP Web Programcılarına HTML, script dili ve kullanıcıdan bağımsız veritabanı işlemlerini rahatlıkla kullanabilme olanağı sağlar. Bu özgürlük ve platform bağımsızlığı, icra gören tüm işlemlerin sunucu tarafında halledilerek sadece sonuçları türetmesi ile sağlanmaktadır. Sonuç salt HTML kodunda türetildiği için browserlar üzerinde tanımsız komut gibi bir problemle karşılaşılmaz. Başka bir deyişle istemci tarafında Internet Explorer, Netscape Navigator, Opera gibi browser seçimi yapmamakta kusursuz çalışmaktadır.
- ASP kendi içinde Javascript, VB Script, Jscript, Perl Script gibi geniş bir yelpazedeki ham kodların kullanılmasına olanak tanımaktadır. Hatta Windows Scripting host ile uyumlu kalmak şartıyla oluşturulan her yeni script dil, Web Server' a tanıtılarak ASP üzerinde kullanılabilir.
- ASP sunucu tarafında uygulama çalıştırmaya imkan sağlamaktadır. Hazırlanan uygulamaları çalıştıran komutlar sadece sunucu tarafında kalır. İstemci yalnızca sonuçları görebilir yani kullanıcılar, inceledikleri sayfayı oluşturan komutları görüntüleyemez. Bu yönüyle ASP, web programcılarına güvenilirlik ve özgünlük sağlamaktadır.



- Uygulama ile ilişkilendirilmiş bir veri tabanı (database) üzerindeki veriler ASP ile istenilen yayınlama stratejisine göre Web sayfalarına aktarılır ve bilginin internet üzerinden güncellenmesi, otomatik hale dönüşebilir. Bu özellik ASP'nin hazırlanan bir formla veri giriş-kontrol arayüzü olduğu gibi aynı zamanda hazırlanan paket programlar ile senkronize (eş zamanlı) çalışacak bir alternatif de sunmaktadır.
- ASP ile hazırlanan projeler kaynakların yönetimi, kontrolü ve uzaktan sistem yönetimine imkan verir. Şirketler arası bilgi paylaşımı için ideal bir ortam oluşturur. Bugün web programcıları geliştirdikleri uygulamaları bir çok avantajı göz önünde bulundurarak ASP ile tasarlamaktadırlar. İnternet Servis Sağlayıcılarının çoğu ASP uygulamalarını Host edebilecek şekilde sunucularını düzenlemişlerdir. Bir çok yazılım firması ASP'ye özgün editörler tasarlamaktadır. Geliştirilen paket programların veri giriş ara yüzleri alternatifli olarak browser üzerinden de sağlanmaya başlanmıştır. Artık üzerinde aktif bilgi sunumlarının gerçekleştiği, anketlerin çeşitli fonksiyonellikte formların, aritmetik ve lojik hesaplamaların yapıldığı portal sitelerin, üyelik sistemlerinin, forumların, online alışveriş yapılabilen sayfaların sıkça karşımıza çıktığını, firmaların hızla intranetlerinde yer alan web siteleri üzerinde firma işleyişini sağlayan uygulamalara yöneldikleri görülmektedir. Elbette ki bu fonksiyonelliğin web sayfalarına yansımada ASP'nin rolü göz ardı edilmemektedir[9].

#### **2.1.7. ASP İle Bileşen Kullanma**

Microsoft'un sunucu teknolojisi, ASP için bir çok ikinci parti bileşen bulundurmaktadır. ASP sayfaları ile uyum içerisinde çalışarak sonuç sayfalarının oluşturulması, web sayfası sahiplerine işlenmiş bilgi aktarılması gibi işlere yarayan bileşenlerden en popüler olanları *ASPMail* ve *RoboHTTP*'dir. Server Objects firmasının bu iki ürünü, sayfadaki formların içeriğinin kullanıcıya e-mail olarak kullanılması, misafir defteri oluşturulması ve bir çok değişik bilginin bir sayfada toplanması gibi işlemleri kolayca halletmektedir[7].

#### **2.1.8. ASP'de Oturum Kavramı**

İnternet Information Server'in en vurucu özelliklerinden olan "oturum"dan ASP teknolojisi sonuna kadar faydalanır. Bir kullanıcının web sitesine bağlandığı andan web sitesinden ayrılacağı ana kadar olan zaman dilimine "oturum/ session" adı verilir. Bu süre içerisinde sunucudaki ASP eklentisi kullanıcıyı takip eder ve yaptığı hareketleri, yolladığı tüm bilgileri kontrol eder. Örneğin bir kullanıcının alışveriş sepetine koyduğu tüm ürünler ASP eklentisi tarafından sunucunun belleğinde tutulur, kullanıcı siteden çıktığı anda bellektekiler eğer aksi belirtilmediyse oturum sonlandığı için silinir. Oturum takibi işi bir ISAPI (İnternet Server Application Programming Interface) filtresi olan Cookie Munger ile yapılabilir, bu

arabirim aynı zamanda kullanıcıya yolladığı çerezler ile eğer silinmemişse sunucu üzerinde bulunan bir oturumun devamını da sağlayabilir.

Sunucuya oturumu hatırlatmak amacıyla kullanıcıya yollanan çerezlerin hiç bir zararı yoktur. Aynı zamanda yollanılan çerezlerin içinde sadece bir hatırlatma numarası olduğundan kullanıcının kendi bilgisayarında sunucuya ait olan bilgileri ele geçirmesi mümkün değildir. Bu yüzden çerezler bir güvenlik açığı oluşturmamaktadır[7].

## 2.2. PHP (Personel Home Page)

Rasmus Lerdorf, internet'in henüz yayıldığı 1990'ların ortalarına doğru iş ararken, hayat öyküsünü bir kişisel sayfada yayınlamak ve başvuracağı yerlere bu sayfanın adresini verebileceğini düşünmüştür. Fakat o dönemde özellikle üniversitelerin Unix ağlarında kurulan Web Sunucularda kişisel sayfa yapmak kolay değildi. Rasmus, kendisi için hazırladığı yazılımın Web'e aşına olmayanlar tarafından da kolayca kullanılabileceğini düşünmüştür. Bu yazılımın büyükçe bir bölümü Perl dilinden alınmıştır. Adına Personal Home Page (Kişisel Ana Sayfa) dediği bu programın çok tutulması üzerine Rasmus Lerdorf, internet sitesi kuran herkesin birinci gün değilse bile ikinci gün farkına vardığı ihtiyacı, yani bir form yoluyla ziyaretçiden gelen bilgileri işlemeyi sağlayan ekleri yazmıştır ve programın adı PHP/FI (Form Interpreter/Form Yorumlayıcı) olmuştur. Kimileri programın bu sürümüne PHP2 adını takmıştır ve bu ad, programın çok değişmesine ve gelişmesine rağmen uzun süre aynen kalmıştır.

Rasmus Lerdorf, 1995'in ortalarında, Zeev Suraski, Stig Bakken, Shane Caraveo ve Jim Winstead ile bir grup kurarak PHP'yi Perl'den ödünç alma rutinlerle iş yapan bir paket olmaktan çıkartarak, Nesne-Yönelimli (Object-Oriented) bir programlama dili haline getirmiştir. Bu grup, bugün PHP4 adı verilen bu dili geliştirmekte ve yeni işlevsellikler kazanması için sürekli çalışmaktadır. PHP, Linux gibi, *Açık Sistem* kurallarına tabidir; isteyen programda istediği değişikliği yapabilir; ancak bu değişiklikten para kazanamaz ve yaptığı değişiklikleri isteyen herkese açıklamak zorundadır.

Linux'un önümüzdeki yıllarda en çok kullanılan işletim sistemi olacağı yolundaki bilgiler kuşkusuz en çok Linus Torvalds'ı şaşırtmıştır. Linux'un bu başarısındaki en önemli anahtar, nokta olan açık kaynak geliştirilmekte ve dünyanın her yerinden yazılım geliştiricilerin desteğini almaktadır. Aynı başarıyı Web yazılım geliştirme dili PHP de göstermektedir. PHP bundan iki yıl öncesine kadar adı duyulmamış basit bir dil idi. Ancak şimdi gerek performans ve gerek destek açısından diğer Web yazılım geliştirme dilleri ile kolayca yarışabilmektedir. PHP, her türlü veritabanıyla sorunsuz çalışabilmektedir. PHP ve açık kaynak olarak geliştirilmeye başlanan MySQL birlikte kullanıldıklarında yakaladıkları etkinlik binlerce dolar verilerek alınan veritabanı ve uygulama dilleriyle yarışmanın ötesinde farklara sahiptir.



Türkiye'de PHP oldukça yeni kullanılmaya başlanmıştır. Dünyadaki PHP kullanıcılarının tercih ettikleri PHP editörü PHPed'i bir Türk'ün geliştirmiş olması, hepimize haklı bir gurur vermektedir[10].

### 2.2.1. PHP Dilinin Yapısı

PHP bir Script dilidir; PHP ile yazılan "programlar" birer düzyazı dosyasıdır. Dil öğelerini ayrıntılı olarak ele almadan önce bir iki yazım kuralından bahsetmek gerekirse; PHP yorumlayıcısı, bu "programı" çalıştırabilmek için dosyanın içinde PHP komutlarını arar. PHP komutları birinci bölümde görüldüğü gibi iki şekilde yazılabilir:

1. `<?PHP .... ?>`

2. `<? .... ?>`

Bunlara *PHP Komut Ayracı* denir; birinci türü uzun veya “*standart ayraç*” sayılır; ikincisine ise “*kısa ayraç*” denir. PHP yorumlayıcısında kısa ayraç kullanılması istenmiyorsa, PHP.ini dosyasındaki, `short_open_tag = On`: satırını `short_open_tag = Off`; yapmak yeterlidir. Bununla birlikte uzun vadede, PHP programlarında HTML yerine XML kullanılmak isteniyorsa, uzun ayraça ihtiyaç duyulur; çünkü XML etiketleri “`<?xml`” şeklinde başlar ve “`?>`” şeklinde biter.

ASP ile çalışan Web programcıları “`<?>`” yerine sık sık “`<%>`” ve “`%>`” yazarlar. PHP dilini geliştirenler, çok yaygın olan bu hatanın PHP programlarının çalışmasına engel olmasını önlemek için bir kolaylık sağlamışlardır. ASP ile çalışan ve daha sonra PHP'ye geçen bir programcı, PHP.ini dosyasını açarak, `asp_tags = Off`; satırını `short_open_tag = On`; yapabilir ve artık rahatça ASP ayracını PHP yazmak için kullanabilir.

PHP komutları daha uzun olmakla birlikte `<SCRIPT>` etiketiyle de kullanılabilir. Yukarıdaki örnekte oluşturulan merhaba.PHP şu şekilde değiştirilerek, merhaba2.PHP adıyla kaydedilir:

```
<HTML>
<SCRIPT LANGUAGE="PHP">
    print ("Merhaba Dünya!");
</SCRIPT>
</HTML>
```

Web sunucusunda bu program açılırsa, aynı sonucun alındığı görülür. Javascript veya VBScript gibi HTML kodlarının içine gömülen ve HTML sayfasının içinde, Web ziyaretçisinin bilgisayarına gönderilen ve orada (istemci-tarafından) icra edilen ve yine aynı şekilde `<SCRIPT>` etiketi içine konulan Script kodları ile burada `<SCRIPT>` etiketi ile ifade edilen PHP kodlarının icra yerini karıştırmamak gerekir. PHP, nasıl bir ayraç ve etiket ile yazılırsa yazılsın, daima sunucuda çalıştırılırlar ve ziyaretçiye asla PHP kodları gösterilmez. merhaba2.PHP'nin çalıştığı

Browser'da kaynak görüntülenirse, istemciye gelen kodların saf ve temiz HTML olduğu görülür. PHP kodları, oluşturulmak istenen sayfanın HTML kodları ile iç içe yazılır. Sayfanın görüntüsünde hiç bir değişiklik olmamakla birlikte, Browser'a giden HTML kodları önemli ölçüde değişmiş olacaktır.

Ciddi PHP programcılığına başlanıldığı zaman, hangi satırda ne tür bir işlem yapıldığını anlamak daima mümkün olmayabilir. Biz kendi programımızı satır satır ezberlesek bile, işbirliği yaptığımız diğer Web programcılarının programımızı anlamalarını sağlamamız gerekir. Bu işlem, yorum satırları ile yapılabilir. PHP ayraçları içinde çok-satırlı ve tek satırlı olmak üzere iki türlü yorum bulunabilir. Birinci grubu, başına "/\*" ve sonuna "\*/" işaretleri koyarak, ikinci grubu ise sadece başına "/\*" işareti koyarak belirtilir:

```
/*
```

Bu satır da PHP'nin çok-satırlı yorum bölümü.

Bunu da PHP yorumcusu asla dikkate almaz

Buraya istenildiği kadar yorum yazılabilir.

```
*/
```

```
// Bu ise PHP'nin tek satırlı yorum bölümü
```

```
// Bu satırları da PHP yorumcusu dikkate almayacaktır.
```

PHP'nin bir diğer yorum işareti ise # (Diyez) işaretidir. Bu işaretin başında olduğu satır da PHP tarafından dikkate alınmaz:

```
# Bu satır PHP'nin çok-satırlı yorum bölümü.
```

```
# Bunu da PHP yorumcusu asla dikkate almaz
```

```
# Buraya istenildiği kadar yorum yazılabilir.
```

Diyez ile yorum haline getirdiğimiz satırların sonuna başka işaret koymaya gerek yoktur[10].

### **2.2.2. PHP'nin Tercih Nedenleri**

- Platform Çeşitliliği (Hem UNIX, hem Linux, hem de Windows için hazır).
- Performans (Zend yorumlayıcı motoru ile şimdi çok hızlı).
- Büyük ölçekli veri tabanı uygulamaları için ideal (ODBC, MySQL, PostgreSQL, Oracle ve diğerleri için arayüz desteği).
- İnternet standartlarına uyumluluk (LDAP, IMAP, FTP, news gibi standartlara sonuna kadar açık bir kapı).

- Gelişmiş Özellikler (Oturum Yönetimi, Semafor ve Paylaşımlı Hafıza kullanım, Sürekli Veritabanı Bağlantıları, Regular Expressions).
- Ekonomik (Bedava).

#### 2.2.2.1. PHP'nin Çalıştırılabileceği Ortamlar

Kaynak kodu açık olarak dağıtılan PHP, Linux, Solaris, HP-UX, IRIX, FreeBSD vb. üzerinde rahatlıkla derlenip çalıştırılabilir. Kendi başına derlenip web serverden cgi programı olarak çağrılabilir. Ancak hız ve güvenlik açısından sakınca oluşturduğu için özellikle Apache web sunucusu ile birlikte modül olarak derlenmesi tavsiye edilmektedir.

Windows-NT ortamında da, Internet Information Server(IIS) ile çalışmak üzere kullanılabilmektedir.

#### 2.2.2.2. Yüksek Performans (Zend'in yorumlayıcı motoru ile şimdi çok hızlı)

PHP 3.x versiyonları popüler olmalarına rağmen ASP yorumlayıcısına karşı yavaş kalıyorlardı. PHP, 4.0 versiyonunda Zend firmasının script yorumlayıcı motorunu kullanılarak tamamen yenilendi ve hız olarak ASP ile yarışır hale geldi. Zend'in kendi yaptığı testlerde PHP3.x ile PHP4.x arasındaki fark gözlenebilmektedir:

100MHz Pentium işlemci ve 24MB hafızalı Linux kurulu bir sistemde şu sonuçlar alınmıştır:

**Tablo 2.1.** PHP3.x ile PHP4.x Performans Analizi

|                                              | PHP3.0.6 | PHP/Zend |                   | PHP/Zend+optimizer |                   |
|----------------------------------------------|----------|----------|-------------------|--------------------|-------------------|
|                                              | Süre     | Süre     | Performans Artışı | Süre               | Performans Artışı |
| <b>QuickSort algoritması (1000 öge)</b>      | 5:26min. | 1:11min. | 460%              | 31sec.             | 1050%             |
| <b>QuickSort algoritması (10 öge)</b>        | 0.1sec   | 0.06sec  | 60%               | 0.06sec            | 66%               |
| <b>MySQL'de tablo listeleme (1000 kayıt)</b> | 4.35sec. | 0.83sec. | 525%              | 0.57sec.           | 765%              |
| <b>MySQL'de tablo listeleme (10 kayıt)</b>   | 1.1sec.  | 0.6sec.  | 185%              | 0.5sec.            | 220%              |

PHP 4.0 pre-alpha versiyonu ile Microsoft'un ASP'si arasında yapılan benchmark sonuçları ilginçtir. QuickSort algoritması çalıştıran özdeş ASP ve PHP scriptleri 128 MB RAM'li P2-233 sistemi üzerinde Windows NT/Service Pack4 ve web server olarak IIS 4.0

kullanılarak çalıştırılmışlardır. Burada PHP yorumlayıcısı ISAPI modülü olarak çalıştırılmıştır. İşletim sisteminin NT ve web serverin IIS olması ASP için bir avantaj olması gerekirken hiç de öyle olmadığı görülmüştür:

**Tablo 2.2.** ASP-PHP Karşılaştırması

|                                              | ASP    | PHP/Zend |                   | PHP/Zend+optimizer |                   |
|----------------------------------------------|--------|----------|-------------------|--------------------|-------------------|
|                                              | Süre   | Süre     | Performans Artışı | Süre               | Performans Artışı |
| <b>Basit Döngü<br/>(10 Milyon iterasyon)</b> | 44sec. | 43sec.   | 2%                | 12.5sec.           | 352%              |
| <b>QuickSort algoritması (200 öge)</b>       | 10sec. | 44sec    | -440%             | 6sec.              | 66%               |

Ayrıca ASP özellikle yoğun hit alınan durumlarda web sunucusunu yavaşlatırken, modül olarak kurulu Apache-PHP ikilisinde bu sorun pek yaşanmamaktadır.

PHP/Zend yorumlayıcısı ücretsizdir. Ancak Zend firması PHP'nin optimizer'li yorumlayıcısını ücretli dağıtmayı düşünmektedir. PHP'nin hızlı olması Ulakbim web sitesinde de parser olarak kullanılması için tercih sebeplerinden biri olmuştur.

#### 2.2.2.3. PHP'in Büyük Ölçekli Veri Tabanı Uygulamalarındaki Konumu

ASP ve Cold Fusion gibi araçlar özellikle veri tabanlarına bağlanıp ve kullanabilme gibi özellikleri ile ön plana çıkmışlardır. PHP de arayüz oluşturduğu veritabanı yönetim sistemlerinin çeşitliliği ile ön plana çıkmaktadır. PHP ile birlikte:

- MySQL
- mSQL
- PostgreSQL
- ORACLE
- MS-SQL Server
- Sybase
- Informix
- InterBase
- Solid

gibi popüler veri tabanları kullanılabilir.

#### 2.2.2.4. İnternet Standartlarına Uyumluluk

PHP açık standartları desteklemektedir. Ona http protokolü ile ilişkisi olmayan pek çok iş yaptırılabilir. Üstelik tüm bu standartların web'e çıkış kapısı olabilir. PHP derlenirken bazı opsiyonlar ve kütüphaneler kullanılarak çok fonksiyonlu bir araç haline getirilebilir.

Kurum içi personel bilgilerinin tutulduğu bir LDAP sunucusuna erişim, sorgulama ve değişiklik yaptırılabilir, tamamen özelleşmiş web mail veya web news arabirimleri oluşturulabilir, FTP ve Telnet istemcilerinin yerini alabilir.

Bunları yaparken muhtemelen internete iyi bir bağlantısı olan ve güçlü donanım özelliklerine sahip bir sunucu tarafında uygulamaları çalıştırıyor olmanın avantajlarından yararlanır.

Üretilen uygulamaların yönetimi de tamamen otomatik veya yine web üzerinden çok az yönetimsel fonksiyon gerektirecek hale getirilebilir.

#### 2.2.2.5. PHP'in Gelişmiş Özellikleri

PHP, oturum yönetimi (session management) konusunda da oldukça iyidir. PHP'de bu özellik Netscape Enterprise Server üzerinde kullanılan ServerSideJavaScript kadar gelişmiş ve kolay kullanılır değildir. NSEnterprise Server, SSJS ile geliştirilmiş uygulamaları çağrılmadan önce hafızaya yüklemekte ve bunların ortak kullanabileceği hiyerarşik session objectleri tanımlamaktadır. Bu ise yapılacak işlemi gerçekten kolaylaştırmaktadır. PHP'ye de aynı işi yaptırılabilir, ancak PHP'de scriptler çağrılınca hafızaya alındığı için bu iş için örneğin paylaşımlı hafıza (shared memory) ve semafor kullanmak gerekebilmektedir.

Sürekli Veritabanı Bağlantısı (Persistent Database Connections) özelliği veri tabanı uygulamalarında hızı oldukça arttıran bir faktördür. Normalde bir kullanıcı veri tabanı ile ilgili bir iş yapmak için web sunucuya birden çok istem iletir. CGI programları aynı kullanıcının her istemi için veri tabanı sunucusuna yeni bir bağlantı kurar. PHP ile sürekli bağlantı kullanıldığında sadece ilk istek (request) için veri tabanı bağlantısı kurulup sonra aynı bağlantı numarası (connection handle) kullanılarak bağlanılabilmektedir.

#### 2.2.2.6. PHP'in Ekonomikliği

PHP, General Public License ile ücretsiz dağıtılmaktadır. Aynı şekilde MySQL'in de kaynak kodları açıktır. İşletim sistemini de ucuza getirmek, bir Linux paketi ile komple bir web tabanlı uygulama ve intranet platformu edinmek demektir. Bu durumda harcamalar neredeyse tamamen donanıma gidecektir. Üstelik oldukça pahalı olan diğer çözümlere yakın (hatta bazı yönlerden daha iyi) bir yetkinlikte ihtiyaçlara cevap verildiği görülecektir[11].

### 2.2.3. PHP ile Uygulama Geliştirme

#### 2.2.3.1. MySQL ile Veri Tabanı Uygulamaları

- MySQL ile Veri Tabanı Yönetimi
- PHP arayüzü

##### a) MySQL ile Veri Tabanı Yönetimi

Burada UNIX sistemi üzerinde MySQL'in kurulup yönetilmesi üzerinde durulacaktır. MySQL derlendikten sonra ilk yapılması gereken 'mysql' adlı veri tabanının server üzerine kurmaktır. Bu veri tabanı, server üzerindeki diğer veri tabanlarına erişim haklarını oldukça ayrıntılı bir şekilde tablolarında tutmaktadır. Bu veri tabanının yaratılması için öncelikle root olarak login olunur ve aşağıdaki komutlar işletilir:

```
>cd <MySQL directory>/bin
```

```
>./mysql_install_db
```

Böylece tüm verilerin tutulacağı '<MySQL directory>/var' dizini de yaratılmış olur.

Artık veri tabanı sunucusu arka planda çalıştırılabilir:

```
>./safe_mysqld &
```

MySQL standart olarak 3306 numaralı portta çalışmaktadır.

Veri tabanı üzerinde her hakka sahip kullanıcı, aynı zamanda sistemin 'root'udur. Şu anda root, MySQL sunucusuna şifresiz bağlanabilir. Güvenlik için bu süper kullanıcıya veri tabanına erişim şifresi verilmesi gerekir:

```
>./mysqladmin -u root -password '<şifre>'
```

Daha sonra bu şifre kullanılarak sunucuya bağlanılabilir:

```
>./mysql -u root -p
```

```
>Enter passwd: <şifre>
```

##### b) PHP Arayüzü

PHP ile MySQL tablolarını kullanmak oldukça kolaydır. İzlenmesi gereken sıra ise;

- Veritabanı sunucusuna bağlanmak (mysql\_connect, vs. gibi komutlar ile) ve bir bağlantı numarası (link identifier) almak,

- Bu bağlantıyı kullanarak belirli bir veritabanı üzerine konumlanmak (`my_sql_select_db`, vs.),
- Yine bu bağlantıyı kullanarak sunucuya bir sorgu yöneltmek (`mysql_db_query`, vs. gibi komutlar ile) ve bir sonuç numarası almak (`result identifier`),
- Bu sonucu (bir nevi `view`) PHP değişkenlerine aktararak kullanmak (`mysql_fetch_row`, vs.),
- Sunucu ile bağlantıyı kesmek (sürekli olmayan bağlantılar için `mysql_close` kullanılabilir veya script işletimi sonlanınca zaten bağlantı kesilir, sürekli bağlantılar ise web server durduruluncaya veya bağlantı sayısı belli bir değeri aşınca kadar açık kalır), şeklindedir.

#### 2.2.3.2. LDAP Kullanımı

LDAP (Lightweight Directory Access Protocol), bir nevi dizin servisi standardıdır. Dizin servisleri hiyerarşik bir yapıda (dizin yapısında), veriye merkezi olarak ulaşılması için düşünülmüş bir nevi veritabanı hizmeti veren sistemlerdir. Açık sistem olması dünya üzerinde global olarak erişilebilir bilgi kaynakları oluşturulabilmesini sağlar. Bunu yaparken coğrafi yapı (ülke, şehir), kurum içi organizasyon ve diğer özellikleri içeren hiyerarşik bir yapı kullanır.

Özellikle kurum içi organizasyon ve personelin kayıtlarının tutulabileceği elverişli bir ortam sunar. Bir kurumda çalışan veya bir üniversitede okuyan öğrencilere çeşitli servisler sunmak gerekir. Bunun için her servisin üzerinde çalıştığı makinada ayrı ayrı kullanıcı hesapları açmak gerekebilir. Böyle olunca sistemin ve kullanıcıların yönetimi zorlaşır. LDAP kullanarak bu sorun çözülebilir.

Kuruma giren personel için LDAP sunucusuna tek bir entry (giriş) eklenmeleri ve LDAP desteği veren servislerin bu sunucu üzerinden kullanıcı bilgilerine erişim ve doğrulama yapmaları sağlanabilir.

PHP'de LDAP için iyi bir arabirim sunmaktadır. Bu sayede web uygulamalarında da LDAP kullanılabilmekte ve özellikle kullanıcılara kendilerine ait verileri güncelleyebilme olanağı sunmaktadır.

- LDAP servisleri
- PHP arayüzü

##### *a) LDAP servisleri*

LDAP dizinleme sisteminde her kayda ait özellikler (attributes) ve bu özelliklerin değerleri (values) vardır. Her kaydın ait olduğu bir nesne sınıfı (object class) vardır. Bu yapı sayesinde dünya üzerindeki her LDAP kaydı kendine özgü olur. Bunun için DNS de olduğu gibi hiyerarşik bir isimlendirme kullanılmıştır. Kullanıcı bu kendine özgün özelliğini DN (Distinguished Name) özelliğinden alır.

Örneğin **dn='cn=Ferhat Bagcaci, o=Ulakbim, c=tr'** benim Türkiye içinde, Ulakbim adlı kuruluştta, adı 'Ferhat Bagcaci' olan kişi olduğumu söyler ve kendine özgüdür.

Personele ait bazı özellikler şunlardır:

- **uid (User ID)** : Sistemdeki kullanıcı adı,
- **cn (Common Name)** : Kullanıcının adı soyadı,
- **sn** : Kullanıcının soyadı,
- **mail** : Kullanıcının e-mail adresi,
- **telephonenumber** : Kullanıcının telefonu

#### *b) PHP Arayüzü*

PHP ile LDAP erişimi oldukça kolaydır. İzlenmesi gereken sıra ise;

- LDAP sunucusuna bağlanmak (ldap\_connect) ve bir bağlantı numarası (link identifier) almak,
- Bu bağlantıyı ve doğrulama bilgileri kullanarak sunucuya login olmak (ldap\_bind),
- Bind edilmiş bu bağlantıyı kullanarak LDAP işlevleri gerçekleştirmek (ldap\_search, ldap\_modify, vs.),
- Elde edilen sonuçları ilgili (associative) dizilere aktararak kullanmak (ldap\_get\_entries, vs.),

#### 2.2.3.3. Oturum Yönetimi

Kullanıcı web sunucusuna bağlandıktan sonra yapacağı her işlem için o an hangi aşamada bulunduğu bilgisi bilinmelidir. Bunun için oturum bilgisinin bir şekilde tutulması gerekir. Bu durum session management kapsamına girmektedir.

Cookiler session management sorununu istemci (client) tarafında çözen bir yoldur. Çerezler (Cookie) isim, değer, zaman aşımı süresi gibi bilgiler içerebilirler.

İstemcilere cookie göndermek için 'setcookie' fonksiyonu kullanılır. İstemcilerden gelen cookieler ise \$HTTP\_COOKIE\_VARS global associative dizisinden cookie adı index olarak verilerek okunabilir.

Cookieleri sisteme tanımlı kullanıcıları ayırtetmekte kullanabilir. LDAP sunucusu bağlantısı için kullanıcı adı ve şifresini tekrar tekrar girmemesini sağlamak için ilk olarak login ekranını gönderen script içinde komutları ile browserlerindeki user ve pwd cookieleri temizlemektedir:

```
setcookie("user");  
setcookie("pwd");
```



Sonra formdan girilen deęerler ikinci bir scriptde browsere set edilir:

```
setcookie("user", $user);  
setcookie("pwd" , $pwd);
```

Doęrulamanın yapılacağı scriptte bu deęerler alınmaktadır:

```
$user = $_HTTP_COOKIE_VARS["user"];  
$pwd = $_HTTP_COOKIE_VARS["pwd"];
```

Bu metod alıřmakla birlikte yalnızca güvenli olduęu düşünlen bir bölgede kullanılmalıdır. Aksi taktirde HTTP üzerinden açık bir şekilde giden kullanıcı adı ve řifresi rahata alınabilir. Bu nedenle doęrulama (authentication) gereken web bölgeleri ve uygulamaları için güvenli soket düzeyi (secure socket layer) kullanılması (HTTPS protokolü üzerinden) önerilir.

Ayrıca setcookie fonksiyonuyla sadece HTTPS kullanılıyorsa, cookie göndermesi sağlanabilir.

NSEnterprise Server'deki session objectlerini karşılayacak yapılar hafıza paylaşımı (shared memory) ve semafor kullanılarak kurulabilir. Hafızada script ömryle sınırlı olmayan alanlar ayırmak için shared memory fonksiyonları kullanılabilir[11].

#### 2.2.4. PHP'de Güvenlik

ok kullanıcı sistemlerin tümnde güvenlik (security), önemli bir sorun olmuřtur. Web uygulamalarında da verilerin ve kaynakların yetkisiz kişilerin erişim ve kullanımına kapatılması gerekebilir. Dolayısıyla bir Web uygulaması gerekleřtiriminde de güvenlięe dikkat edilmesi gerekmektedir.

Güvenlik, birkaç farklı kavramı ierisine alır. Bunlar arasında **doęrulama** (authentication), **yetkilendirme** (authorization) ve **gizlilik** (privacy) sayılabilir.

Bir Web-mail uygulamasında posta kutusu mesajları ile IMAP ve send mail kaynaklarına sadece ilgili posta kutusu sahibinin erişebilmesine izin verilmelidir. Burada Web sunucu ile posta kutusu sahibi arasında bir diyalog gerekleřecek ve her iki taraf da birbirine kendisini tanıtacaktır. Bunlardan kullanıcı kendisinin söyledięi kişi olduęuna delil olacak bilgiyi karşı tarafa iletecek ve karşı taraf doęrulama sağlarsa oturumu açacaktır. İşte burada örneklenen, **doęrulama** (authentication) sürecidir.

Bir veritabanından sorgulama yapabilecek kişiler ile veritabanında değişiklik yapabilecek kişilerin yetkilerinin web uygulaması tarafından doğru bir şekilde ele alınabilmesi gerekmektedir. Burada ise *yetkilendirme* (authorization) sorunu ile karşılaşilmektedir.

Web-mail uygulamasına dönmek gerekirse, kullanıcının postalarını tarayıcısına (browser) çekerken üçüncü bir kişinin araya girip bunları görebileceği açıktır. Burada ise *gizlilik* (privacy) ön plana çıkmakta ve bu sorun da daha çok çeşitli *şifreleme* (encryption) metodları ile çözülmeye çalışılmaktadır[11].

### 2.2.5. PHP ile Veritabanı

PHP'nin varlık sebeplerinin başında, ticarî Unix ve Microsoft Windows işletim sistemlerinden ayrı, ticarî olmayan Açık Sistem ortamlarında çalışan bir Script dili ile bu dille kullanılabilecek yine Açık Sistem ürünü bir veritabanı erişim ve yönetim aracına sahip olmak gelir. Perl, gerçi PHP'den çok önce Açık Sistem ürünü olarak, isteyen Web Programcısının ücretsiz edineceği bir CGI programlama dili olarak Web'in ilk gününden beri Web Programcısı tarafından zaten kullanılmaktadır. Ama Perl, metin dosyalarından yazı çekmek ve bunları biçimlendirerek rapor haline getirmek amacıyla geliştirilen bir dil olduğu için, özellikle veritabanına dayalı işlemlerde programcıya çok güçlük çıkartıyordu. ASP ise (ASP desteği sağlayan ve ücretsiz Web sitesi veren firmaların giderek artmasına ve veritabanı konusunda hem kolaylık, hem de büyük bir etkinlik sağlamasına rağmen) çoğunlukla ücretli Web sitesi aldığımız ticarî amaçlı evsahibi (Hosting) firmalarının sağladığı bir teknoloji olarak görülmüştür. PHP, tasarımcılarının MySQL ve PostgreSQL teknolojilerini geliştiren kurum ve firmalarla yaptığı yakın işbirliği sonucu ikinci sürümünden itibaren veri-yönlendirmeli Web uygulaması alanında önemli bir araç olarak belirmiştir. Dolayısıyla, PHP kurulumu dendiği zaman, kendiliğinden, MySQL kurulumu da kastedilmiş olur.

PHP, bir dildir; MySQL ve PostgreSQL ise bir demet sürücüdür. PHP ile, Web Server'a sayfasını internet ziyaretçisine göndermeden önce bir takım komutları icra etmesini, bir takım veri dosyalarını açıp içindeki bilgileri alıp bunları HTML kodlarıyla bezeyip, tarayıcıya (browser) bilinen klasik HTML sayfası olarak göndermesi talimatı verilebilir. MySQL ve PostgreSQL ise, Windows sistemlerinde Denetim Masası'nda görülen ODBC Veri Kaynağı Yönetmeni adlı araç gibi bir veritabanı sunucusu, yani Database Server programlarıdır. Yakın zamana kadar bu iki veritabanı yönetim aracı arasında fazla bir çekişme yoktu. Ancak şimdi özellikle Linux kullanıcıları topluluklarında iki program arasında verimlilik testleri yapılmakta ve hangisinin daha iyi olduğuna ilişkin yoğun bir tartışma sürdürülmektedir. PHPBuilder.com uzmanları, yaptıkları bir çok denemede, MySQL'in daha hızlı ve daha etkin çalıştıkları sonucuna varmışlardır. MySQL, bir kullanıcı lisansı 200 Dolar olan ticarî bir program iken 2000

yılı ortalarında GPL (GNU General Public License/GNU Kamusal Lisans) uygulamasına geçmiş ve ücretsiz dağıtılır hale gelmiştir. Buna göre MySQL'i bir ticarî uygulamanın içinde motor olarak kullanılacaksa üretici firmaya lisans ücreti ödemek gerekir; bunun dışındaki uygulamalar için program internetten indirilerek kullanılabilir. PHP için MySQL'e gerek yoktur. PHP, bir NT veya Win9x tabanlı Web Server'da çalıştırılıyorsa, Microsoft'un ODBC sürücülerini ile çalışabilir ve ODBC'nin okuduğu bütün veritabanlarından veri çekebilir (ODBC'nin Unix-Linux sürümü de vardır), ayrıca PHP için Adabas, dBase, Empress, FilePro, Informix, InterBase, mSQL, Oracle, Solid, Sybase, Velocis ve bir çok Unix veritabanı yöneticisi için geliştirilmiş add-on (sonradan eklenen) modüller vardır. PHP, bir çok LDAP istemci programı için yazılmış API'lere de sahiptir. PHP ile IMAP, SMTP gibi internet elektronik mektup protokollerini kullanmak da mümkündür.

#### 2.2.5.1. PHP-MySQL İlişkisi

PHP programlarında veritabanından yararlanabilmek için programın önce Web sunucusu aracılığıyla veritabanı dosyası ile bağlantı kurması gerekir. Başka bir deyişle, PHP programının veri ile arasında bir yol açması gerekir. MySQL açısından ise bu bağlantı, veri sunucusunda yeni bir oturum açılması anlamına gelir. İki program arasındaki bu ilişkiyi PHP'nin `mysql_connect()` fonksiyonu yapar. Bu fonksiyonun alabileceği üç parametre vardır:

```
$veri_yolu = mysql_connect ("localhost" , "root" , "parola" );
```

Burada "localhost" yerine MySQL programının parçası olarak çalıştığı sunucunun adı yazılır. "root" bu MySQL sunucusunda açılacak oturumun kimin adına açılacağını belirtir. "root" kelimesi, sunucunun yönetici olarak oturum açılacağı anlamına gelir: "parola" kelimesinin yerine de MySQL'i kurarken belirlenen bir kullanıcı parolası varsa, o yazılır. Bu komutta yer alan `$veri_yolu` değişkeni, açılacak veri yolunun, PHP ile MySQL veritabanı sunucusu arasındaki bağı tanıttığı işareti olacaktır. Bu bağlantı kurulduktan sonra, açılan veri yolundan gelecek bilgiler ve veritabanına gidecek bilgiler bu değişken aracılığıyla gerçekleşecektir. Veri sunucusu ile veri yolu bağlantısı kurulursa, bu değişken değer tutar hale gelir; bağlantı kurulamazsa bu değişken boş kalır. `mysql_connect()` fonksiyonunun başarılı olup olmadığını bu değişkenin durumu sınanarak anlaşılabilir. Örneğin:

```
$veri_yolu =mysql_connect("ferhat-murat", "root");  
if ( ! $veri_yolu) die ("MySQL ile veri bağlantısı kurulamıyor!");
```

Burada veri sunucusunun bulunduğu Web sunucusunun adı "ferhat-murat"tır ve oturum "root" yetkileriyle açılmıştır. İkinci satırdaki if deyimi, \$veri\_yolu değişkeninin değer içerip içermediğine bakmakta ve değişkende bir değer yoksa, bağlantı kurma girişini durdurarak, ziyaretçiye hata mesajı göndermektedir.

Bağlantı başarıyla kurulduktan sonra PHP programı, bu yoldan, veritabanı sunucusuna, hangi veritabanı dosyasından yararlanmak istediğini bildirmelidir. Buna veritabanı dosyası seçme işlemi denir ve mysql\_select\_db() fonksiyonu ile yapılır:

```
mysql_select_db( "veritabanı_adi" , $veri_yolu ) or die ("Veritabanı  
açılamıyor!".mysql_error() );
```

Bu fonksiyonun başarıyla icra edilip edilmediği, fonksiyondan dönen değer true/doğru veya false/yanlış olmasından anlaşılır. Bu değer false ise bu deyimin die() bölümü icra edilecek ve tarayıcı penceresine veritabanının açılmadığı mesajıyla birlikte MySQL'in hata mesajı da gönderilecektir. PHP'nin MySQL veritabanını seçememesi çoğu zaman kullanıcı yetkilerinin internet ziyaretçilerini kapsayacak şekilde düzenlenmemiş olmasından kaynaklanır. Bu durum gerçek Web sunucusunda ortaya çıkarsa, Web sunucusu yönetimine başvurmak gerekir.

PHP'nin MySQL ile veritabanı yönetimi için yapılabilecek 20'ye yakın fonksiyonu vardır. MySQL de bu fonksiyonlar yardımıyla çok sayıda iş yapabilir[10].

### 2.3. CGI (Common Gateway Interface)

CGI (Common Gateway Interface), Web Servisleri ile bu servislerin dışındaki programlar arasında etkileşim (ortak çalışma) platformu oluşturmak için geliştirilmiş bir standarttır.

CGI programcıları Web sunucusu üzerinde çalışan dış kaynaklı programlardır. CGI programcılarını yüksek etkileşimli Web yayınları yaratmak için kullanabilir. Web sunucuları üzerinde kullanılan yabancı kaynaklı programların standartları ve ortak ağ geçit arayüzünün diğer uygulamalarla etkileşiminin nasıl olması gerektiği belirlenmiştir. Ortak, ağ geçit ve arayüz sözcükleri bu standartların anahtar sözcükleridir ve nasıl çalıştığını tanımlar.

CGI; programcığın erişilebilir olmasını sağlamak için ortak bir yol tanımlayarak, herhangi bir kimsenin kullandığı ortam ne olursa olsun bir CGI programcığına bilgi geçirmesini sağlar.

CGI; programcık, sunucu ve diğer uygulamalar arasında bir bağlantı ya da ağgeçit tanımlayarak dış kaynaklı programları genelleştirilmiş girişler olarak kabul edilebilmesini ve diğer uygulamalara bilgi aktarılabilmesini mümkün kılar.

CGI; kullanıcıların dış kaynaklı programlara erişebilmesi için bir arayüz ya da bir yol tanımlayarak dış kaynaklı programlar için gerçekleştirilen karmaşık arayüz işlemlerini birkaç basit prosedüre indirger.

CGI'yi geliştirenler Web yayımcılarının okuyucularını karmaşıklıklardan korumasını sağlamak için güçlü ve eklenebilir nitelikte özellikler yaratmak üzere bir CGI standardı oluşturmak amacıyla bu anahtar kavramlar doğrultusunda çalışmışlardır. Okuyucunun tek yapması gereken iş, bir görüntü haritasındaki bir alana tıklamak ya da doldurma formunu tamamladıktan sonra teslim etmek olacaktır. Her şey farenin tuşuna dokunur dokunmaz otomatik yapılıyor gibi görünür ve kullanıcı nasıl veya niçin gibi sorular sorma durumuyla karşılaşmaz, her şey kendi kendine yürür.

Bir Web yayımcısı için özellikle güçlü Web yayımları yaratırken CGI programcıklarının sunduğu avantajlardan faydalanmak istiyorsa, CGI programcığının nasıl çalıştığını bilmek çok önemlidir.

Kullanıcı sadece yaptığı işlerin sonuçlarını görür, arkadaki manzaradan habersizdir. Programcıklar yürütülürken arka planda gerçekleşen olaylar şöyledir:

- Okuyucunun inceleyicisi girişleri Web sunucusuna aktarır.
- Sunucu sistem sırası geldiğinde girişleri CGI programcığına aktarır.
- CGI; programcığı girişi işler ve eğer başka bir uygulama gerekiyorsa uygulamayı programcığa geçirir ve çıkışı Web sunucusuna gönderir.
- Web sunucusu okuyucunun cevabını onun inceleyicisine geri yollar. Bir CGI programcığının çıkışı; bir veri tabanı sorgulamasının sonucundan, okuyucunun cevabına göre yaratılmış yeni bir dokümana kadar her şey olabilir.
- UNIX sistemlerinde CGI programcıkları usr dosya sistemi içinde cgi-bin olarak adlandırılan kataloglara yerleştirilmiştir ve CGI uygulamaları da yine usr dosya sistemi içinde cgi-scr olarak adlandırılan kataloglara yerleştirilmiştir[12].

### ***2.3.1. CGI İle Hazırlanan Kullanıcı Arayüzlerinin Başlıca Avantajları ve Dezavantajları***

#### **2.3.1.1. Avantajları:**

- *Kolay ve Hızlı Hazırlanabilmesi:* CGI arayüzlerinde, diğer kullanıcı yüzlerindeki birçok çabaya gerek yoktur. Kullanıcıyla ilişkiyi www tarayıcısı (web browser) yaptığından, web tarayıcısı için çoktan hazırlanmış karmaşık kullanıcı arayüzü işlemleriyle uğraşmak zorunda kalınmamaktadır.

- *İstenilen Herhangi Bir Programlama Dilinin Kullanılabilmesi:* CGI, kullanıcıları tek bir programlama diline mahkum kılmamaktadır. Perl, C/C++, VisualBasic gibi diller kullanılabilir.
- *Kolay Kullanılır, Kullanıcının Alışık Olduğu Arayüzler:* CGI arayüzleri internet kullanan herkesin alışık olduğu Netscape, Lynx, Mosaic gibi web tarayıcılarından yararlanmaktadır.
- *Değişik Ortamlarda Çalışabilmesi:* Programınızın asıl çalıştığı sistem web sunucusunun çalıştığı sistem olmasına karşın programa, bilgisayar ağına bağlı herhangi bir bilgisayardan da ulaşılmaktadır.
- *Dağıtık Ortamlarda Kullanılabilme Olasılığı:* Programın kullanıcı arayüzü (web tarayıcısı) başka bir ortamda, web HTML formları başka bir sistemde, CGI programları başka bir sistemde kullanılabilir.

#### 2.3.1.2. Dezavantajları:

- Dikkatsiz yazılmış CGI programları güvenlik açıklarına neden olabilmektedir.
- Kullanıcı arayüzleri çok kolay ve hızlı oluşturulmalarına karşın, web tarayıcısının yetenekleriyle sınırlıdır.

CGI programcıları (script) kullanılarak okuyucu ile gerçek bir etkileşim içinde güçlü, kişisel ve profesyonel web yayımları oluşturulabilir. CGI programcıları web sunucusu ile diğer uygulamalar arasında bir ağıgeçit gibi davranan dış kaynaklı programlardır. CGI programcılarını okuyuculardan gelen girişleri işlemek için kullanabilir. Böylelikle okuyucu ile çift yönlü bir iletişim yolu kurulabilir. Okuyucu girişleri doldurma formlarının verileri şeklinde olabildiği gibi veri tabanı sorgulamaları için anahtar sözcükler ya da okuyucunun inceleyicisinin ya da bağlantının tanımladığı değerler şeklinde de olabilmektedir[13].

#### **2.3.2. CGI'nin Önemi ve İşlevi**

Web sayfaları tasarımında sayfaları kaç kişinin ziyaret ettiğini öğrenmek, ziyaretçilerin mesaj bırakmalarını ya da bir sipariş vermelerini sağlamak ve buna benzer istekler olabilmektedir. CGI, tüm bunları ve daha fazlasını yapmak için birtakım imkânlar sunmaktadır. Günümüzde CGI, basit bir sayaç veya e-posta programından tutun da daha karmaşık ve veri tabanı kullanan pek çok siteye kadar her yerde kullanılmaktadır. Örneğin; genel olarak bilinen amazon.com, yahoo.com veya hotmail.com sitelerinin gerisinde hep CGI programlamanın olduğu görülmektedir.

Bir CGI programı genel olarak 4 adımdan geçmektedir, ancak bunların bazıları her uygulamada olmayabilir:



- Kullanıcıdan bilgi alma,
- Alınan verilerin bir veri tabanına işlenmesi,
- İstenen verilerin veri tabanından bir sorgu sonucu alınması,
- Kullanıcıya işlenmiş bilgilerin gönderilmesi[13],

Bu adımları kullanabilecek uygulama sayısı sınırsız gibidir. Birkaç örnek aşağıda verilmiştir.

*Sayaç:* Web sitesinde bulunan ziyaretçi sayısını görmek için kullanılan uygulamalardır. Text ve image olmak üzere iki türü vardır.

*Web Site Trafik Analizi:* Bu uygulama yardımıyla web sitesine giren ziyaretçiler hakkında çok ayrıntılı bilgilere sahip olunabilir. Ziyaretçilerin web sitesine hangi gün ve saatte, hangi arama motorlarını kullanarak, hangi sitelerden geldiklerini ve ziyaretçiler hakkında daha birçok ayrıntılı bilgiyi bu sistem ile öğrenebilir.

*Anket Formu:* Ziyaretçilerin belli aralıklarla belli konulardaki görüşlerini toplu olarak almak ve bu görüşlerin değerlendirilmesi için kullanılır.

*Site İçi Arama:* Ziyaretçiler geniş içerikli bir sitede aradıkları bilgiye ulaşamayabilir veya ulaşmak için uzun zaman harcamak zorunda kalabilirler. Bu problemin engellenmesi için web sitelerine site içi arama motoru adapte edilir. Bu sayede ziyaretçiler web sitesi içinde arama yapabilirler ve aradıkları bilgiye çok hızlı bir şekilde ulaşabilirler.

*Kontak ve İstek Formu:* Ziyaretçilerin web sitesi üzerinden site sahibi ile iletişim kurması için hazırlanan uygulamalardır. Ziyaretçi site üzerinde dolduracağı bir form vasıtasıyla isteklerini ve görüşlerini site sahibine iletir.

*Mailing List:* Web üzerinden toplu mail yollama sistemidir. Bu sistem yardımıyla tek mouse tıklamasıyla siteye üye olan ziyaretçilere toplu olarak ve çok hızlı bir şekilde mail yollanabilir.

*Ziyaretçi Defteri:* Ziyaretçilerin web sitesi üzerinden dolduracağı bir form vasıtasıyla site ile ilgili görüşlerini belirtebileceği sistemdir. Ziyaretçilerin yazdıkları mesajlar, diğer tüm ziyaretçiler tarafından da okunabilir özelliktedir.

*Forum Sistemi:* Ziyaretçilerin belirli konularda aralarında fikir alışverişinde bulunması için kullanılan uygulamalardır. İstenirse üyelik sistemi ile senkronize çalışması sağlanabilir.

*Portal Modülleri:* Portal modülleri, İnternette sıkça karşımıza çıkan portalların tüm özelliklerini kendi bünyesinde toplayan yazılım parçalarıdır. Bu yazılım parçalarının hepsinin veya istenen bazılarının kullanımı ile başarılı ve son derece profesyonel portallar oluşturulabilir.

- Açık arttırma
- E-kart
- Genel arama motoru
- Kişiselleştirme
- Elektronik bülten yollama
- Haber, finans, havadurumu
- Sohbet
- Üyelik ve üye yönetim
- Site yönetim
- Anket
- E-ticaret (Kredi kartı process, sipariş takip vs.)
- Raporlama
- Banner yönetim
- Ajanda
- Oyun
- Email sistemleri

*Web Tabanlı Mail Sistemi:* İnternet üzerinden ziyaretçilere email hesabı veren sistemdir. Bu sistem yardımıyla Hotmail benzeri hatta çok daha komplike özelliklere sahip bir sistem kurulabilir. Mesala kullanıcılara mesajlarını sms olarak bildirebilir veya sisteme wap özellikleri entegre edilebilir.

*Web Tabanlı Site Yönetimi:* Bu sistem yardımıyla web sitesindeki her türlü veriyi, tasarım bilgisi ve teknik bilgi sahibi olunmasa dahi kolaylıkla yönetilebilir.

*Veri Tabanı Yönetim Sistemi:* Bir veritabanının web tabanlı olarak yönetimi için kullanılan sistemlerdir. Bu sistemler kullanılarak SQL gibi karmaşık veritabanı dilleri bilmeden veritabanı üzerinde istenilen değişiklikler yapılabilir.

*Elektronik Ticaret Sistemleri:* Bu sistemler vasıtasıyla ürünler internet üzerinden pazarlanabilir, stok ve cari hesap takibini yapılabilir kısacası firma internete taşınabilir.

*Online Destek Sistemi:* Bu sistem müşterilerle internet üzerinden yazılı ve sözlü olarak iletişim kurmak ve onlara teknik destek ve bilgilendirme hizmetinde bulunmak için kullanılır.

*Hesaplama Sistemleri:* Her türlü matematiksel hesaplamanın internet üzerinden otomatik olarak gerçekleştirilmesi için kullanılırlar. Örneğin bir ürünün bazı özelliklerinin girilmesi ve çıktı olarak bir başka özelliğinin alınması gibi.



*Banner Yönetim Sistemi:* Web sitelerinde yayınlanan bannerların istatistiki olarak takip edilmesi için kullanılan uygulamalardır. Bu uygulamalar sayesinde yayınlanan bannerların gösterim sayıları, tıklanma sayıları ve daha bir çok istatistiki bilgiyi alabilir ve yayınlanan bannerların gösterim periyotları ayarlanabilir[14].

### **2.3.3. CGI Programcıları İçin Bir Programlama Dili Seçme İşleminin Gerçekleştirilmesi**

CGI programları için istenen birçok dil kullanılabilir. Ancak internet geliştiricilerinin kabul ettiği de facto standart Perl, arkasından C'dir. Perl, neredeyse CGI'in doğumundan beri kendine bu alanda pek rakip tanımamaktaydı. Bunun birkaç sebebi vardı:

*Perl'in Avantajları:*

- Perl öğrenmesi kolay bir dildir (yine de bazı programcılar kodlarını anlaşılması çok zor şekilde yazmaktadır).
- Yukarıdaki sebeple, dünya üzerinde birçok geliştiricisi vardır ve karşılaşılan sorunların çözümünü internet üzerinden bulmak kolaydır.
- Yine aynı sebeple, varolan uygulamaları geliştirmek için, Perl bilen üniversite öğrencileri, mühendisler vb. birçok kişi görevlendirilebilir.
- CGI uygulamalarında en çok karşılaşılan görevlerden biri olan karakter dizilerinin işlenmesi (string manipulation) için Perl çok güçlü bir dildir.
- Kendini kabul ettirmiş olması nedeniyle, Perl için yazılmış birçok modül bulunmaktadır ve bunlar uzun zamandır kullanıldığı için karşılaşılabilen birçok sorunun çözümü test edilmiş ve sağlamlaşmışlardır[12].

Genel olarak hangi dilin seçileceği, aslında bir mühendislik problemi gibidir ve şu soruların cevaplarında saklıdır:

- Platform Seçimi: Bu genel olarak bir işletim sistemi ve Web sunucusu seçimidir. Bazı diller, işletim sistemi seçildiği anda dışarıda kalmaktadır. Örneğin, MacOS dışında bir seçim yapıldığı anda AppleScript de dışarıda bırakılmaktadır. Genel olarak Web sunucularında, sağladıkları yüksek güvenlik ve sağlamlık nedeniyle, Unix ve türevi işletim sistemleri seçilmelidir. Örneğin, Solaris, HP-UX, Linux ve FreeBSD sunucu olarak en çok kullanılan Unix sistemleridir.
- Dil Seçimi: Kendini kabul ettirmiş birçok dil, birden fazla işletim sistemine taşınmıştır. Örneğin Perl ve C varolan bütün işletim sistemlerinde vardır.
- Uygulamanın Çalıştırılma Sıklığı: Eğer yazılacak uygulama, sunucunun bütün trafiğini önişlemden geçiriyorsa, ya da 1 gün içinde 100,000 gibi bir sıklıkta çalışmak zorundaysa, yorumlanan (interpreted) diller yerine (Perl, Python, Tcl, PHP, kabuk programları vb)

derlenen (compiled) dillerin (C/C++ vb) kullanılması daha yerinde olur. Tersî durumu ele alırsak, uygulamaların, sunucuya çok fazla yük bindirmeyeceđi biliniyorsa, yorumlanan diller işin daha kısa sürede yapılmasını sağlarlar, çünkü yorumlanan dillerin yapısı genel olarak daha basittir. Ancak bu durumda dilin basitliđi nedeniyle kendine fazla güvenen acemi geliştiricilerin hata yapma olasılığı artar.

#### **2.3.4. CGI Programcıklarının Kullanılma Sebepleri**

Bir programlama dilini öğrenmek kolay değildir, ancak; hiçbir zaman bir programın tamamı hazırlamak zorunda kalınmayabilir. Çok sayıda kullanıma hazır CGI programcığı web üzerinde kullanıma serbest şekilde bulunmaktadır. İhtiyaçları karşılamak için bu mevcut programlardan faydalanılabilir.

CGI programcığı kullanmak, okuyucu için oldukça faydalıdır. Böylelikle okuyucu kolaylık, cevaplarına otomatik cevaplar, geri postalamanın kolay yolları ve hızlı arama yapma imkânı gibi avantajlar kazanır. Ağgeçit programcıkları iş emirlerini, sorgulamalarını ve başka birçok şeyi otomatik olarak yapmayı sağlamaktadır. CGI programcıklarının genel kullanım alanları şunlardır:

- Web sayfalarının ziyaretçilerini izlemek ve erişildikçe güncellenen numaraları sürekli olarak postalamak,
- Girişleri işlemek, örneğın bir katar aramak ve aramanın sonuçlarını taşıyan dokümanı çıkış olarak göndermek,
- Kullanıcı tanımlamalarını ve şifre bilgilerini geçerli kılmak ve web üzerinde girilmesi sınırlı alanlara yetkili okuyucuların erişimini sağlamak,
- Görüntü haritaları girişlerini işlemek ve okuyucuyu ilişkili dokümanlara yönlendirmek,
- Okuyucunun kullandığı inceleyicinin tipine dayalı dokümanlar oluşturmak,
- Bir veri tabanına ya da indekse gönderilen cevapları incelemek ya da okuyucudan gelen geri beslemeye eklemek[13].

#### **2.3.5. CGI Programlamanın Özellikleri**

*CGI İçin Dil Seçimi:* CGI programları için istenen birçok dil kullanılabilir. Ancak yaygın olarak PERL ve C kullanılmaktadır.

*CGI Programlarında En Önemli Konu:* İşlenen verinin ham veri değil, bir kullanıcının girdiđi bilgiler olduğudur. Dolayısıyla programcı, karşı tarafta rasgele bir bilgi kaynağı olmadığını unutmamalıdır.

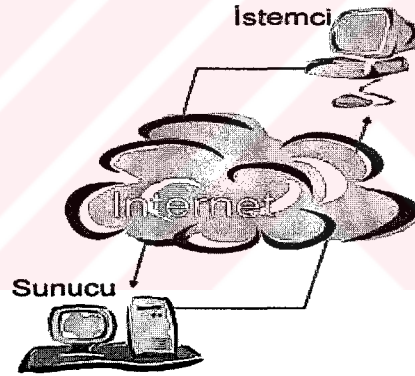
*Güvenlik:* CGI programları Web sunucusunun çalıştığı makineyi normale göre daha güvensiz hale getirebilir. İnternet'te bulunan birçok makine CGI programlarının güvenlik açıkları nedeniyle kırılmıştır yada geçici süreler için işlemez hale getirilmiştir.

*Hız ve Bellek Kullanımı:* CGI programları yazılırken mümkün olduğu kadar hızlı çalışması ve az bellek kullanması için optimizasyon yapılmalıdır. Sunucu, bir istemci için gerekli CGI programını çalıştırırken, diğer on istemcinin isteğini bekletmemelidir[9].

*Taşınabilirlik:* Uygulamalar yazılırken mümkün olduğu kadar kullanılan işletim sistemi ve Web sunucudan bağımsız olmasına dikkat edilmelidir. Bu şekilde, kullanılacak işletim sistemi veya sunucunun değiştirilmesi durumunda programların küçük değişikliklerle çalıştırılması mümkün olur.

*Kodların Tekrar Kullanımı:* Uygulama birden fazla program kullanıyorsa, ortak parçaların tek bir parça altında toplanması yararlı olacaktır. Bu şekilde hem kodların yönetimi daha kolay olur, hem de bazı kritik hataları her programcıkta düzeltmek gerekmez [10].

*Siteye Bütünleştirme:* Mümkün CGI uygulamaları sitenin genel görünümünü bozmayacak çıktılar üretecek şekilde programlanmalıdır. Bunu yaparken şablonlama sistemi kullanılırsa, sitenin görünümü değişince, CGI uygulamaları yeni görünüme kolayca adapte edilebilir.



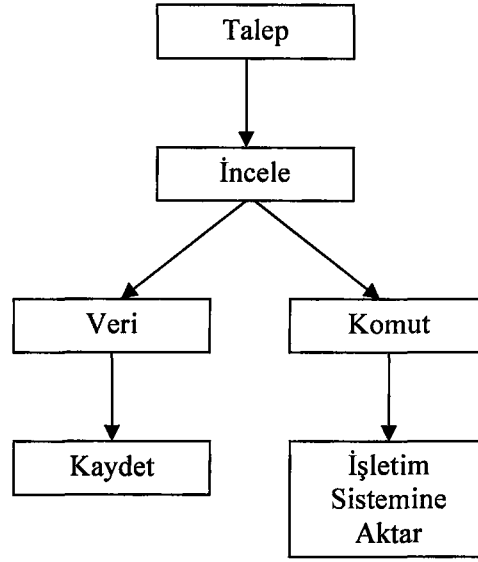
Şekil 2.2. İstemci Sunucu Çalışması

Şekil 2.2.'de görüldüğü üzere Web sunucu programı, istemci programdan kendisini çalıştıran bilgisayara gönderilen komutlar için bir geçit noktasıdır.

İnternet'te istemciler ve sunucular vardır. İstemci, bir internet tarayıcı programıdır. Bu program, kullanıcı olarak, istenilen internet adresini bulmak ve bu adresteki HTML belgesini tarayıcı penceresinde görüntülemekle görevlidir. İnternet'in diğer ucunda bulunan sunucu da tarayıcı programı olup, bir bilgisayarda çalışır. İstemci olarak gönderilen talep, sunucuya ulaştığında;

- Önce Web sunucu programı, talebi inceler.

- Taleple birlikte kendisine gelen birçok bilgiyi kaydedeceği bir ortam oluşturur.
- Sonra talep edilen HTML dosyasını kendi bilgisayarında bulur ve istemciye gönderir.



Şekil 2.3. Sunucuda Taleplerin Yorumlanması

Sunucu, kendisine gelenleri inceler ve ikiye ayırır: Veriler ve komutlar. Veriler, bizim için o anda oluşturulan ortamda kaydedilir komutlar ise çalıştırılmak üzere işletim sistemine aktarılır. Web sunucu programının çalıştığı işletim sistemi, kendisine Web sunucu tarafından iletilen komutu inceler ve gereğini yerine getirir.

Bu anlamda CGI, istemcinin Web sunucu ve onun işletim sistemine "iş yaptırttığı" noktadır. CGI programı, bu işleri belirten programdır. PERL, bu programları yazıldığı ve çağrıldığı programı yazmakta kullanılan bir dildir.

PERL veya hangi dille yazılırsa yazılsın, CGI programı kendi başına iş yapmaz, Web sunucuya o işin yapılmasını bildirir. Başka bir deyişle PERL ile yazılacak CGI programı aslında sadece Web sunucuyu "programlamaya" yarar.

### 2.3.6. Ortam Değişkenleri

CGI programcıklarında giriş ortam değişkenleri şeklinde olmaktadır. Bu değişkenler tarayıcının sunucudan istemiş olduğu bilgi ile ilgilidir. Büyük-küçük harfe duyarlıdır.

Sunucu bu değişkenleri üç değişik kaynaktan oluşturmaktadır:

**Gelen HTTP İstek Paketi:** Bu istekte bulunan bilgisayarın IP (Internet Protocol) adresini sağlamaktadır.

**Web Sunucusu Kendisi:** Örneğin, SERVER\_SOFTWARE değişkeni web sunucusu hakkında bilgi sağlamaktadır.

**HTTP İstek Başlığı Web Sunucusuna ve İstemciden Bilgi Sağlayan Değişkenler:** İstek başlığından bilgi alan QUERY\_STRING.REQUEST\_METHOD gibi ortam değişkenlerine istek başlığından bilgi sağlanabilmektedir[15].

Ortam değişkenleri CGI programı çalıştırıldığı zaman tarayıcının (ve web server'ın) programa gönderdiği bilgilerdir. Ortam değişkenleri %ENV isimli Hash değişkende saklanır.

**Tablo 2.3.** CGI Ortam Değişkenleri

| CGI Ortam Değişkenleri |                                                                        |
|------------------------|------------------------------------------------------------------------|
| DOCUMENT_ROOT          | Server'ınızın kök dizinini verir.                                      |
| HTTP_COOKIE            | Eğer varsa ziyaretçinin "cookie"sini verir.                            |
| HTTP_HOST              | Server'ınızın "hostname"ini verir.                                     |
| HTTP_REFERER           | Scriptin çağrıldığı sayfanın adresini verir.                           |
| HTTP_USER_AGENT        | Ziyaretçinin browser'ının tipini verir.                                |
| PATH                   | Server'ınızın "system path"ini verir. (Genelde /bin, /usr/sbin vs.)    |
| QUERY_STRING           | GET metodu ile yollanan bilgiyi verir.                                 |
| REMOTE_ADDR            | Ziyaretçinin IP adresini verir.                                        |
| REMOTE_HOST            | Ziyaretçinin "hostname"ini verir.                                      |
| REMOTE_PORT            | Ziyaretçinin web server'ınıza bağlandığı portu verir.                  |
| REMOTE_USER            | Ziyaretçinin kullanıcı adını verir (.htaccess korumalı sayfalar için). |
| REQUEST_METHOD         | Bilgi gönderme metodunu verir (GET ya da POST).                        |
| SCRIPT_FILENAME        | Scriptin tam yolunu verir.                                             |
| SERVER_ADMIN           | Server'ınızın webadmin'inin e-mail adresini verir.                     |
| SERVER_NAME            | Server'ınızın tam domain adını verir (Örn: cc.sau.edu.tr).             |
| SERVER_PORT            | Server'ınızın dinlemede olduğu portu verir.                            |
| SERVER_SOFTWARE        | Serverdaki web server programını verir (Örn: Apache 1.3).              |

Eğer server'da "reverse-name-lookups" yoksa REMOTE\_HOST ziyaretçinin IP adresini verir. Bazı server'larda bunlardan başka değişkenler de olabilmektedir. Bazı değişkenler server'la ilgili, bazıları ise ziyaretçi ile ilgili bilgileri içermektedir. Server'la ilgili olanların programa bildirdiği bilgiler o server'da çalışan bütün CGI programları için aynı olacaktır. Fakat ziyaretçi ile ilgili bilgiler program her çalıştırıldığında farklı olabilir.

Her CGI çalıştırıldığında bütün ortam değişkenleri atanmaz. REMOTE\_USER değişkeni sadece htaccess dosyası ile korunan dizin ya da altdizinler için atanır. Bu durumda REMOTE\_USER değişkeninin alacağı değer .htaccess dosyasındaki kullanıcı adıdır.

%ENV değişkeni CGI çalıştırıldığında otomatik olarak atanır ve istenirse tamamı istenirse sadece belli değerleri kullanılabilir[16].

### **2.3.7. Standart CGI Girdisi**

Web sunucusuna gönderilen çoğu girdi ortam değişkenlerini ayarlamak için kullanılmaktadır. Bir kullanıcı bir CGI programcığı tarafından işlenmek üzere gerçek veriyi gönderdiğinde, bu veri bir URL kodlu arama katarı olarak ya da standart bir giriş katarı yoluyla alınır. Sunucu da gerçek verinin veriyi gönderme yöntemi aracılığı ile nasıl işleneceğini bilmektedir.

Standart bir giriş olarak veri göndermek, veri iletiminin en doğrudan yoludur. Kısaca; sunucu, CGI'a standart girişten kaç byte okuyacağını söylemektedir. Programcık standart giriş akımını başlatmakta ve belirtilen miktarda veriyi okumaktadır. Uzun URL kodlu arama katarları kesilip kısaltılmasına rağmen, standart giriş akımı üzerinden gönderilen veriler kesilmemektedir. Sonuç olarak, standart giriş akımı veri aktarmak için tercih edilmektedir[15].

### **2.3.8. CGI Başlıkları**

CGI başlıkları sunucuya ilişkin bazı talimatlar içerir. Üç geçerli sunucu talimatı vardır:

- Content Type (İçerik tipi)
- Location (Konum)
- Status (Durum)

Tek bir başlık sunucu talimatlarının birini ya da tümünü taşıyabilir. Bu talimatlar CGI programcığını sunucuya aktarır. Normalde başlığı mesaj gövdesinden ayıran boş bir satır takip eder, ancak buradaki çıkışın mesaj gövdesi yoktur.

#### **2.3.8.1. CGI Başlıklarında Kullanılan Content-Type'lar**

Bir CGI başlığındaki Content\_Types (içerik tipi) alanları istemciye geri gönderilen verinin MIME(Multipurpose Internet Mail Extension) tipini tanımlar. Genellikle programcığın

veri çıktısı bir HTML dokümanı gibi tam formatlı bir dokümandır. Bu formatı başlık içinde aşağıdaki gibi belirlenebilir:

**Content-Type : [tip]/[alt tip]**

**Content-Type: text/html**

Sık kullanılan MIME tipleri aşağıda Tablo 2.4’de görülmektedir.

**Tablo 2.4. Sık Kullanılan MIME Tipleri**

| <b>Tip/ Alt tip</b>          | <b>Tanım</b>                |
|------------------------------|-----------------------------|
| <i>Application/msword</i>    | Microsoft word dokümanı     |
| <i>Audio/basic</i>           | Wav formatında ses verisi   |
| <i>Image/gif</i>             | Gif formatında görüntü      |
| <i>Image/jpeg</i>            | Jpeg formatında görüntü     |
| <i>Multipart/alternative</i> | Alternatif formatlarda veri |
| <i>Text/html</i>             | HTML formatlı metin         |
| <i>Text/plain</i>            | Düz metin                   |
| <i>Video/mpeg</i>            | Mpeg formatında video       |
| <i>Video/quicktime</i>       | QuickTime formatında video  |

#### 2.3.8.2. CGI Başlıklarında Kullanılan Location’lar

CGI programcığının çıktısı, programcığın içinde oluşturulan bir doküman olmak zorunda değildir. Location (konum) alanını kullanarak web üzerindeki herhangi bir dokümana erişebilir. Location alanı, URL aracılığı ile bir dosyaya erişmek için kullanılmaktadır. Sunucu, dosyanın konumuna göre doğrudan ya da dolaylı olarak konum referanslarını işler. Sunucu dosyayı yerel olarak bulabilirse istemciye aktarır. Aksi halde sunucu istemcinin URL’sini yeniden yönlendirir ve istemci dosyayı alır. Location bir programcık içinde aşağıdaki gibi belirlenebilir:

Location: <http://www.tvpress.com/>

#### 2.3.8.3. CGI Başlıklarında Kullanılan Status’lar

Bu alan, istemciyi izlemek üzere sunucuya bir durum satırı aktarmak amacıyla kullanılmaktadır. Durum kodları genellikle durumun ne olduğunu açıklayan bir katarla ve bunu



takip eden üç rakamlı bir kod kullanılarak ifade edilmektedir. Bu rakamlar ve karşılıkları aşağıda listelenmektedir[15].

**Tablo 2.5. CGI Başlık Statüsları**

|            |                       |
|------------|-----------------------|
| <b>1xx</b> | Henüz yerleştirilmedi |
| <b>2xx</b> | Başarılı              |
| <b>3xx</b> | Yeniden yönlendirme   |
| <b>4xx</b> | İstemci hatası        |
| <b>5xx</b> | Sunucu hatası         |

### **2.3.9. CGI'in Alternatifleri**

Aslında CGI'in birçok "alternatifi" vardır. Bunların bir kısmı temel olarak CGI prensibiyle çalışmasına rağmen, farklı isimler almışlardır. Bu nedenle aşağıdaki teknolojileri, CGI'dan farklı nitelendirilmesinin sebebi, genel olarak CGI dendiğinde Apache/Perl ya da Apache/C ikililerinin kastediliyor olmasıdır.

- *SSJS - ServerSide JavaScript*: Netscape firmasının ilk defa Livewire ürünüyle birlikte ortaya attığı bir teknolojidir. İstemci tarafında çalışan ve büyük kabul gören JavaScript dilinin sunucu tarafında da çalışabilir bir sürümüdür. Programlar, sunucu tarafında HTML sayfalarına gömülebilen JavaScript kodlarından oluşur. Sunucu, bu kodları yorumlar, çıktıyı üretir ve istemciye saf HTML kodu gönderir[33].
  - Avantajları: İstemci tarafında çalışan JavaScript'ün tüm özellikleri yanısıra, dosya okuma/yazma ve gelişmiş oturum yönetimi (session management) fonksiyonları sağlar. Veritabanı bağlantısı olarak ODBC, Oracle, Informix ve Sybase arabirimleri sağlar. Birçok Unix ve Windows sunucu ortamlarında çalışabilir.
  - Dezavantajları: Uygulamalar, site genelinde çalıştırılmaz. Her uygulamanın kendine has bir dizini olmalıdır ve sadece Netscape Web sunucularında (Netscape Fasttrack, Enterprise Server, yeni adıyla iPlanet Web Server vb) çalışabilir. Her ne kadar derlense de SSJS de aynı JavaScript gibi yorumlanan bir dildir. Ayrıca, uygulamaların çalıştırılmadan önce derlenmesi ve yönetim konsolundan tekrar başlatılması gerekir.
- *ASP - Active Server Pages*: Microsoft firmasının aynı SSJS gibi sunucu tarafında çalışan script dili çözümüdür. ASP de belli bir programlama diline bağlı değildir, IIS üzerinde çalıştırılabilen her dili kullanabilir[17].



- Avantajları: ASP, SSJS'den farklı olarak kendisi için aktarılan her dille (ör. JavaScript, JScript, VBScript, hatta gerekli modüller kurulursa Perl, PHP vb.) geliştirilebilir. Uygulamalar gerekirse bileşen (component) haline getirilebilir ve diğer uygulamalar içinden çağırılabilir. Ayrıca, ASP programlamaya, Windows NT işletim sistemiyle gelen IIS sunucusunu kurar kurmaz başlanabilir.
- Dezavantajları: Sadece Microsoft'un Web sunucularında çalışır, Windows NT ve türevlerine bağımlıdır. ASP de SSJS gibi yorumlandığı için yerleşik koddan (native code) yavaş çalışır. Şu aralar geliştirilmeye çalışılsa da, Apache Web sunucusunda çalışmamaktadır. (Visual Studio 6.0 içindeki) 2.0 sürümü itibarıyla CORBA, SMTP, POP, IMAP, LDAP ve XML desteği yoktur.
- *Lotus Notes/Domino:* IBM-Lotus firmasının önceleri Groupware için geliştirdiği, sonradan internet ortamına uyarlamaya çalıştığı hızlı uygulama geliştirme platformudur. Kendisine has bir döküman yönetim sistemine sahiptir ve kısıtlı da olsa bir veritabanı gibi kullanılabilir[18].
  - Avantajları: Lotus Notes/Domino, geliştiricilere basitleştirilmiş bir arabirim sunar ve birçok uygulama, klavyeye dahi dokunmadan yazılabilir. Ayrıca, Domino sunucu ailesinde birçok paket çözümler sunulmaktadır.
  - Dezavantajları: Lotus Notes, öncelikle kendi içine kapalı bir sistemdir. Uygulama geliştiricilerin, sunucuda birçok alana müdahalesi, veritabanındaki bilgileri SQL uyumlu bir başka veritabanına aktarması zordur. Genel olarak, İntranet uygulamalarına yönelik olduğu için Extranet'lerin bazı taleplerini karşılayamamaktadır. Son olarak, bazı uygulamalar, Web tarayıcılarında çalışmamaktadır, bu uygulamalar için Lotus Notes istemcisi kullanılmak zorundadır.
- *ColdFusion:* İnternet Allaire firmasının üzerinde dikkatlice düşünerek piyasaya sunduğu bir sunucu tarafı uygulama geliştirme platformudur. ColdFusion uygulamaları ASP ve SSJS gibi, HTML dosyalarının içine yerleştirilen ColdFusion etiketleri kullanılarak yazılır, bu dile de CFML (ColdFusion Markup Language) denilmiştir[19].
  - Avantajları: ColdFusion, Windows NT haricinde, belli başlı Unix işletim sistemlerinde de çalışabilir ve programları sorunsuzca aktarılabilir. Ayrıca, SQL, IMAP, POP, SMTP, LDAP, CORBA, XML destekleri yanında, Sunucu kümeleme (server clustering), varolan Web sunucusuna (ör. Apache) entegrasyon, CGI, ISAPI, NSAPI gibi teknolojilere de destek vermektedir. Son

olarak, piyasadaki en iyi HTML düzenleyicilerinden Hometown'la iyi bir bütünleşikliği vardır.

- **Dezavantajları:** Yüksek maliyet (Haziran 2000 itibariyle 4995\$), Unix sürümünün zaman zaman göçmesi ve zayıf bellek yönetimi.
- ***Java:*** Sun Microsystems firmasının internet'de devrim yaratan dilidir. Asıl olarak, taşınabilir cihazlar (avuçiçi bilgisayarlar, cep telefonları vb) ve elektronik aletlerde (televizyon, çamaşır makinesi, müzik seti vb) kullanılmak üzere tasarlandığı için platform-bağımsızlığı en büyük vaatlerinden biridir. Ancak, daha sonra hedef büyütülerek, Java önce istemci tarafından sonra da sunucu tarafında çalışabilir bir hale getirilmiştir[20].
  - **Avantajları:** Java herşeyden önce platform-bağımsızdır. Yazılan uygulamalar, işletim sistemi değiştirilse de hiçbir değişiklik yapılmadan yenisinde de çalışabilir. Java, NYP-Nesne Yönelimli Programlama (Object Oriented Programming)'ya dayanan bir dil olduğu için NYP'nin tüm avantajlarını taşır. Dolayısıyla öğrenme süreci, NYP'ye tanıdık olanlar için çok yüksektir ve internet üzerinde geniş bir destekçi kitlesi vardır. Bu nedenle yeni programların geliştirilmesi çok hızlı olmaktadır. Java, zamanla geliştirilerek Java Server Pages (JSP) ve Servlet (Server ve Applet kelimelerinden türetme) teknolojilerini destekler hale gelmiştir. JSP, ASP'ye çok benzeyen bir teknolojidir. Ayrıca, daha önce yazılmış Java, programların yenilerine bütünleştirilmesi de çok kolay olmaktadır. Piyasada, hemen her tür isteği karşılayabilecek modüller bulunmaktadır.
  - **Dezavantajları:** Java, iddia edebildiği şeylerin hemen hepsini yapabilmesine rağmen, halen çok yavaştır. Özellikle derlenmiş kodlara göre 20:1 gibi bir fark oluşmaktadır. Ayrıca, internet tarayıcılarının, özellikle Netscape Communicator ve İnternet Explorer'ın, Java destekleri birbirinden farklı olduğu için, geliştiriciler istemci tarafı çalışan uygulamalarında sıkıntı çekmektedirler. Bu normal olarak, sunucu tarafını etkilemese de iki tarafın arasındaki iletişimin istemci tarafı Java'ya dayandırılmaması akılcı olur.

#### 2.3.10. CGI'nin Çalıştırıldığı Web Sunucular

Piyasadaki birçok Web sunucusu CGI programlarının çalıştırılmasına izin vermektedir. Belli başlıları:

- ***Apache Web Sunucusu:***  
Apache Software Foundation'ın geliştirdiği bu sunucu, temellerini NCSA'nın Web sunucusundan almıştır, ancak zamanla artan ilgi nedeniyle, geliştiriciler bu sunucuyu

sürekli geliştirmiş, site yöneticileri de hem bedava olması, hem birçok özelliği desteklemesi, hem de performansının iyi olması nedeniyle Apache'ye çok rağbet göstermişlerdir. Kaynak kodunun açık olması nedeniyle, güvenlik delikleri sadece kendi geliştiricileri tarafından değil, 3. grup geliştiriciler tarafından da hızla kapatılmaktadır. Birçok site, kaynak kodu kendilerine göre değiştirerek, ihtiyaçlarına daha iyi cevap vermesini sağlamaktadır. Apache, CGI programlarının çalıştırılması için de ideal ve hızlı bir sunucudur. Ayrıca, Java, XML, Perl, PHP gibi dillere, yüklediği modüllerle yerleşik destek (native support) de sunabilmektedir. 2.0 sürümüyle Apache'ye multi-threading özelliği de eklenecektir. Bu nedenlerle Apache, şu an dünyada %62 gibi bir pazar payına sahiptir[21].

- *Microsoft IIS:*

Microsoft, Windows NT makineleriyle bedava olarak dağıttığı IIS'le küçümsenmeyecek bir pazar payına kavuşmuştur. ASP teknolojisinin de, IIS'in kabul görmesinde rolü olmuştur. IIS, kolay kullanımı ve epey yüksek performansı ile dikkat çekmektedir. Ayrıca, Frontpage HTML düzenleyicisi için de yerleşik destek sunmakta ve bu düzenleyiciyle yazılmış sayfaların sunucuya aktarılması çok kolay olmaktadır. Ancak, IIS'in en büyük sorunlarından biri kendisinden kaynaklanan güvenlik problemleri, diğeri de çalışmak zorunda olduğu Windows işletim sisteminden kaynaklanan problemleridir [22].

- *Netscape Enterprise ve Fasttrack Sunucuları:*

Netscape, istemci tarafında başlattığı akımı sürdürememesine rağmen, sunucu tarafında daha başarılı olmuştur. Netscape sunucularının iki sürümü vardır ve her ikisi de varolan birçok işletim sisteminde çalışabilirler. Bu ailenin birincisi, küçük firmalar ve intranet'ler için tasarlanan Fasttrack'dir. Fasttrack, aşağıda da bahsedilen SSJS teknolojisiyle birlikte gelmektedir. Bütün yönetimi Web tabanlı bir arayüzden yapılabilmektedir. Enterprise'sa (yeni adıyla iPlanet Web Server) durağan sayfa performansı yüksek, ancak CGI performansı aynı oranda yüksek olmayan bir sunucudur. Ayrıca birden fazla sanal alanın (virtual domain) olduğu bir ortamda performansı gittikçe düşmektedir. Dünyadaki kullanım açısından, Apache ve IIS'in arkasından, 3. sıradadır. Her iki sunucu da *multi-threaded* çalışmaktadır[23].

- *WebSite Pro:*

O'Reilly firmasının yazdığı birçok özelliği destekleyen ancak yeterince taraftar toplayamamış bir sunucudur. 2.0 sürümüyle beraber özel (custom) sunucu tarafı HTML etiketleri, ISAPI (Microsoft'un IIS için geliştirdiği bir API) desteği, Java Server Pages

ve Servlet, VBScript, JScript, Perl ve Python desteği de sunmaktadır. WebSite'in benzeri olan IIS'in gerisinde kalmasının en büyük nedeni, IIS'in bedava, WebSite'in ise 800\$ dolar civarında olmasıdır. WebSite Pro üzerinde, CGI programları orta denebilecek bir performans da çalışmaktadır[7].

- *Zeus Web Sunucusu:*

Zeus, Web sunucusu, yanına bile yaklaşılamayan yüksek bir performans sunmaktadır. CGI/Fast CGI, ISAPI, NSAPI, Servlet, RMI, Frontpage Eklentileri, 128 bit güvenlik vb gibi birçok teknolojiyi desteklemektedir. Zeus, son araştırmalara göre pazar payını gittikçe yükseltmektedir. 30 günlük bir deneme sürümü mevcuttur[22,24].

## **2.4. ASP, PHP, CGI Kurulumu ve Çalıştırılması**

### **2.4.1. ASP'nin Çalıştırılması**

#### **2.4.1.1. PWS Kurulumu**

PWS' nin kurulumu 3 adımda gerçekleşir.

- 1) Kurulumdan önce gerekli kontrollerin yapılması,
- 2) Kurulumun gerçekleştirilmesi,
- 3) Kurulum sonrası ayarların yapılması.

#### *Kurulumdan Önce Gerekli Kontrollerin Yapılması:*

1. Sistem Windows 2000 Professionel veya Windows 2000 Server ise:

Web Sunucu programını(IIS) kendiliğinden kurar.

2. Sistem Windows 95/98, NT4 Workstation veya NT4 Server ise:

Kişisel Web Server'ı kurmak gerekir.

Windows 95/98 için;

Bilgisayarım>Denetim Masası>Program Ekle/Kaldır>İnternet Araçları > Personel Web Server Seçeneğinin işaretlenmiş olması gerekir. Eğer Windows'u normal seçeneği ile kurulmuş ise bu seçenek işaretli demektir. Bilgisayara bir kimlik vermek gerekir. Bunun için Bilgisayarım > Denetim Masası > Ağ'ı tıklayarak, gelen diyalog kutusunun ikinci sekmesi olan "Tanımlama" seçilir ve "Bilgisayar Adı" kutusuna istenilen bir isim verilir. Bilgisayarın IP yapılandırmasının doğru çalışıp çalışmadığını kontrol etmek içinde Başlat>Çalıştır seçilip, "winipcfg" yazarak kontrol edilebilir.

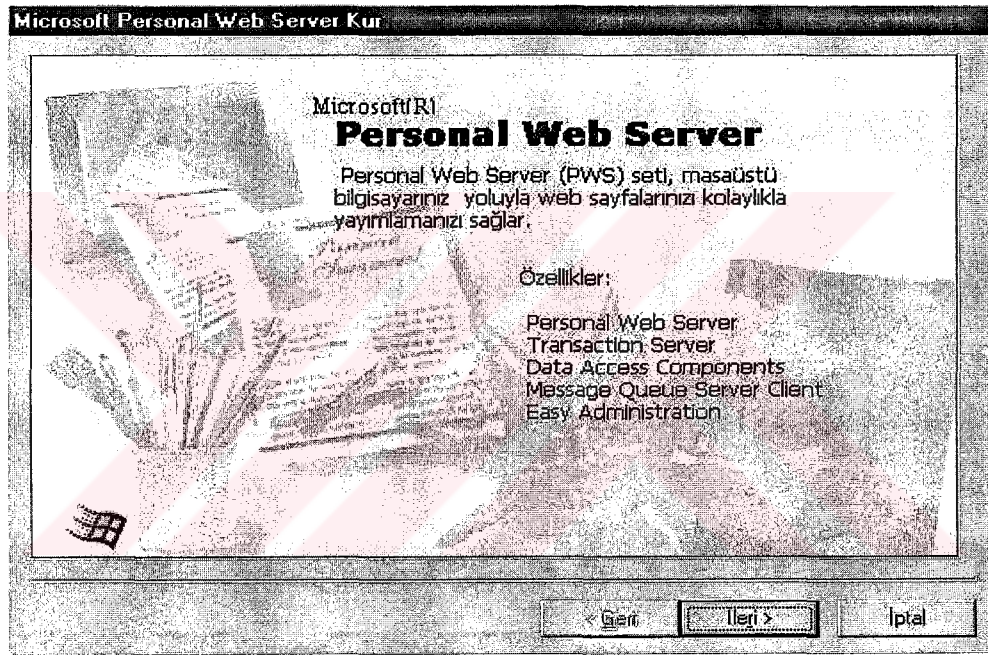
NT4 Workstation veya Server ise;

Option Pack CD-ROM'undaki "default.htm" çalıştırılır ve açılacak tarayıcı penceresinde ISS kurma seçeneğini tıklamak yeterlidir.

#### *Kurulumun Gerçekleştirilmesi:*

PWS'yi kurabilmek için Windows 95/98 CD-ROM'unun içindeki Add-ons klasörünün içerisindeki kur programını çalıştırmak gerekmektedir. Eğer Sistem, NT4 Workstation veya NT4 Server ile çalışıyorsa aynı işlemler ISS için yapılmalıdır. ISS'i kurmadan önce bilgisayara Service Pack 3 ve/veya üstü uygulanır.

Kur işlemini başlattıktan sonra ekrana aşağıdaki gibi kur işleminin ilk penceresi gelecektir:

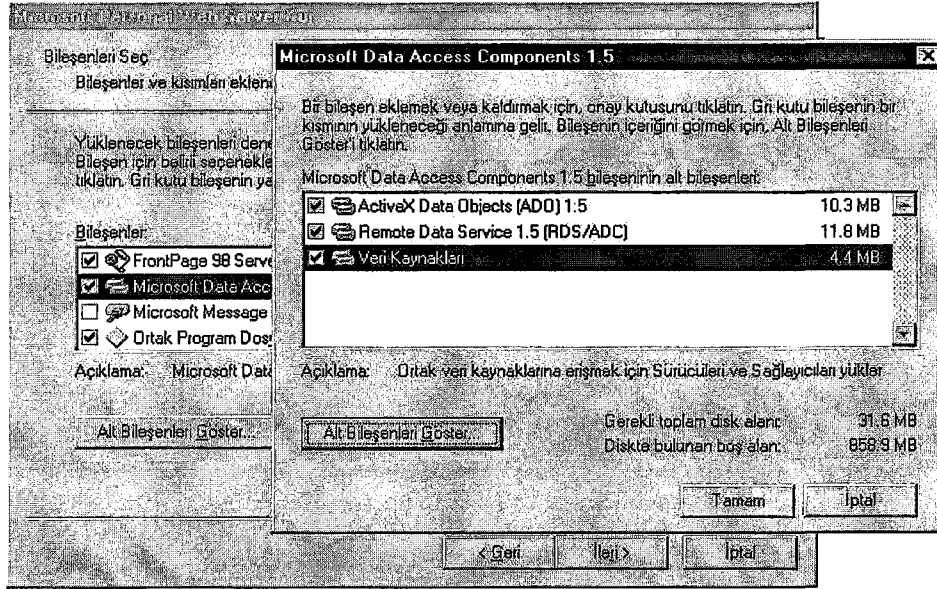


**Şekil 2.4. PWS Kurulumu**

Bu ilk pencerede ileri butonuna basılarak Personel Web Server'a ait kurulumun çeşidi ekrana gelir. Burada Kısa, Normal ve Özel olmak üzere 3 çeşit kurulum vardır. PWS içerisindeki ayarları kontrol edebilmek için "özel kurulum" seçilir.

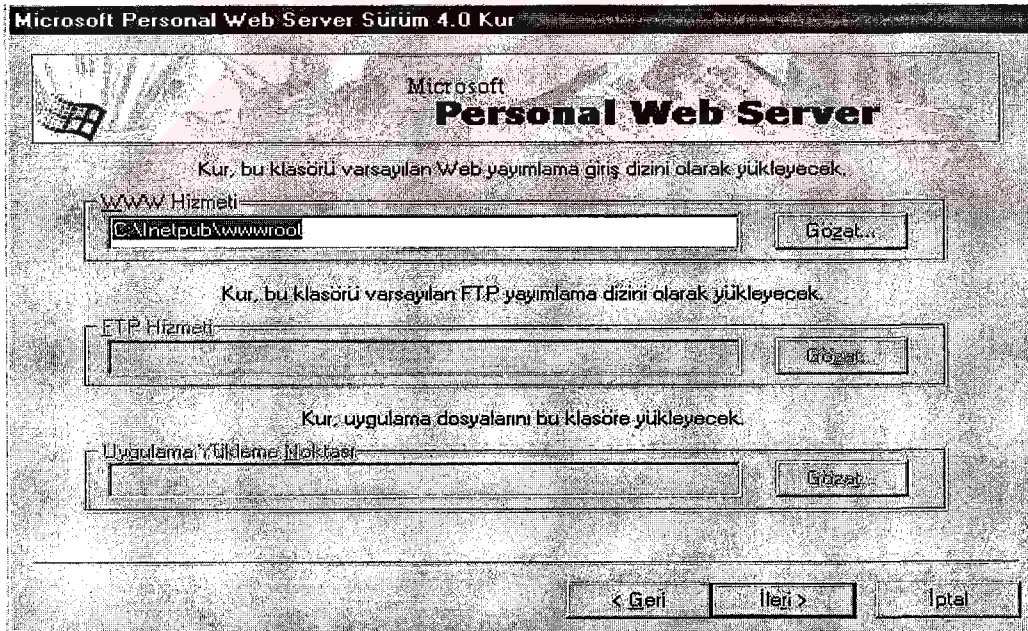
Özel seçeneği seçildikten sonra kontrol edilecek bölüm, Microsoft Data Access Components bölümüdür. Microsoft Data Access Components seçeneği seçili durumda iken Alt Bileşenleri Göster butonunu kullanarak, içerisindeki bileşenlerin seçili olup olmadığını kontrol etmek gerekir. Çünkü ADO kullanılarak hazırlanacak programlar bu bileşenleri kullanır:





Şekil 2.5. PWS Kurulumu

Bu basamakta site dosyalarının bulunacağı dizini belirlemek gerekir. Burada istenirse PWS'nin varsayılan dizini olan C:\inetpub\wwwroot'a, yada Gözet butonu kullanılarak herhangi bir dizin belirtilebilir:



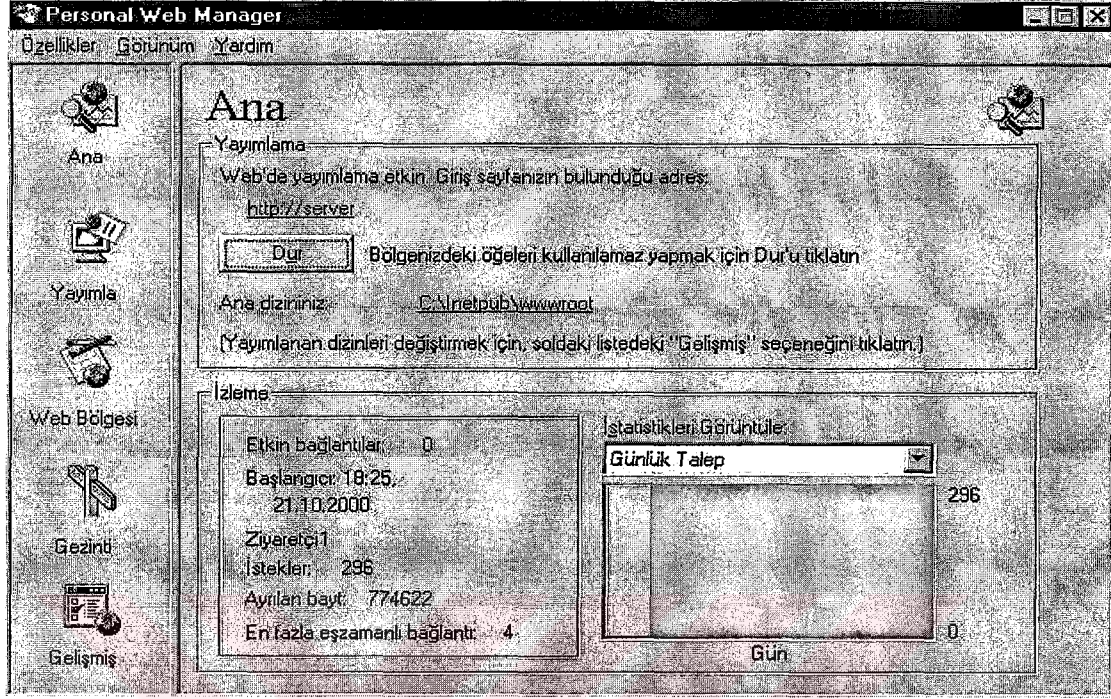
Şekil 2.6. PWS Kurulumu

Bu işlemten itibaren son butonuna basılarak kurulum bitirilir. Kurulum bittikten sonra bilgisayar yeniden başlatılır.



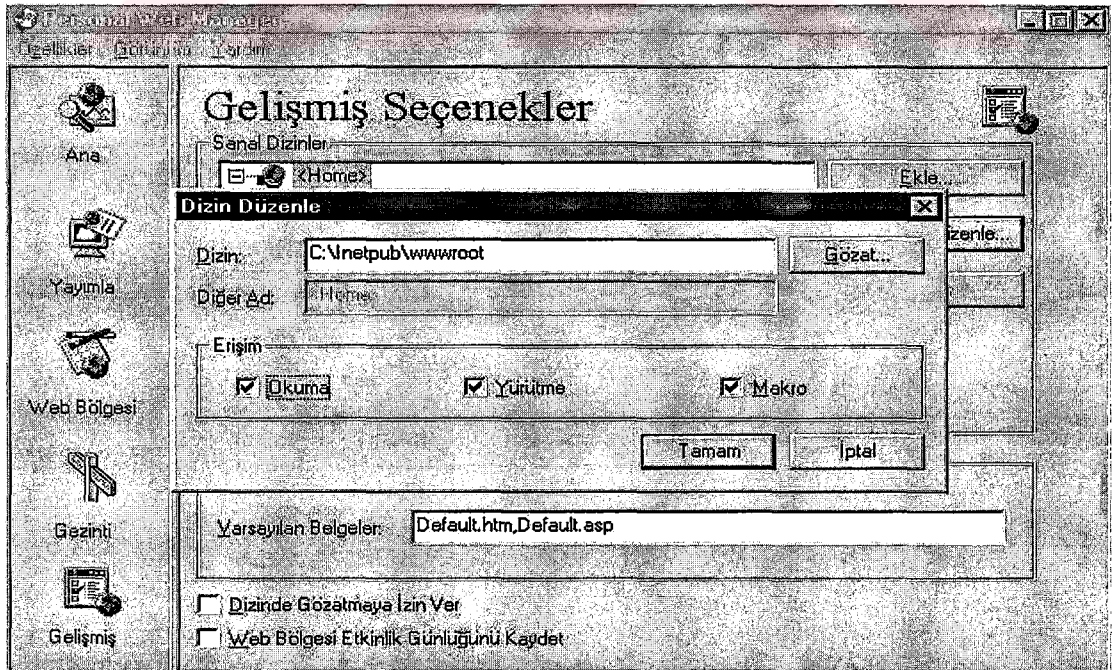
### Kurulum Sonrası Ayarların Yapılması:

PWS'nin masaüstünde oluşturduğu ikon'u tıklayarak veya görev çubuğundaki PWS simgesine tıklayarak, Personel Web Manager'ı şekildeki gibi açılır.



Şekil 2.7. Kurulum Sonrası Ayarların Yapılması

Burada sol alt köşedeki Gelişmiş seçeneğine tıklanır, bu pencerede de Özellikleri Düzenle butonuna basılarak ekrana aşağıdaki gibi bir pencere gelecektir:



Şekil 2.8. Kurulum Sonrası Ayarların Yapılması

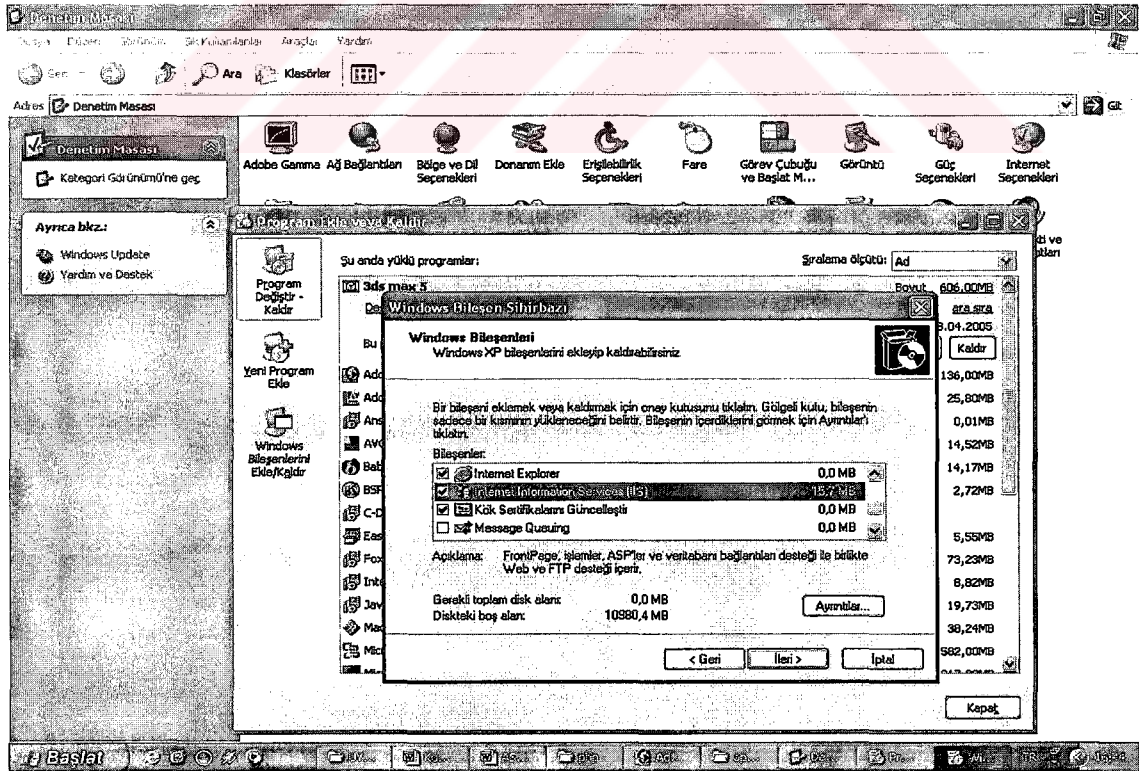


Özellikleri Düzenle butonuna basıldığında yukarıdaki gibi gelen pencerede Okuma, Yürütme ve Makro seçeneklerinin işaretlenmiş olması gerekmektedir. Örneğin Yürütme bölümü işaretlenmemiş olursa yazılan asp programlarını PWS kurulu olduğu halde tarayıcıda çalıştırılmaz. Bu da kontrol edildikten sonra PWS'nin kurulum işlemi bitmiş olur. Artık bir ASP sayfası çalıştırmak için, önce sayfayı belirlenen dizine kaydetmeli (bu normalde c:\inetpub\wwwroot dizinidir), sonra da tarayıcıyı açıp adres bölümüne **http://bilgisayar\_adı/sayfa\_adı.asp** yazmak gerekecektir. Bu adresteki bilgisayar\_adı kullanılan bilgisayarın adı (yani PWS ana penceresinde görülen isim), sayfa\_adı da yine hazırlanıp yayınlama dizinine (yani PWS ana penceresinde görülen yayın dizini) kopyalanan sayfa adı olacaktır. Ayrıca bilgisayar\_adı kısmına “localhost” yazılsa da asp sayfası yine çalışacaktır.

#### 2.4.1.2. IIS Kurulumu

Eğer kullanılan işletim sistemi XP gibi NT tabanlı ise o zaman IIS (Internet Information Services) kurmak gerekir. Bunun için iki yol vardır;

1- Başlat/Denetim Masası/Program Ekle-Kaldır/Windows Bileşenlerini Ekle-Kaldır seçeneği seçilerek ekrana “Windows bileşen sihirbazı” gelecektir. Bu pencerede Internet Information Services(IIS) kutucuğu işaretlenip ileri düğmesine tıklanır. Daha sonra Windows XP Kurulum CD’sini istenecektir, CD yi takıp ilerleyerek kurulum bitirebilir.



Şekil 2.9. IIS Kurulumu

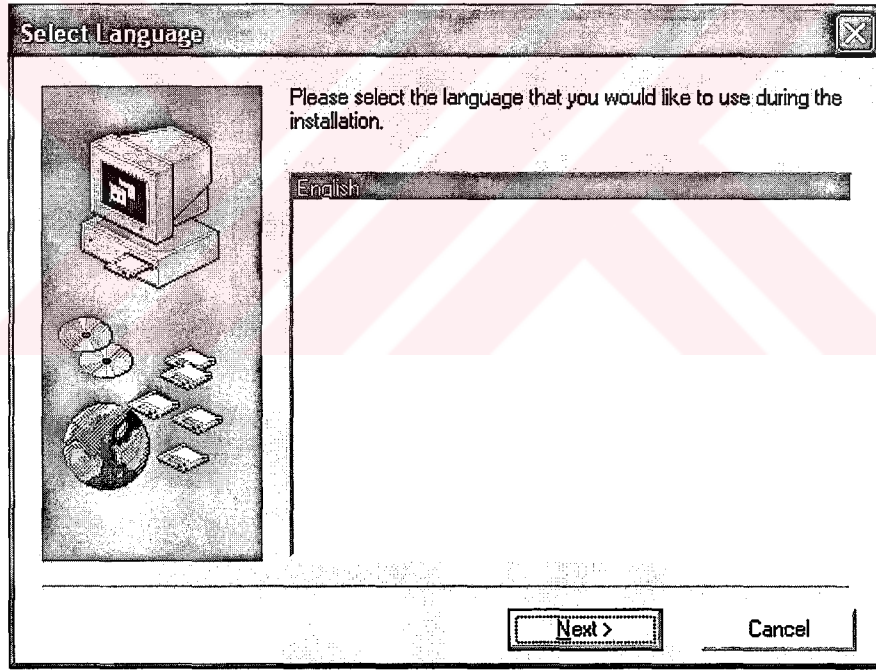


2- Windows XP Kurulum CD'sini CD/Rom'a takıp gelen pencereden "Windows bileşenlerini ekle/kaldır" seçeneği seçilerek Internet Information Services(IIS) kurulumu tamamlanır.

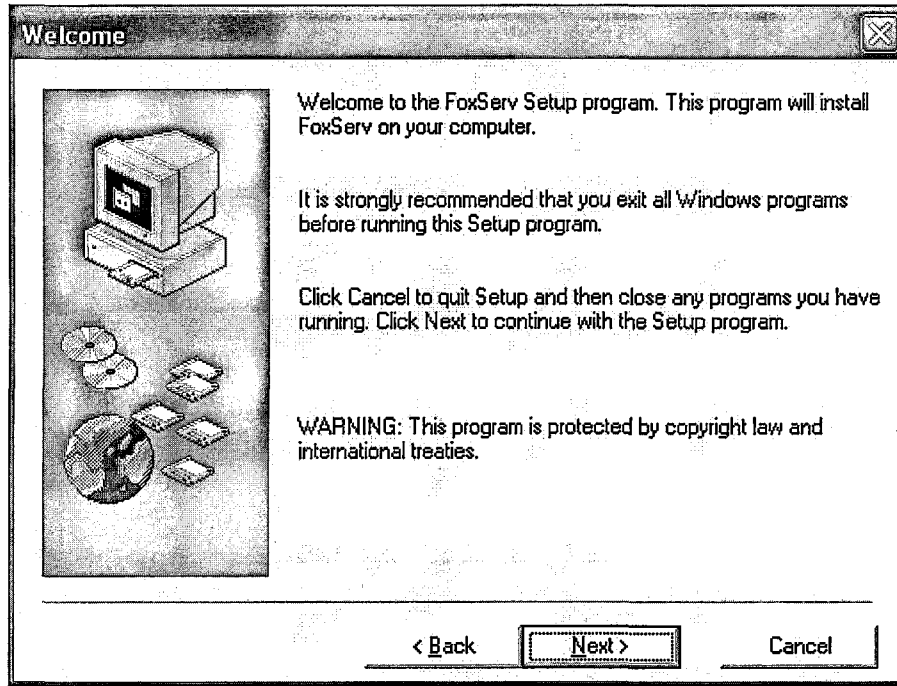
Kurulum bittiğinde eğer işletim sistemi C sürücüsüne kurulu ise C sürücüsü altında "Inetpub" klasörü oluşturulacaktır. Çalıştırılmak istenilen "asp" uzantılı dosya, Inetpub dizini altındaki wwwroot dizini içerisine kopyalanmalıdır. Daha sonra Internet Explorer adres alanına [http://localhost/dosya\\_adı.asp](http://localhost/dosya_adı.asp) yazılarak enter tuşuna basılır. Burada "localhost", kullanılan bilgisayar adı yerine geçiyor ve gösterdiği yer "C:\inetpub\wwwroot\" dur.

#### 2.4.2. PHP'nin Çalıştırılması

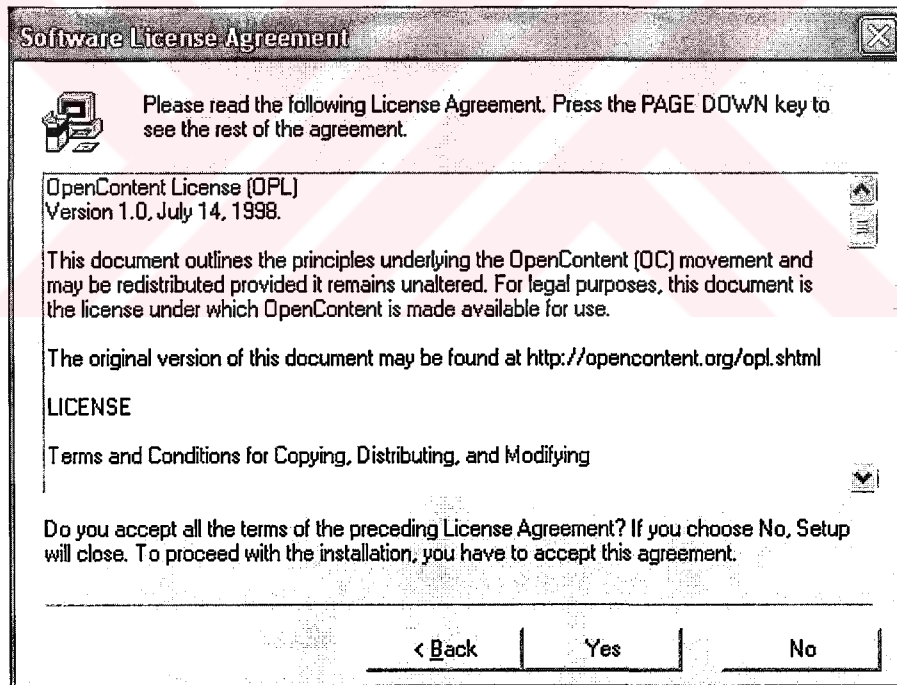
PHP'nin çalıştırılması için Apache server sistemde kurulu olmalıdır. PHP'yi çalıştırmak için bir çok program bulunmaktadır. Bu programlarla hem PHP, hemde CGI programları çalıştırılabilir. Biz ise uygulamalarımızı yaparken "FoxServ3.1Beta1" adındaki programı kullandık. Bu program ile hem PHP hemde CGI örnekleri çalıştırılabilir. Bunun için "FoxServ3.1Beta1" kurulum dosyası çift tıklanarak kurulum başlatılır.



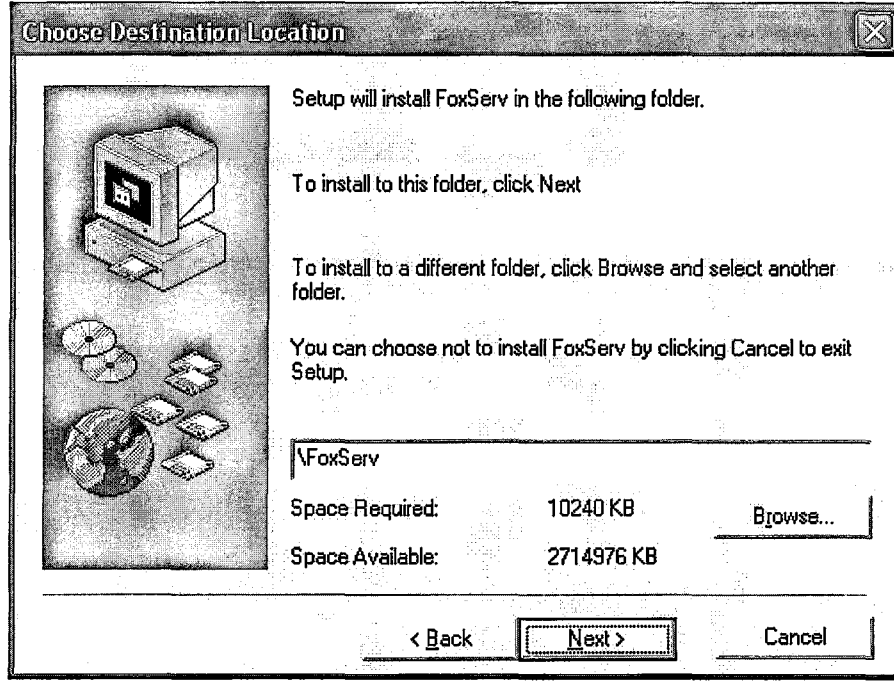
Şekil 2.10. FoxServ 3.1 Kurulumu



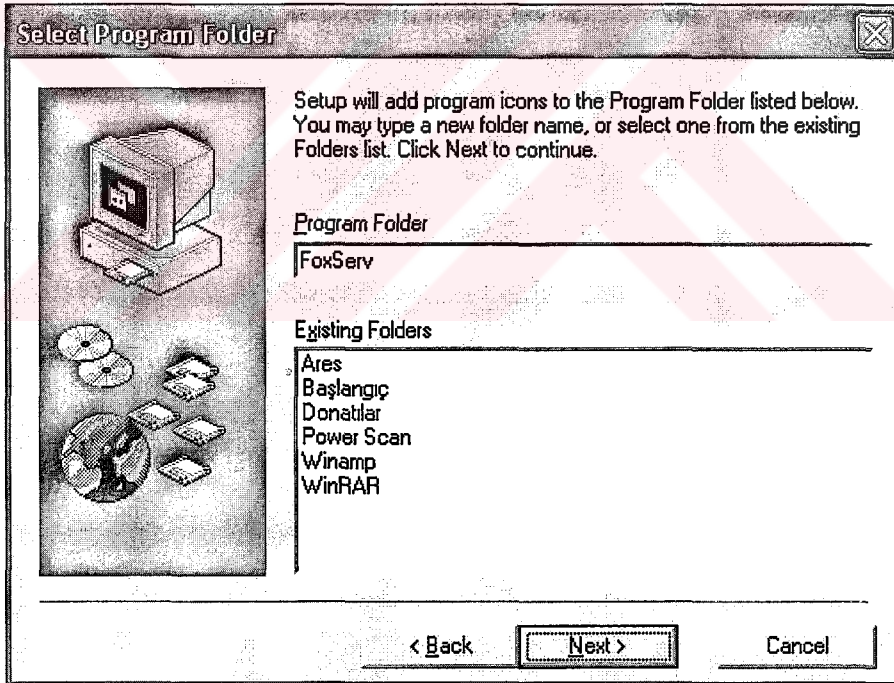
Şekil 2.11. FoxServ 3.1 Kurulumu



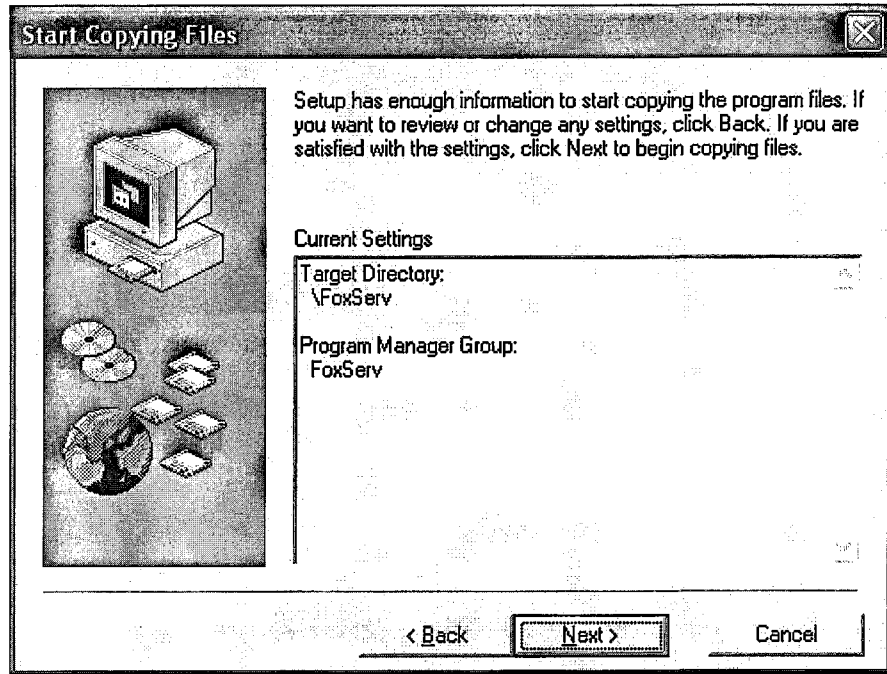
Şekil 2.12. FoxServ 3.1 Kurulumu



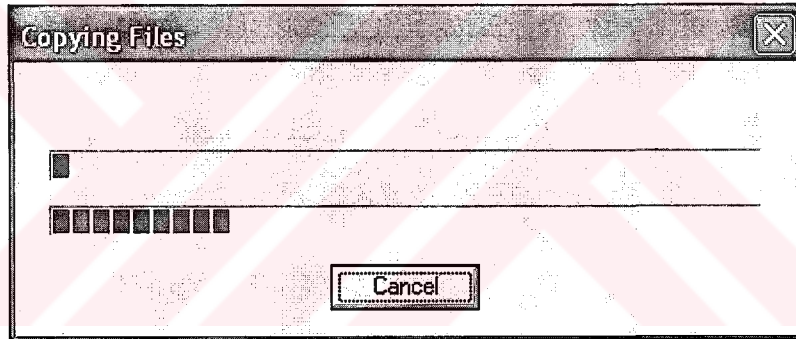
Şekil 2.13. FoxServ 3.1 Kurulumu



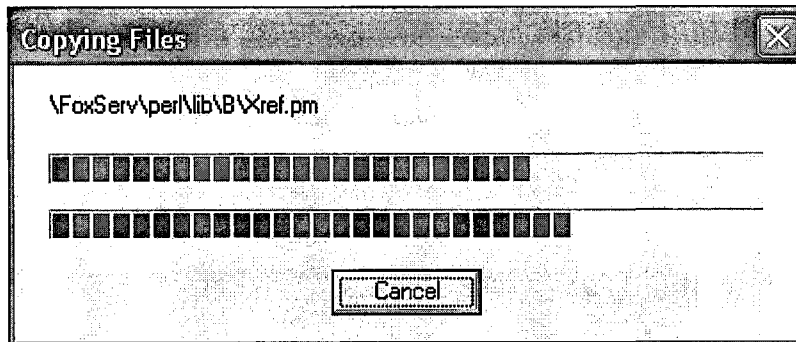
Şekil 2.14. FoxServ 3.1 Kurulumu



Şekil 2.15. FoxServ 3.1 Kurulumu

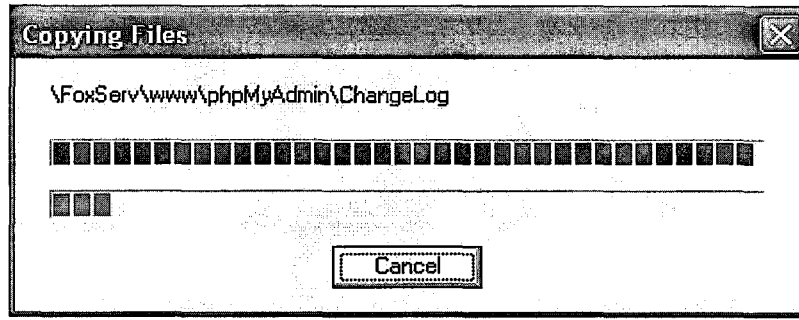


Şekil 2.16. FoxServ 3.1 Kurulumu

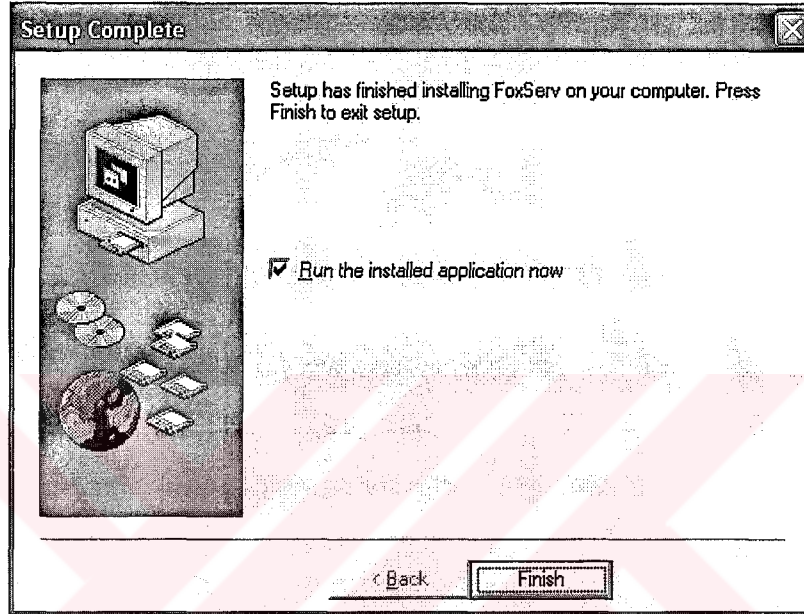


Şekil 2.17. FoxServ 3.1 Kurulumu

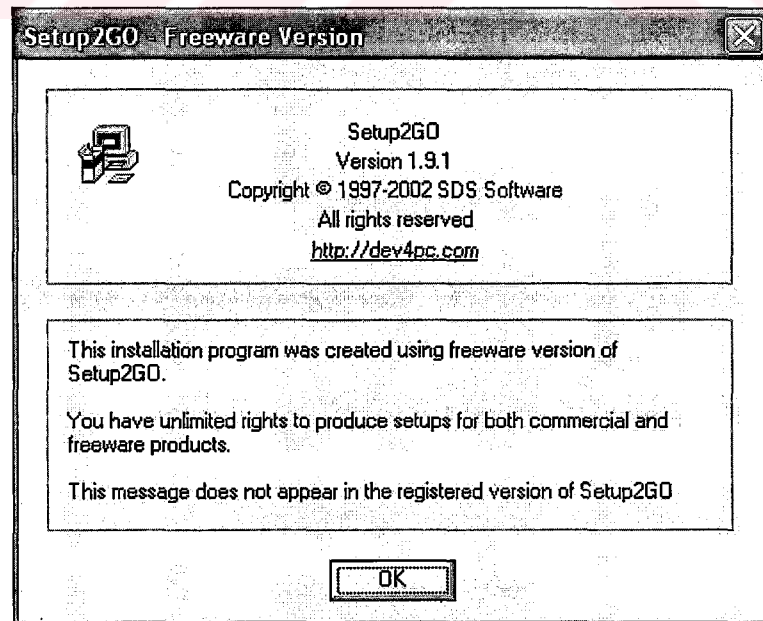




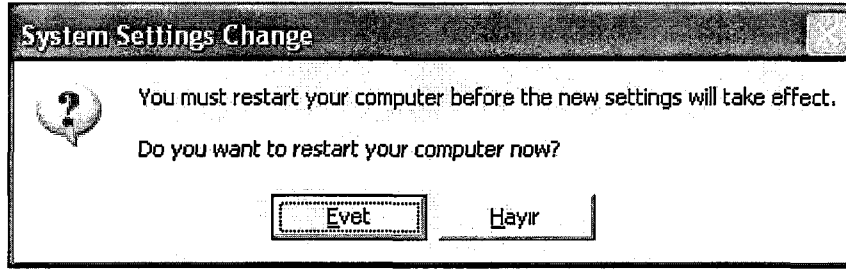
Şekil 2.18. FoxServ 3.1 Kurulumu



Şekil 2.19. FoxServ 3.1 Kurulumu



Şekil 2.20. FoxServ 3.1 Kurulumu



**Şekil 2.21.** FoxServ 3.1 Kurulumu

Program kurulumu bittikten sonra “C:\FoxServ\apache\conf” yolu içinde bulunan “httpd” isimli CONF dosyasını not defteri ile açılıp içindeki “easy” kelimeleri “fox” olarak değiştirilir ve dosya kaydedilerek kapatılır.

Çalıştırılmak istenilen “.php” uzantılı PHP dosyası C:\FoxServ\www\ altına kopyalanır, sonra C:\FoxServ\ altındaki “start” isimli dosyaya çift tıklanarak Apache server çalıştırılır, internet explorer açılarak [http://localhost/dosya\\_adi.php](http://localhost/dosya_adi.php) yazıp enter’a basılarak PHP dosyası çalıştırılır. Buradaki localhost, C:\FoxServ\www\ yolunu göstermektedir. Eğer PHP uzantılı dosya C:\FoxServ\www\ altında başka bir klasör oluşturularak onun içine kopyalanırsa, örneğin C:\FoxServ\www\test\ gibi, o zaman internet explorer’a yazılacak adres şöyle olmalıdır: [http://localhost/test/dosya\\_adi.php](http://localhost/test/dosya_adi.php)

#### **2.4.3. CGI’nin Çalıştırılması**

CGI ile yapılan sayfaları çalıştırmak için yine FoxServ isimli program kullanılmıştır. Programı yukarıda anlatıldığı gibi kurduktan sonra “C:\FoxServ\apache\conf” yolu içinde bulunan “httpd” isimli CONF dosyası not defteri ile açılıp içindeki “easy” kelimeleri “fox” olarak değiştirilip dosya kaydedilerek kapatılır. CGI uzantılı dosyaları çalıştırmak için “C:\FoxServ\www\cgi-bin” içerisine kopyalamak gerekmektedir. C:\FoxServ\ altındaki “start” isimli dosyayı yada C:\FoxServ\apache altında bulunan apache.exe dosyasını çalıştırarak apache server aktif hale getirilir. İnternet explorer adres satırına şu satır yazılır; “[http://localhost/cgi-bin/dosya\\_adi.cgi](http://localhost/cgi-bin/dosya_adi.cgi)” ve enter tuşuna basılır. Buradaki “dosya\_adi” kısmı cgi uzantılı dosyaya verilen isimdir.



### 3. WEB TABANLI YAZILIMLARIN KARŞILAŞTIRILMASI

#### 3.1. ASP, PHP, CGI'ın Karşılaştırılması

##### 3.1.1. ASP.NET ve Avantajları

ASP.NET, güçlü web uygulamalarının hızla hazırlanabilmesi için Microsoft tarafından hazırlanan en son programlama çatısı olarak özetlenebilir. ASP.NET bir programlama dili olmayıp bir programlama çatısıdır. ASP.NET'i, ASP'nin son sürümü olarak düşünmektense, yeni bir dönemin başlangıcı olarak görmek daha doğru olmaktadır[42].

##### 3.1.1.1. Daha Hızlı Web Uygulamaları

ASP.NET, derlenmiş kod ve saklanma işlemi ile hız kazanmaktadır. Daha önceki ASP sürümlerinde bir kullanıcı web sitesine girdiği zaman bilgisayar üzerinde anlık derlemeler yapılır ve bu hız kaybına neden olurdu. Artık ASP.NET ile hazırlanmış bir kod, bir ziyaretçi tarafından hiçbir zaman ziyaret edilmese de derlenebilmektedir. Saklama işlemiyle de hafızaya bilgi depolanıp, bu sayede kullanıcılara verilerin daha hızlı yollanması sağlanabilmektedir.

ASP.NET ile programcılar sayfalarını her kullanımda yeniden aratmak yerine bir kısmını 'caching' yani tamponlama işlemi sayesinde daha hızlı erişilebilir hale getirebilmektedir. Bir diğer değişiklik ise ASP.NET ile veri tabanındaki belirli verilerin saklanabilir hale getirilebilmesidir. Bu sayede çok değişmeyen bir veritabanını her seferinde tekrar yüklemek yerine, hızlı erişilebilir forma sokulmuş olmaktadır[42].

##### 3.1.1.2. Güçlü Veritabanı Fonksiyonları

ASP gibi ASP.NET'de programcılar veri tabanlarıyla etkileşimli çalışabilen web uygulamaları yapabilmesine izin vermektedir. ASP.NET'in asıl avantajı bir çok programlama aracına imkan veren Nesne Tabanlı (Object Oriented) programlama ile çalışmasıdır. Bu sayede daha fonksiyonel ve daha hızlı tasarıma imkan sağlamaktadır[42].

##### 3.1.1.3. Hafıza Sızıntısı ve Çökme Koruması

ASP.NET'teki en çok beğenilen yeniliklerin başında hafıza sızıntısı ve çökme koruması özellikleri gelmektedir. Bazı yazılımların en büyük problemlerinden birisi, dinamik bellek yönetimidir. Yazılımda işaretçi (pointer) kullanılarak dinamik olarak bellek ayrıldıktan sonra o bellek ile iş bitirildiğinde, ayrılan belleğin mutlaka bellek yöneticiye özel yöntemlerle (destructor, delete vb.) iade edilmesi gerekir. Aksi takdirde bellek sızıntısı oluşur ve bu bir süre sonra yazılımın ve işletim sisteminin beklenenden daha farklı davranmasına yol açar.

Yazılımların bazıları az da olsa bellek sızıntısı içerir ve bu tür sızıntıların tespiti zordur. ASP.NET ile bu hatalar otomatik olarak düzeltilerek, kullanıcıların her zaman web sitesine ulaşabilmesi sağlanmaktadır[42].

#### 3.1.1.4. Çoklu Dil Desteği

Programcılar artık 25'ten fazla .NET dili ile ASP.NET'i kullanabilmektedir. Bu dillerden bazıları VB.NET, C# ve Jscript NET. Geniş dil desteği sayesinde birçok programcı web uygulamalarını kolayca hizmete sunabilmektedir[42].

#### 3.1.1.5. .Net'in Yapısı

.NET çatısı 3400'den fazla sınıftan oluşan bir yapıya sahiptir ve ASP.NET'de .NET'in bir parçası olduğu için neredeyse web uygulaması olarak yapılamayacak hiçbir şey yok gibi gözükmemektedir[42].

#### 3.1.1.6. HTML'in ASP'deki Durumu

Önceki sürüm ASP'lerde göze çarpan sorunlardan biri HTML sayfasında programlamanın da görünmesi idi. Diğer bir deyişle, scriptler ve HTML aynı sayfada bulunmaktaydı. Artık gösterilen kod ile script kodu ayrılmış olduğundan, bu sorun giderilmiştir[42].

### **3.1.2. ASP.NET'in Dezavantajları**

#### 3.1.2.1. Windows Zorunluluğu

ASP.NET halen Windows işletim sistemi kullanan bir sunucu üzerinde yapılandırılmak zorundadır. Bu nedenle platform bağımsız değildir. ASP'de de olan bu sorunu çözmek için Linux sistemlerinde çalışan birkaç ASP hosting sunucusu geliştirildiyse de pek başarılı sonuçlar alınamamıştır. ASP.NET'de hali hazırda Windows ve IIS (Internet Information Server) sistemlerine bağımlı olarak hizmet vermektedir[42].

#### 3.1.2.2. Güvenlik

Güvenlik açıkları IIS'ten kaynaklanmaktadır. IIS güvenlik sorunlarının sık yaşandığı bir yazılımdır. ASP.NET'in bu açıkların azaltılacağı sanılmaktadır[42].

#### 3.1.2.3. Lisans Gereksinimi

Temelde ASP.NET bedava değildir. Bilgisayarlarda ASP.NET'i kullanabilmek için (lisanslı) bir Windows sürümüne ihtiyaç vardır[42].

#### 3.1.2.4. Eklenti Sorunu

ASP.NET kaynak olarak bedava gibi görünebilir ama eklentileri lisans gerektirmektedir. Bir işlemi yapmak için gereken eklentiler genelde ücretli olarak satılmaktadır[42].

#### **3.1.3. PHP (Personal Home Page)**

PHP, şu anda Dünya’da bireysel kullanıcılar için en popüler internet programlama dilidir. Bundan 2-3 yıl öncesinde PHP’nin adından bile söz edebilecek bilgi birikimine sahip kimse bulunmamaktaydı. Zaten bilinse bile hazırlanan web sayfasını yayımlayacak bir host şirketi bile bulmak zordu. Artık hemen hemen tüm host firmalar, PHP’ye destek sağlamaktadır.

Fiyat açısından internet programlama dillerine bakıldığında linux işletim sistemi, Apache web sunucusu ile PHP internet programlama dilini kullanırken herhangi bir lisans ücreti verilmemektedir. Sadece mySQL veritabanı için küçük bir meblağ lisans ücreti ödenmektedir. Zira bu lisans ücreti programı ticari amaç için kullananları kapsamaktadır. Onun içindir ki PHP ve teknolojilerinin internet hayatına girmesiyle birlikte hosting ücretlerini büyük ölçüde düşürmüştür. Sadece PHP destekli sunucuların fiyatlarını değil, genel itibariyle bakıldığında sunucu fiyatlarını da düşürmüştür. Ayrıca dünyada en çok kullanılan server programı olan Apache’nin PHP teknolojisine destek vermesi, PHP’ye avantajlar sağlamaktadır.

Performans açısından PHP’in diğer dillere karşı üstünlük sağladığı anlaşılmaktadır. Çeşitli şirketlerin laboratuvarlarında yapmış oldukları hız testlerinde PHP’nin önde olduğu gözükmiştir. PHP, diğer dillerden daha performanslıdır. Bunu gerek döngü yapısıyla, gerek veritabanından veri alış-verişi sırasında göstermektedir. Yalnız PHP ve ASP teknolojileri “Access veya MsSQL” kullanıldığında, ASP daha performanslı olabilmektedir. Genel itibariyle PHP, Linux türevli bir dildir ve Linux-Apache-PHP-MySQL dörtlüsünün bir araya gelmesiyle hız konusunda rakiplerini geçmektedir.

PHP’yi geliştiren Zend Firması; büyük hit alan web sayfalarının ziyaretçilerine kesintisiz hizmet sağlamak için geliştirmiştir. Gerek Zend Optimizer programı gerekse sunucular için geliştirdiği Accelerator programı ile PHP’nin hızına hız katarak sunucuların yüklü trafiklerde dahi sunucu hataları vermesini büyük ölçüde önlemiştir.

PHP, birçok platformda “Windows, Linux, Solaris, HP-UX, MacOS vb.” ve de birçok sunucuda çalışabilmesi “Apache, IIS/PWS, Xitami, IPlanet, OmniHTTPd, vb.”, PHP’ye multiplatform özelliğinden dolayı artılar sağlamaktadır. Buna karşılık Microsoft firması sadece

kendi yaptığı işletim sistemi ve sunucu programlarına destek sağlamaktadır. Bu açıdan bakıldığında, PHP gerçekten rakiplerinden ASP'ye bir üstünlük sağlamaktadır.

PHP'nin resmi sitesinde PHP dilinin referans el kitapçığını birçok dilde dağıtılması, hangi komutun nasıl çalıştığının gösterilmesi, programın çabucak öğrenilmesinde büyük bir avantaj sağlamaktadır. Çok kısa bir sürede dünyada en çok bilinen CGI script arşivlerinden daha fazla bir arşiv sağlanmıştır. Dünyaca ünlü script arşivi sitesi <http://www.hotscripts.com/>'de istenilen dillerde bedava scriptler dağıtılmaktadır.

Yukarıda da değinildiği gibi, PHP multiplatform özelliğini içerisinde barındırır. Hemen hemen tüm teknolojilere, “ODBC, COM, LDAP, OpenSSL, GD, PDF, Flash vb.” tüm SQL programlarına kolaylıkla uyum sağlamaktadır. Dilin açık kaynak koduyla dağıtılması, programın beğenilmeyen bir özelliğinin kolayca değiştirilmesine olanak sağlamaktadır.

Buraya kadar sadece PHP internet programlama dilinin artılarını verilmiştir. ASP'yi istenilen bir dille yazmamız mümkündür. Aynı olay JSP'de de mevcuttur. Ama PHP bir dil olduğundan dolayı, PHP'yi farklı bir dille yazmamız mümkün değildir. Onun için sadece PHP dilini öğrenmek gerekmektedir. PHP; C, Java ve Perl'den esinlenerek geliştirilmiştir. PHP dili bu 3 dilin birçok sistematüğını içerisinde barındırmaktadır. PHP'nin SQL programlarına bağlantı sağlamasının büyük bir dezavantajı vardır. Örneğin; bir program MySQL ile yazılmış olsun, daha sonra bu programın veritabanı değiştirilmek istendiğinde, tüm programın yeniden yazılması gerekecektir. Çünkü PHP her SQL programına sağladığı veri alış-verişinde farklı komutlar kullanmaktadır. Mesela MySQL'e bağlanmak için *mysql\_connect(host,user,pass)* komutu kullanılırken; PostgreSQL'e bağlanmak için ise *pg\_connect(host= port= dbname= user= password=)* şeklinde birbirinden bağımsız komutlar kullanması, PHP'nin en büyük dezavantajlarından biridir. Bu sorunu giderici küçük programcıklar hazırlansa da yeterli sonucu verememektedir. Böyle olunca da geliştirilecek olan programı eğer çoklu veritabanı desteğı ile donatmak istenirse, programı oluşturmak için daha fazla zaman ayırmak gerekecektir. Mesela böyle bir sorun JSP'de yoktur.

İşin başka bir boyutu da şirketlerin geliştirdikleri programları lisanslı olarak dağıtabilmeleri için dosyaları şifrelemeleri gerekmektedir. Bu iş için ASP kullanıcılara dosyaları DLL olarak yazıp şifreleyebilmektedir. Yine JSP kullanıcıları ise dosyalarını servlet yapabilmekteler. Bütünleşik bir kod çözücü program ile dosyaları şifreleyerek oluşturulan projeler, lisanslı olarak dağıtılabilmektedir. PHP'de ise dosyaları şifreleyici bütünleşik bir program bulunmamaktadır. Ama Zend Firması yazılım firmaları için bir kod çözücü programı

geliştirmiştir. Yalnız bu programın bazı dezavantajları vardır. Örneğin bu program lisanslı olarak dağıtılmaktadır. Ayrıca Zend Encoder ile şifrelenen dosyaların sunucuda görüntülenebilmesi için Zend Optimizer programının da sunucuda kurulu olması gerekmektedir. Aksi takdirde PHP yorumlayıcısı şifrelenmiş dosyaları çözememektedir. Bu da PHP için bir dezavantaj olarak gösterilebilir. PHP için yöneltilebilecek bir dezavantaj da C programlama diliyle yapısından kaynaklanan hafıza şişkinliğidir. PHP’de çöp toplayıcısı özelliğinin olmasından dolayı hafızayı boşaltma tamamen programcıya bırakılmaktadır. Ama Java tabanlı dillerde “JSP, ColdFusion MX” çöp toplayıcısı özelliği bulunmasından dolayı, hafızayı boşaltma işi otomatikman gerçekleşmektedir[41].

#### 3.1.3.1. Çöp Toplayıcısı

Bir program yazıldığı zaman çeşitli değişkenlere değerler yüklenir. Bu değişkenlere yüklenen her değer de programı çalıştıran bilgisayarın hafızasında bir yer kaplamaktadır. Bir programı oluştururken özellikle döngülerde değişkenler oluşturulmaktadır. Bu oluşturulan değişkenleri kullandıktan sonra değişkenleri silmek gerekmektedir. Aksi takdirde hafıza boşa kullanılmış olur ve bilgisayar yavaşlar. İşte Java programlama dilinin özelliği; bu kullanılan değişkenlerdeki değerleri tespit edip bilgisayarın hafızasını boşaltmaktır. Dolayısıyla kullanıcılara verilen değişkenleri silme görevini alır ve kendisi halleder. Eğer PHP kullanılıyorsa ve özellikle SQL’de ki veriler bir döngü ile alınıyorsa muhakkak döngünün sonunda hafızayı boşaltıcı komutları kullanmak gerekmektedir. Bu da hızı artıracaktır.

#### 3.1.3.2. PHP’nin Avantajları

Personel Home Page’in kısaltmasından oluşan PHP zaman içinde çok fazla taraftar bulması sonucu insanlar arasında adı PHP, Hypertext Preprocessor’ kısaltması olarak da değiştirilmiştir[42].

#### *Açık Kaynak Kodu:*

Açık kaynak tabanlı her şey çok hızlı gelişmektedir. Açık kaynak kodu sayesinde herhangi bir hata ya da açık bulunduğu anda, saatler içinde yamalar yapılabilmektedir[42].

#### *Platform Bağımsızlığı:*

PHP ile bir işletim sistemine mahkum kalınmamaktadır. Unix, Linux, Solaris ve Windows altında PHP sunucusu kurulabilir. Ayrıca PHP dünya üzerinde kullanılan birçok web sunucusu çözümünde desteklenmektedir. Örneğin Apache, Microsoft IIS ve Microsoft PWS PHP desteklidir[42].

#### *Maliyet Hesapları:*

PHP açık kaynak kodu ile yazıldığı için herhangi bir maliyeti bulunmamaktadır. Bunun yanında şirketler PHP'nin açık kaynak kodu üzerinde çalışma yapıp gerekli değişiklikleri gerçekleştirerek, istedikleri sonucu alabilmektedir. İnternet üzerinde binlerce sayıda PHP eklentileri ücretsiz olarak bulunabilmektedir. PHP ile hazırlanmış bir portal, saniyeler içinde çalışır hale getirilebilir[42].

#### *MySQL Faktörü:*

MySQL veritabanı, PHP ile mükemmel sonuç vermektedir. Bütünleşikmiş izlenimi veren bu birliktelik, performansı çok fazla etkilemektedir. İnternet üzerinde MySQL yönetimi ve ayarlaması için PHP ile hazırlanmış birçok kaynak vardır. Ayrıca MySQL birçok veritabanında bulunmayan fonksiyonlara sahiptir[42].

#### *Ekonomiklik:*

- Tüm eklentileri ile birlikte ücretsizdir.
- Değişebilir değişkenler
- Kurallı ifadeler
- Semafor ve paylaşılan hafıza
- Yerleşik oturum desteği
- Çıktı yastıklama
- Serileştirilme
- Java desteği
- Windows COM desteği
- XML ve SOAP desteği
- Şifreleme ve OpenSSL[43]

#### *Güvenlik:*

PHP çok güçlü bir dil ve yorumlayıcıdır. Gerek sunucunun içinde bir modül olarak ve gerekse ayrı bir CGI programı olarak çalışırken; komutları yorumlama, dosyalara erişme ve ağ bağlantıları açabilmektedir. Normalde bu özellikler bir web sunucusu üzerinde çalışan herhangi birşey için düşük güvenlik anlamına gelir. Fakat PHP, Perl ve C ile yazılmış CGI programlarından çok daha güvenli olacak şekilde tasarlanmıştır. Özellikle derleme/kurulum aşaması seçeneklerinin ve çalışma zamanı seçeneklerinin doğru şekilde ayarlanması, ihtiyaç olan özgürlük ve güvenliğin en barizini verecektir. Özelleştirme için verilmiş olan çok büyük bir seçenekler listesi sayesinde, bir çok amaç için rahatlıkla ve güvenle kullanılabilir[42].



### 3.1.3.3. PHP'nin Dezavantajları

#### *Hatalarla Baş Etmek:*

PHP hatalarla başa çıkmak konusunda ASP.NET veya ColdFusion kadar başarılı ve güçlü değildir. PHP'de yapılan bir hata, kolayca birden çok sayfada soruna neden olabilmektedir[42].

#### *Detaylı Programlama Zorluğu:*

PHP bir işaretleme dili olduğu için bir PHP programından cevap alma süresi uzayabilir. Kısacası, ileri seviye programlama işaretleme dili ile yapılabilecek bir işlem değildir[42].

### 3.1.3.4. Performans

- 8 saniye kuralı. Web'de ziyaretçiler genellikle yüklenmesi 8 saniyeyi geçen sayfalardan vazgeçerler. Buna karşın çoğu kez bant genişliği sınırlıdır. Toplam 35KB büyüklüğünde olan bir sayfa 56Kbps'lik bir modemle 7-8 saniyede çekilir. Bu süreye sunucu cevap süresi eklenecektir.
- Cevap süresi (response time). Normalde web sunucusu dinamik bir sayfada script işlemini bitene kadar cevap göndermez. Bu sürenin de 1-2 saniyeyi aşmaması zorunludur.
- CGI kullanan uygulamalar işletim sistemi tarafından görevlerin paralel olarak hafızaya yüklenmesi ve bunlar arasında geçişler gerektirdiği için CPU, RAM, zaman harcaması fazladır ve tutarsız davranırlar. Her seferinde yeniden veri tabanına bağlantı gerektirirler.
- Server API'si kullanan modüller ise CGI'ya göre 15-20 kat hızlı çalışabilirler. PHP'nin SAPI kullanan modül versiyonu da bu avantajdan yararlanır.
- Zend optimizör ile ara kod iyileştirmesi kullanılarak buna ek olarak %40 - %100 hızlanma elde edilebilir.
- Yerleşik (native) veritabanı desteği verdiği için ODBC kullanan sistemlere göre de hızlıdır.
- Sürekli veritabanı bağlantıları (persistent connections). Java servetlerindeki connection pooling mantığını programcıya saydam tutarak kullanır.
- PHP standart görevler için ASP'de olduğu gibi COM nesneleriyle iletişim kurmak zorunda kalmamaktadır. Bu tür işlevler sunucunun kendi hafıza alanı içinden çağrılır.
- Önbellekleme. Zend Cache ve benzeri yazılımlar ile scriptler derlenmiş olarak hafızada tutulabilir.
- Açık kaynak kodlu. Bir çok ülkeden gönüllü geliştiricilere sahiptir. Hatalar çok çabuk bulunup düzeltilmektedir. Pek çok web sunucusu ve işletim sistemine port edilmiş durumdadır.

- Dünya üzerinde PHP kullanan domain sayısı her geçen gün çok hızlı bir şekilde büyümektedir[43].

#### 3.1.4. ASP/PHP Karşılaştırılması

- PHP ve ASP arasında performans ve fiyat açısından açık farklar vardır. Fiyat konusu haricinde (Php ücretsiz dağıtılmaktadır.) ASP kullanıcılarının kabul etmedikleri şey, hız faktörüdür. Belki de en çok tartışılan bu konu hakkında en basit karşılaştırma; PHP + MySql ve ASP + MSSQL oranlarında çıkmaktadır. Elde edilen verilere göre 5000 satırlık bir bilgiyi database' e; aynı makina üzerinde Php ve MySql 3 ( Üç ) saniyede girerken, ASP + MSSQL 28 saniyede girmektedir.
- PHP'nin açık kaynak kodlu oluşu, hataların veya yeni özelliklerin güncellenmesi/eklenmesi gibi olaylarda büyük avantajlar sağlamaktadır.
- Kişisel kullanıcılar; Php ile web sayfalarında daha çok kullanıcı etkileşimli kodlar kullanmaktadır. Örnek olarak; Ziyaretçiler için oluşturulan, forum, sayaç, anket gibi kodlarda Php çok rağbet görmektedir.
- Günümüzde büyük şirketlerde internet ve intranet uygulamalarında, büyük paya sahip Microsoft firması, kendi ürünü olan ASP için büyük çabalar sarfetmektedir. Yeni teknolojiler (Özellikle .Net) ile ASP'ye hergün yeni zeminler hazırlamakta ve iş istihdamı konusunda ASP'yi bir adım öne taşımaktadır. Sunucu fiyatları olarak PHP'ye göre yerine göre üç kat daha fazla oranda olmasına rağmen, büyük şirketler tarafından tercih edilmektedir[14].

#### 3.1.5. CGI (Common Gateway Interface)

##### 3.1.5.1. CGI'da Güvenlik

Siteyi geliştirirken, veritabanlarından bilgi alma, onlara bilgi yazma gibi gereksinimler doğabilir. Bunun için de CGI uygulamaları yazılabilir. Web sunucusuna bağlanmaya çalışıldığında makineye birilerinin girdiği veya "kötü birşeyler" yaptığı fark edilebilir.

Bu bir senaryodan çok sıkça karşılaşılan bir durumdur. Bir bilgisayarın güvenlik riskleri fişi takıldığı andan itibaren başladığı halde, güvenlik hala ihmal edilmektedir. Bunun birkaç sebebini saymak gerekirse:

- Yetersiz bilgi
- Başkaları tarafından bilgisayara erişimin mümkün olamayacağı yanılgısı
- Hiç kimsenin sunucuyla uğraşmaya tenezzül etmeyeceğini düşünme
- Programları yazarken yeteri kadar sınamama
- Tembellik[4].

#### *Güvenlik Deliklerinin Başlıca Sebepleri:*

Bütün bilgisayarların ve programların insan elinden çıktığı düşünülürse, onlara zorla girebilecek başka kişilerin varolması da çok normaldir. Web sunucuları gibi çoklu kullanıcı ve dış dünyaya açık olması zorunluluk durumlarında, bu risk katlanmaktadır. Genel olarak bir Web sunucusunun güvenlik açıklarının sebepleri şunlardan bir kısmı veya hepsidir:

- Karşı uç
- Kullanıcı verileri
- Web sunucusu
- Web tarayıcısı

CGI, ya da genel olarak sunucu tarafı çalışan programlar ve onların çağırdığı programlar;

*Karşı uç:* Hattı dinleyen, sistemdeki açıkları taratan bir saldırgan ya da virüs gibi sisteme istenmeden yerleşen bir program olabilir.

*Kullanıcı verileri:* İstemciden alınan veriler; bu verileri her ne şekilde işlenirse işlensin, kontrolden geçirmek faydalı olacaktır.

*Web sunucusu:* Kullanılan Web sunucusunun çeşitli açıkları biliniyor olabilir. Güvenlik sitelerinde araştırma yaparak, kullanılan sunucu sürümüne özel, ya da eski sürümlerde olup kullanılan sürümde hala kapatılmamış açıkların bir listesini çıkarmak faydalı olacaktır[4].

*Web tarayıcısı:* 2. maddeyle birleşik düşünülebilir. Güvenliğin, kullanıcıdan gelen verileri doğrulamak için istemci tarafı tekniklere (Ör: JavaScript'le form doğrulama) dayandırılmamalıdır.

CGI, sunucu tarafı programlar: Web sunucusu, CGI programlarıyla saldırıya daha açık hale gelebilir, ancak bu her zaman daha güvensiz olacaktır şeklinde de yorumlanmamalıdır.

## 4. UYGULAMALAR

### 4.1. Uygulama-1 (0-10.000.000 Arası Sayan Döngü)

#### 4.1.1. Amaç

Bu uygulamada amaç, PHP, ASP ve CGI kodlarının işletilme hızlarını karşılaştırmaktır.

#### 4.1.2. Yapılış

Bu uygulamada her üç programda aynı işi yapacak, yani 0-10.000.000 arası sayacak ve döngü 0'da iken sistem saati alınıp ekrana yazdırılacak, sonra da döngü 10.000.000 da iken sistem saati alınıp ekrana yazdırılacaktır. Her üç program döngü işleme hızları bakımından karşılaştırılırken, eşit çalışma şartları sağlanmaya çalışılmıştır. Böylece döngünün başında ve sonunda alınan sistem saatleri arasındaki fark o programın 0-10.000.000 döngüsünü ne kadar zamanda işleyerek istemciye cevap verdiğini bulmaya ve buna göre programları karşılaştırmaya imkan verecektir. Ayrıca uygulamalar çalışırken Windows görev yöneticisinin fotoğrafı çekilmiş, bu fotoğraf Photoshop programı kullanılarak CPU'nun kullanılma grafiğinin mm cinsinden değeri bulunarak karşılaştırma için somut veriler elde edilmiştir.

Uygulamada test1.asp, test1.php ve test1.cgi adında 3 tane dosya hazırlanmış bu dosyalardan her biri dörder kez çalıştırılarak elde edilen sonuçların aritmetik ortalaması alınmıştır. Aşağıda programların içerikleri, nasıl çalıştırıldıkları ve sonuçları anlatılmaktadır.

#### 4.1.3. Test1.asp

```
<HTML>
<HEAD>
<TITLE>TEST1 ASP</TITLE>
</HEAD>
<BODY bgcolor="#C0C0C0">
<% for sayac=0 to 10000000
if sayac=0 then
s1=sayac
saat1=time()
end if
if sayac=10000000 then
s2=sayac
saat2=time()
end if
```

next

%>

<p>&nbsp;</p>

<p>&nbsp;</p>

<p>&nbsp;</p>

<p>&nbsp;</p>

<p>&nbsp;</p>

<div align="center">

<center>

<table border="5" cellpadding="1" width="460" height="37" id="AutoNumber1">

<tr>

<td width="460" height="31" colspan="2" bgcolor="#FFFFFF">

<p align="center"><font color="#FF0000" size="4">0 dan 10.000.000 a kadar olan döngü sonuçları</font></td>

</tr>

<tr>

<td width="234" height="16" bgcolor="#FFFFFF">

<p align="center"><b><font size="3" color="#0000FF">Döngü değeri</font></b></td>

<td width="220" height="16" bgcolor="#FFFFFF">

<p align="center"><b><font size="3" color="#0000FF">Sistem saati</font></b></td>

</tr>

<tr>

<td width="234" height="22" bgcolor="#FFFFFF"><font size="3"><p align="center"><b><%response.write(s1)%></b></font></td>

<td width="220" height="22" bgcolor="#FFFFFF"><font size="3"><p align="center"><b><%response.write(saat1)%></b></font></td>

</tr>

<tr>

<td width="234" height="21" bgcolor="#FFFFFF"><font size="3"><p align="center"><b><%response.write(s2)%></b></td>

<td width="220" height="20" bgcolor="#FFFFFF"><font size="3"><p align="center"><b><%response.write(saat2)%></b></td>

</tr>

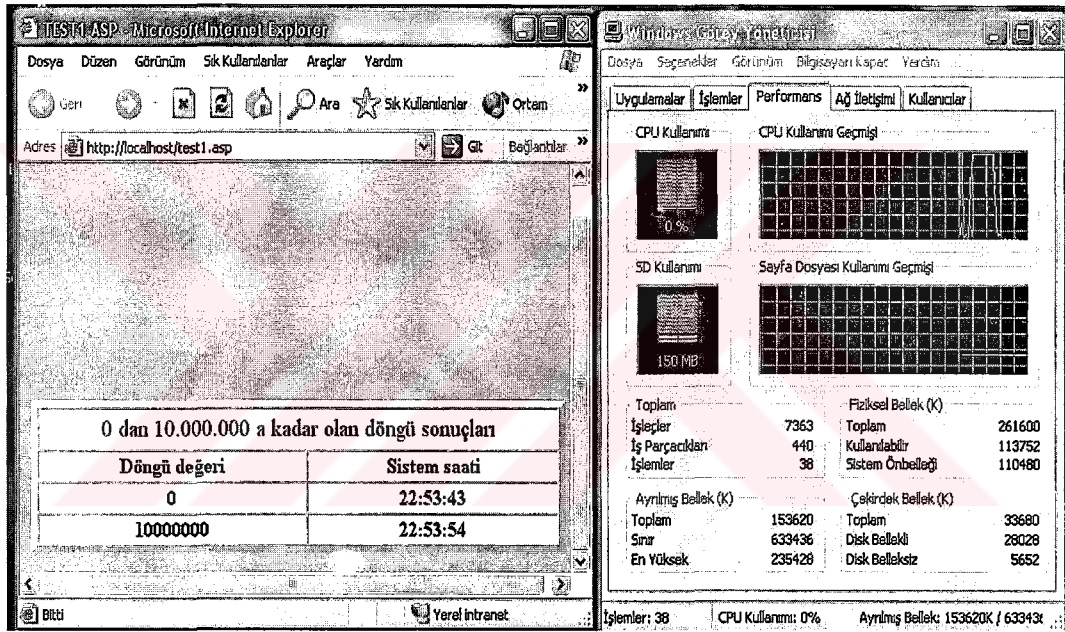
```

</table>
</center>
</div>
</BODY>
</HTML>

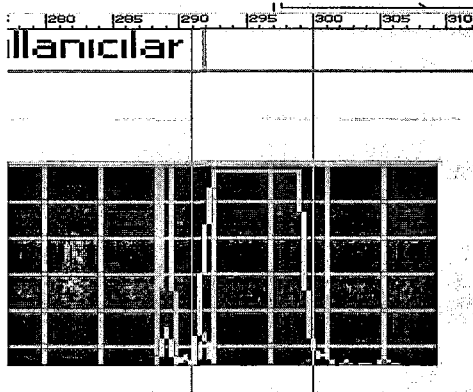
```

#### 4.1.3.1. Çalıştırılması

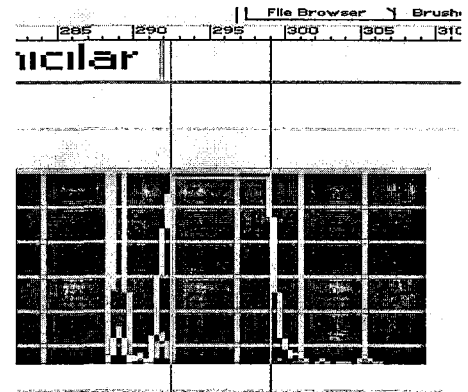
İçeriği yukarıdaki gibi oluşturulan test1.asp dosyası, kurulumu önceki bölümlerde anlatılan IIS kullanılarak çalıştırılmıştır. Şöyle ki test1.asp dosyası c:\inetpub\www\ dizinine kopyalanarak internet explorer açılıp adres satırına http://localhost/test1.asp yazılıp enter tuşuna basılarak çalıştırılmıştır. Dosyanın çalışması bittikten sonraki görüntüsü aşağıda verilmiştir.



Şekil 4.1. Test1.asp Sayfa Görüntüsü ve Windows Görev Yöneticisi



Şekil 4.2. CPU Kullanımı



Şekil 4.3. CPU Kullanımı(%100)



#### 4.1.3.2. Sonuç

Şekilde de görüldüğü gibi döngü değeri 0 da iken sistem saati 22:53:43, döngü değeri 10.000.000 da iken ise 22:53:54 olarak alınmıştır. Bu verilere dayanarak komut satırı yukarıdaki şekilde olan bir asp dosyası, yine özellikleri yukarıda verilen bir makine de çalıştırıldığında 0-10.000.000 arasındaki bir döngüyü 11 sn'de işletmektedir şeklinde bir sonuca varılır.

Burada dikkat edilmesi gereken diğer bir kısım ise dosyanın işlemciyi nasıl kullandığıdır. Şekil 4.2. incelendiğinde işlemcinin başlama anı ile bitiş anı arasında 9 mm olduğu ve Şekil 4.3.'de ise CPU'nun %100'lük kullanımının 6.5 mm olduğu görülmektedir.

#### 4.1.4. Test1.php

```
<html>
<head>
  <title>TEST1 PHP</title>
</head>
<BODY bgcolor="#C0C0C0">
  <?
  for($i=0;$i<=10000000;$i++) {
  if ($i==0){
    $s1=$i;
    $saat1 = date("H:i:s");
  }
  if ($i==10000000){
    $s2=$i;
    $saat2 = date("H:i:s");
  }
  }
  ?>

  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <div align="center">
```

```

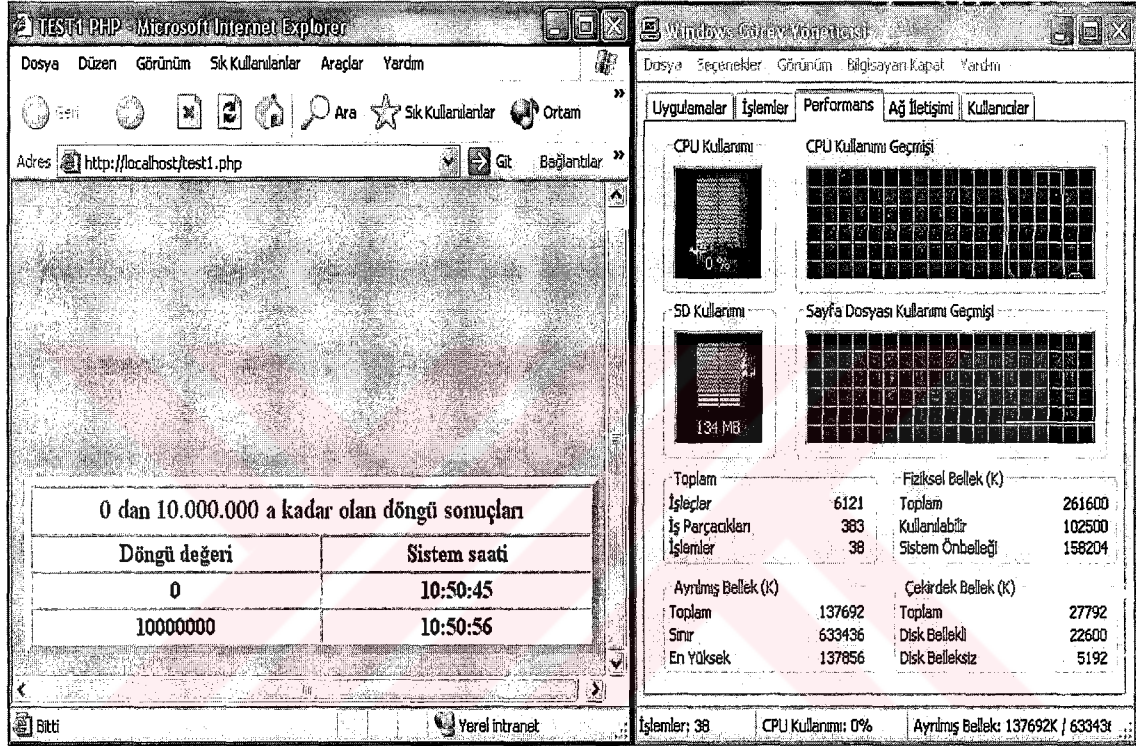
<center>
<table border="5" cellspacing="1" width="460" height="37" id="AutoNumber1">
  <tr>
    <td width="460" height="31" colspan="2" bgcolor="#FFFFFF">
      <p align="center"><font color="#FF0000" size="4">0 dan 10.000.000 a kadar olan döngü
sonuçları</font></td>
    </tr>
    <tr>
      <td width="234" height="16" bgcolor="#FFFFFF">
        <p align="center"><b><font size="3" color="#0000FF">Döngü değeri</font></b></td>
      <td width="220" height="16" bgcolor="#FFFFFF">
        <p align="center"><b><font size="3" color="#0000FF">Sistem saati</font></b></td>
      </tr>
      <tr>
        <td width="234" height="22" bgcolor="#FFFFFF"><font size="3"><p
align="center"><b><?echo $s1?> </b></font></td>
        <td width="220" height="22" bgcolor="#FFFFFF"><font size="3"><p
align="center"><b><?echo $saat1?></b></font></td>
      </tr>
      <tr>
        <td width="234" height="21" bgcolor="#FFFFFF"><font size="3"><p
align="center"><b><?echo $s2?></b></td>
        <td width="220" height="20" bgcolor="#FFFFFF"><font size="3"><p
align="center"><b><?echo $saat2?></b></td>
      </tr>
    </table>
  </center>
</div>

</body>
</html>

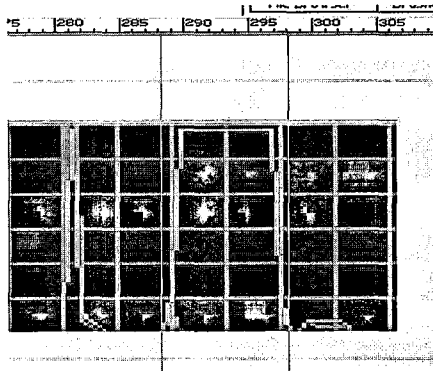
```

#### 4.1.4.1. Çalıştırılması

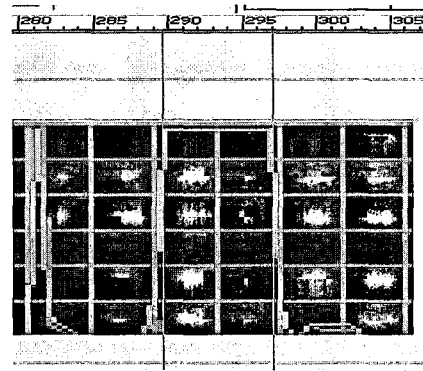
Sayfayı çalıştırmak için kurulumu önceki bölümlerde anlatılan “FoxServ” isimli program kullanılmıştır. Bu program kurulduktan sonra test1.php adlı sayfa c:\FoxServ\www\ dizinine kopyalanır. Sonra c:\FoxServ\apache\ dizini altındaki apache.exe dosyası çalıştırılır ve böylece apache server aktif duruma gelmiş olur. İnternet Explorer adres satırına http://localhost/test1.php yazılıp enter tuşuna basılarak program çalıştırılır. Programın çalışması bittikten sonraki görüntüsü aşağıda verilmiştir.



Şekil 4.4. Test1.php Sayfa Görüntüsü ve Windows Görev Yöneticisi



Şekil 4.5. CPU Kullanımı



Şekil 4.6. CPU Kullanımı(%100)

#### 4.1.4.2. Sonuç

Alınan değerlere göre döngü değeri 0 da iken sistem saati 10:50:45, döngü değeri 10.000.000 iken sistem saati 10:50:56'dır. Buradan şu sonuca ulaşabilir; yukarıda içeriği verilen test1.php sayfası yine yukarıda verilen makine özelliklerine göre 0-10.000.000 arasındaki döngüyü 11 sn'de işletmektedir. Şekil 4.5.'de CPU kullanımının başlangıç ve bitiş değerleri arasında 10 mm'lik fark olduğu ve Şekil 4.6.'da ise CPU'nun %100'lük kullanımının 7.5 mm olduğu görülmektedir.

#### **4.1.5. Test1.cgi**

```
#!/C:/FoxServ/perl/bin/perl.exe
print "Content-type: text/html", "\n\n";
for($sayac = 0 ; $sayac <= 10000000 ; $sayac ++)
{
    if($sayac == 0)
    {
        $s1=$sayac;
        ($saniye1,$dakika1,$saat1) = (localtime(time))[0,1,2];

    }
    if($sayac == 10000000)
    {
        $s2=$sayac;
        ($saniye2,$dakika2,$saat2) = (localtime(time))[0,1,2];
    }
}

print "<HTML><HEAD><TITLE>TEST1 CGI</TITLE></HEAD>";

print "<BODY bgcolor=#C0C0C0>";
print "<p>&nbsp;</p>";
print "<p>&nbsp;</p>";
print "<p>&nbsp;</p>";
print "<p>&nbsp;</p>";
print "<p>&nbsp;</p>";
```

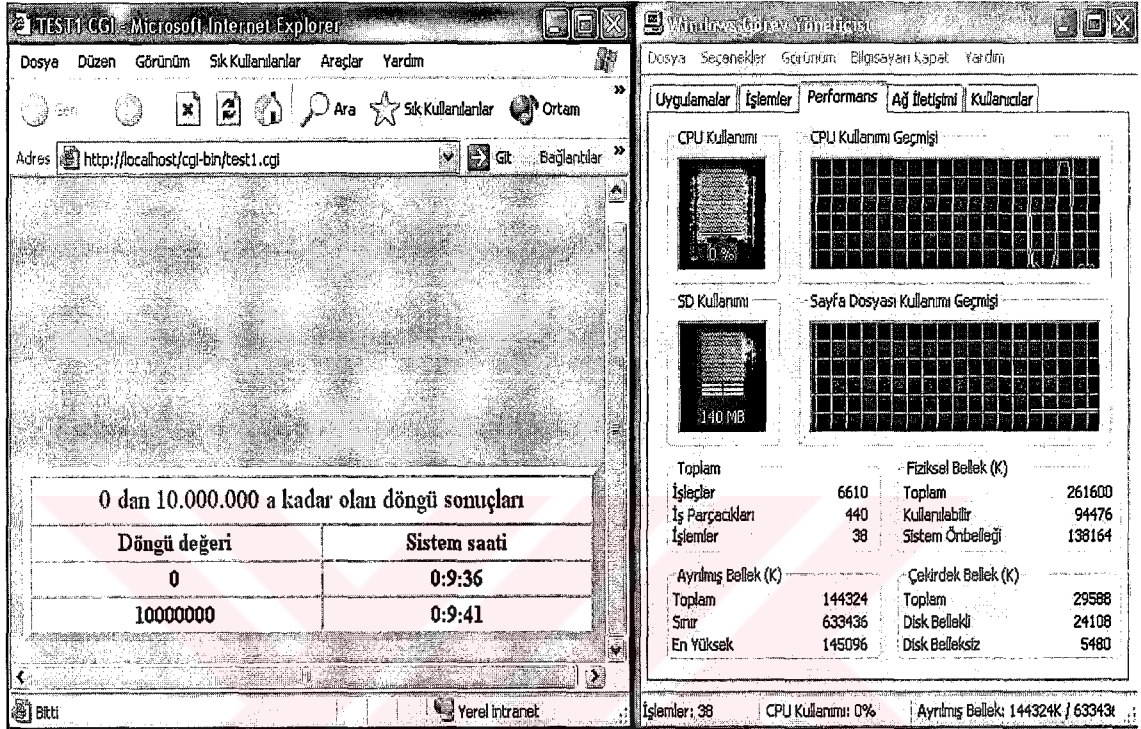
```

print "<div align=center>";
print "<center>";
print "<table border=5 cellpadding=1 width=460 height=37 id=AutoNumber1>";
print "<tr>";
print "<td width=460 height=31 colspan=2 bgcolor=#FFFFFF>";
print "<p align=center><font color=#FF0000 size=4 >0 dan 10.000.000 a kadar olan döngü sonuçları</font></td>";
print "</tr>";
print "<tr>";
print "<td width=234 height=16 bgcolor=#FFFFFF>";
print "<p align=center><b><font size=3 color=#0000FF>Döngü değeri</font></b></td>";
print "<td width=220 height=16 bgcolor=#FFFFFF>";
print "<p align=center><b><font size=3 color=#0000FF>Sistem saati</font></b></td>";
print "</tr>";
print "<tr>";
print "<td width=234 height=22 bgcolor=#FFFFFF><font size=3><p align=center> <b> $s1  
</b> </p> </font> </td>";
print "<td width=220 height=22 bgcolor=#FFFFFF><font size=3><p align=center> <b>  
$saat1:$dakika1:$saniye1</b></p></font></td>";
print "</tr>";
print "<tr>";
print "<td width=234 height=21 bgcolor=#FFFFFF><font size=3><p align=center><b>$s2  
</b> </p></font></td>";
print "<td width=220 height=20 bgcolor=#FFFFFF><font size=3><p align=center><b>  
$saat2:$dakika2:$saniye2</b></p></font></td>";
print "</tr>";
print "</table>";
print "</center>";
print "</div>";
print "</BODY>";
print "</HTML>";

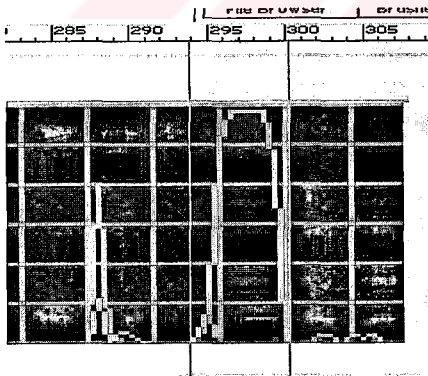
```

#### 4.1.5.1. Çalıştırılması

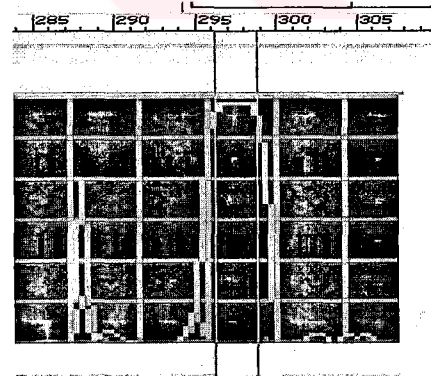
Sayfayı çalıştırmak için test1.cgi dosyası, c:\FoxServ\www\cgi-bin\ dizini altına kopyalandıktan sonra internet explorer adres satırına <http://localhost/cgi-bin/test1.cgi> yazılır ve enter tuşuna basılır. Sayfa çalıştırıldıktan sonra alınan sonuç aşağıda verilmiştir.



Şekil 4.7. Test1.cgi Sayfa Görüntüsü ve Windows Görev Yöneticisi



Şekil 4.8. CPU Kullanımı



Şekil 4.9. CPU Kullanımı(%100)

#### 4.1.5.2. Sonuç

Elde edilen sonuçlara göre cgi sayfası, döngü değeri 0 da iken sistem saati 00:09:36, 10.000.000 da iken ise 00:09:41 olarak görülmektedir. Böylece 0-10.000.000 arasındaki olan döngü yaklaşık 5 sn'de işletilmiştir. Şekil 4.8.'de CPU kullanımının başlangıç ve bitiş değerleri



arasında 6 mm'lik fark olduğu ve Şekil 4.9.'da ise CPU'nun %100'lük kullanımının 3 mm olduğu görülmektedir.

#### 4.1.6. Genel Uygulama Sonucu

Yukarıda üç sayfanın test sonuçları Tablo 4.1.'de görülmektedir.

Tablo 4.1. ASP, PHP, CGI Sayfalarının Test Sonuçları(Döngü)

|                  | <i>İşletilme süresi</i>   | <i>CPU Kullanım Durumu</i> | <i>CPU'nun %100'lük Kullanım Durumu</i> |
|------------------|---------------------------|----------------------------|-----------------------------------------|
| <i>Test1.asp</i> | 11 saniye                 | 9 mm                       | 6.5 mm                                  |
| <i>Test1.php</i> | 11 saniye                 | 10 mm                      | 7.5 mm                                  |
| <i>Test1.cgi</i> | 5 saniye                  | 6 mm                       | 3 mm                                    |
| <b>SONUÇ</b>     | <b>CGI &gt; ASP = PHP</b> |                            |                                         |

Bu verilerden çıkarılan sonuca göre ASP ve PHP sayfaları birbirine yakın iken, CGI sayfası daha hızlı görülmektedir. Farklı donanıma sahip makinelerde bu uygulama gerçekleştirilirse farklı değerler elde edilebilir. Ancak aradaki hız ilişkisi yine bu sonuçlara yakın olacaktır. Yani cgi döngü bazında, php ve asp'den hızlı çıkacaktır.

## 4.2. Uygulama-2 (20.000 Adet JPG Resmi Yükleme)

### 4.2.1. Amaç

Bu uygulamada PHP, ASP ve CGI dilleriyle hazırlanan sayfalara resimlerin yüklenmesi hususunda hızlarını karşılaştırmak amaçlanmıştır.

### 4.2.2. Yapılışı

Uygulama için test2.asp, test2.php ve test2.cgi adında 3 sayfa oluşturulmuştur. Bu sayfaların içerikleri, çalıştıklarında 20.000 adet resmi yükleyecek şekilde tasarlanmıştır. Ayrıca TEST.swf adındaki küçük flash programıyla asp, php yada cgi sayfalarının sunucudan istenme zamanları da elde edilmektedir. TEST.swf adlı program bünyesindeki TEST butonuna tıklandığında tarayıcı (internet explorer) çalıştırılarak sayfanın sunucu tarafından yorumlanıp ekrana gelmesi sağlanmaktadır. İlk olarak TEST butonuna tıklandığında sistem saati alınarak ekrana yazdırılmakta, ikinci olarak da sayfa görüntülediği anda sistem saati sayfaya yazdırılmaktadır. Böylece sayfa talep edildiği andaki sistem saati ile sayfanın açıldığı andaki sistem saati arasındaki zaman farkı bize sayfanın yüklenme süresini vermiş olacaktır. Ayrıca uygulamalar çalışırken Windows görev yöneticisinin fotoğrafı çekilmiş bu fotoğraf Photoshop

programı kullanılarak CPU'nun kullanılma grafiğinin **mm** (milimetre) cinsinden değeri bulunarak karşılaştırma için somut veriler elde edilmeye çalışılmıştır.

Aşağıda önceden bu uygulama için oluşturulmuş ASP, PHP ve CGI sayfalarının içerikleri ve çalıştırmalarıyla ilgili bilgiler verilmektedir.

#### 4.2.3. Test2.asp

```
<html>
<head>
<title>TEST2 ASP</title>
</head>
<body bgcolor=c0c0c0>
  <center>
    <table border="5" width="866" height="46" cellspacing="1" >
      <tr>
        <td width="866" height="24">
          <p align="center"><b><font color="#FF0000" size="5">20.000 adet JPG resminin
yüklenme süresi</font></b></p></td>
        </tr>
        <tr>
          <td width="866" height="18">
            <p align="center"><b><font color="#0000FF" size="4">Sistem saati=&nbsp; <%
response.write (time)%>
            </font></b></td>
          </tr>
        </table></center>
        <div align="center">
          <center>
            <table border="5" width="854" height="358" cellspacing="1" >
              <tr>
                <td width="889" height="19" colspan="3">&nbsp;</td>
              </tr>
              <tr>
                <td width="280" height="133">
```

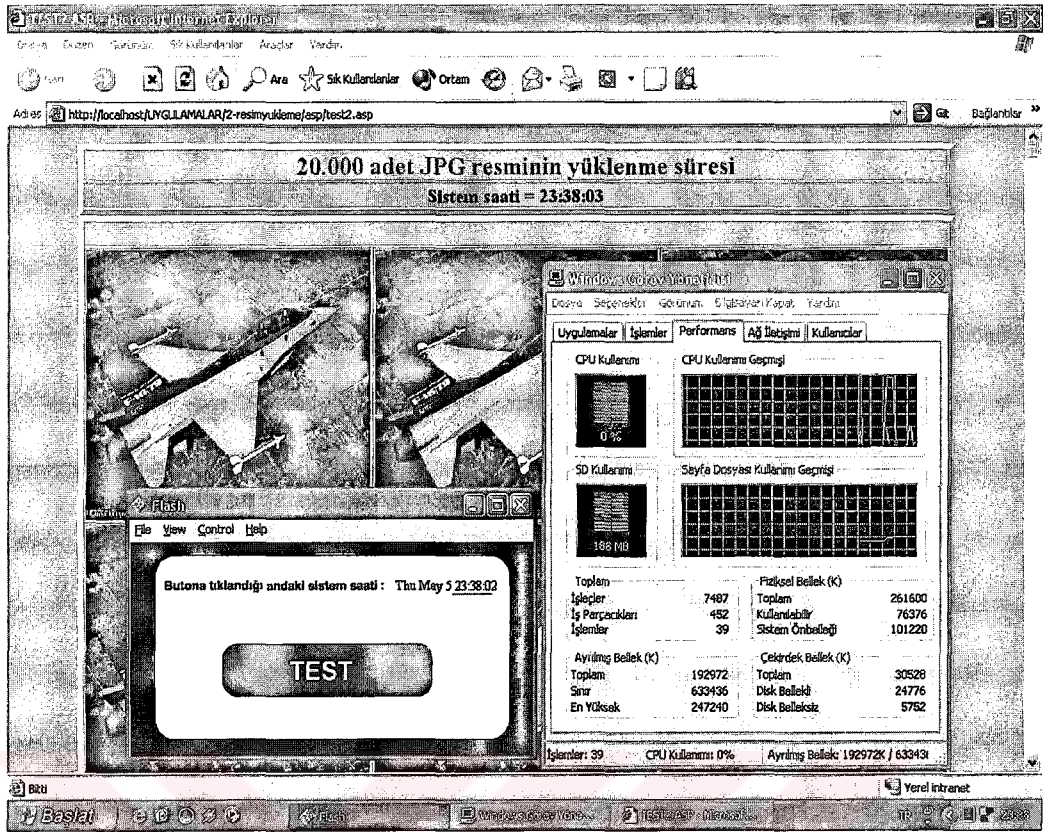
```

</td>
<td width="280" height="133">
</td>
<td width="277" height="133">
</td>
</tr>
.....
.....
.....
.....
.....
.....
<tr>
<td width="280" height="201">
</td>
<td width="280" height="201">
</td>
<td width="277" height="201">
</td>
</tr>
</table>
</center>
</div>
</body>
</html>

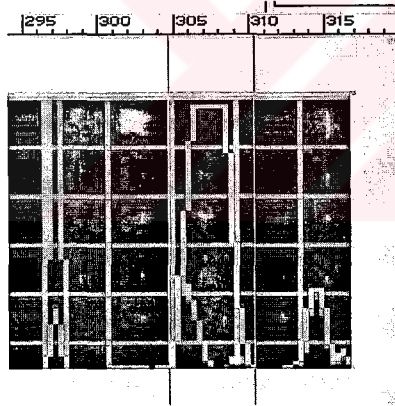
```

#### 4.2.3.1.Çalıştırılması

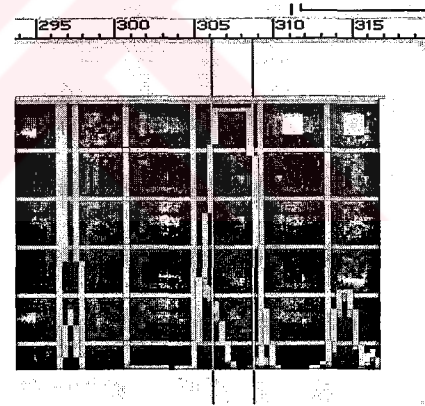
UYGULAMALAR isimli klasör, c:\inetpub\wwwroot\ dizini altına kopyalandıktan sonra C:\inetpub\wwwroot\UYGULAMALAR\2-resimyukleme\asp\ dizini altındaki Test.swf programı çift tıklanarak çalıştırılır. Windows görev yöneticisi çalıştırılır ve sayfanın çalışması için gerekli olan programlar dışındaki tüm programlar sonlandırılır. Görev yöneticisinin performans kısmı aktif yapılır böylece sayfa yüklenirken CPU'nun kullanım grafiği elde edilmiş olacaktır. Test.swf programındaki TEST butonuna tıklanarak sayfa çalıştırılır. Çalıştırılma sonucu aşağıda görülmektedir.



Şekil 4.10. Test2.asp Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi



Şekil 4.11. CPU Kullanımı



Şekil 4.12. CPU Kullanımı(%100)

#### 4.2.3.2. Sonuç

Şekilde de görüldüğü gibi TEST butonuna tıklandığı anda sistem saati 23:38:02, sayfanın kodları yorumlanıp ekranda görüntülediği andaki sistem saati ise 23:38:03 olarak tespit edilmiştir. Yine resimde sayfanın tam olarak yani resimlerin tümünün yüklendiği ana kadar geçen zaman dilimi içindeki CPU kullanımı görülmektedir. Buradaki sistem saatleri arasındaki zaman farkı 1 saniye olarak görülmektedir. Ancak bu zaman dilimi sayfanın tam olarak yüklendiği ana kadar geçen zamanı vermemektedir. Yani sayfa yorumlanıp

görüntülediği anda sistem saati ekrana yazılmış, ancak; 20.000 adet resmin tamamı daha sonra yüklenmiştir. Peki sayfanın 20.000 resimle olan yüklenme süresi nasıl bulunur gibi bir soru akla gelebilir. Bu sorunun çözümü için de Görev yöneticisi kullanılmıştır. Yani tam olarak yüklenme süresini saniye cinsinden elde edilemese de sayfaların talep edildiği andan, tam olarak (20.000 resmin tamamı yüklenerek) yüklendiği ana kadar geçen zaman içinde CPU kullanımının grafiğinden yararlanarak ASP, PHP ve CGI sayfalarının ayrı ayrı CPU grafikleri karşılaştırılarak hangisinin daha hızlı ya da daha yavaş yüklendiği konusunda bir fikir edinilebilir. Şekil 4.11.'de CPU kullanımının başlangıç ve bitiş değerleri arasında 5.5 mm lik fark olduğu ve Şekil 4.12.'de ise CPU'nun %100 lük kullanımının 2.5 mm olduğu görülmektedir.

#### 4.2.4. Test2.php

```
<html>
<head>
<title>TEST2 PHP</title>
</head>
<?
$saat = date("H:i:s");
?>
<body bgcolor=c0c0c0>
  <center>
    <table border="5" width="866" height="46" cellpadding="1" >
      <tr>
        <td width="866" height="24">
          <p align="center"><b><font color="#FF0000" size="5">20.000 adet JPG resminin
yüklenme süresi</font></b></td>
        </tr>
        <tr>
          <td width="866" height="18">
            <p align="center"><b><font color="#0000FF" size="4">Sistem saati =<?echo $saat?>
</font></b></td>
          </tr>
        </table></center>
      <div align="center">
```

```

<center>
<table border="5" width="854" height="358" cellspacing="1" >
  <tr>
    <td width="889" height="19" colspan="3">&nbsp;</td>
  </tr>
  <tr>
    <td width="280" height="133">
      </td>
    <td width="280" height="133">
      </td>
    <td width="277" height="133">
      </td>
    </tr>
    .....
    .....
    .....
    .....
  <tr>
    <td width="280" height="201">
      </td>
    <td width="280" height="201">
      </td>
    <td width="277" height="201">
      </td>
    </tr>

</table>
</center>
</div>
</body>
</html>

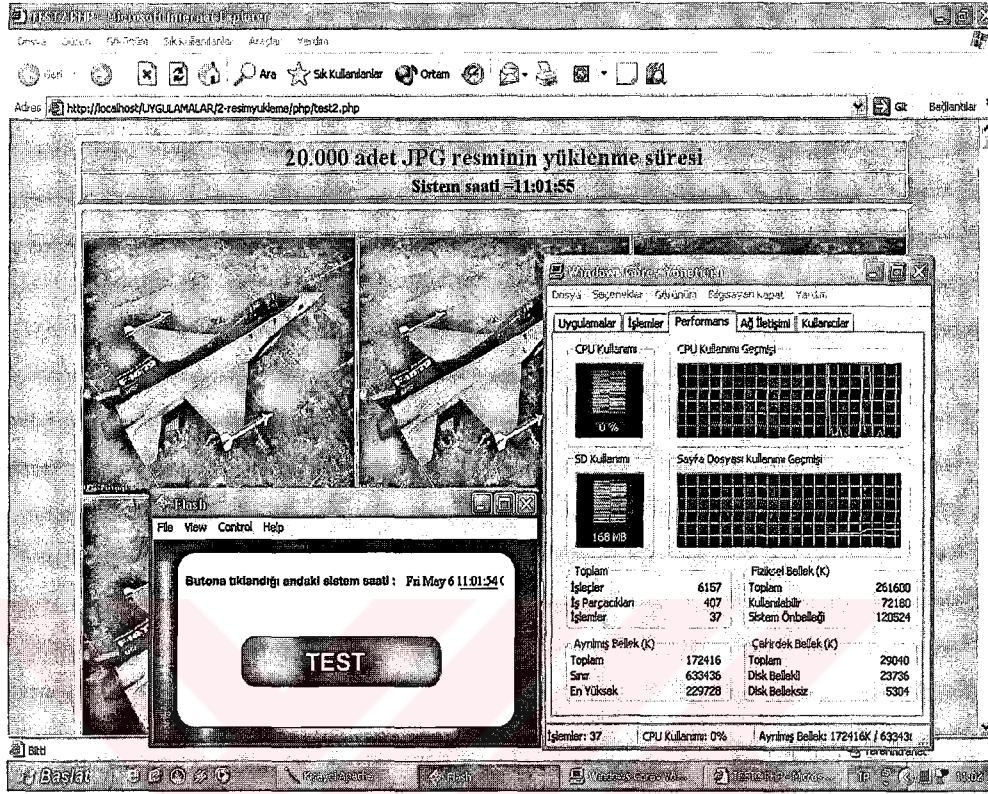
```

#### 4.2.4.1. Çalıştırılması

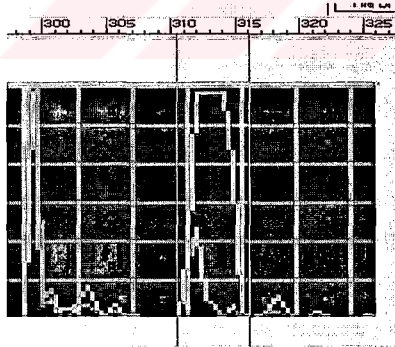
UYGULAMALAR isimli klasör “C:\FoxServ\www\” dizinine kopyalanır ve “C:\FoxServ\www\UYGULAMALAR\2-resimyukleme\php\” dizini altındaki TEST.swf



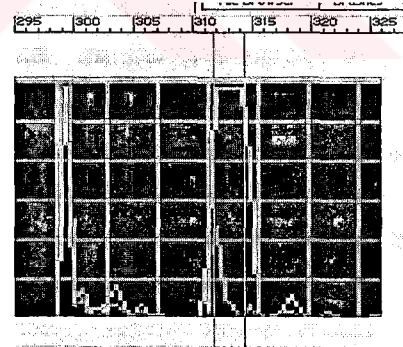
programı çalıştırılır. “C:\FoxServ\apache\” klasörü altındaki “apache.exe” dosyası çalıştırılarak böylece sayfadaki php kodlarını yorumlayıp HTML sayfası şeklinde sunacak olan sunucu (server) aktif edilmiş olur. Sayfanın çalıştıktan sonraki ekran görüntüsü aşağıda verilmiştir.



Şekil 4.13. Test2.php Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi



Şekil 4.14. CPU Kullanımı



Şekil 4.15. CPU Kullanımı(%100)

#### 4.2.4.2. Sonuç

Şekil incelendiğinde butona tıklanma anı ile sayfanın görüntülendiği an arasındaki zaman farkı 1 saniye olarak görülmektedir. Daha önceden belirtildiği gibi bu süre 20.000 adet resmin tümünün yüklenme zamanı değildir. Çünkü sayfa çalıştırıldıktan ve sayfadaki sistem saati yazıldıktan sonra da resimler yüklenmekte ve CPU’da %100 oranında kullanılmaktadır.

Tabi ki biz hız karşılaştırmasını yaparken sadece bu zaman farkına göre değil aynı zamanda elde edilen CPU kullanım grafiğini de kullanılacaktır. Çünkü bu grafik daha doğru sonuçlar verecektir. Şekil 4.14.'de CPU kullanımının başlangıç ve bitiş değerleri arasında 5.5 mm'lik fark olduğu görülmektedir. Şekil 4.15.'de ise CPU'nun %100'lük kullanımının 2 mm olduğu görülmektedir.

#### 4.2.5. Test2.cgi

CGI sayfası, PERL programlama dili kullanılarak yazılmıştır. Bu bütün uygulamalarda böyledir. PERL dili kullanıldığı için sayfa içindeki perl komutlarının yorumlanıp tekrar HTML olarak gönderilmesi için aşağıdaki ilk program satırı olan “#!C:\FoxServ\perl\bin\perl.exe” kullanılmıştır. Buradaki “perl.exe” dosyası ile perl komutları yorumlanıp işletilmektedir.

```
#!C:\FoxServ\perl\bin\perl.exe
print "Content-type: text/html", "\n\n";

($saniye,$dakika,$saat) = (localtime(time))[0,1,2];
$yol="C:/FoxServ/www/cgi-bin/UYGULAMALAR/2-resimyukleme/cgi/F16.jpg";
print "<HTML>", "\n";
print "<BODY bgcolor=c0c0c0>", "\n";
print "<center>";
print "<table border=5 width=866 height=46 cellpadding=1 >";
    print"<tr>";
    print"<td width=866 height=24>";
        print"<p align=center><b><font color=#FF0000 size=5>20.000 adet JPG resminin
yüklenme süresi</font></b></td>";
    print"</tr>";
    print"<tr>";
        print"<td width=866 height=18>";
            print"<p align=center><b><font color=#0000FF size=4>Sistem saati =
$saat:$dakika:$saniye";
                print"</font></b></td>";
            print"</tr>";
print"</table></center>";
print"<div align=center>";
print"<center>";
print"<table border=5 width=854 height=358 cellpadding=1 >";
```

```

print"<tr>";
    print"<td width=889 height=19 colspan=3>&nbsp;  </td>";
print"</tr>";
print"<tr>";
    print"<td width=280 height=133>";
    print"<img border=0 src=$yol width=280 height=250></td>";
    print"<td width=280 height=133>";
    print"<img border=0 src=$yol width=280 height=250></td>";
    print"<td width=277 height=133>";
    print"<img border=0 src=$yol width=280 height=250></td>";
print"</tr>";

.....
.....
.....
.....
print"<tr>";
    print"<td width=280 height=133>";
    print"<img border=0 src=$yol width=280 height=250></td>";
    print"<td width=280 height=133>";
    print"<img border=0 src=$yol width=280 height=250></td>";
    print"<td width=277 height=133>";
    print"<img border=0 src=$yol width=280 height=250></td>";
print"</tr>";

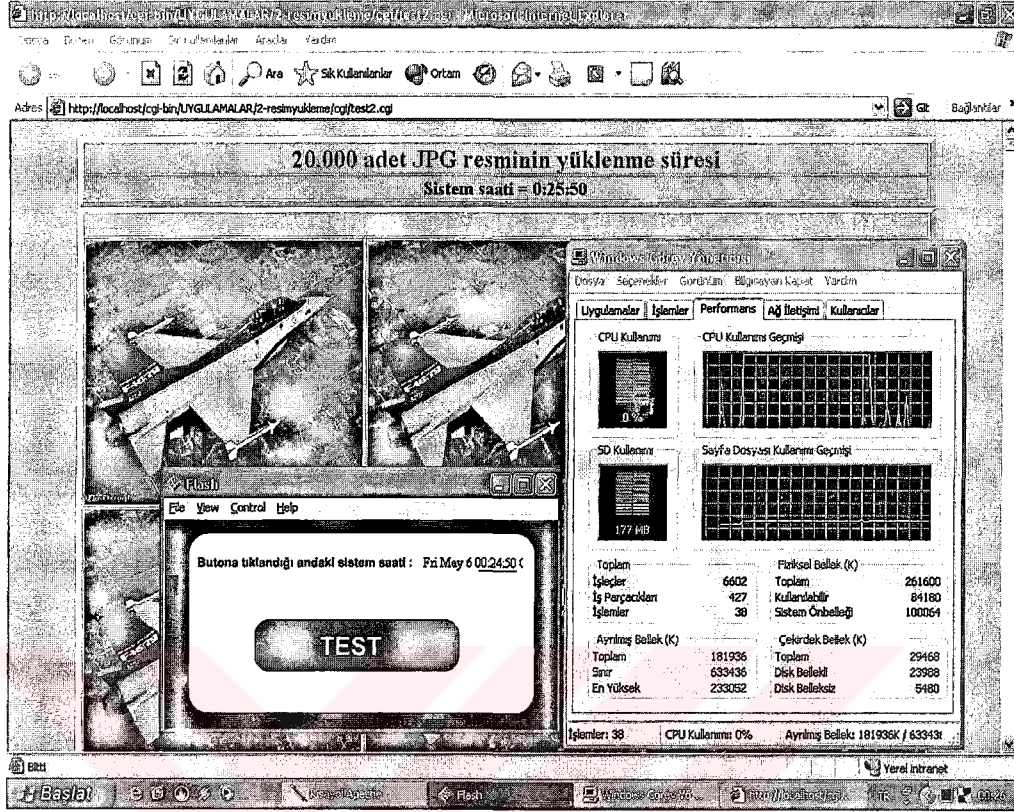
print"</table>";
print"</center>";
print"</div>";
print "</BODY></HTML>", "\n";

```

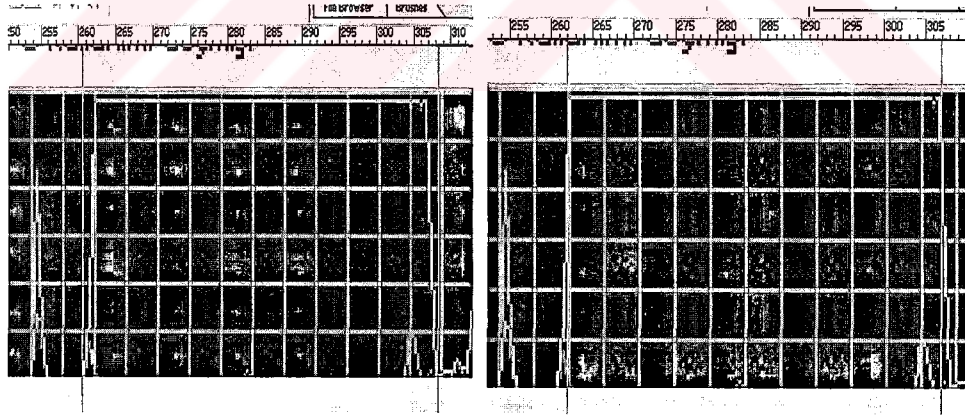
#### 4.2.5.1. Çalıştırılması

UYGULAMALAR isimli klasör “C:\FoxServ\www\cgi-bin\” dizinine kopyalanır ve “C:\FoxServ\www\cgi-bin\UYGULAMALAR\2-resimyukleme\cgi\” dizini altındaki TEST.swf programı çalıştırılır. “C:\FoxServ\apache\” klasörü altındaki “apache.exe” dosyası çalıştırılır ve

böylece sayfadaki cgi kodlarını yorumlayıp HTML sayfası şeklinde sunacak olan sunucu (server) aktif edilir. Sayfanın çalıştıktan sonraki ekran görüntüsü aşağıda verilmiştir.



Şekil 4.16. Test2.cgi Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi



Şekil 4.17. CPU Kullanımı

Şekil 4.18. CPU Kullanımı(%100)

#### 4.2.5.2. Sonuç

Sayfanın talep edildiği an ile görüntülendiği an arasındaki zaman farkı **1 dakika (60 sn.)** olarak saptanmıştır. Ayrıca talep anından, 20.000 adet resmin tam olarak yüklendiği ana kadar geçen zaman diliminde ki CPU kullanım grafiği de ekranda görülmektedir. Şekil 4.17.'de



CPU kullanımının başlangıç ve bitiş değerleri arasında **48 mm**'lik fark olduğu ve Şekil 4.18.'de ise CPU'nun %100'lük kullanımının **45 mm** olduğu görülmektedir.

#### 4.2.6. Genel Uygulama Sonucu

Yukarıda üç sayfanında test sonuçları Tablo 4.2.'de görülmektedir. Buna göre 20.000 adet jpeg resmin yüklenmesini ASP ve PHP hemen hemen aynı sürelerde yaparlarken, CGI bu işlem için çok uzun süre harcamaktadır.

**Tablo 4.2.** ASP, PHP, CGI Sayfalarının Test Sonuçları(JPG Resmi Yükleme)

|                  | <i>İşletilme süresi</i>   | <i>CPU Kullanım Durumu</i> | <i>CPU'nun %100'lük Kullanım Durumu</i> |
|------------------|---------------------------|----------------------------|-----------------------------------------|
| <i>Test2.asp</i> | 1 saniye                  | 5.5 mm                     | 2.5 mm                                  |
| <i>Test2.php</i> | 1 saniye                  | 5.5 mm                     | 2 mm                                    |
| <i>Test2.cgi</i> | 60 saniye                 | 48 mm                      | 45 mm                                   |
| <b>SONUÇ</b>     | <b>ASP = PHP &gt; CGI</b> |                            |                                         |

### 4.3. Uygulama-3 (Avi yükleme)

#### 4.3.1. Amaç

Bu uygulamada PHP, ASP ve CGI dilleriyle hazırlanan sayfalara avi formatı gibi (hem görüntü hemde ses) çoklu ortam uygulamalarının yüklenme hızlarını karşılaştırmak amaçlanmıştır.

#### 4.3.2. Yapılışı

Uygulama için test3.asp, test3.php ve test3.cgi adında 3 sayfa oluşturulmuştur. Bu sayfaların içerikleri poligon.avi adlı dosyayı yükleyip oynatacak şekilde düzenlenmiştir. Çalıştırma işlemi için yine TEST.swf adlı program kullanılmıştır. TEST.swf programındaki TEST butonuna tıklama anı ile sayfanın yüklenme anı arasındaki zaman farkı alınarak sayfaların yüklenme süreleri bulunmaya çalışılmıştır.

#### 4.3.3. Test3.asp

```
<html>
<head>
<title>TEST3 ASP</title>
</head>
```

```

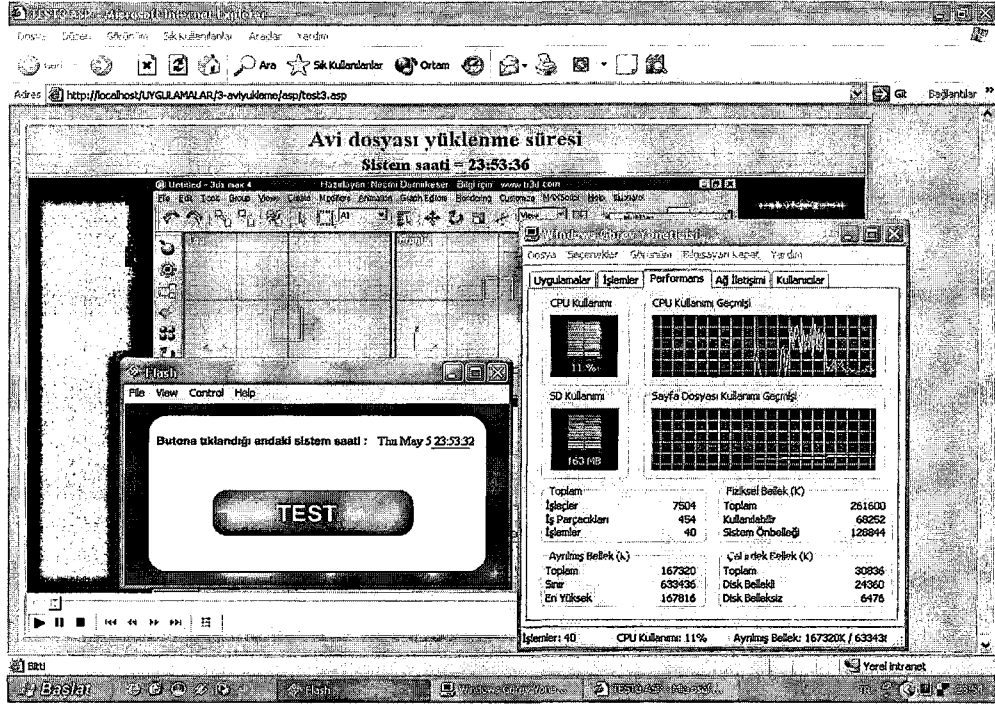
<body bgcolor=c0c0c0>
<table border="5" width="866" height="46" cellspacing="1" >
  <tr>
    <td width="866" height="24">
      <p align="center"><b><font color="#FF0000" size="5">Avi dosyası yüklenme
süresi</font></b></td>
    </tr>
    <tr>
      <td width="866" height="18">
        <p align="center"><b><font color="#0000FF" size="4">Sistem saati =&nbsp;
<%response.write(time)%>
        </font></b></td>
      </tr>
    <tr>
      <td width="866" height="500">
        <embed src="polygon.avi" width="866" height="500"></embed></td>
      </tr>
    </table></center>
</body></html>

```

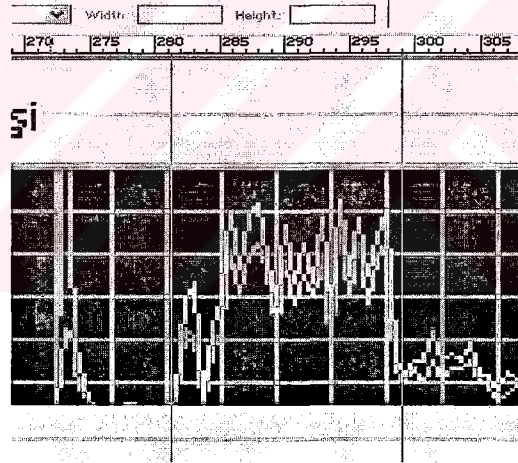
#### 4.3.3.1. Çalıştırılması

UYGULAMALAR isimli klasör c:\inetpub\wwwroot\ dizini altına kopyalandıktan sonra C:\inetpub\wwwroot\UYGULAMALAR\3-aviyukleme\asp\ dizini altındaki Test.swf programı çift tıklanarak çalıştırılır. Görev yöneticisinin performans kısmı aktif yapılır, böylece sayfa yüklenirken CPU'nun kullanım grafiği elde edilmiş olacaktır. Test.swf programındaki TEST butonuna tıklanarak sayfa çalıştırılır. Çalıştırılma sonucu aşağıda görülmektedir.





Şekil 4.19. Test3.asp Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi



Şekil 4.20. CPU Kullanımı

#### 4.3.3.2. Sonuç

Sayfamızın talep edildiği andan (TEST butonuna tıklandığı an) görüntülendiği ana kadar geçen süre **4 saniye** olarak görülmektedir. Talep anı ile sayfamızın avi dosyasını ses ve görüntü olarak oynattığı ana kadarki CPU kullanımı Şekil 4.20.'de görüldüğü gibi **18 mm**'dir.

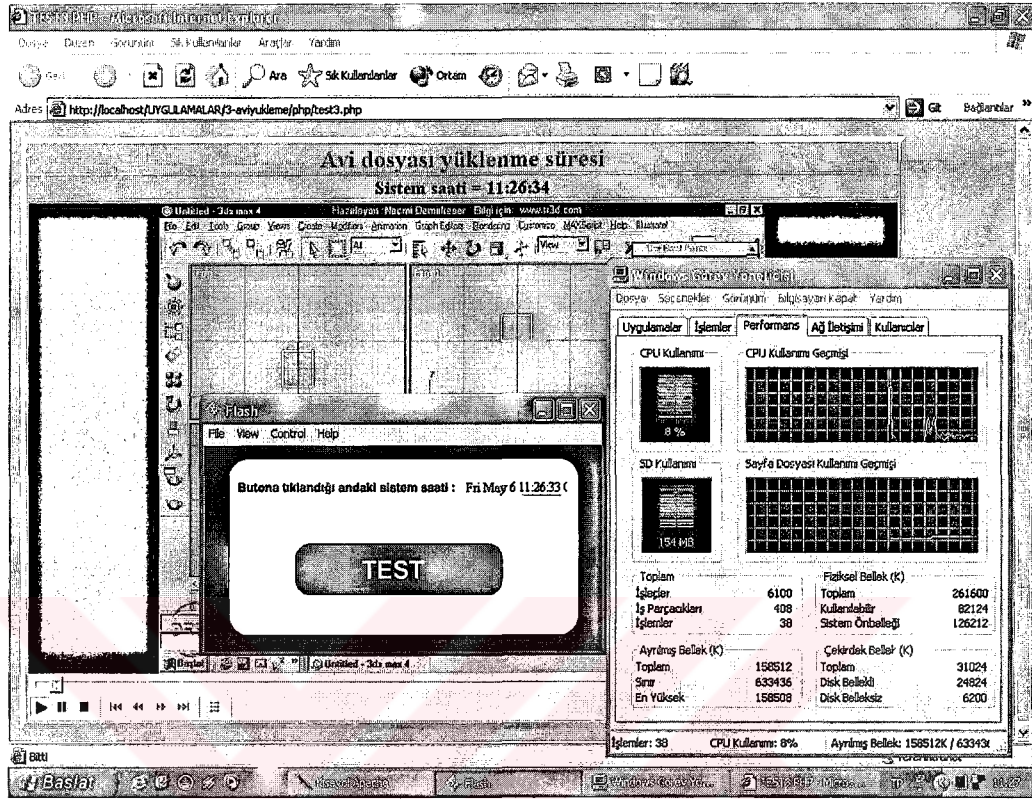
#### 4.3.4. Test3.php

```
<html>
<head>
<title>TEST3 PHP</title>
</head>
<body bgcolor=c0c0c0>
<?
$saat = date("H:i:s");
?>
<table border="5" width="866" height="46" cellspacing="1" >
  <tr>
    <td width="866" height="24">
      <p align="center"><b><font color="#FF0000" size="5">Avi dosyası yüklenme
süresi</font></b></td>
    </tr>
    <tr>
      <td width="866" height="18">
        <p align="center"><b><font color="#0000FF" size="4">Sistem saati =&nbsp;<?echo
$saat;?>
        </font></b></td>
      </tr>
    <tr>
      <td width="866" height="500">
        <embed src="poligon.avi" width="866" height="500"></embed></td>
      </tr>
    </table></center>
</body>
</html>
```

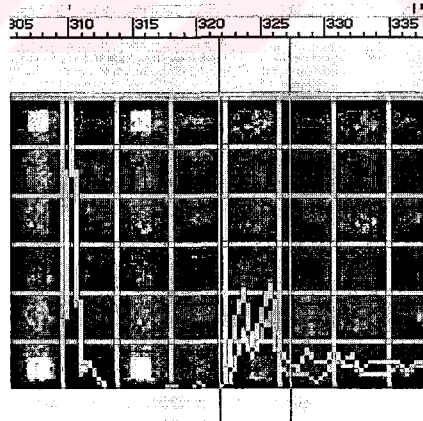
##### 4.3.4.1. Çalıştırılması

UYGULAMALAR isimli klasör "C:\FoxServ\www\" dizinine kopyalanır ve "C:\FoxServ\www\UYGULAMALAR\3-aviyukleme\php\" dizini altındaki TEST.swf programı çalıştırılır. Görev yöneticisi çalıştırılarak sayfanın çalışması için gerekli olan programlar

dışındaki tüm programlar sonlandırılır. Sayfa çalıştıktan sonraki ekran görüntüsü aşağıda verilmiştir.



Şekil 4.21. Test3.php Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi



Şekil 4.22. CPU Kullanımı

#### 4.3.4.2. Sonuç

Şekil incelendiğinde butona tıklanma anı ile sayfanın görüntülendiği an arasındaki zaman farkı **1 saniye** olarak görülmektedir. Talep anı ile sayfanın avi dosyasını ses ve görüntü olarak oynattığı ana kadarki CPU kullanımı Şekil 4.22.'de görüldüğü gibi **5 mm**'dir.

#### 4.3.5. Test3.cgi

```
#!/C:/FoxServ/perl/bin/perl.exe
print "Content-type: text/html", "\n\n";
($saniye,$dakika,$saat) = (localtime(time))[0,1,2];
$video="C:/FoxServ/www/cgi-bin/UYGULAMALAR/3-aviyukleme/cgi/poligon.avi";
print "<HTML>", "\n";
print "<BODY>", "\n";
print "<center>";
print "<table border=5 width=866 height=46 cellpadding=1 >";
    print "<tr>";
    print "<td width=866 height=24>";
    print "<p align=center><b><font color=#FF0000 size=5>Avi dosyası yüklenme";
    print "süresi</font></b></td>";
    print "</tr>";
    print "<tr>";
    print "<td width=866 height=18>";
    print "<p align=center><b><font color=#0000FF size=4>Sistem saati =";
    print "$saat:$dakika:$saniye";
    print "</font></b></td>";
    print "</tr>";
print "</table></center>";
print "<div align=center>";
print "<center>";
print "<table border=5 width=866 height=500 cellpadding=1 >";
print "<tr>";
    print "<td width=850 height=500>";
    print "<embed src=$video width=850 height=500></embed></td>";

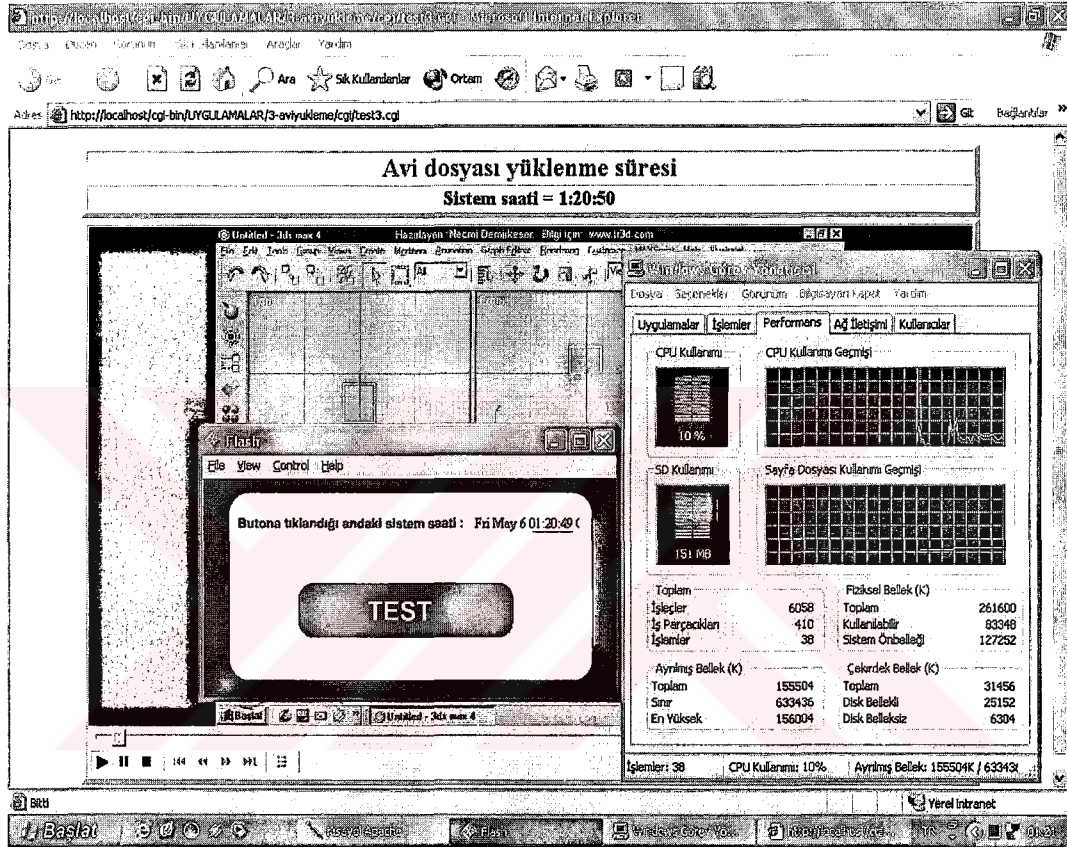
print "</tr>";

print "</BODY></HTML>", "\n";
```

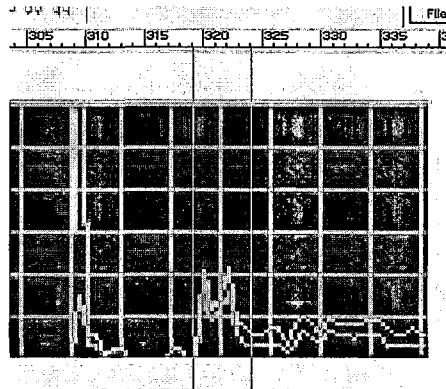


#### 4.3.5.1. Çalıştırılması

UYGULAMALAR isimli klasör “C:\FoxServ\www\cgi-bin\” dizinine kopyalanır ve “C:\FoxServ\www\cgi-bin\UYGULAMALAR\3-aviyukleme\cgi\” dizini altındaki TEST.swf programı çalıştırılır. “C:\FoxServ\apache\” klasörü altındaki “apache.exe” dosyası çalıştırılır ve böylece sayfadaki cgi kodlarını yorumlayıp HTML sayfası şeklinde sunacak olan sunucu (server) aktif edilir. TEST.swf programındaki TEST butonuna tıklanarak sayfa çalıştırılır. Sayfanın çalıştıktan sonraki ekran görüntüsü aşağıda verilmiştir.



Şekil 4.23. Test3.cgi Sayfa Görüntüsü, Test.swf Programı ve Görev Yöneticisi



Şekil 4.24. CPU Kullanımı

#### 4.3.5.2. Sonuç

Talep anı ile görüntülenme anı arasında **1 saniye** geçmiştir. Talep anı ile sayfamızın avi dosyasını ses ve görüntü olarak oynattığı ana kadarki CPU kullanımı Şekil 4.24.'de görüldüğü gibi **5 mm**'dir.

#### 4.3.6. Genel Uygulama Sonucu

Yukarıda üç sayfanında test sonuçları Tablo 4.3.'de görülmektedir. Buna göre göre avi formatında bir dosyanın yüklenmesini PHP ve CGI aynı sürelerde gerçekleştirirken, ASP aynı işlem için dört misli zamana gereksinim duymaktadır.

**Tablo 4.3.** ASP, PHP, CGI Sayfalarının Test Sonuçları(Avi yükleme)

|                  | <i>İşletilme süresi</i>   | <i>CPU Kullanım Durumu</i> |
|------------------|---------------------------|----------------------------|
| <i>Test3.asp</i> | <b>4 saniye</b>           | <b>18 mm</b>               |
| <i>Test3.php</i> | <b>1 saniye</b>           | <b>5 mm</b>                |
| <i>Test3.cgi</i> | <b>1 saniye</b>           | <b>5 mm</b>                |
| <b>SONUÇ</b>     | <b>PHP = CGI &gt; ASP</b> |                            |

#### 4.4. Uygulama-4 (10.000.000 Defa 100 Faktöriyel Hesaplama)

##### 4.4.1. Amaç

Bu uygulamada amaç, PHP, ASP ve CGI kodlarının işletilme hızlarını karşılaştırmaktır.

##### 4.4.2. Yapılış

Uygulamada yine 3 adet php, asp ve cgi sayfası oluşturulmuştur. Sayfaların içerikleri, 100 faktöriyeli 10.000.000 defa hesaplayacak şekilde düzenlenmiştir. Burada 1-10.000.000 arasındaki döngünün önce 1 de iken sistem saati, sonra 10.000.000 da iken sistem saati alınmıştır. Böylece aynı işlemi yapacak olan ASP, PHP ve CGI sayfalarının hangisinin bu işlemi daha hızlı yaptığı kıyaslanmıştır.

##### 4.4.3. Test4.asp

<HTML>

<HEAD>

<TITLE>TEST4 ASP</TITLE>

</HEAD>



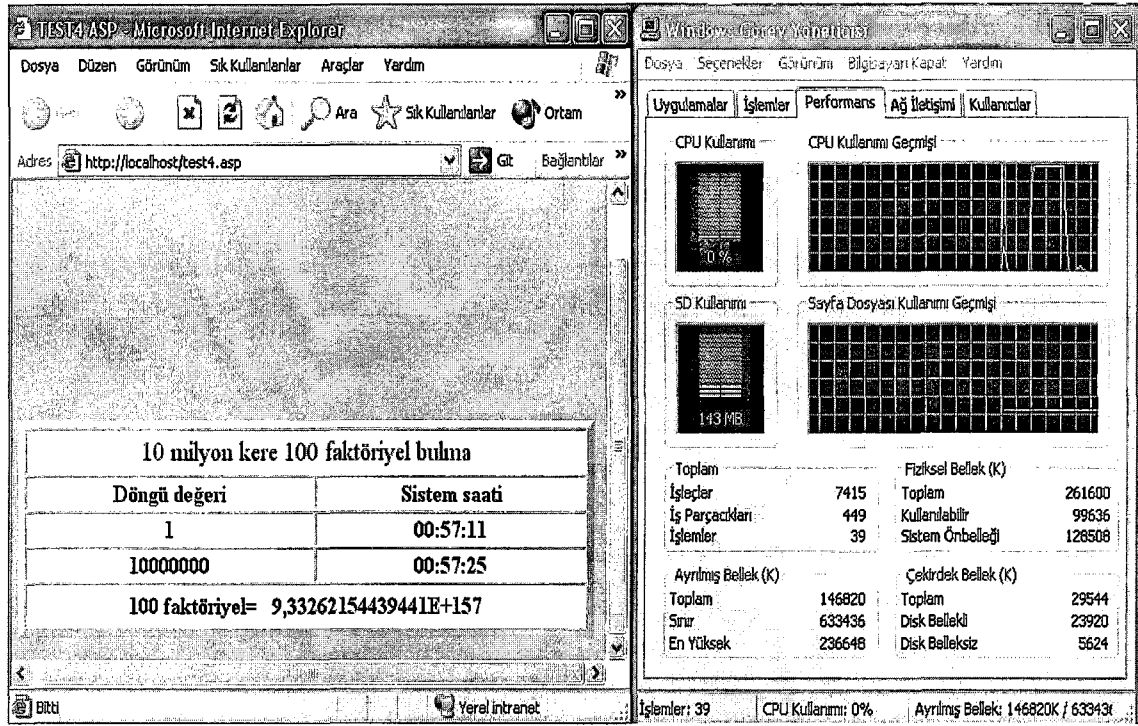
```

<BODY bgcolor="#C0C0C0">
<%
fak=1
sayac=1
for say = 1 to 10000000
if say = 1 then
s1=say
saat1=time()
end if
while sayac <=100
fak=fak*sayac
sayac=sayac+1
wend
if say = 10000000 then
s2=say
saat2=time()
end if
next
%>

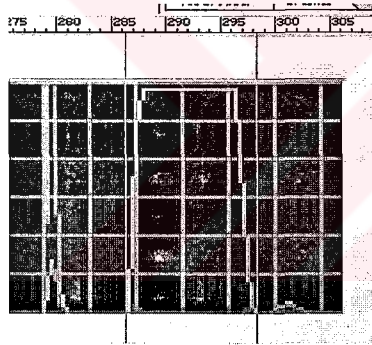
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<div align="center">
<center>
<table border="5" cellpadding="1" width="460" height="37" id="AutoNumber1">
<tr>
<td width="460" height="31" colspan="2" bgcolor="#FFFFFF">
<p align="center"><font size="4" color="#FF0000">10 milyon kere 100
faktöriyel bulma</font></td>
</tr>
<tr>
<td width="234" height="16" bgcolor="#FFFFFF">

```

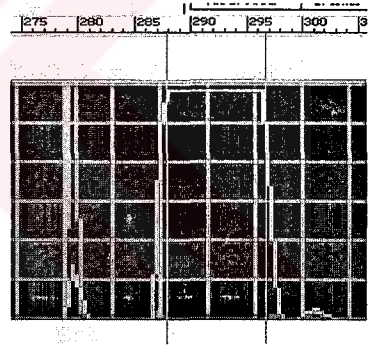




Şekil 4.25. Test4.asp Sayfa Görünümü ve Windows Görev Yöneticisi



Şekil 4.26. CPU Kullanımı



Şekil 4.27. CPU Kullanımı(%100)

#### 4.4.3.2. Sonuç

Sayfanın çalışma sonuçlarına göre döngü **14 saniyede** son bulmaktadır. Şekil 4.26.'da CPU kullanımının başlangıç ve bitiş değerleri arasında **12 mm**'lik fark olduğu görülmektedir. Şekil 4.27.'de ise CPU'nun %100'lük kullanımının **8.5 mm** olduğu görülmektedir.

#### 4.4.4. Test4.php

<HTML>

<HEAD>

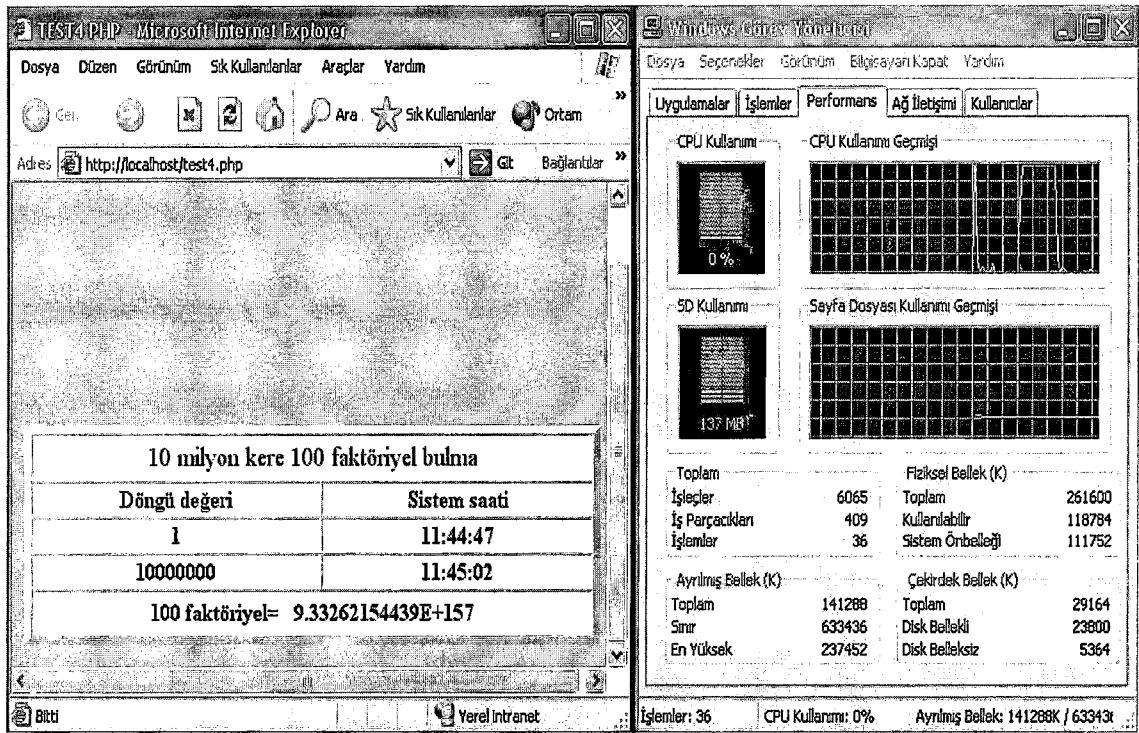
<TITLE>TEST4 PHP</TITLE>

```

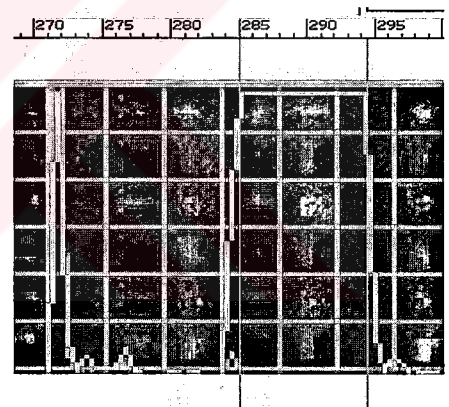
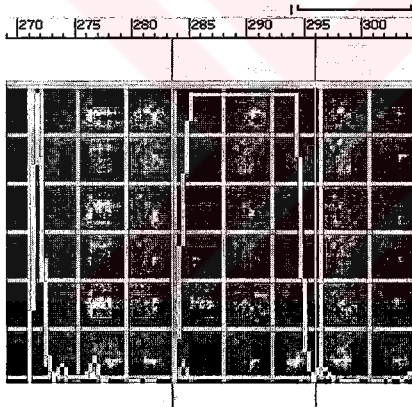
</HEAD>
<BODY bgcolor="#C0C0C0">
  <?
$fak=1;
$i=1;
for($say=1;$say<=10000000;$say++)
{
    if ($say==1)
    {
        $saat1 = date("H:i:s");
        $s1=$say;
    }
    while($i<=100)
    {
        $fak=$fak*$i;
        $i++;
    }
    if ($say==10000000)
    {
        $saat2 = date("H:i:s");
        $s2=$say;
    }
}
  ?>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <div align="center">
    <center>
      <table border="5" cellspacing="1" width="460" height="37" id="AutoNumber1">
        <tr>
          <td width="460" height="31" colspan="2" bgcolor="#FFFFFF">
            <p align="center"><font size="4" color="#FF0000">10 milyon kere 100

```





Şekil 4.28. Test4.php Sayfa Görünümü ve Windows Görev Yöneticisi



#### 4.4.4.2. Sonuç

Sonuçlara göre her defasında 100 faktöriyeli hesaplayan ve 1-10.000.000 arasında sayan php sayfası **15 saniyede** sona ermiştir. Şekil 4.29.'da CPU kullanımının başlangıç ve bitiş değerleri arasında **12.5 mm**'lik fark olduğu görülmektedir. Şekil 4.30.'da ise CPU'nun %100 lük kullanımının **9.5 mm** olduğu görülmektedir.



#### 4.4.5. Test4.cgi

```
#!/C:\FoxServ\perl\bin\perl.exe
print "Content-type: text/html", "\n\n";
$fak=1;
$sayac=1;
for ($say=1;$say<=10000000;$say++)
{

    if($say == 1)
    {
        ($saniye1,$dakika1,$saat1) = (localtime(time))[0,1,2];
        $s1=$say
    }
    while($sayac <=100)
    {

        $fak=$fak*$sayac;
        $sayac++;
    }
    if($say == 10000000)
    {
        ($saniye2,$dakika2,$saat2) = (localtime(time))[0,1,2];
        $s2=$say
    }
}

print "<HTML><HEAD><TITLE>TEST4 CGI</TITLE></HEAD>";
print "<BODY bgcolor=#C0C0C0>";
print "<p>&nbsp;</p>";
print "<p>&nbsp;</p>";
print "<p>&nbsp;</p>";
print "<p>&nbsp;</p>";
print "<p>&nbsp;</p>";
```

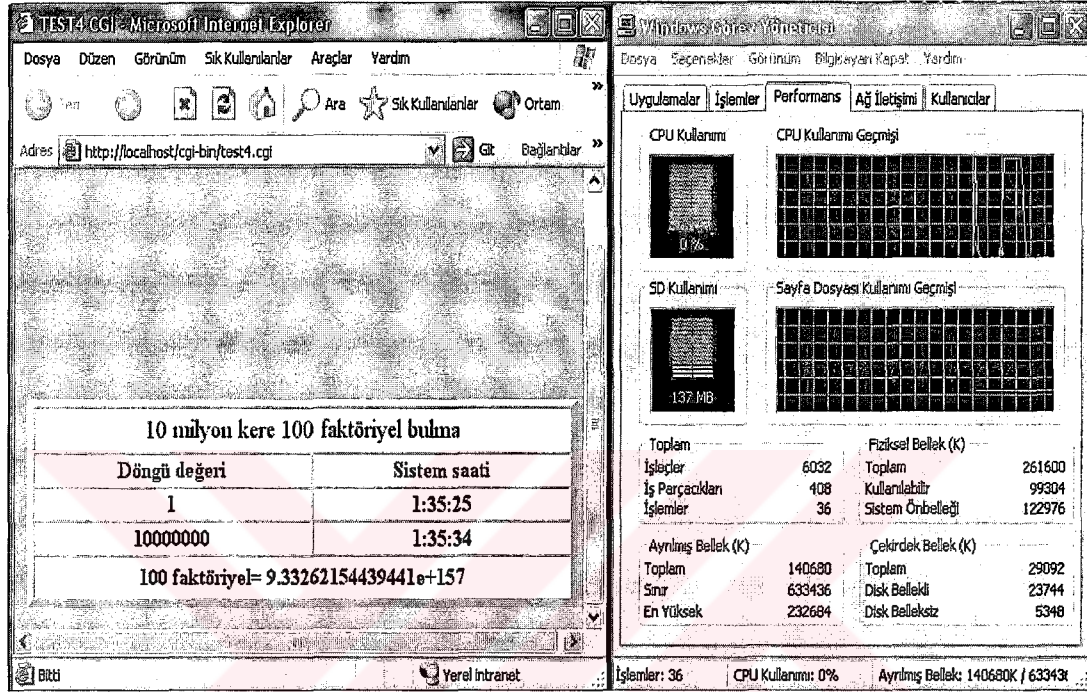
```

print "<div align=center>";
print "<center>";
print "<table border=5 cellpadding=1 width=460 height=37 id=AutoNumber1>";
print "<tr>";
print "<td width=460 height=31 colspan=2 bgcolor=#FFFFFF>";
print "<p align=center><font color=#FF0000 size=4 >10 milyon kere 100 faktöriyel  
bulma</font></td>";
print "</tr>";
print "<tr>";
print "<td width=234 height=16 bgcolor=#FFFFFF>";
print "<p align=center><b><font size=3 color=#0000FF>Döngü değeri</font></b></td>";
print "<td width=220 height=16 bgcolor=#FFFFFF>";
print "<p align=center><b><font size=3 color=#0000FF>Sistem saati</font></b></td>";
print "</tr>";
print "<tr>";
print "<td width=234 height=22 bgcolor=#FFFFFF><font size=3><p align=center> <b> $s1  
</b> </font> </td>";
print "<td width=220 height=22 bgcolor=#FFFFFF><font size=3><p align=center> <b>  
$saat1:$dakika1:$saniye1 </b></font></td>";
print "</tr>";
print "<tr>";
print "<td width=234 height=21 bgcolor=#FFFFFF><font size=3><p align=center> <b> $s2  
</b> </td>";
print "<td width=220 height=20 bgcolor=#FFFFFF><font size=3><p align=center><b>  
$saat2:$dakika2:$saniye2</b></td>";
print "</tr>";
print "<tr>";
print "<td width=460 colspan=2 height=30 bgcolor=#FFFFFF><font size=3><p  
align=center><b>100 faktöriyel= $fak </b></td>";
print "</tr>";
print "</table>";
print "</center>";
print "</div>";
print "</BODY>";
print "</HTML>";

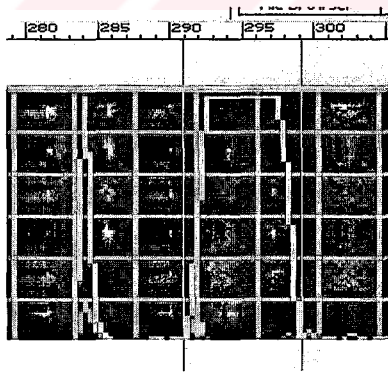
```

#### 4.4.5.1. Çalıştırılması

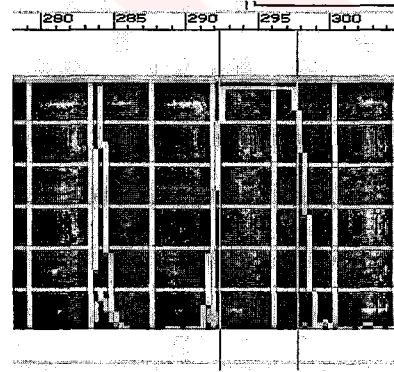
Sayfayı çalıştırmak için test4.cgi dosyası, “c:\FoxServ\www\cgi-bin\” dizini altına kopyalanır. İnternet Explorer adres satırına “http://localhost/cgi-bin/test4.cgi” yazılıp enter tuşuna basılır. Sayfanın çalıştırıldıktan sonra alınan ekran görüntüsü aşağıda verilmiştir.



Şekil 4.31. Test4.cgi Sayfa Görünümü ve Windows Görev Yöneticisi



Şekil 4.32. CPU Kullanımı



Şekil 4.33. CPU Kullanımı(%100)

#### 4.4.5.2. Sonuç

Şekil incelendiğinde döngünün 9 saniye sürdüğü görülmektedir. Şekil 4.32.’de CPU kullanımının başlangıç ve bitiş değerleri arasında 8 mm’lik fark olduğu görülmektedir. Şekil 4.33.’de ise CPU’nun %100’lük kullanımının 4.5 mm olduğu görülmektedir.

#### 4.4.6. Genel Uygulama Sonucu

Yukarıda üç sayfanında test sonuçları Tablo 4.4.'de görülmektedir. Buna göre 10 milyonluk döngü içerisinde 100 faktöriyel işleminin yüklenme istatistikleri sonucunda CGI; ASP ve PHP'den daha hızlı işlem yapmaktadır.

**Tablo 4.4.** ASP, PHP, CGI Sayfalarının Test Sonuçları(Faktöriyel Hesaplama)

|                  | <i>İşletilme süresi</i>      | <i>CPU Kullanım Durumu</i> | <i>CPU'nun %100'lük Kullanım Durumu</i> |
|------------------|------------------------------|----------------------------|-----------------------------------------|
| <i>Test4.asp</i> | 14 saniye                    | 12 mm                      | 8.5 mm                                  |
| <i>Test4.php</i> | 15 saniye                    | 12.5 mm                    | 9.5 mm                                  |
| <i>Test4.cgi</i> | 9 saniye                     | 8 mm                       | 4.5 mm                                  |
| <b>SONUÇ</b>     | <b>CGI &gt; ASP &gt; PHP</b> |                            |                                         |

#### 4.5. Uygulama-5 (8.96 MB Büyüklüğündeki Yeni.txt Dosyasından Bilgi Okutma)

##### 4.5.1. Amaç

Bu uygulamada PHP, ASP ve CGI dilleriyle hazırlanan programlarla dosyadan bilgi okutma hususunda hızlarını karşılaştırmak amaçlanmıştır.

##### 4.5.2. Yapılışı

Uygulama için okuma.asp, okuma.php ve okuma.cgi adında 3 sayfa oluşturulmuştur. Bu sayfaların içeriklerinin, tarayıcı (internet explorer) çalıştırılıp sayfanın sunucu tarafından yorumlanıp ekrana gelmesi amaçlanmaktadır. Böylece tarayıcı 8,96 MB büyüklüğündeki Yeni.txt dosyasının içeriğini okuyup ekranda görüntülemektedir.

Aşağıda önceden bu uygulama için oluşturulmuş ASP, PHP ve CGI sayfalarının içerikleri ve çalıştırılmalarıyla ilgili bilgiler verilmektedir.

##### 4.5.3. Okuma.asp

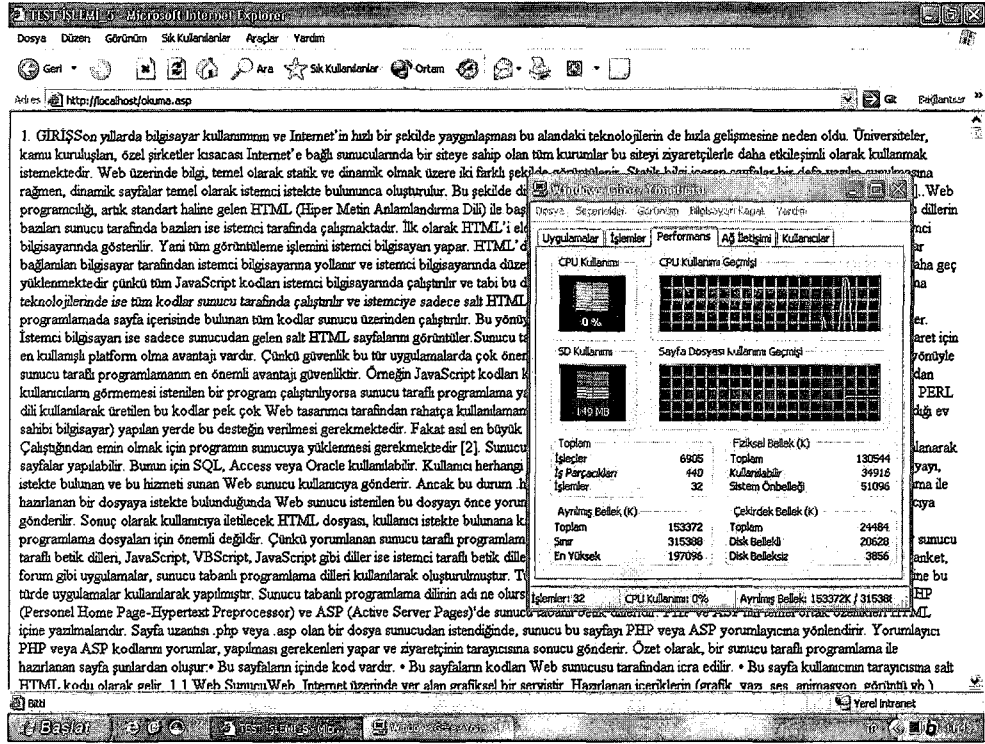
```
<% Option Explicit %>
<HTML>
<HEAD>
<TITLE>TEST İŞLEMİ_5</TITLE>
</HEAD>
<BODY>
<%
```

```
Dim DosyaSistemi, MetinDosyasi, Satir
Set DosyaSistemi = CreateObject("Scripting.FileSystemObject")
Set MetinDosyasi = DosyaSistemi.OpenTextFile("c:\Inetpub\wwwroot\yeni.txt",1, 0)
Do
Satir = MetinDosyasi.ReadLine
%>
<%=Satir%>
<%
Loop Until MetinDosyasi.AtEndOfStream
MetinDosyasi.Close
%>
</BODY>
</HTML>
```

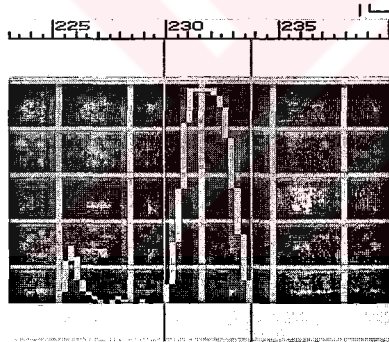
#### 4.5.3.1. Çalıştırılması

UYGULAMALAR isimli klasör “c:\Inetpub\wwwroot\” dizini altına kopyalandıktan sonra tarayıcının adres satırına “http://localhost/UYGULAMALAR/5-dosyaokuma/asp/okuma.asp” yazılarak sayfa çağırılır. Windows görev yöneticisi çalıştırılır ve sayfanın çalışması için gerekli olan programlar dışındaki tüm programlar sonlandırılır. Görev yöneticisinin performans kısmı aktif yapılır ve böylece sayfa yüklenirken CPU’nun kullanım grafiği elde edilmiş olacaktır. Çalıştırılma sonucu aşağıda görülmektedir.

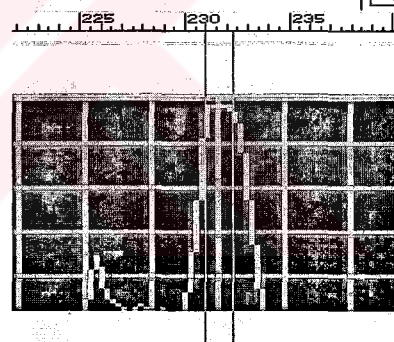




Şekil 4.34. Okuma.asp Sayfa Görünümü ve Görev Yöneticisi



Şekil 4.35. CPU Kullanımı



Şekil 4.36. CPU Kullanımı(%100)

#### 4.5.3.2. Sonuç

Yukarıdaki şekilde sayfanın tam olarak Yeni.txt dosyasını kuduğu ana kadar geçen zaman dilimi içindeki CPU kullanımı görülmektedir. Sayfanın tam olarak yükleme süresi belirlenemediği için Görev Yöneticisini kullanarak performans grafiğinden yararlanarak ASP, PHP ve CGI sayfalarının ayrı ayrı CPU grafikleri karşılaştırılarak hangisinin daha hızlı yada daha yavaş yüklendiği konusunda bir fikir edinilebilir. Şekil 4.35 incelendiğinde işlemcinin başlama anı ile bitiş anı arasında 4 mm olduğu görülmektedir. Şekil 4.36.'da ise CPU'nun %100 lük kullanımı 1.5 mm olarak görülmektedir.



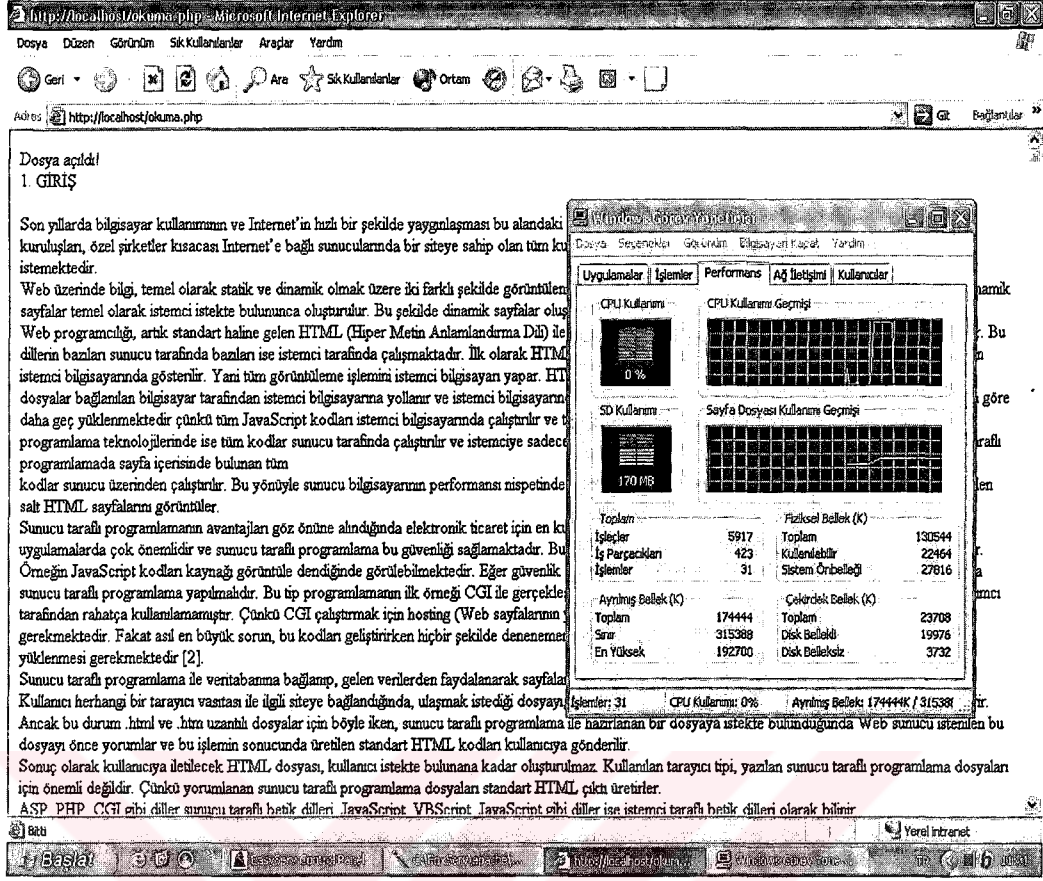
#### 4.5.4. Okuma.php

```
<?php
$dosya_dizin = "/FoxServ/www/";
if ($dosya = (fopen ("$dosya_dizin/yeni.txt" , 'r') ) ) {
    print ("Dosya açıldı!<br>");
}
else {
    print ("Dosya açılmadı!");
}
while ( ! feof ($dosya ) ) {
    $satir = fgets ( $dosya, 1024 ) ;
    print ("$satir<br>");
}
fclose ($dosya);
?>
```

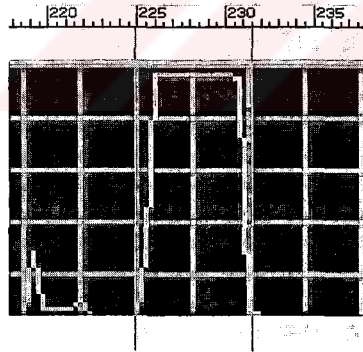
##### 4.5.4.1. Çalıştırılması

UYGULAMALAR isimli klasör “c:\FoxServ\www\ ” dizini altına kopyalandıktan sonra “C:\FoxServ\apache\” klasörü altındaki “apache.exe” dosyası çalıştırılır ve böylece sayfadaki php kodlarını yorumlayıp HTML sayfası şeklinde sunacak olan sunucu (server) aktif edilir.

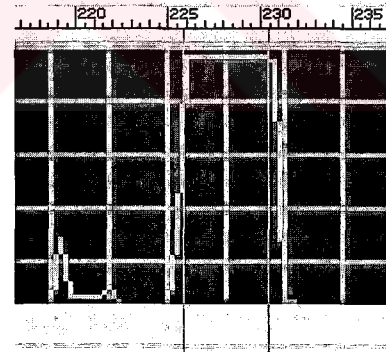
Tarayıcı                      adres                      satırına                      “http://localhost/UYGULAMALAR/5-dosyaokuma/php/okuma.php” yazılarak sayfa çağırılır. Görev yöneticisinin performans kısmı aktif yapılır ve böylece sayfa yüklenirken CPU’nun kullanım grafiği elde edilmiş olacaktır. Çalıştırılma sonucu aşağıda görülmektedir.



Şekil 4.37. Okuma.php Sayfa Görünümü ve Görev Yöneticisi



Şekil 4.38. CPU Kullanımı



Şekil 4.39. CPU Kullanımı(%100)

#### 4.5.4.2. Sonuç

Şekil 4.38. incelendiğinde işlemci kullanımının başlama anı ile bitiş anı arasında 6.5 mm olduğu görülmektedir. Şekil 4.39.'da ise CPU'nun %100'lük kullanımı 4.5 mm olarak görülmektedir.

#### 4.5.5. Okuma.cgi

CGI sayfası PERL programlama dili kullanılarak yazılmıştır. Bu bütün uygulamalarda böyledir. PERL dili kullanıldığı için sayfa içindeki perl komutlarının yorumlanıp tekrar gönderilmesi için aşağıdaki ilk program satırı yani “#!C:\FoxServ\perl\bin\perl.exe” kullanılmıştır. Buradaki “perl.exe” dosyası ile perl komutları yorumlanıp işletilmektedir.

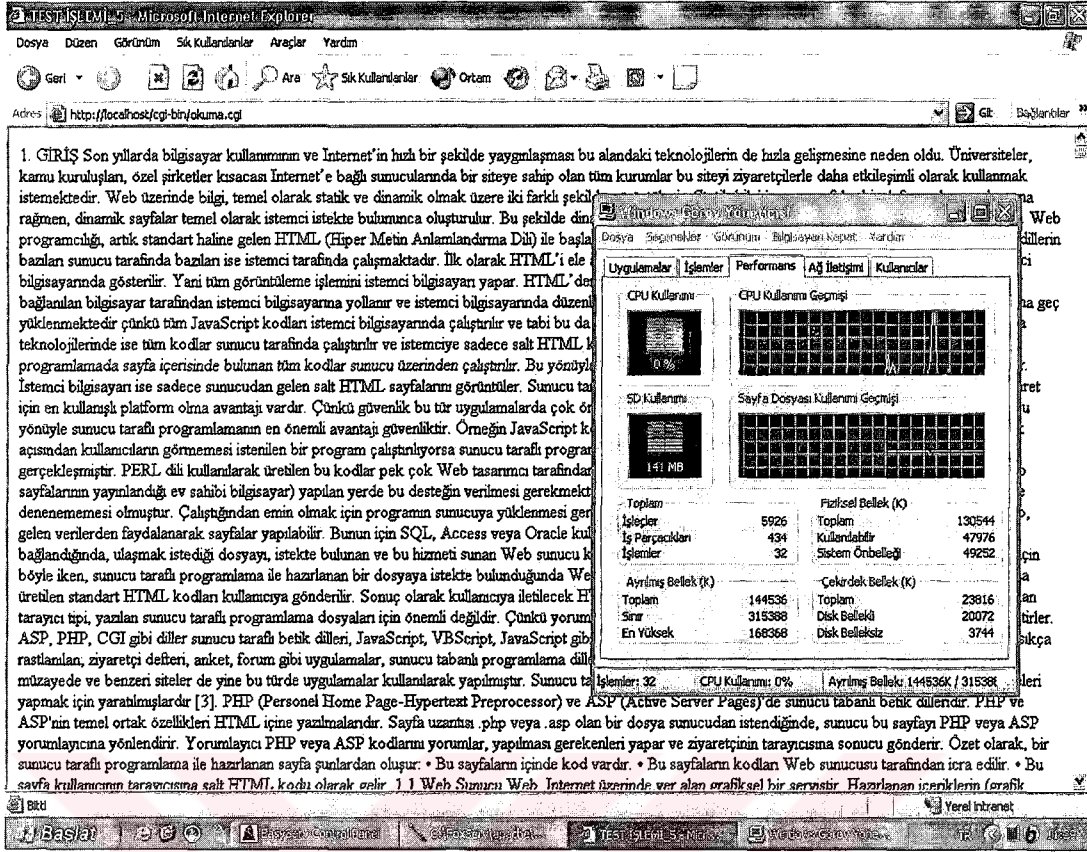
```
#!C:\FoxServ\perl\bin\perl.exe
print "Content-type:text/html\n\n";
print"<HTML><HEAD><TITLE>TEST İŞLEMİ_5</TITLE></HEAD>";
print"<BODY>";
if (open(DOSYAM, "yeni.txt")) {
    $kayit = <DOSYAM>;
    while ($kayit ne "") {
        print ($kayit);
        $kayit =<DOSYAM>;
    }
}

print "</BODY></HTML>", "\n";
```

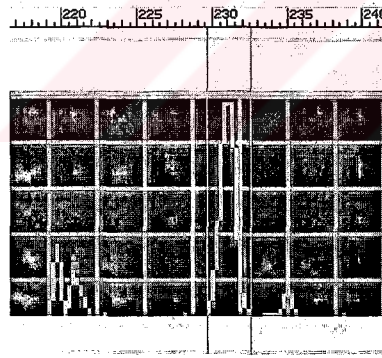
#### 4.5.5.1. Çalıştırılması

UYGULAMALAR isimli klasör “c:\FoxServ\www\ cgi-bin\” dizini altına kopyalandıktan sonra “C:\FoxServ\apache\” klasörü altındaki “apache.exe” dosyası çalıştırılır ve böylece sayfadaki php kodlarını yorumlayıp HTML sayfası şeklinde sunacak olan sunucu (server) aktif edilir.

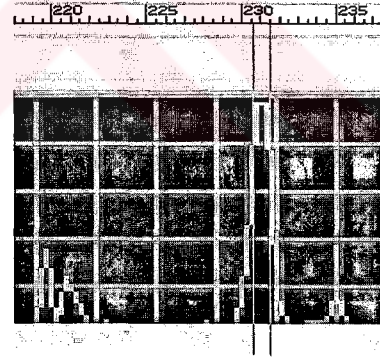
Tarayıcı adres satırına “http://localhost/UYGULAMALAR/5-dosyaokuma/cgi/okuma.cgi” yazılarak sayfa çağırılır. Çalıştırılma sonucu aşağıda görülmektedir.



Şekil 4.40. Okuma.cgi Sayfa Görünümü ve Görev Yöneticisi



Şekil 4.41. CPU Kullanımı



Şekil 4.42. CPU Kullanımı(%100)

#### 4.5.5.2. Sonuç

Sayfanın talep edildiği andan yüklendiği ana kadar geçen zaman diliminde ki CPU kullanım grafiği ekranda görülmektedir. Şekil 4.41. incelendiğinde işlemci kullanımının başlama anı ile bitiş anı arasında 3 mm olduğu görülmektedir. Şekil 4.42.'de ise CPU'nun %100 lük kullanımı 1 mm olarak görülmektedir.

#### 4.5.6. Genel Uygulama Sonucu

Yukarıda üç sayfanında test sonuçları Tablo 4.5.'de görülmektedir. Buna göre herhangi bir txt dosyasından bilgi okutma işleminin yüklenme istatistiklerine göre, CGI; ASP ve PHP'den daha kısa sürede yükleme işlemini tamamlamaktadır.

Tablo 4.5. ASP, PHP, CGI Sayfalarının Test Sonuçları(Dosya Okuma)

|                  | <i>CPU Kullanım Durumu</i>   | <i>CPU'nun %100'lük Kullanım Durumu</i> |
|------------------|------------------------------|-----------------------------------------|
| <i>okuma.asp</i> | 4 mm                         | 1.5 mm                                  |
| <i>okuma.php</i> | 6.5 mm                       | 4.5 mm                                  |
| <i>Okuma.cgi</i> | 3 mm                         | 1 mm                                    |
| <b>SONUÇ</b>     | <b>CGI &gt; ASP &gt; PHP</b> |                                         |



## 5. SONUÇ VE DEĞERLENDİRMELER

Bu tez çalışmasında, web tabanlı programlama dilleri konusunda yapılan araştırmalar ve yöntemler incelenmiştir. İncelemeler sonucunda genelde web tabanlı programlama için ASP, PHP, ve CGI dillerinin kullanıldığı görülmüştür. Bu araştırma sonucunda, web tabanlı programlama dilleri hakkında genel bir bilgi edinilerek, bunların birbiri ile karşılaştırılması yapılmıştır.

Yapılan değerlendirmeler neticesinde, PHP programlama dilinin şu anda Dünya’da bireysel kullanıcılar için en popüler programlama dili olduğu, performans açısından bakıldığında diğer dillere bir üstünlük sağlamakta olan PHP’nin, çeşitli şirketlerin laboratuvarlarında yapmış oldukları hız testlerinde önde olduğu gözükümüştür. Ancak PHP bir dil olduğundan dolayı, ASP gibi istenilen bir dille yazılması mümkün değildir. Bu nedenle sadece PHP dilini öğrenmek gerekmektedir. Bu da PHP dilinin önemli bir dezavantajı olarak karşımıza çıkmaktadır. Ayrıca aşağıda verilen Tablo 5.’den de görüleceği üzere, işlem çeşidine göre PHP’nin diğer dillere olan üstünlüğünde, değişimler olabileceği gerçeği saptanmıştır.

ASP diline bakıldığında ise Java, C++, PHP gibi bir dile gereksinim duymamaktadır. ASP, bir teknolojidir. Sadece ASP için bir dil seçilmesi yeterli olup büyük avantaj sağlamaktadır. ASP için en büyük dezavantaj ise Microsoft’un çıkarmış olduğu işletim sistemi web sunucuda çalışması olarak gösterilmektedir. Hız konusunda da gene sıralama, yapılan işlem çeşidine göre farklılıklar gösterecektir.

CGI ile hazırlanan kullanıcı arayüzlerine bakıldığında ise kolay ve hızlı hazırlanabilmesi, istenilen herhangi bir programlama dilinin kullanılabilmesi, değişik ortamlarda çalışabilmesi ve dağıtık ortamlarda kullanılabilir olması, CGI’nin önemli avantajları arasında sayılabilmektedir. Ancak dikkatsiz yazılmış CGI programaları güvenlik açıklarına neden olabilmektedir ve işlem çeşidi burada da önemli bir parametre olarak karşımıza çıkmaktadır.

Bu tez çalışmasında kıyaslamalar yapılırken farklı uygulamalar kullanılarak, söz konusu yazılımların aynı işlevi ne kadar süratle yaptıkları bulunmuştur. Ortaya Tablo 5.1.’de verilen ilginç durumlar çıkmıştır.



**Tablo 5.1.** ASP, PHP ve CGI Yazılımlarının Farklı İşlem Çeşitleri Karşısındaki İşletme Süreleri

| <i>İşlem Çeşidi</i>                             | <i>İşletme Süresi<br/>(Saniye)</i> |            |            | <i>Sonuç</i>    |
|-------------------------------------------------|------------------------------------|------------|------------|-----------------|
|                                                 | <i>ASP</i>                         | <i>PHP</i> | <i>CGI</i> |                 |
| <i>0-10.000.000 Arası Sayan Döngü</i>           | 11                                 | 11         | 5          | CGI > ASP = PHP |
| <i>20.000 Adet JPG Resmi Yükleme</i>            | 1                                  | 1          | 60         | ASP = PHP > CGI |
| <i>Avi yükleme</i>                              | 4                                  | 1          | 1          | PHP = CGI > ASP |
| <i>10.000.000 Defa 100 Faktöriyel Hesaplama</i> | 14                                 | 15         | 9          | CGI > ASP > PHP |

Tablo 5.1. dikkatli incelendiğinde, ASP ve PHP yazılımlarının farklı işlem çeşitleri karşısında aynı sonucu elde etmek için harcadıkları sürelerin birbirine yakın oldukları görülmektedir. Ancak bu yazılımların işletme hız performansları ve birbirine üstünlük sıraları, işlem çeşidine bağlı olduğu gerçeği ortaya çıkmaktadır. Yani örneğin 0-10.000.000 arası sayan bir döngü işleminde CGI en süratli yazılım olarak karşımıza çıkarken, 20.000 adet JPG resmi yüklemeye en hızlı sırayı ASP ve PHP birlikte paylaşabilmektedir. Eğer işlem çeşidi Avi ise, ilk sırayı PHP almaktadır.

Sonuç olarak web ortamında hizmet veren bir yazılım ile ilgili hız konusunda karar verilecekse, işlem türünün çok önemli bir parametre olarak karşımıza çıkacağı gerçeği asla gözden uzak tutulmamalıdır. Bir işlem çeşidi için bir diğer yazılım ilk sırayı alırken, bu hız sıralamasında diğer bir işlemde bir başka yazılım öne çıkabilmektedir. Bu nedenle hızlı işleme konusunda karar verirken, işlem çeşidinin ne olduğu çok açık bir biçimde ortaya konulmalıdır.

## KAYNAKÇA

- [1] VAROL, A.; ALKAN, T. Eğitimde İnternet'in Yeri, Türk Cumhuriyetleri ve Asya Pasifik Ülkeleri Uluslararası Eğitim Sempozyumu, 24-26 Eylül 1997, Elazığ, S: 172-184
- [2] VAROL, A., ALKAN, T. İnternet'e Genel Bakış, Uzaktan Eğitim, Kış 1998, S: 10-16
- [3] <http://www.core.gen.tr>
- [4] <http://www.ulakbim.gov.tr>
- [5] <http://www.gaziantep.edu.tr>
- [6] STANEK, William Robert. HTML, JAVA, CGI, SGML, VRML, (İstanbul: Sistem Yayıncılık,2000).
- [7] <http://www.odevsitesi.com>
- [8] <http://www.lavinya.net>
- [9] <http://www.pcakademi.com>
- [10] <http://www.mycgiserver.com/~ocal/>
- [11] [hadi@ulakbim.gov.tr](mailto:hadi@ulakbim.gov.tr)
- [12] [cu@ulakbim.gov.tr](mailto:cu@ulakbim.gov.tr)
- [13] GÜNEŞ, A.: Perl ile CGI Programlama Kılavuzu,1. Baskı (İstanbul: Alfa Yayınları Mayıs 2000).
- [14] ÇİNİ Ö.: Web Tasarım Teknikleri, PPT sunu
- [15] ACAR, Ö.: PERL/CGI, 1. Baskı (İstanbul: Pusula Yayıncılık Ocak 2001).
- [16] <http://kitap.selcuk.edu.tr>
- [17] <http://msdn.microsoft.com/>
- [18] <http://www.lotus.com/>
- [19] <http://www.allaire.com/Products/ColdFusion/>
- [20] <http://java.sun.com/>
- [21] <http://www.apache.org>
- [22] <http://www.microsoft.com/iis/>
- [23] [www.yazilim.mikrobeta.com.tr](http://www.yazilim.mikrobeta.com.tr)
- [24] <http://www.zeus.co.uk/>
- [25] ŞAMLI, M.: PHP ile Web Programcılığı, 1. Baskı (İstanbul: Pusula Yayıncılık ve İletişim Ltd. Mayıs 2002)
- [26] GÜNEŞ, A.: Perl İle CGI Programlama Kılavuzu, 1. Baskı (ALFA Basım Yayım Dağıtım Ltd.Şti.)
- [27] ACAR, Ö.: PERL CGI, 1. Baskı (Pusula Yayıncılık ve İletişim Ltd. Ocak 2001)
- [28] DEMİRKOL, Z.: ASP İle Web Programcılığı ve Elektronik Ticaret, 6. Baskı (Pusula Yayıncılık ve İletişim Ltd. Mayıs 2002)

- [29] ÇUBUKÇU, F.: ASP 3.0 ile Veritabanı Yönetimi, 1 Baskı (ALFA Basım Yayım Dağıtım Ltd.Şti. Eylül 2001)
- [30] <http://www.oksinet.com>
- [31] <http://www.oreilly.com/catalog/webpro2/>
- [32] KARAHOCA, A.: ASP ile WEB Programcılığına Giriş, 3. Baskı, (İstanbul: Beta Yayınevi Mart 2001)
- [33] <http://developer.iplanet.com/tech/ssjs/>
- [34] BARENGİ, R.: DELPHI 5'E BAKIŞ, 1. Baskı (Ankara: Seçkin Yayınevi Şubat 2000).
- [35] CALVERT, C.: Delphi 4, 1. Baskı (İstanbul: Sistem Yayıncılık Eylül 1999).
- [36] [www.ulakbim.gov.tr](http://www.ulakbim.gov.tr)
- [37] [www.delphiturk.com](http://www.delphiturk.com)
- [38] [www.seckin.com.tr](http://www.seckin.com.tr)
- [39] [www.core.gen.tr](http://www.core.gen.tr)
- [40] <http://www.ipplanet.com/products/infrastructure/webrowsers/>
- [41] <http://www.aspnet.net>
- [42] <http://www.e-hack.org/index.php?id=220>
- [43] <http://www.serdar.ktg.com>
- [44] <http://www.programlama.com>
- [45] CANTU, M.: DELPHI 5 Uygulama Geliştirme Kılavuzu, 1. Baskı (İstanbul: Alfa Yayınları Ocak 2000).
- [46] <http://www.istanbul.edu.tr>

## ÖZGEÇMİŞ

Ferhat BAĞÇACI

ferhatb@firat.edu.tr

Tlf: 424 2370000 / 6726

Fırat Üniversitesi Teknik Eğitim Fakültesi

Elektronik-Bilgisayar Eğitimi Bölümü

ELAZIĞ

1978 yılında Elazığ'da doğdu. İlk, orta ve lise öğrenimini Elazığ'da tamamladıktan sonra 1996 yılında Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik-Bilgisayar Eğitimi Bölümü Bilgisayar Öğretmenliği Bölümünü kazandı. 2000 yılında bu bölümden mezun oldu. Eylül 2000 – Mart 2002 tarihleri arasında Kastamonu Endüstri Meslek Lisesi'nde Bilgisayar Öğretmeni olarak görev yaptı. 2001 yılı Kasım ayında Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik Bilgisayar Eğitimi Bölümü tarafından yapılan araştırma görevliliği sınavını kazanarak 2002 yılı Mart ayında göreve başladı. Nisan 2002 – Aralık 2002 tarihleri arasında Van Jandarma Asayiş Komutanlığında 285. kısa dönem er olarak vatani görevini yaptı. 2003 yılı Şubat ayında Fırat Üniversitesi Fen Bilimleri Enstitüsü Elektronik Bilgisayar Eğitimi Ana Bilim Dalı Bilgisayar Sistemleri Bilim Dalı'nda yüksek lisans eğitimine başladı. Halen Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik-Bilgisayar Eğitimi bölümünde araştırma görevlisi olarak çalışmaktadır. Yabancı dili İngilizcedir.