

## ÖZET

Gelişen ve değişen dünyanın ürünü olan web sayfalarında, değişime uyum sağlanarak çok hızlı bir gelişme sürecine girmiştir. Özellikle son yıllarda Web Sayfası sadece devlet kurumları, üniversiteler ve büyük şirketler için gerekli olmak tan çıkmış olmazsa olmaz bir koşul haline gelmiştir. Buna ek olarak kişiye ait sayfalar bile neredeyse gereklilik durumuna ulaşmıştır. Buna en güzel örnek iş başvuru formlarında kişisel web sayfası bölümünün bulunmasıdır.

Tabi bu çok hızlı gelişim karşısında web sayfası üretim şeklide değişmek zorunda kalmıştır. Eskiden web sayfaları geliştirmek için HTML dosyaları üretimi benimsenmişti. Artık bu yöntem kabul görmemektedir. Çünkü böylesine hızlı bir gelişimin devam edebilmesi için çok kısa zamanda çok fazla iş yapılması gerekmektedir. HTML dosyaları üreterek değişimlerin hızlı yapılabilmesi mümkün değildir. Bu üretim yöntemi ile bir web sayfası güncellenmek istenildiğinde sayfada yer alan bütün tarih, isim, telefon tarzı bilgilerin tek tek kodlarının bulunarak değiştirilmesi gerekmektedir.

İşte bu nedenlerden dolayı HTML'ye ek ve yardımcı yazılımlar geliştirilmek zorunda kalmıştır. Hatta dahada ileri gidilerek veritabanı bağlantılı Web Sayfalar'ı üretilmesi amacıyla, veritabanı bağlantılı yeni diller geliştirilmiştir. Son günlerde bu dillerin en gözdesi PHP (personal home page) 'dir. Bunun arkasında reklam desteği değil, iyi performans, kolay öğrenim ve açık kaynak felsefesi yatmaktadır. PHP dili yine açık kaynaklı yazılım felsefesinin bir ürünü olan MySql ile biraraya geldiğinde gerçekten rakip tanımaz bir hal almıştır. Bu ikili ortak çatı altında çalıştırılmak sureti ile tek bir program gibi hareket etmektedir. Durum böyle olunca hem sunucu tarafında çalışan bir script dil hemde bir veritabanı birleştirilerek tek bir yazılımı gibi kullanma olanağı ortaya çıkıyor. İşte bu avantaj hem açık kaynak felsefesinin başarısını hemde Web Sayfalarında yeni yöntemlerin ortaya çıkmasını hızlandırmıştır.

Bugünlerde Web Sayfaları üretmekte sadece içeriğe yönelinmesi amacıyla açık kaynak web portalları dünyanın kullanımına sunulmuştur. Bu düşüncenin altında, internet dünyasını statik yapıdan kurtarmak ve yeni Web Yazılımcılar'ının deneyleri ile bu dünyayı bir çöplük haline getirmesini engellemek yatmaktadır.

İşte bütün bu gelişmelerin sonucunda Astronomi Bölümü Web Sayfası'nın statik yapıdan arındırılıp, dinamik bir hale getirilmesi amaçlanmış ve bu çalışmamda bunu gerçekleştirmiş bulunuyorum. Yapılan iş; Astronomi Bölümü Web Sayfası'nı PHP ve MySQL ikilisini kullanarak dinamikleştirmektir.

Bu kitapçıkta, PHP ve MySQL dillerinin anayapısı ve geliştirilen web sayfası ile ilgili ayrıntılı bilgilerin anlatıldığı bölümler yer almaktadır.

## BÖLÜM I

### 1. PHP

#### 1.1 Kısa Tarihçe

Rasmus Lerdorf, internet tarihin ilk dönemlerinde yani internet'in henüz yayıldığı 1990'ların ortalarına doğru iş arıyordu; hayat öyküsünü bir kişisel sayfada yayınlamak ve başvuracağı yerlere bu sayfanın adresini verebileceğini düşündü. Fakat o dönemde özellikle üniversitelerin Unix ağlarında kurulan Web Sunucularda kişisel sayfa yapmak kolay değildi. Lerdorf, kendisi için hazırladığı yazılımın Web'e aşina olmayanlar tarafından da kolayca kullanılabilceğini düşündü. Bu yazılımın büyükçe bir bölümü Perl dilinden alınmıştı. 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 yazdı ve programın adı PHP/FI (Form Interpreter/Form Yorumlayıcı) oldu. Kimileri programın bu sürümüne PHP2 adını taktı; ve bu ad, programın çok değişmesine ve gelişmesine rağmen uzun süre değişmeden kaldı.

Rasmus Lerdorf, 1995'in ortalarında, Zeev Suraski, Stig Bakken, Shane Caraveo ve Jim Winstead ile bir grup kurdu ve PHP'yi Perl'den ödünç alma rutinlerle iş yapan bir paket olmaktan çıkartıp, nesne-yönelimli (Object-Oriented) bir programlama dili haline getirdi. 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.

## 1.2 PHP 'ye Giriş

PHP, bir Script dilidir; yani kodları düz yazı dosyaları halinde kaydedilir ve kullanılacağı ortamda bir yorumlayıcı tarafından yorumlanır. Bu, PHP ile yazacağınız programları, derlemek yani ortaya bir "exe" veya çalıştırılabilir başka bir dosya çıkartmaya gerek olmadığı anlamına gelir. Fakat PHP Script 'lerinizi çalıştırabilmek için bu dili bilen bir programa ihtiyacınız vardır. Bu programın tek başına (komut istemci penceresinde veya terminal ekranında) çalışması mümkün olduğu gibi, Web Server tarafından da çalıştırılabilir olması gerekir. Başka bir deyişle Web Sunucu programının PHP anlar hale getirilmesini sağlamak zorunluluğu vardır.

PHP yorumcusu, bugün "Zend-çekirdeği" adı verilen bir "Scripting teknoloji"ine dayalı olarak, Zeev Suraski ile Andi Gutmans tarafından sıfırdan, tamamen yeniden yazılmış bulunuyor. Zend, tıpkı Windows Script Host ve mesela Visual Basic for Applications (VBA) gibi, işletim sistemine kendisi ile uyumlu Script dilleriyle yazılmış programları çalıştırma imkanı kazandıran bir teknolojidir (Zend konusunda ek bilgiyi Internet'te [www.zend.com](http://www.zend.com) adresinde bulabilirsiniz.). Bugün bu dilin resmi adı "PHP: Hypertext Preprocessor" (Hiper -metin ön-işlemcisi) olarak değiştirildi, ve çeşitli Web Server'ların PHP dili anlaması için gerekli yorumlayıcıları yeniden üretildi. 20 Ocak 2000'de İsrail 'de PHP'nin ilk uluslararası konferansı yapıldı ve 4'ncü sürümünün betası ortaya kondu.

PHP ile artık oturup her türlü işlevselliği sahip programlar yazılabilir; yani PHP başlı başına uygulama alanları bulabilecek düzeye ulaşmıştır. Başka bir deyişle PHP ile oyun programından tutun veritabanı yönetimine kadar hemen her türlü programı yazılabilir. Bu tez konusunu ilgilendiren tarafla PHP, Web Sunucu'ya bir takım işler yaptırmak için program yazma dilidir. Tıpkı CGI/Perl veya ASP ya da ColdFusion, Java (Javascript değil!) gibi. PHP kodları, oluşturacağımız HTML sayfalarında HTML etiketlerinin arasında kendi özel ayracı içinde yazılır:

```

<html>
<?PHP
    echo ("Merhaba !");
?>
</html>

```

İster kişisel bilgisayarlarda, ister gerçek Internet ortamında olsun, bir Web Sunucu, ziyaretçinin (İnternet istemci programının, Browser'ın) talep ettiği dosya adının uzatması olarak ".PHP," ".PHP2," ".PHP3," veya ".PHP4" görünce, kendine kendine, "Yine bana iş çıktı" diye düşünür! Çünkü Web Server, uzatması ".htm" veya ".html" olan bir dosyayı, kendi sabit disklerinden birinde bulur ve zahmetsizce, ziyaretçinin bilgisayarına gönderir. Fakat, ".asp," ".pl," ".cfm" gibi, ".PHP" uzatması, Web Sunucu'ya bu sayfayı olduğu gibi, alıp ziyaretçinin Browser'ına göndermek yerine, önce PHP yorumlayıcıyı çağırarak, ona teslim etmesi gerektiğini bildirir. PHP yorumlayıcısı (yani Windows ortamında PHP.exe veya 4'ncü sürümle gelen PHP4isapi.dll, Unix ve türevlerinde çalıştırılabilir PHP dosyası), kendisine teslim edilen bu belgede önce "<?PHP" ve ">" ayraçları arasındaki PHP kodlarını seçerek alır ve gereği ne ise onu yapar. Bu ayraçların içinde kalan kodlar, bizim yapılmasını istediğimiz işlemin komutlarıdır. HTML ile yetinmeyip, PHP kullanılmasının sebebi, bu komutların HTML'in yapamayacağı şeyleri yapmasıdır.

HTML'in yapamayacağı şeylerin başında, Web Sunucusu'nda yapılacak işler vardır. Sözelimi HTML etiketlerini kullanarak, Web Sunucusu'ndaki bir veritabanı dosyasını açıp, içindeki verileri okuyamaz. HTML yoluyla Web Sunucusu'nun sabit disklerindeki düz yazı dosyalarını da okuyamaz ve bu disklere dosya yazamaz. HTML etiketleri ile Web ziyaretçisinden bilgi edinebilir ve bunlar Sunucu'ya yollanabilir. Bu bilgileri işe yarar şekle sokmak için ise Server'da çalışan bir program gerekir. Server'da çalışan program, EXE ve DLL veya JSP (Java Server Pages) gibi bir "gerçek program"; veya Perl, ASP ve PHP gibi Script diliyle yazılmış bir program olabilir.

Web Sunucu, ".PHP" uzatmasını görünce telaşa kapılır, bu dosyayı PHP yorumlayıcısına verir ve o da "<?PHP" ve "?>" ayrıçaları içindeki kodları icra eder. Bu icraatın sonunda ortaya çıkan ürün eğer ziyaretçiye gönderilecekse, HTML etiketleri içinde gönderilir; yani ziyaretçi asla PHP kodlarını göremez. Kimi zaman PHP programının çalışması sonucu ortaya ziyaretçiye gönderilecek bir ürün çıkmaz; elde edilen sonuç ya başka bir programa (örneğin elektronik ileti gönderen bir Sendmail programına) hammadde olarak verilir, ya da Server'da sabit diske yazılır. Fakat her durumda, Web ziyaretçisi yazılan PHP kodlarını göremez; bu Web sayfalarında PHP kodu kullanılması halinde, sayfaların Browser tarafından tanınması veya tanınmaması gibi bir durumun ortaya çıkmaması demektir. Ziyaretçi ne tür Browser kullanırsa kullansın, kendisine Sunucu'dan daima saf HTML kodu geleceği için, sayfaları rahatça görebilecektir. Tabi bir nokta var: Sayfalarda herhangi bir Browser'ın arızalı yorumladığı veya hiç yorumlayamadığı HTML etiketleri bulunabilir; ya da sözcüğü ziyaretçiye giden HTML'in içinde Browser'da çalıştırılacak olan başka Script kodları bulunabilir; ve Browser bu kodları anlamayabilir. Örneğin, HTML sayfasında VBScript kodları kullanılmış ise, Netscape ile siteye bağlanmış olan kişiler bu kodun oluşturmasını istediği etkiyi göremeyeceklerdir; çünkü Netscape VBScript anlamaz. PHP ile yazılan Web programlarının (eski deyimiyle "sayfaların") oluşturacağı HTML belgesinin Browser türleri ve sürümleri ile uyumluluğunu sağlamak yine Web programcısının sorumluluğundadır. Özetlersek, PHP bir CGI programlama dilidir; bu dille, Web Sunucusu ile Web ziyaretçisi arasındaki buluşma noktası olan CGI'da bilgi alışverişi yapılabilir; Sunucu'da bulunan başka programlar çalıştırılabilir ve böylece Web sayfaları HTML'in sınırlamalarının ötesinde hareket ve ziyaretçi ile etkileşme olanağı kazanır.

### 1.3. PHP ve Veritabanı

PHP'nin varlık sebeplerinin başında, ticari Unix ve Microsoft Windows işletim sistemlerinden ayrı, ticari olmayan Açık Sistem ortamlarında çalışan bir Script dili ile bu dille kullanılacak 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 Programcısının emrinde idi. 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 ticari amaçlı evsahibi (Hosting) firmalarının sağladığı bir teknoloji olarak görüldü. 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 belirdi

#### 1.4. PHP Dili

PHP bir Script dilidir; PHP ile yazdığınız "programlar" birer düzyazı dosyasıdır. Dil öğelerini ayrıntılı olarak ele almadan önce bir iki yazım kuralından söz edelim.

PHP yorumlayıcısı, bu "programı" çalıştırmak için dosyanın içinde PHP komutlarını arar. PHP komutları iki şekilde yazılabilir:

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

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

Bunlara PHP komut ayracı denir; birinci türü uzun veya standart ayraç sayılır; ikincisine ike "kısa ayraç" denir. PHP yorumlayıcısının kısa ayraç kullanmasını istemiyorsak, PHP.ini dosyasındaki, `short_open_tag = On`: satırını `short_open_tag = Off`; yapmak yeter.

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 sağlıyorlar. Eğer ASP ile çalışan ve daha sonra PHP'ye geçen bir programcı iseniz, PHP.ini

dosyasını açarak, asp\_tags = Off; satırını short\_open\_tag = On; yapabilirsiniz. Şimdi artık rahatça ASP ayracını PHP yazmak için kullanabilirsiniz.

PHP kodları, oluşturulmasını istenilen sayfanın HTML kodları ile tabir yerinde ise içiçe yazılır. Bunun bir örneği olarak aşağıdaki dosya merhaba adıyla kaydedilirse:

```
<HTML>
<HEAD>
<TITLE>PHP ile Merhaba</TITLE>
</HEAD>
<BODY>
<CENTER>
<B>
<H1>
<?PHP
    echo "Merhaba Dünya!";
?>
</H1>
</B>
</CENTER>
</BODY>
</HTML>
```

Kaydedilen dosyayı çağırıldığında başlığı “PHP ile Merhaba” olan ve sayfa başında “Merhaba Dünya!” yazan bir tarayıcı sayfası belirir. PHP dilinde “echo” veya “print” komutu ekrana yazılması istenilen değerleri yazdırmak için kullanılır. Aynı dosyanın tarayıcı üzerindeki kaynak kodlarına bakıldığında ise hiçbir PHP kodu görülemeyecektir. Çünkü PHP kodundan gelen bilgiyi sunucu yorumlamış ve sonucu taracıya HTML kodları içinde göndermiştir.

PHP programcılığına başlandığında, hangi satırda neye niyet edildiğini anlamak daima mümkün olmayabilir. Biz kendi programımızı satır satır ezberlesek bile, işbirliği yaptığımız diğer Web programcılarının programımızı



anlamalarını sağlamamız gerekir. Bu, yorum satırları ile yapılır. PHP ayrıçaları içinde iki türlü yorum bulunabilir: Çok-satırlı yorumlar ve tek-satırlı yorumlar. Birinci grubu, başına "/\*" ve sonuna "\*/" işaretleri koyarak, ikinci grubu ise sadece başına "//" işareti koyarak belirtiriz:

```
<?PHP
```

```
/*
```

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

```
Bunu PHP yorumcusu asla dikkate almaz
```

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

```
*/
```

```
    print "Merhaba Dünya!";
```

```
// Bu i se PHP'nin tek satırlı yorum bölümü
```

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

```
?>
```

## 1.5. PHP 'nin Yapı Taşları

PHP, bir programlama dili olarak, değişkenler, değişkenlerin değerleriyle bir işlem yapmayı sağlayan işlemciler, işlemcilerle oluşturulan deyimler ve nihayet bunların tümünün toplu kullanımını sağlayan işlemlere (fonksiyonlara) sahiptir. PHP, nesne-yönelimli (object-oriented) bir dil olduğu için, nesne oluşturma imkanına ve bunların kullanılmasını sağlayan yöntemlere de sahiptir. Bütün bu olanakları kullanarak, PHP ile bir veritabanından veri alarak, bunlar HTML etiketlerinin değerleri olarak kullanılabilir; Web sitesini ziyaret eden kişilerden bilgi alınabilir, bu bilgilerle işlemler yapılabilir. PHP'nin çeşitli komutlarını, deyimlerini ve fonksiyonlarını kullanarak, programın çalıştığı Web sunucusunun bulunduğu bilgisayara da çeşitli dosya işlemleri yaptırılabilir.

### 1.5.1. Değişkenler

Programcılıkta işlemler değişkenlerle yapılır. Değişken bir kap gibi düşünülebilir. Sözelimi "Gün," değişkenin adı ise bu değişkenin değeri Pazar,

Pazartesi, Salı, vb., olabilir. Her değişken, türüne göre, ya bir ya da daha fazla değer tutar. Adından da anlaşılacağı gibi değişkenin değeri değişir. Bu değişikliği biz yapabiliriz veya programın kendisi yapabilir.

PHP'de de, bir çok başka bilgisayar programlama dilinde olduğu gibi değişkenlerin içine bir değer konmadan önce tanımlanması mümkündür; fakat gerekli değildir. Değişkenler adının önüne \$ işareti koyarak tanımlanır:

```
$adi;
```

```
$soyadi;
```

```
$123;
```

```
$sevdigiRenk;
```

gibi. Değişkenler, harf, rakam veya alt çizgi (\_) ile başlayabilirler; bu karakterleri içerebilirler; ama içinde boşluk veya diğer işaretler bulunamaz. PHP değişkenleri her türlü değişken tutabilirler: bir değişkenin adının yazılış şekli onun tutabileceği değer niteliğini belirlemez. Dolayısıyla, "\$adi" değişkenin değeri "Talip" te olabilir, "1255" de olabilir.

PHP'de genellikle değişkenler değeri atanarak belirlenir:

```
$adi = "Talip";
```

```
$soyadi = "Kaycı";
```

```
$123 = 123;
```

```
$sevdigiRenk = "mavi";
```

Değişkenler, kullandıkları işleme, taşıdıkları değeri verirler:

```
print $adi;
```

komutu, eğer değeri "Talip" ise "Browser penceresine "Talip" kelimesini yazdırır.

PHP'de özel bir değişkene değişken adı olarak kullanılacak değerler de atanabilir:

```
$adi = "Talip";
```

```
$degisken = "adi";
```

```
print $$degisken;
```

Burada Browser penceresine yine "Talip" kelimesi yazılacaktır; çünkü PHP \$degisken adlı değişkenin "adi" adlı değişkeni tuttuğunu bilecek ve iki Dolar

işaretini görünce, \$degisken'in değerini değil, onun tuttuğu değişkenin değerini yazacaktır. Bu teknik şöyle de kullanılabilir:

```
$degisken = "adi";
$$degisken = "Talip";
print "$adi";
print $$degisken;
print "${$degisken}";
print "${'adi'}";
```

Buradaki dört "print" komutu da Browser'a "Talip" kelimesini yazdıracaktır.

### 1.5.2. Veri Türleri

PHP, değişkenlere, tuttukları değere göre farklı bellek alanı tahsis eder; bu bakımdan verilerin türü etkin bir programcılık açısından önem taşır. Ayrıca PHP, diğer bütün programlama dilleri gibi belirli veri türleri ile belirli işlemleri yapar veya yapamaz. Örneğin, ancak iki sayı arasında aritmetik toplama yapılabilir. İki metin tutan değişken aritmetik olarak toplanamaz; ancak içerdikleri değerler ardarda eklenebilir. Ne var ki, PHP aritmetik toplama mı, yoksa ardarda ekleme mi yapılmak istendiğini bilmek durumunda değildir.

PHP açısından dünyada altı tür değer vardır:

Tamsayı (Integer): 5,124, 9834 gibi

Çift yoğunluklu(Double): 3,567 gibi

Alfanümerik (String): "Talip" gibi

Mantıksal (Boolean): doğru (true)/yanlış (false) gibi

Nesne (Object)

Dizi (Array)

### 1.5.3. Fonksiyon

PHP'nin echo() ve print() gibi, kendi içinde bulunan kullanılmaya hazır bir çok fonksiyonu vardır. Programcılık dilinde bir fonksiyonu kullanmaya "fonksiyon

çağırma" denir. Fonksiyonlar da fonksiyon çağırabilirler. Çağrılan bir fonksiyon ya kendisinden beklenen işi yapar, ya da kendisini çağıran işleme veya fonksiyona bir değer sunar. Buna fonksiyondan dönen değer denir. Ayrıca PHP'de olmayan fonksiyonlarda yaratabilme imkanı vardır (tıpkı excel'de macro yazarak Jülyen Gün hesabı fonksiyonunun yaratılması gibi). Buna PHP'de fonksiyon tanımlama va çağırma denir.

PHP'de fonksiyonlar function komutu ile oluşturulur. Tanımlanan fonksiyon, kendisini göreve çağırarak komuttan, yapacağı işlemde kullanmak üzere değer alacaksa, bu değerlere verilen değişken isimleri fonksiyon adının yanında parantez içinde gösterilir. Fonksiyon birden fazla değer bekleyecekse, bunların değişken adlarının arasına virgül konulur. Fonksiyona ulaştırılan değerlere argüman denir. Fonksiyon, kendisine bir değer ulaştırılmasını beklemese bile içi boş parantez konulması gerekir. Buna göre PHP'de fonksiyon şöyle yazılır:

```
function fonksiyonun_adi (argüman1, argüman2, ... argümanN) {
    fonksiyonun yapacağı işe ilişkin komutlar
}
```

#### 1.5.4. Escape

```
print "İkinci değişkenin adı: \$alfanumerik<br>";
```

Tek veya çift tırnak içine de alsak, PHP, bir değişken adını gördüğü zaman onun yerine o değişkenin tuttuğu değeri yazar. PHP bunu, değişken adının başındaki \$ görerek yaparak. \$ işareti gibi PHP için özel anlamı olan işaretlerin anlamlandırılmasını önlemek ve bu işaretleri düz metin saymasını sağlamak için bu işaretlerin önüne ters-bölü işareti konulur. Buna o karakteri kurtarma veya "ESCaping" denir. PHP'nin anlamlı işaretleri ve bunların ESCape-yazılışı şöyledir:

```
\      Tek tırnak
\"     Çift tırnak
\\     Ters-bölü
\$     Dolar işareti
```

\n	Yeni Satır (New Line)
\r	Satır Başı (Return)
\t	Sekme (Tab) karakteri

### 1.5.5. İşlemciler (Operatörler)

PHP'nin aritmetik, birleşik-atama, birleştirme, karşılaştırma ve mantıksal sınıma operatörleri vardır.

Aritmetik işlemciler:

+	Toplama	6+5	=	11
-	Çıkartma	6-5	=	1
/	Bölme	6/5	=	1.2
*	Çarpma	6*5	=	30
%	Kalan (Modul)	6%5	=	1

Aritmetik işlemleri gerçek sayılarla yapıldığı gibi, tuttuğu değer sayı olan değişkenlerle de yapılabilir.

PHP'de değişken tanımlar ve bunlara değer atarken sayı olan değişkenleri tırnak işareti kullanmadan, alfanümerik değişkenleri ise tırnak içinde yazılmıştı. PHP, tırnak kullanmasanız da bir değişkenin sayı olup olmadığını anlar; daha sonra bu değişkenlerle aritmetik işlem yapabilir. PHP ayrıca rakamla başlayan alfanümerik değişkenleri sayı olarak kullanmaya kalktığınızda, bu değerlerin başındaki bütün rakamları sayı olarak kabul eder. PHP açısından harf veya işaretle başlayan alfanümerik değişkenlerin sayı olarak değeri sıfırdır.

Örnek:

```
$a = 6;
```

```
$b = "22/A"
```

```
print ($a + $b)
```

sonuç olarak 28'i verecektir; veya

```
$a = 6;
```

```
$b = "A/22"
```

```
print ($a + $b)
```

sonuç olarak 6 çıkacaktır.

### 1.5.6. Sabit Değerler

İster Web, ister Web-dışı amaçlı olsun, bir programın bazen başından sonuna kadar değeri değişmeyen değişkene ihtiyacı olabilir. PHP'de sabit değerler, Script boyunca değişmeden kalır.

Sabit değerler, değişkenlerden farklı şekilde oluşturulur. Bunun için PHP'nin define() fonksiyonunu kullanılır. Bu fonksiyonun yazım kuralı şöyledir:

```
define ("SABIT_DEGER", değer);
```

Burada SABIT\_DEGER yerine, tanımlanmak istenilen sabit değere verilecek isim, değer yerine de sabit değer yazılır. Örnek:

```
<?php
    $Dolar_miktar = 125;
    define ( "DOLAR_KURU", 625675);
    $TL_Tutar = $Dolar_miktar * DOLAR_KURU;
    print ($TL_Tutar);
?>
```

Burada tanımlanan DOLAR\_KURU sabit-değeri, tıpkı bir değişken gibi kullanılabilir. Sabit değer ile aynı ismi taşıyan değişken oluşturmak mümkündür.

Tanımlanmış olan bir sabiti yeniden oluşturulamaz. Bir sabit değer oluşturulmuş olup olmadığını defined() fonksiyonu ile kontrol edilir:

```
<?php
    $Dolar_miktar = 125;
    if (defined( "DOLAR_KURU" )) {
        echo ("Sabit değer daha önce tanımlanmıştı.<br>");
    }
    define ( "DOLAR_KURU", 625675);
    $TL_Tutar = $Dolar_miktar * DOLAR_KURU;
    print ($TL_Tutar);
?>
```

Diğer sabitler şunlardır:

`__FILE__` : O anda çalışmakta olan PHP dosyasının adı (kelimenin önünde ve sonunda iki alt-çizgi var);

`__LINE__` : Bu ifadenin yer aldığı dosyadaki satır sayısı (kelimenin önünde ve sonunda iki alt-çizgi var);

`PHP_VERSION`: PHP'nin sürümü;

`PHP_OS`: PHP'nin çalıştığı işletim sistemi;

Bunlara ek olarak, `E_ERROR`, `E_WARNING`, `E_NOTICE` ve `E_PARSE` şeklinde hata durumunda hatanın çeşitli özelliklerini bildiren sabit değerler de kullanılabilir.

PHP hakkında çok daha geniş bilgi `phpinfo()` fonksiyonunu çağırılarak alınabilir.

#### 1.5.7. Tarih Bilgisi

PHP, o andaki zaman bilgisini, saat, dakika, saniye ve salise olarak; tarih bilgisini yıl, ay, gün (sayı veya isim olarak), programın herhangi bir yerinde bildirebilir. Bu bilgiyi Web sunucusunda istenilen anda, muhtemelen sunucunun bulunduğu bilgisayarın sistem saatinden alacak olan PHP, sunucu programında farklı bölgesel ayarlar için gerekli düzenleme yapılmışsa, bu imkandan yararlanarak sunucunun değil, arzu edilen bölgenin saat ve tarihini bildirebilir.

PHP'nin zaman ve tarih belirlemekte kullanabilecek başlıca fonksiyonu `getdate()`; ise şöyle kullanılır.

`getdate()` Tarih ve saat bilgisini alır ve verilen bir isimdeki dizi değişkende kaydeder. Örnek:

```
$saat_tarih = getdate()
```

Bu durumda, `$saat_tarih` dizi değişkeninde sırasıyla şu bilgiler yer alır:

32 saniye

57 dakika

6 saat

30 ayın gün sayısı (1-31)

0 haftanın gün sayısı (1-7)  
 7 ayın sayısı (1-12)  
 2000 yıl  
 211 yılın kaçınıcı günü  
 Sunday günün adı  
 July ayın adı

PHP'nin tarih ve zaman bilgisini kullanılmasını sağlayan getdate() fonksiyonunun yanı sıra, elde edilen bilgileri çok daha esnek bir şekilde biçimlendirmeyi sağlayan date() fonksiyonu da sık kullanılır. Bu fonksiyon çağrıldığı yerde Unix sisteminin zaman-tarih damgasını verir. Bu verinin biçimlendirilmesi için çeşitli parametreler vardır. Bu parametreler, gün adlarının dili ve tarihlerin yazılışı bakımından Web sunucusunun bulunduğu bilgisayardaki bölgesel ayarlara göre sonuç verir.

date() \$tarih date( biçimlendirme\_parametreleri );

Burada biçimlendirme\_parametreleri yerine şunlar yazılabilir:

- a 12 saat esasına dayanan Anglo-Sakson sistemlerinde öğleden önce ("am") veya öğleden sonra ("pm") işaretinin verilmesini sağlar.
- A Aynı işaretlerin büyük harfle yazılmasını sağlar.
- d İki haneli gün sayısı, tek haneli günlerin önüne sıfır konur: "01" - "31"
- D Üç haneli gün adı kısaltması: "Cum"
- F Uzun ay adı: "Ocak"
- h 12 saatlik sistemde saat: "01" - "12"
- H 24 saatlik sistemde saat: "00" - "23"
- g 12 saatlik sistemde tek haneli saatlerin önüne sıfır konmadan saat: "1" - "12"
- G 24 saatlik sistemde tek haneli saatlerin önüne sıfır konmadan saat: "0" - "23"
- i Dakika: "00" - "59"
- j Tek haneli sayıların önüne sıfır konmadan gün sayısı "1" - "31"
- l (küçük L harfi) Uzun gün adı: "Cuma"



- L Artık yıl olup olmadığına ilişkin Boolean (doğru/yanlış) değişken. Artık yıl ise 1, değilse 0.
- m Tek hanelilerin önüne sıfır konarak ay sayısı: "01" - "12"
- n Tek hanelilerin önüne sıfır konmadan ay sayısı: "1" - "12"
- M Kısaltılmış ay adı: "Şub"
- s Saniye: "00" - "59"
- S İngilizce ('ncı anlamına) 2 karakter ek: "th", "nd"
- t Belirtilen ayın gün sayısı; "28" - "31"
- w Haftanın gün sayısı: "0" (Pazar veya Pazartesi) - "6" (Cumartesi veya Pazar)
- Y Dört haneli yıl: "2000"
- y İki haneli yıl "00"
- z Yılın gün sayısı: "0" - "365"

Örnek:

```
print (date ("l dS of F Y h:i:s A"));
```

Browser penceresine İngilizce bölgesel ayarlar yapılmış bir bilgisayara kurulu Web sunucusunda: "Sunday 30th of July 2000 07:51:08 AM" yazdırır.

#### 1.5.8. PHP 'de Mantıksal Denetimler

Bilgisayar programı, belirli değerlere göre kendi izleyeceği yolu kendisi bulmalıdır. Bir Web programı da aynı mantıkla tasarlanır. PHP, belirli durumlarda belirli istikametlere gidecek, yeri geldiğinde geri dönecek, yeri geldiğinde başka bölümlere sıçrayacak bir akış planı tasarlanmasına olanak verir.

Web programlarında, ziyaretçilerin sistemde yaptıkları veya yapmak istediklerini bildirdikleri işe göre sayfalar, bu işlere göre sayfa içerikleri sunulur. Web programları ziyaretçinin hareket tarzına ve amacına uygun olarak ilerler. Bir programın akış planını kontrol eden sağlayan unsurlar çoğu zaman programın kendi kendine karar vermesini sağlayan araçlardır. PHP'de program denetim unsurları, programın gidiş yönünün değiştirilmesini veya olduğu yerde durmasını ve beklemesini sağlar.

### 1.5.8.1. If Deyimi

If deyimi birçok programlama dilinde varolan mantıksal kıyaslama işlemini gerçekleştirir. Günlük hayatta kullanılan birçok kıyaslama gibi, örneğin bir yolda karşıdan karşıya geçerken, sola ve sağa bakarak gelen aracın olup olmadığını kontrol eder ve eğer gelen araç yoksa karşıdan karşıya geçme işlemi gerçekleştirilir. Eğer araç geliyorsa karşıdan karşıya geçme işlemi beklemeye alınır. İşte if deyimi, bilgisayarda tam olarak buna benzer kıyaslamalar yapar. PHP 'de if deyimi şöyle yazılır:

```
if ( koşullar ) {
koşullar doğru ise yapılacak işlere ilişkin komutlar
}
elseif (diğer koşullar) {
diğer koşullar doğru ise yapılacak işlere ilişkin komutlar
}
else {
diğer her durumda yapılacak işlere ilişkin komutlar
}
```

PHP, if ifadesinin doğru olması halinde, ifadeye ait ilk süslü parantezin içindeki komutları icra eder; bu şartlar doğru değilse, elseif deyimi ile ileri sürülen diğer bir grup şartın doğru olup olmadığı sınıranır. Bu şartlar yerine geliyorsa, PHP, elseif bölümündeki işleri icra eder. Nihayet diğer her hangi bir durumda icra edilmesi istenilen komutlar else bölümünde belirtilir. PHP, if deyimi doğru değilse, deyime ait süslü parantezin içindeki işleri yapmadan yoluna devam eder; varsa elseif deyiminin şartlarını sınar; elseif bölümü yok fakat else bölümü varsa bu bölümdeki işleri icra eder. Deyimde else bölümü de yoksa, PHP yoluna if deyiminden sonraki ilk adımdan devam eder.

### 1.5.8.2. Switch Deyimi

PHP'de program akışını yönlendirmekte kullanılan bir diğer unsur, switch deyimidir. Adı anahtar anlamına gelen switch deyimini, verilen bir değişkenin değerinin sıralanan koşullardan hangisine uygun olduğunu sınırlar ve o koşula ilişkin komutları icra eder. PHP'nin yaptığı işi bitirdikten sonra switch deyiminin dışına çıkmasını sağlayan, break komutu vardır:

```
switch ( değişken ) {
    case KOŞUL-1 ;
        Koşul-1 doğru ise yapılacak işlere ilişkin komutlar
    break;
    case KOŞUL-2 ;
        Koşul-2 doğru ise yapılacak işlere ilişkin komutlar
    break;
    case KOŞUL-3 ;
        Koşul-3 doğru ise yapılacak işlere ilişkin komutlar
    break;
    case KOŞUL-4 ;
        Koşul-4 doğru ise yapılacak işlere ilişkin komutlar
    break;
    .....
    .....
    default:
        diğer her durumda yapılacak işlere ilişkin komutlar
}
```

### 1.5.9. Döngüler

Yukarıdaki iki denetim ögesi de programın ileri doğru akışını bozmayan araçlardır. Oysa bazen programın bir koşul gerçekleşinceye kadar yerinde sayması ve koşul gerçekleşmedikçe ilerlememesi gerekebilir. Kimi zaman da programın bir işi bitirmeden ileri gitmemesini biz bildiririz. Program akışının

bu tür denetimini döngü deyimleri sağlar. PHP'de iki tür döngü vardır: bir durum devam ettiği veya etmediği sürece yapılan ve kaç kere döneceği bilinemeyen while döngüleri ve tekrar sayısı belirli for döngüleri.

#### 1.5.9.1. While döngüsü

Bir değişkenin içeriğinin belirli bir şartı karşılaması veya karşılamaması halinde icra edilir. Burada dikkat edilmesi gereken nokta, programın icrası sırasında değişkenin içeriğinin veya koşulun değişmesinin sağlanmasıdır. Aksi takdirde program sonsuz döngüye girer ve muhtemelen çöker. While döngüsü şöyle yazılır:

```
while (koşul) {
    Koşul doğru ise yapılacak işlere ilişkin komutlar
}
```

#### 1.5.9.2.For döngüsü

PHP'de Web programları, döngünün belirli bir sayıda olmasını ve mesela bu sayının istenilen basamaklarda artmasını gerektiriyorsa, döngü for deyiimiyle kurulabilir. Ayrıca for deyiimi sayaç gibi şartın yerine gelmesini sağlayacak arttırma veya eksiltme işlemlerini kendisi yapacağı için, sonsuz döngüye girme tehlikesi de hemen hemen yoktur. Bu döngü şöyle yazılır:

```
for ( $yeni_degisken atama ; koşul ; artım miktarı ) {
    Koşul doğru ise yapılacak işlere ilişkin komutlar
}
```

for deyiminin ilk ögesi, döngü sayacı olarak kullanılacak bir değişken oluşturulması ve buna değer atanmasına ilişkindir; sonra bir noktalı virgül ile bu sayacın sınırı olan koşul belirtilir. Yine noktalı virgül ile ayırarak, sayaç değişkeninin artım miktarı verilir.

### 1.5.10. Dosya İşlemleri

PHP kullanılarak site geliştirirken, belli fonksiyonlar sitede her PHP dosyasında kullanılıyor olabilir. Her PHP dosyasında tekrar tekrar o fonksiyonun tanımlanması yerine, bu fonksiyonun bir ortak dosyada tutularak diğer PHP dosyaları tarafından çağırılması tercih edilebilir. Bu sayede, bir fonksiyonda yapılacak değişiklik için, tüm PHP dosyalarının açılarak gereken düzeltmelerin yapılması gerekliliği ortadan kalkar. PHP’de ortak dosya kullanımı için include veya require komutu kullanılır. Bu komutlar sayesinde, PHP dosyalarına dışarıdan fonksiyon kütüphanesi dosyaları yada başka ortak bilgiler içeren dosyalar katılabilir. Kullanım Şekli : Include(“dosya yeri ve adı”); şeklindedir. Örneğin: Localhost/astronomi/astro klasöründe bulunan “index.php” dosyasının içine, daha önceden tarih bilgisi almak için hazırlanmış olan, localhost /astronomi /fonksiyonlar/ klasöründe bulunan “tarih.txt” dosyasını koymak için: Include(“../fonksiyonlar/tarih.txt”); yazılması yeterlidir.

### 1.5.11. Metin Dosyası Okuma ve Yazma

PHP’de herhangi bir dosya üzerinde işlem yapmadan önce, üzerinde okuma yada yazma işlemi yapılacak dosyayı bir tanımlayıcıyla eşleştirmek gerekir. Bu eşleştirme işlemi fopen() komutuyla yapılır. Bu komuta yerel dizindeki bir dosyanın adı verilebileceği gibi, web üzerindeki bir adres de verilebilir. Örnek: \$dosya = fopen(“dosya.txt”,”r”);

yada

\$dosya = fopen(“http://www.ankara.edu.tr/science/astronomy/index.php”,”r”);

fopen() komutunun parametreleri:

- 1- r : Sadece okuma işlemi yapar.
- 2- r+ : Hem yazma hem okuma işlemine izin verir.
- 3- w : Sadece yazma işlemi yapar. Eğer sistemde belirtilen adda bir dosya varsa silinerek yenisi oluşturulur.

- 4- w+ : Hem yazma hem okuma işlemi yapar. Eğer sistemde belirtilen adda bir dosya varsa silinerek yenisi oluşturulur.

#### 1.5.12. PHP 'de Metin Fonksiyonları

PHP'nin metin olarak gördüğü değişkenlere alfanümerik (String) türü değişkenler denir. PHP'nin alfanümerik fonksiyonları, bu tür değişkenlerin değerlerinin içinden bir bölümü alma, atma veya değiştirme imkanı sağlar.

`substr()`

Bir alfanümerik değişkenin değerinin veya bir metnin tanımladığınız bölümünü verir. İkisi zorunlu, biri seçmeli üç parametre ile kullanılır. Yazılışı:

`substr($degisken, başla, [boyut] );`

`trim()`

Bir alfanümerik değişkenin değerinin baş ve son tarafındaki boşlukları atar.

Yazılışı:

`trim($degisken);`

`chr()`

Parametre olarak ASCII değerini belirttiğimiz karakteri sağlar. Örnek:

`echo (chr(34));`

`ord()`

Parametre olarak yazdığımız karakterin ASCII değerini sağlar. Örnek:

`echo (ord("A"));`

`strlen()`

Bir alfanümerik değişkenin değerinin kaç karakter içerdiğini bildirir. Yazılışı:

`strlen($degisken);`

`strtolower()`

Bir alfanümerik değişkenin içindeki tüm harfleri küçük yapmak için kullanılır.

`strtolower($degisken);`

`strtoupper()`

Bir alfanümerik değişkenin içindeki tüm harfleri büyük yapmak için kullanılır.

`strtoupper($degisken);`

### 1.5.13. PHP'de Matematiksel Fonksiyonlar

Genel olarak çok fazla başvurulmasada PHP'de ileri matematiksel fonksiyon desteği vardır. Bunlardan bazıları aşağıda belirtilmiştir:

#### Number\_format()

Bu fonksiyon, üç farklı parametre olarak sayıyı ekrana dökmekte seçenekler sağlar. Örnek:

```
$sayi = 12345678900;
```

```
echo number_format($sayi,0,"",".");
```

koduyla \$sayi değişkeniyle gönderilen sayının ondalık hane sayısını 0 olarak ayarlayıp, binlik grupta kullanılacak karakteri de nokta olarak belirtiliyor. Bu sayede, sayı ekrana 12.345.678.900 şeklinde formatlanmış olarak basılıyor.

#### Floor() ve ceil()

Bu iki fonksiyonda sayı yuvarlama işlemi yapılır. Ceil() fonksiyonu sayıyı yukarıya yuvarlarken floor() ise aşağı yuvarlar.

#### Max() ve Min()

Bu iki fonksiyon verilen sayılar arasında en büyük ve en küçük olan değerleri bulurlar. Max(sayi1,sayi2,sayi3,sayi4) sayıları arasından en büyük olan değeri alır. Min(sayi1,sayi2,sayi3,sayi4) ise en küçük değeri alacaktır.

#### rand()

Bu fonksiyon verilen iki değer arasında rastgele değer üretmek için kullanılır. Rand(1,100); bu durumda fonksiyon 1 ile 100 arasında rastgele bir sayı üretecektir.

## BÖLÜM II

### 2. PHP İLE VERİTABANI (MySQL)

#### 2.1. Niçin MySQL ?

PHP 'nin yaygınlaşmasındaki kuşkusuz en büyük yardımcılarından biride MySQL oldu. Windows ve Unix versiyonları ücretsiz olarak dağıtılan MySQL'in PHP ile beraber gelişmesi sayesinde hem PHP hem de MySQL karlı çıktılar. Gelişmiş veritabanı sistemlerinde görülen ilişkisel veritabanı mantığıyla geliştirilen MySQL, çoklu bağlantı desteği ve performansı ile uzun süreden beri İnternet'te adını duyurmayı başarmıştı. GPL lisansı ile kaynak kodları açıklanan MySQL'in PHP ile stratejik işbirliği sayesinde, PHP 4 ve MySQL birlikte diğer veritabanı ve dil ikililerine karşı konulmaz performans üstünlüğü sağlıyor.

PHP'nin MySQL ve Apache'yle birlikte derlenerek kullanıldığı sistemlerde veritabanı ve web sunucusu arasında kurulan kalıcı bağlantılar sayesinde web sunucusu ve veritabanı tek bir yazılım gibi çalışabiliyor. Bu sayede çalışan iki ayrı yazılıma göre %40'lere varan performans üstünlüğü sağlamış oluyor.

On senelik bir mazisi olmayan Linux'ün masaüstü ve sunucu pazarında yükselen grafiği, açık kaynak olarak geliştirilmesinden kaynaklanıyordu. Apache web sunucusunun da dünya web sunucusu pazarında %60'ların üzerinde tercih edilmesi Apache'nin açık kaynak olarak üretilen ve dünyaca güvenilen en güçlü yazılım olduğunu kullanıcılara ispatlıyor. MySQL, özellikle PHP birlikteliğiyle dünyanın son zamanlarda adından oldukça sık söz ettiren en popüler veritabanı yazılımı. Dünyada kullanıcıların en çok tercih ettiği sitelerden biri olan Yahoo'nun bazı bölümlerinin altyapısında MySQL kullanılıyor olması bile, MySQL'i tanımak ve kullanmak için yeterince güçlü bir gerekçe.



NASA'nın 2001 ocak ayı başında yaptığı açıklamaya göre, Marshal Hava Üssü'nde artık Oracle yerine MySql kullanılıyor olması, bu gelişmeler arasında en kuvvetli olanı. Oracle gibi, dünyada çok kullanılan ve oldukça güçlü bir altyapıya sahip veritabanı karşısında, tüm kodları açık olarak geliştirilen MySql , dünyanın belkide en hassa bilgilerinin depolandığı NASA gibi bir uygulama alanında güvenle kullanılıyor.

Sybase veritabanının kendi geliştirme dili olan PowerDynamo'yla performansı ve Microsoft'un SQL server veritabanıyla ASP dilinin performansı gibi, MySql de PHP ile birlikte her türlü uygulamada üstün performans sağlıyor. Özellikle günümüzde her türlü uygulamanın web'e taşınmasının sonucunda, birçok yazılım firması stok takibinden, muhasebe uygulamalarına kadar birçok yazılım ürünlerini artık web üzerinde çalışır hale getirmeye çalışıyor. MySql'in web uygulamalarına yönelik tasarlanmış olması, onun genel uygulama amaçlarına hitap eden klasik veritabanları arasından üstün performansla sıyrılmasını sağlıyor."Otaner(2001)".

## 2.2. MySql

SQL, (Structured Query Language, Yapısal Sorgu Dili) veritabanı endüstrisinin standart dilidir. Fakat HTML konusunda farklı firmaların sadece kendi Browserlarının tanıdığı türden etiketler geliştirmeleri gibi, SQL alanında da farklı firmaların ekleri ve dilde yaptıkları değişiklikler vardır

MySQL, sunucuda çalışır ve arzu eden programa, bildireceği (ve erişim hakkı bulunan) veritabanı dosyasından veri çekerek, sunar. Bir veritabanında farklı sayıda tablolar olabilir. Bir veritabanı dosyasında yer alan tablolarda, sütunlar ve satırlar vardır: bunların kesiştikleri yerlere veriler yazılır. Bir SQL veritabanı dosyasında her bir kutunun ne tür veri kabul edeceği önceden belirlenir; dolayısıyla türü sayı olan verinin yazılacağı kutuya, alfanümerik değer yazılmaz. MySQL, sayısal verilerin türünü INT, belirli bir sınıra kadar

değişen sayıda karakterden oluşacak verilerin türünü VARCHAR komutuyla belirler.

### 2.3. MySql veri Türleri ve Temel Komutlar

MySQL'de bir çok veri türü oluşturulabilir. Ancak Web programları açılışından önemli olan bir kaç ve özellikleri şöyle saralanabilir:

INT Tamsayı: -2147483648'den 2147483647 kadar değişen diziye "signed" (işaretili), 0'dan 4294967295'e kadar değişenine "unsigned" (işaretsiz) denir.

VARCHAR(n): n sayısını geçmemek şartıyla değişen boyutta karakter olabilir.

CHAR(n): Kesinlikle n sayısı kadar karakter olabilir.

TEXT : En fazla 65535( $2^{16}-1$ ) karakter alabilen metin alanı.

MEDIUMTEXT: En fazla 16777215( $2^{24}-1$ ) karakter alabilen metin alanı.

DATE : 1000-01-01'den 9999-12-31'e kadar değişebilen tarih alanı.

TIMESTAMP : 1 Ocak 1970'den 18 Ocak 2038'e kadar olan ve Yıl+Ay+Gün+Saat+Dakika+Saniye biçimindeki zaman bilgisi.

MySQL'de bir tablo oluşturmak için gerekli CREATE TABLE komutu şöyle kullanılır:

```
CREATE TABLE uyeler (adi VARCHAR(30), soyadi VARCHAR(30), üye_no INT ) ;
```

Bu komutla, "uyeler" isimli üç sütunlu bir tablo oluşturulur: birinci ve ikinci sütunlarda en fazla 30, karakterlik değişen boyutta alfanümerik değerler yer alırken, üçüncü sütunda sadece tam sayı olan değerler bulunabilir. Bu komutla oluşturulan tabloya INSERT INTO komutuyla veri girilebilir:

```
INSERT INTO uyeler (adi, soyadi, üye_no) VALUES ('Talip','Kayıcı','1234')
```

Bir tablonun oluşturulması ile içine veri yerleştirilmesi komutları ayrı ayrı zamanlarda, ayrı işlemler olarak yapılabileceği gibi, toplu bir metin halinde, otomatik olarak da yapılabilir.

MySQL veritabanından bilgi edinmek için SELECT komutunu kullanılır:  
 SELECT \* FROM uyeler ;

Bu, MySQL'e, uyeler adlı tablodaki bütün değerlerin okunmasını bildirir. Buradaki "\*" işareti, "bütün sütunlardaki bütün değerler" anlamına gelir. Yukarıda oluşturulan tablonun sadece "adi" ve "soyadi" sütunlarındaki bilgileri alınmak istenseydi, bu komut şöyle olurdu:

```
SELECT adi soyadi FROM uyeler ;
```

Bir veritabanındaki bilgilerin yenileriyle değiştirilmesini, yani veritabanı dosyasının güncelleştirilmesini UPDATE komutu sağlar. Bu komut kullanarak veritabanındaki bazı kutucukların içindeki bilgiler değiştirebilir. Veritabanı dosyalarını güncelleştirme zorunluğu bulunması ise veritabanı tasarımının çok önemli olduğunu gösterir. Örneğin:

```
UPDATE uyeler SET adi = "Mehmet" ;
```

Bu komut, veritabanındaki bütün satırlarda, birinci sütundaki değerleri "Mehmet" olarak değiştirmekle sonuçlanırdı. Amaç bu ise, sorun değil; ancak çoğu kez MySQL'e hangi satırda (veritabanı tekniğindeki terimle söylersek, hangi kayıtlarda) değişiklik yapılacağını daha ayrıntılı söylemek gerekir. Veritabanı dosyamızı oluştururken, her kaydın diğer kayıtlarda olmayan (unique) bir sütun (bunu da veritabanı tekniğindeki terimle söylersek. alan) bulunmalıdır, ki MySQL'e yapılacak değişikliğin tam yerini söyleyelim. Örneğin

```
UPDATE uyeler SET adi = "Mehmet" WHERE uye_no = 1234;
```

MySQL bu komutu alınca sadece üye numarası 1234 olan kişinin (yani uye\_no alanındaki değer 1234 olan kaydın) "adi" alanındaki değeri silecek ve yerine verdiğimiz yeni değeri yazacaktır. Böyle birincil alanı bulunan, iyi düşünülmüş bir veritabanından seçim yapmak da kolay olur. Örneğin:

```
SELECT adi soyadi FROM uyeler WHERE uye_no >= 123;
```

deyimi ile tabloda bulunan kayıtlardan sadece üye numarası 123'den büyük olanları seçer.

Bir MySQL veritabanındaki kaydı silmek için DELETE komutu kullanılır:

```
DELETE FROM uyeler WHERE uye_no = 1234;
```

Veritabanında sadece bir kayıta üye numarası 1234 olacağı için bu komutla sadece bir satır silinecektir. Bu komut, üyelik kaydını yenilememiş kişilerin tümünü silmek için de kullanılabilir. Veritabanında üyelik kaydının yenilenme tarihini gösteren bir alan bulunduğunu varsayalım:

```
DELETE FROM uyeler WHERE yenileme_tarihi < 2000-01-31;
```

Bu komutla, üyeliğini yenileme tarihi 31 Ocak 2000'den eski olan bütün üyelerin kaydı veritabanından silinmiş olur.

#### 2.4. 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 belirter. "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 bağın tanıtıcı işareti olacaktır. Bu bağlantı kurulduktan sonra, açtığımız 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 durumunu sınanarak anlaşılabilir. Örneğin:

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

```
if ( ! $veri_yolu) die ("MySQL ile veri bağlantısı kurulamıyor!");
```

İkinci satırdaki `if` deyimi, `$veri_yolu` değişkeninin değer içerip içermediğine bakıyor ve değişkende bir değer yoksa, bağlantı kurma girişini durdurarak, ziyaretçiye hata mesajı gönderiyor.

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( "veritabanının_adi" , $veri_yolu ) or die ("Veritabanı açıla-  
mıyor!".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 Browser 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.

## BÖLÜM III

### 3. AÇIKLAMALI SİTE İÇERİĞİ

#### 3.1. GİRİŞ

Astronomi bölümü web sitesinde, PHP ve MySQL ikilisi kullanılarak yapısal güncelleme yapılmıştır. Buradaki yapısal güncellemeden kasıt, mevcut sitenin iç yapısı hemen hemen sabit kalmak sureti ile sitenin omurgası sayılabilecek ana sayfanın yapısındaki değişikliklerdir. Bu yapısal değişiklik temel anlamda ana sayfanın veritabanına bağlanması ile gerçekleştirilmiştir. Ana sayfanın veritabanı bağlantısı sayesinde yapı, değişkenler zinciri haline getirilmiştir.

Ortalama bir web sayfasında yapı; resimler, animasyonlar, linkler ve yazılardan oluşmaktadır. İşte bu çalışmada yapılan şey, sayfa yapısındaki herşeyin bir değişkenmiş gibi (ki öyle) düşünülmesi ve daha sonra bunların veri olarak kabul edilip veritabanına bağlanması şeklindedir. Bu düşünce çerçevesinde ana sayfa görüntüsü, bir veritabanı bağlantısı sonucu, veritabanından veri çekip bu verileri gerekli yerlere yerleştirmek sureti ile elde edilmiştir. Bu işlemin detaylı açıklaması ilerleyen bölümlerde yer alacaktır.

Bu şekildeki bir ana sayfa yapısının site sahibine yararı, kısa sürede ve zahmetsiz bir şekilde site yapısının güncellenmesidir. Buna iyi bir örnek olarak; ilk bilgisayar algoritmasının öğretildiği “basic”, “fortran” ve “c” gibi dillerde hep bir sayı oyununa değinilir. O oyunun amacı bilgisayarın hafızada bir sayı tutması ve kullanıcının sayıyı bulmasıdır. Burada önemli olan kullanıcının kaç denemede sayıyı bulması gerektiğinin belirlenmesi ve eğer bulamazsa kullanıcıya oyunu kaybettiği mesajının verilmesidir. Eğer siz kullanıcının deneme sayısını programın içine sabit değer olarak yazarsanız, oyunu zorlaştırmak veya kolaylaştırmak için her seferinde oyunun kodlarına girip deneme sayısını de - ğiştirmeniz gerekir. Oysaki deneme sayısını bir değişkene bağlar ve bu değişkenin değerini oyuna başlamadan bir soruya cevap şeklinde belirlerseniz, artık

oyunun kodlarına girip deęişiklik yapmanıza gerek kalmaz. Aslında bu çalışmada yapılan şeyin özü budur. Dinamik bir internet sayfası yapısında bu oyuna benzer onlarca program çalışır. Doğal olarak her program bazı sabit veya deęişkenler içerir. Bir programı kullanışlı kılan şey ise güncellenebilme yani deęişkenlerin deęerlerini deęiştirebilme özelliğidir. Sonuç olarak bu çalışmada yapılan, yüzlerce deęişkenin veritabanı yapısına bağlanarak site sahibinin sadece veri tabanına ulaşmak kaydı ile deęişken deęerlerini deęiştirebilmesidir. Bu sayede her güncellemede anasayfa kodlarının içerisinde deęişken arayıp o deęişkenleri deęiştirme zahmetinden arınılmış olur.

Güncelleme işlemlerinin hepsi site içerisinde “localhost (ana sayfanın url’si)/admyn “ klasörü altında gerçekleştirilebilmektedir. Bu klasöre girebilmek için şifreyi bilmek gerekir. Bu işlemlerle ilgili ayrıntılı bilgi admin bölümünde verilecektir.

Site ile ilgili ayrıntılı açıklamalar site yapısı başlığı altında, çeşitli resim ve çizelgelerle anlatılmaya çalışılacaktır.

### 3.2.Site Yapısı

#### 3.2.1. Anasayfa (Giriş sayfası)

Anasayfa şekil 1 de görünen yapıdadır. Site içeriğinin mümkün olduğu kadar açık bir şekilde ana sayfada görüntülenmesi amaçlanmış ve dolayısıyla sayfa içi ana linklerin altlink yapılarında ana sayfaya taşınmıştır. Bu sayede kullanıcının ulaşmak istediği bilgiye doğrudan yönlendirilmesi amaçlanmıştır. Sayfada görünen hemen hemen herşey veritabanında kayıtlı verilerdir . PHP - nin MySql ile etkileşim yeteneği kullanılarak, veritabanındaki verilerin sunucuda derlendikten sonra, HTML kaynak kodları halinde tarayıcıya sunulup sayfanın son şekliyle görüntülenmesi sağlanmıştır.

Anasayfa taban üstünde üç tablodan oluşmaktadır. Bunlardan ikisi altmesaj tabloları olup sayfanın en altında beyaz renktedirler (isteğe bağlı pasif veya aktif edilebilir). Diğeri ise Bütün içeriğin görüntülediği tablodur (ana tablo). Bu tablo: Logo'nun yer aldığı resimdeki en üst bölüm (birinci tablo); sayfanın sol tarafındaki linklerin ve link başlıklarının yer aldığı bölüm (ikinci tablo); site içi arama motoru ve sayfa ortasındaki resmin yer aldığı bölüm (üçüncü tablo) ve sayfanın sağındaki link ve link başlıklarının yer aldığı bölüm (dördüncü tablo) olmak üzere toplam dört bölümden oluşmaktadır. Bunlara ek olarak bölüm tarafından belirlenen tarihlerde ana tablo ile alt mesaj tabloları arasında 150 pixel genişliğinde duyuru tablosu yer alacaktır. Bu tablo içerisinde bölüm duyuruları dikey doğrultuda akar şekilde görüntülenecektir. Belirlenen tarihler dışında bu tablo kendini otomatik olarak yok edecektir.



şekil 1. Ana sayfanın tarayıcı (browser /MS.I.E. ) üzerindeki görüntüsü.



Tablolarla ilgili detaylı bilgi tablo numaraları kullanılarak ( birinci,ikinci,ana tablo gibi) aşağıda anlatılmaktadır.

### 3.2.2. Anasayfada Veritabanından Çağırılan Veriler ve Tabloları

Bu bölümde anlatacağımız konunun daha iyi anlaşılması için parantez içerisindeki bilgilerin veritabanındaki alan isimleri olduğu unutulmamalıdır..

Önceki bilgiler dahilinde anasayfa yapısını oluşturmak için kullanılan veritabanı tablo ve alanları aşağıda açıklanmıştır:

Tüm sistemde “Astronomy” adında tek bir veritabanı kullanılmıştır. Anasayfayı oluşturmak için kullanılan veri tabloları : “Tasarım”, “duyuru”, “alt\_mesaj1”, “alt\_mesaj2”, ”meta”, “table1”, ”table2”, ”table3”, “table4”, “basliklar”, “alan1”, “alan2” ve “alan3” olmak üzere toplam 13 tablo kullanılmıştır.

Tabloların içerdiği bilgi listesi:

#### I- Tasarım

Anasayfa tasarımı için kullanılan tablodur. Bu tablo 9 bilgi alanından oluşmaktadır. Bu bilgi alanları; anasayfanın arkaplan rengi(bodycolor), tablonun rengi(tablecolor), tablonun eni ve boyu(table width ,tableheight), tablonun hücre boşluğu(tablecellspac) ve site başlığının(title) yer aldığı bilgileri içermektedir.

#### II- Table 1

Anatablonun birinci bölümünün tasarımı için kullanılan tablodur. Bu tablo 5 bilgi alanından oluşmaktadır. Bu bilgi alanları; birinci tablodaki logo(logo), tablonun arkaplan rengi(color) ve tablonun hücre(cellspac) boşluğunun yer aldığı bilgileri içermektedir.

#### III- Table 2

Anatablonun ikinci bölümünün tasarımı için kullanılan tablodur. Bu tablo 4 bilgi alanından oluşmaktadır. Bu bilgi alanları; ikinci tablonun rengi (color), ikinci tablonun yatay ve dikey ortalama(valign,align) değerlerinin yer aldığı bilgileri içermektedir.

#### IV- Table 3

Anatablonun üçüncü bölümünün tasarımı için kullanılan tablodur. Bu tablo 4 bilgi alanından oluşmaktadır. Bu bilgi alanları; üçüncü tablonun rengi (color), üçüncü tablonun yatay ve dikey ortalama(aligned,valigned) değerlerinin yer aldığı bilgileri içermektedir.

#### V- Table 4

Anatablonun dördüncü bölümünün tasarımı için kullanılan tablodur. Bu tablo 4 bilgi alanından oluşmaktadır. Bu bilgi alanları; dördüncü tablonun rengi(color), dördüncü tablonun yatay ve dikey ortalama(aligned,valigned) değerlerinin yer aldığı bilgileri içermektedir.

#### VI- Meta

Anasayfa meta etiket düzeni için kullanılan tablodur. Bu tablo 4 bilgi alanından oluşmaktadır. Bu bilgi alanları; “meta keywords”(keywords), “meta description”(description), ve “meta refresh”(refresh) değerlerinin yer aldığı bilgileri içermektedir.

#### VII - Basliklar

Anatablonun ikinci ve dördüncü bölümünde yer alan link başlıklarının çağırıldığı tablodur. Bu tablo 4 bilgi alanından oluşmaktadır. Bu bilgi alanları; başlığın yer aldığı alan(baslik), başlığın hangi bölümde yayınlanması gerek-

tiğini gösteren alan(baslikid) ve başlık renginin(reng) yer aldığı bilgileri içermektedir.

#### VIII- Alan 1

Anatablonun ikinci bölümünde yer alan linklerin çağırıldığı tablodur. Bu tablo 6 bilgi alanından oluşmaktadır. Bu bilgi alanları; linkin site içi url'si (link), linkin hangi başlık altında yayınlanması gerektiğini gösteren alan (başlık), linkin görünmesi gereken adı(linkadi), linkin rengi(linkrenk) ve o linke ait sayfanın arama motoruna kaydı için gerekli kelimelerin(keywords) yer aldığı bilgileri içermektedir.

#### IX- Alan 2

Anatablonun üçüncü bölümünde yer alan sayfanın çağırıldığı tablodur. Bu tablo 2 bilgi alanından oluşur. Bu bilgi alanları; ana tablonun üçüncü bölümünün içerdiği sayfa veya yazı değerinin site içi url'sinin(url) yer aldığı bilgileri içermektedir.

#### X- Alan 3

Anatablonun dördüncü bölümünde yer alan linklerin çağırıldığı tablodur. Bu tablo 6 bilgi alanından oluşmaktadır. Bu bilgi alanları; linkin site içi url'si (link), linkin hangi başlık altında yayınlanması gerektiğini gösteren alan (başlık), linkin görünmesi gereken adı(linkadi), linkin rengi(linkrenk) ve o linke ait sayfanın arama motoruna kaydı için gerekli kelimelerin(keywords) yer aldığı bilgileri içermektedir.

#### XI- Alt\_mesaj 1

Anasayfanın en altındaki iki tablodan birincisinin çağırıldığı tablodur. Bu tablo 5 bilgi alanından oluşmaktadır. Bu bilgi alanları; tablonun rengi (tablecolor),

yazının rengi(fontcolor), tablo içerisindeki yazı veya şeklin HTML kodları (mesaj) ve tablonun yayınlanıp yayınlanmayacağını belirten varlık değerinin (varlik) yer aldığı bilgileri içermektedir.

## XII- Alt\_mesaj 2

Anasayfanın en altındaki iki tablodan ikincisinin çağırıldığı tablodur. Bu tablo 5 bilgi alanından oluşmaktadır. Bu bilgi alanları; tablonun rengi (tablecolor), yazının rengi(fontcolor), tablo içerisindeki yazı veya şeklin HTML kodları (mesaj) ve tablonun yayınlanıp yayınlanmayacağını belirten varlık değerinin (varlik) yer aldığı bilgileri içermektedir.

## XIII- Duyuru

Anasayfada, anatablo ile alt mesaj tabloları arasında kalan bölgede site duyurularının çağırıldığı tablodur. Bu tablo 6 bilgi alanından oluşmaktadır. Bu bilgi alanları; duyurunun içerdiği bilgi(ilan), duyurunun yayımlandığı tarih (girstarih), duyurunun yayımdan kalkacağı tarih(cikistarih), duyuru tablosunun rengi(tablecolor) ve duyurunun yazı renginin(fontcolor) yer aldığı bilgileri içermektedir.

Anlatıldığı gibi ana sayfa yapısı 13 veri tablosu ve 64 veri alanından oluşmaktadır. Bu alanların depolayacağı bilgi sınırı sistemin kurulu olduğu sunucunun yeteneği ile ilgili olup MySql tarafından herhangi bir sınırlama getirilmemiştir.

Site içerisinde yer alan veritabanı bağlantılı diğer sayfalar Admin İçin Kullanma Kılavuzu Bölümünde anlatılacaktır.

## BÖLÜM IV

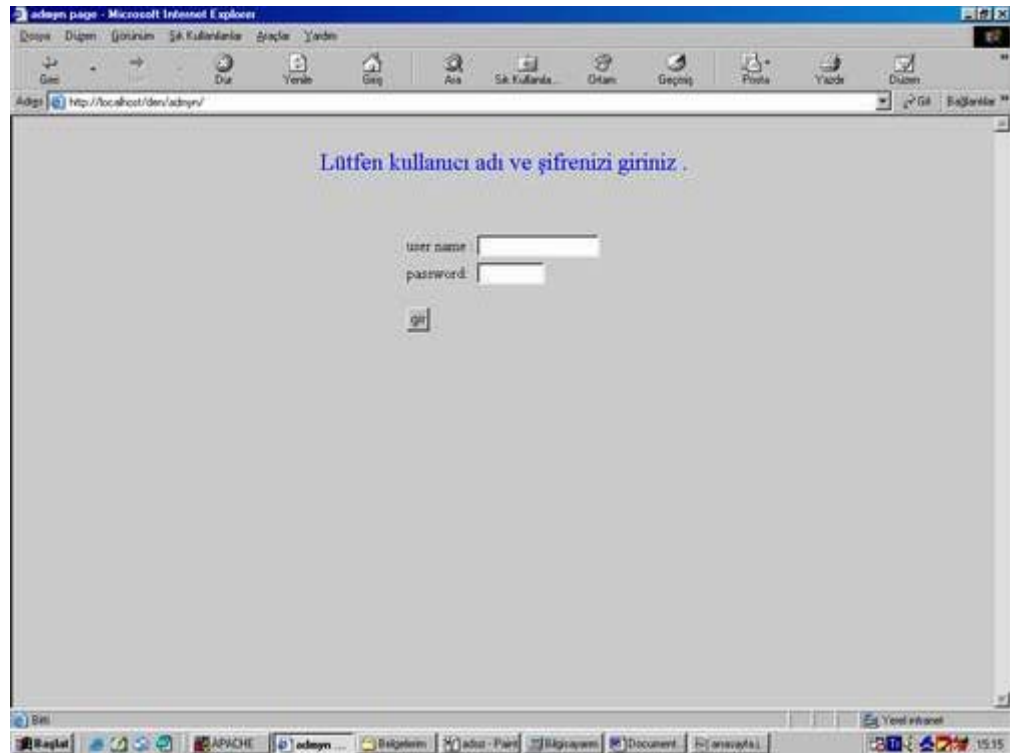
### 4. ADMİN İÇİN KULLANMA KILAVUZU

#### 4.1. Admin Klasör İçeriği

Site içerisinde veritabanıyla etkileşim halinde bulunan bütün sayfalar, yine site içi bir klasör olan Admyn klasöründen kontrol edilebilmektedir. Bu kontrol olanakları; yeni veri girişi, varolan veriyi değiştirme veya silme şeklindedir. Bütün bu işlemleri gerçekleştirmek için ilk olarak tarayıcının adres çubuğuna admyn'in Url'si yazılmalıdır. Bu Url sitenin konulduğu yere göre değişecektir. Şimdiki sitenin adresini temel alacak olursak Url:

<http://www.ankara.edu.tr/science/astronomy/admyn>

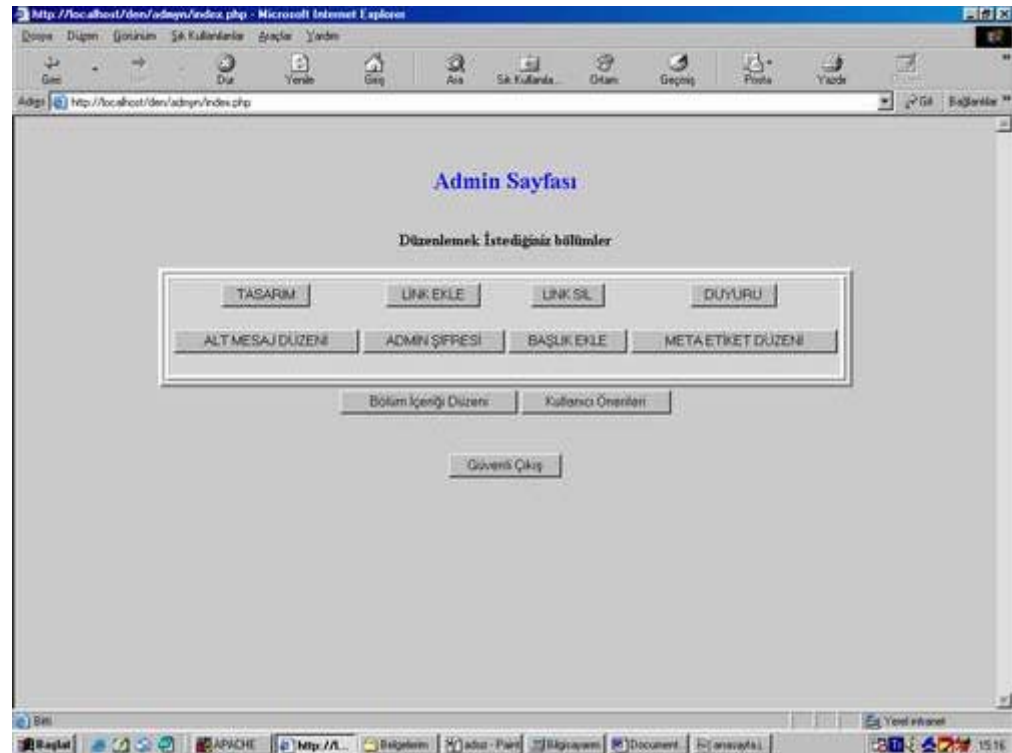
Şeklinde olacaktır. Bu adrese yöneldiğinizde karşınıza şekil 2 'deki ekran görüntüsü çıkacaktır.



Şekil 2. Admin girişi browser görüntüsü

Bu sayfa kullanıcı adı ve şifre istemektedir. Gerekli bilgiler yazıldıktan sonra gir butonuna basıldığında (eğer yazılan şifre doğruysa) admin için hazırlanan giriş sayfasına ulaşılabilir.

Admin sayfası Şekil 3 'te görüldüğü gibidir. Bu sayfada veritabanı ile etkileşimli olan bölümler sıralanmıştır. Bunlar sırasıyla; tasarım, linkekle, linksil, duyuru, altmesaj düzeni, admin şifre, başlık ekle, meta etiket düzeni, bölüm içeriği düzeni, kullanıcı önerileri ve güvenli çıkış şeklindedir. Sitede hangi düzen ile ilgili güncelleme yapılacaksa o güncellenmenin yapılabileceği bölüm düğmesine basılmalıdır. Burada önemli olan konu, bütün işlemler bittikten sonra mutlaka “güvenli çıkış” düğmesine basılarak admin sayfası terk edilmelidir. Aksi takdirde sitenin ana sayfası görüntülenemeyecektir. Bu konu daha detaylı olarak güvenlik bölümünde ele alınacaktır. Şimdi sırasıyla hangi Düğmenin hangi işlemlere izin verdiğine bakalım.

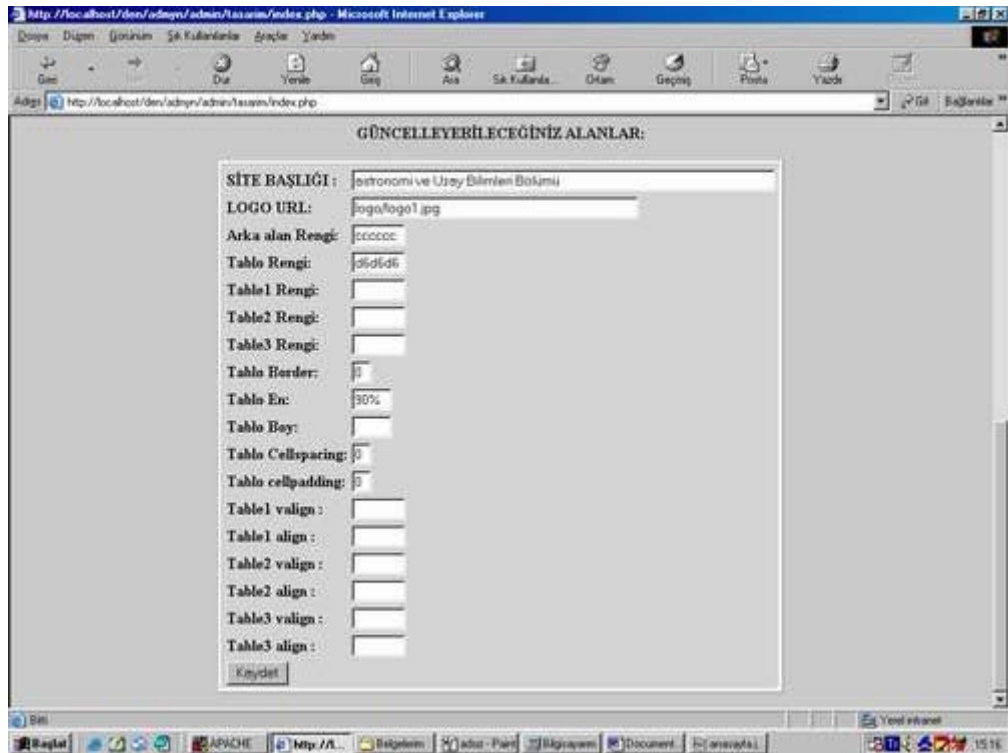


Şekil 3. Admin sayfası browser görüntüsü

#### 4.1.1. Tasarım

Bu bölüm ana sayfanın tasarımı ile ilgilidir. Tasarım butonuna basılarak ulaşılabilecek bu bölümün tarayıcı üzerindeki görüntüsü şekil 5 'te olduğu gibidir. Şekildende görüldüğü üzere kayıtlı değerler form içerisine yazılarak gelir. Bu değerler üzerinde değişiklik yapıp kaydet butonuna basılarak tasarım güncellenmesi tamamlanmış olur.

Burada güncellenebilecek alanlar sayfa içerisindeki form başlıklarıyla belirtilmiştir. Bunlar sırasıyla aşağıdadır :



Şekil 4. Tasarım bölümü browser görüntüsü.

a- Site başlığı : Bu alana sitenin başlığının ne olması isteniyorsa o yazılacaktır.

b- Logo Url : Bu alan ana sayfadaki site logosunun Url'sini içerir.

c- Arka alan rengi : Bu bölüm ana sayfa arka alan renginin değiştirebileceği alandır. Bu değiştirme işlemi HTML renk kodlarıyla yapılmalıdır. Eğer yeni giriş yapılmak istenilen rengin ingilizce karşılığı 6 ( altı) karakteri geçmiyorsa direkt ingilizce değeride yazılabilir. Örneğin; yellow, red, blue, white gibi.

d- Tablo rengi : Bu alana ana sayfadaki ana tablonun rengi yazılmalıdır. Burada da HTML kod ve 6 (altı) karakter ingilizce kuralları geçerlidir.

e- Table1 rengi : Bu alan ana sayfada ana tablonun ikinci bölümünün renk değerini içerir( ana sayfada sol taraftaki linklerin yer aldığı tablo). Yine aynı renk kuralları geçerlidir.

f- Table2 rengi : Bu alan ana sayfada ana tablonun üçüncü bölümünün renk değerini içerir (ana sayfada ortada yer alan resim ve arama motorunun olduğu yer). Renk kuralları geçerlidir.

g- Table3 rengi : Bu alan ana sayfada ana tablonun dördüncü bölümünün renk değerini içerir (ana sayfada sol taraftaki linklerin yer aldığı tablo). Renk kuralları geçerlidir.

h- Tablo border : Bu alan ana tablonun kalınlık değerini içerir. 0,1,2,3 gibi değerler girilerek tablo kalınlığı güncellenebilir.

ı- Tablo en : Bu alan ana tablonun tarayıcı penceresindeki enini belirlemek için kullanılır. Burada değer yüzde biçimiyle girilmesi tavsiye edilir (aksi takdirde farklı çözünürlükte farklı ekran görüntüsü ile karşılaşma olasılığı ortaya çıkacaktır). Örnek olarak; 90%, 75% gibi. Burada 90% tarayıcı penceresi eninin yüzde doksanı en olarak kabul edeceği anlamına gelir.

i- Tablo boy : Bu alan ana tablonun tarayıcı penceresindeki boyunu belirlemek için kullanılır. Tablo en 'deki bütün kurallar burada da geçerlidir.



j- Table cellpadding : Bu alan ana tablonun içeriği ile çerçevesi arasındaki boşluk miktarını içerir. 0,1,2,3 gibi sayı değerleri alır.

k- Table cellspacing : Bu alan ana tablonun hücrelerinin arasındaki boşluk miktarını içerir. 0,1,2,3 gibi sayı değerleri alır.

l- Table1 valign : Bu alan ana tablonun ikinci bölümündeki verilerin dikey hizalama değerini içerir. Değerleri; middle(orta), bottom(alt), top(üst) gibidir.

m- Table1 align : Bu alan ana tablonun ikinci bölümündeki verilerin yatay hizalama değerini içerir.

n- Table2 valign : Bu alan ana tablonun üçüncü bölümündeki verilerin dikey hizalama değerini içerir. Değerleri; middle(orta), bottom(alt), top(üst) gibidir.

o- Table2 align : Bu alan ana tablonun üçüncü bölümündeki verilerin yatay hizalama değerini içerir.

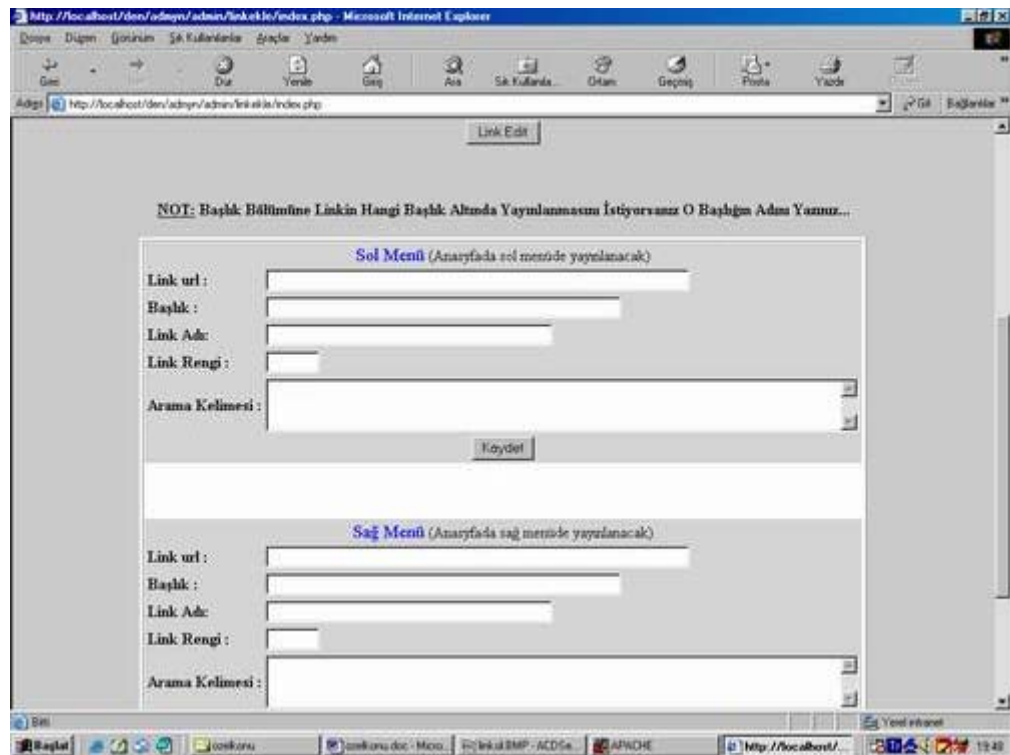
p- Table3 valign : Bu alan ana tablonun dördüncü bölümündeki verilerin dikey hizalama değerini içerir. Değerleri; middle(orta), bottom(alt), top(üst) gibidir.

r- Table3 align : Bu alan ana tablonun dördüncü bölümündeki verilerin yatay hizalama değerini içerir.

#### 4.1.2. Link Ekle

Bu bölüm ana sayfa, ana tabloda ikinci ve dördüncü bölümlerin yani sol ve sağ taraftaki tabloların içeriği ile ilgilidir. Sağ ve sol taraftaki bölümlere link ekleyebilmek ve var olan linkleri düzenleyebilmek amacıyla yapılmıştır(site içi bir sayfanın veya site dışı bir sayfanın linki olabilir).

Bu bölümün tarayıcı üzerindeki görüntüsü şekil 5 'teki gibidir.



Şekil 5. Link ekle bölümü browser görüntüsü

Bu bölümde yeni link eklemek için kullanılabilir sağ ve sol menü olmak üzere iki tablo vardır. Sağ menüye ana sayfada sağ tarafta, sol menüye ise ana sayfada sol tarafta yayınlanması istenilen linklerin girişi yapılmalıdır. Bu yeni veri girişleri form başlıklarındaki değerlere göre olmalıdır. Bunlar :

a- Link url : Bu alana yeni giriş yapacağınız linkin (ana sayfanın sunucudaki yeri temel alınarak kaydıyla ) adresi (bulunduğu yer : site içi veya site dışı ) yazılır.

b- Başlık : Bu alana yeni giriş yapılan linkin ana sayfada hangi başlık altında görüntülenmesi isteniyorsa o başlığın adı yazılır ( bu işlem yapılırken ana sayfada var olan başlığın adı eksiksiz yani boşluklar ve büyük küçük harf duyarlılığı göz önüne alınarak yazılmalıdır).

c- Link adı : Bu alana giriři yapılan linkin ana sayfada görünmesi istenilen adı yazılmalıdır(örneğin; “ www.ankara.edu.tr “linkinin adına “Ankara Üniversitesi” atanarak linkin görüntü deęerinin Ankara Üniversitesi olması sağlanır).

d- Link rengi : Bu alana ana sayfada yayımlanacak linkin karakter rengi yazılır. Daha önceki renk kuralları geçerlidir.

e- Arama kelimesi : Bu alan site içi arama motoru ile ilgili bir düzendir. Buraya sayfaya koymak istenilen linkin içerięi ile ilgili yazı yazılmalıdır. Bu sayede link kayıt olurken aynı zamanda arama motoru için gerekli bilgilerde kayıt altına alınır. Burada önemli husus yazılan yazının hem link ile ilgili anahtar kelimeleri içermesi hemde düz yazı kurallarına uygun halde yazılmasıdır. Çünkü : Site içi arama motorundan bir deęer sorgulandıęında arama motoru bu arama kelimelerinin içerisinde arama yapmakta ve kelimeyi bulduęu zamanda linki ile beraber browser’a göndermektedir. İşte bu işlem sırasında bulunan link’in (sayfanın) içerięi olarak burada girilen “arama kelimesi” deęeri görüntülenmektedir. Örneęin astronomi final tarihleri içeren bir sayfa hazırlandı ve siteye link olarak eklenecek, linkin adı, rengi, Url’si gibi deęerler girildikten sonra arama kelimesi yerine “Bu sayfada astronomi bölümü final tarihleri yer almaktadır.” Gibi bir açıklama girilmelidir ki arama motoru bu linki bulduęunda sayfa açıklaması olarak linkin altında bu deęeri gösterdięi zaman anlamlı bir görüntü olsun.

Bu bölüm içerisinde ayrıca “Anasayfa Orta Bölüm “ adı altında bir form var. Bu formda sadece “Link Url” bilgi alanı yer almaktadır. Bu alan ana tablonun ortası yani arama motorunun yer aldıęı bölümün yayımı ile ilgilidir. Burada varsayılan deęer olarak “resim.txt” adında bir düzyazı dosyası var. Bu dosya, veritabanından çağırılarak anasayfada “her dakika deęişen resim” olayı görüntülenmektedir. İstenildięi takdirde sitenin herhangi bir yerine yeni bir düz yazı (.txt) dosyası konularak bu dosyanın çağırılması sağlanabilir. Yeni konulacak düz yazı dosyası HTML veya PHP dilinde yazılmış olabilir bu konuda

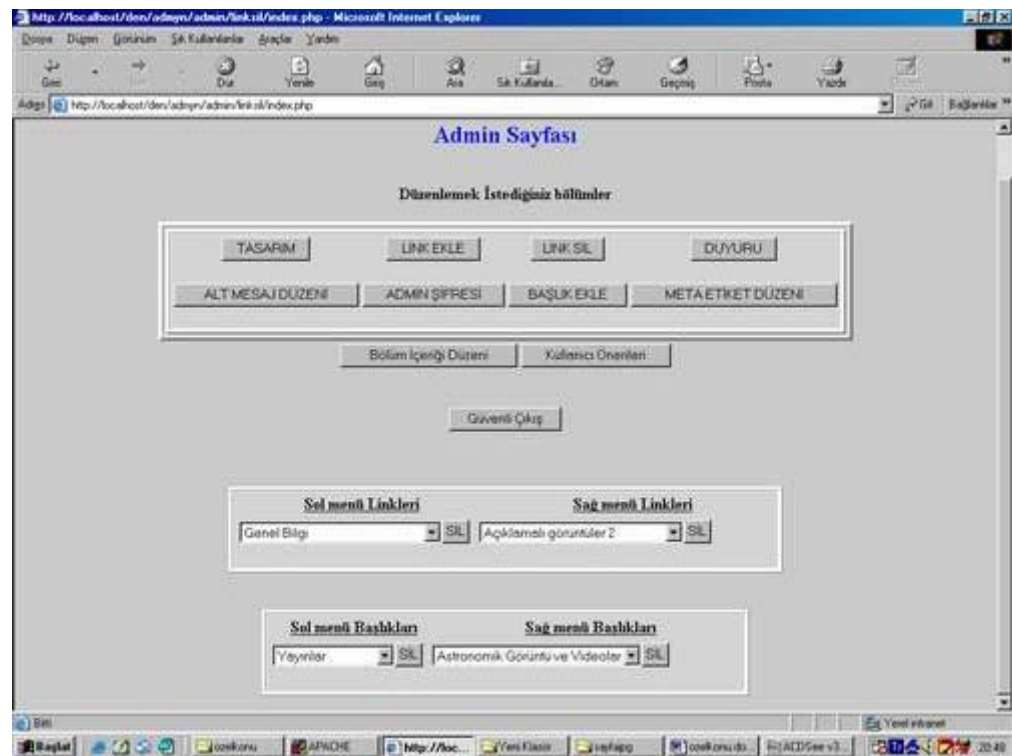
herhangi bir sınırlama olmaksızın sistem bunu derleyip ana tablonun ortasına yerleştirir.

#### 4.1.2.1. Link edit

Link ekle bölümünden “Link Edit” butonuna basılarak ulaşılan bu bölümde yanlış girilmiş linklerin düzeltilmesi işlemi yapılır. İlk önce çok seçmeli kutudan düzenlenmek istenilen link seçilir ve daha sonra gelen formlarda düzenleme işlemi yapılır. Burada tek sınır “Link Url”sidir. Budeğer asla düzenlenemez. Eğer bir “Link Url”si yanlış girilmişse tek yol o linki tümünden silip yeniden giriş yapmaktır.

#### 4.1.3. Link Sil

Bu bölüm kayıtlı linklerin ve başlıkların silinmesi işlemi için yapılmıştır.



Şekil 6. Link sil bölümü browser görüntüsü

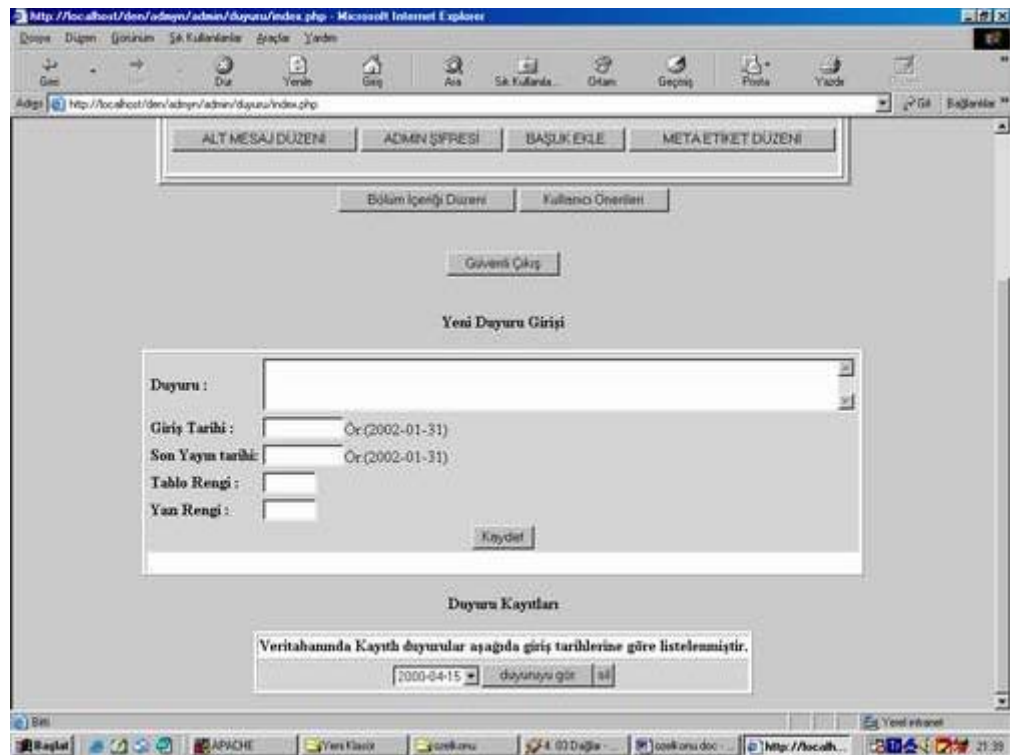
Tarayıcıdaki görüntüsü şekil 6'daki gibidir.

Bu bölümde silinmek istenen link veya başlık çoktan seçmeli formdan seçilerek sil butonuna basılmak sureti ile silinir. Burada önemli bir nokta; başlık silerken o başlık altındaki bütün linklerde veritabanından silinir.

#### 4.1.4. Duyuru

Bu bölüm anasayfda yer alan duyuru tablosu ile ilgilidir. Buradan yeni duyuru girişi, var olan duyuruların görüntülenmesi ve silinmesi işlemleri yapılabilir.

Bu bölümün görüntüsü şekil 7'deki gibidir.



Şekil 7. Duyuru bölümü browser görüntüsü

Yeni duyuru girişi yapmak için “yeni duyuru girişi” yazısının altındaki form doldurulmalıdır. Form alan başlıklarına göre doldurulur. Buna göre :

a- Duyuru

Bu alana yayımlanması istenilen duyuru yazılmalıdır. Herhangi bir karakter sınırlaması yoktur.

b- Giriş tarihi

Bu alana duyurunun yazıldığı tarih yazılmalıdır. Formatı; 2002-01-31 şeklinde olmak zorundadır.

c- Son yayın tarihi

Bu alana duyurunun en son yayımlanması istenilen tarih yazılmalıdır. Bu tarihten sonra ana sayfadaki duyuru tablosu kendisini otomatik olarak yok edecektir. Formatı; 2002-01-31 şeklinde olmak zorundadır.

d- Tablo rengi

Bu alana duyuru tablosunun renk değeri yazılır. Daha önceki renk kuralları buradada geçerlidir.

e- Yazı rengi

Bu alana duyurunun yazı rengi yazılmalıdır. Renk kuralları buradada geçerlidir.

Var olan duyuruları görmek veya silmek için “duyuru kayıtları” adlı yazının altındaki tablo kullanılır. Bu tabloda duyurular giriş tarihlerine göre sıralanmış ve görüntülenmiştir. Çoktan seçmeli kutudan duyuruyu seçilerek “duyuruyu gör” butonuna basılırsa, seçilen duyurunun kayıt görüntüleri ekrana gelir. Eğer sil butonuna basılırsa duyuru veritabanından silinir. Burada ilk önce duyuruyu görüp ondan sonra silinmesi tavsiye edilir çünkü çoktan seçmeli kutu içerisinde

duyuruların sadece giriş tarihleri yer almaktadır. Eğer yanlış duyuru silinirse bu işlemin geri dönüşü yoktur.

#### 4.1.5. Alt Mesaj Düzeni

Bu bölüm ana sayfanın en altında yer alan iki mesaj tablosunun düzenini içerir. Tarayıcı görüntüsü şekil 8'deki gibidir.

Bu bölümde alt mesaj tablolarını pasif veya aktif yapılabilir ayrıca varolan alt mesaj tabloları içerisindeki değerleri değiştirilebilir.

Tabloları aktif veya pasif yapmak için sayfanın üst bölümündeki "aktif" "pasif" butonları kullanılır. Bu butonlar veritabanı ile etkileşim halindedir. Eğer mesaj alanı aktif ise buton "pasif" olarak görüntülenecek, pasif ise "aktif" olarak görüntülenecektir. Bu sayede hangi mesaj alanının aktif hangisinin pasif olduğu durumundan haberdar olunması amaçlanmıştır.

Alt mesaj alanlarına yeni veri girişi içinse butonların altındaki form alanları kullanılır. Buradan yeni veri girişi için form, başlıklarına göre doldurulmalıdır. Bu başlıklar:

##### a- Mesaj

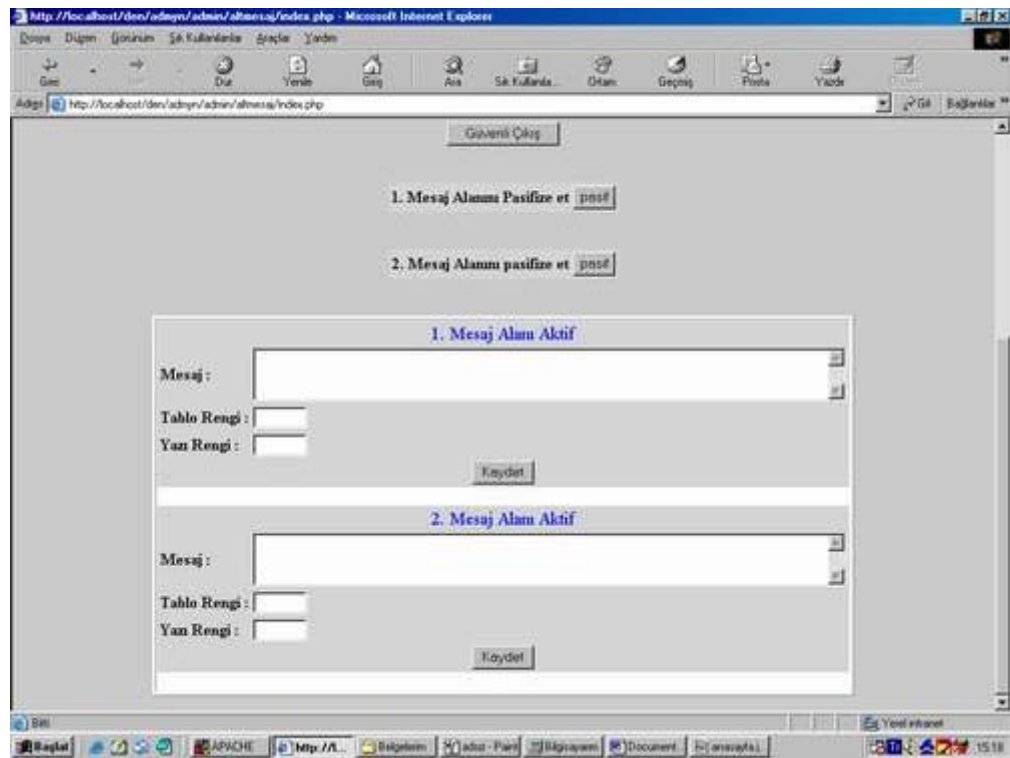
Bu alana tablolarda görüntülenmesi istenilen mesaj yazılır. Bu işlem için hiçbir kural yoktur. PHP dilinde HTML dilinde hatta ikisi karışık bir şekilde kullanılarak ta yazılabilir.

##### b- Tablo rengi

Bu alana mesaj tablosunun rengi yazılır. Renk kuralları geçerlidir.

##### c- Yazı rengi

Bu alana mesaj tablosu içerisindeki yazıların varsayılan karakter rengi yazılır. Bu işlem mesaj alanı içerisine yazılan ve farklı renklere kodlanmış yazıları etkilemez, sadece o tablo için “yapısal varsayılan yazı rengi (body textcolor)” değerini verir.



Şekil 8. Alt mesaj düzen bölümü browser görüntüsü

#### 4.1.6. Admin Şifresi

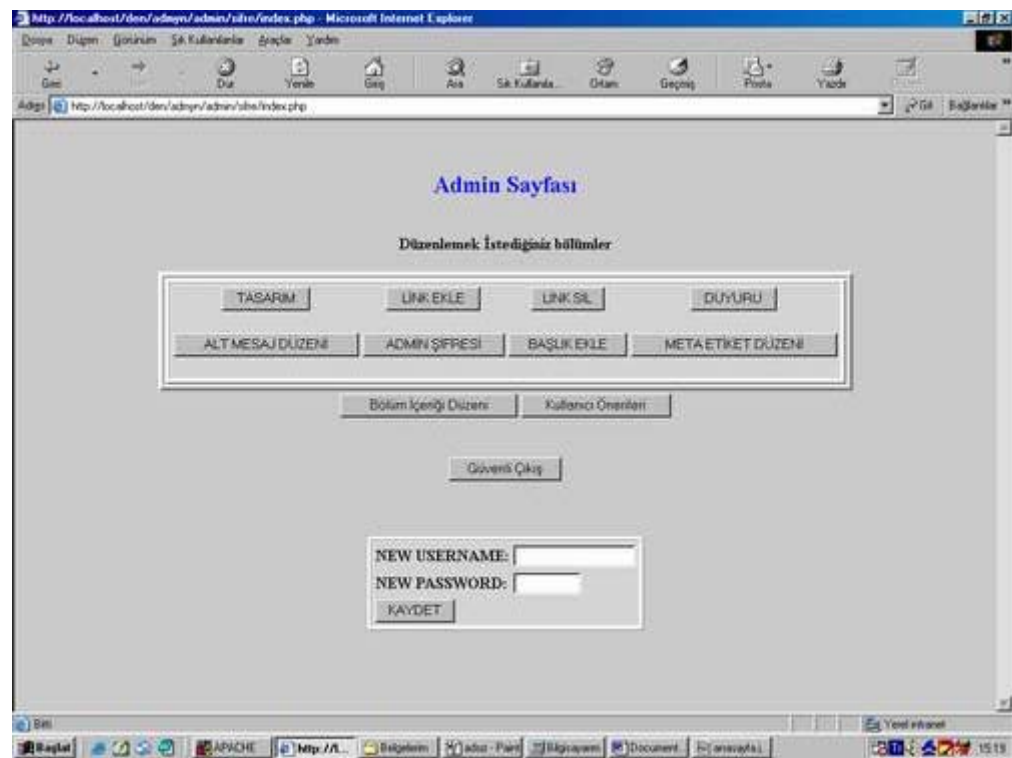
Bu bölüm admin sayfası giriş şifresini güncelleme işlemi yapmaktadır. Buradaki amaç belirli aralıklarla admin şifresini güncelleyebilmektir. Bu işlem güvenlik açısından tavsiye edilir.

Şifre güncelleme işlemi sayfadaki form aracılığıyla gerçekleştirilir. Bu formda “new user name” alanına yeni kullanıcı adını, “new password” alanına ise yeni şifre yazılmalıdır. Bu işlemlerden sonra kaydet butonuna basıldığında yeni bir



sayfada, yenilenmiş kullanıcı adı ve şifre değerleri görüntüsü gelecektir. Bu görüntüdeki değerlerin dikkatlice okunulup bir yere not edilmesi çok önemli bir husustur. Çünkü klavyede yapılacak bir tuşa basma hatası düşünülen şifre veya kullanıcı adından farklı bir değer veritabanına kayıt edilmesine sebep olabilir. Kaydetme işleminden sonra geri dönen değer veritabanından çağırılan değer olduğu için kesin sonuç oradadır.

Admin şifresi bölümünün tarayıcı görüntüsü şekil 9'daki gibidir.

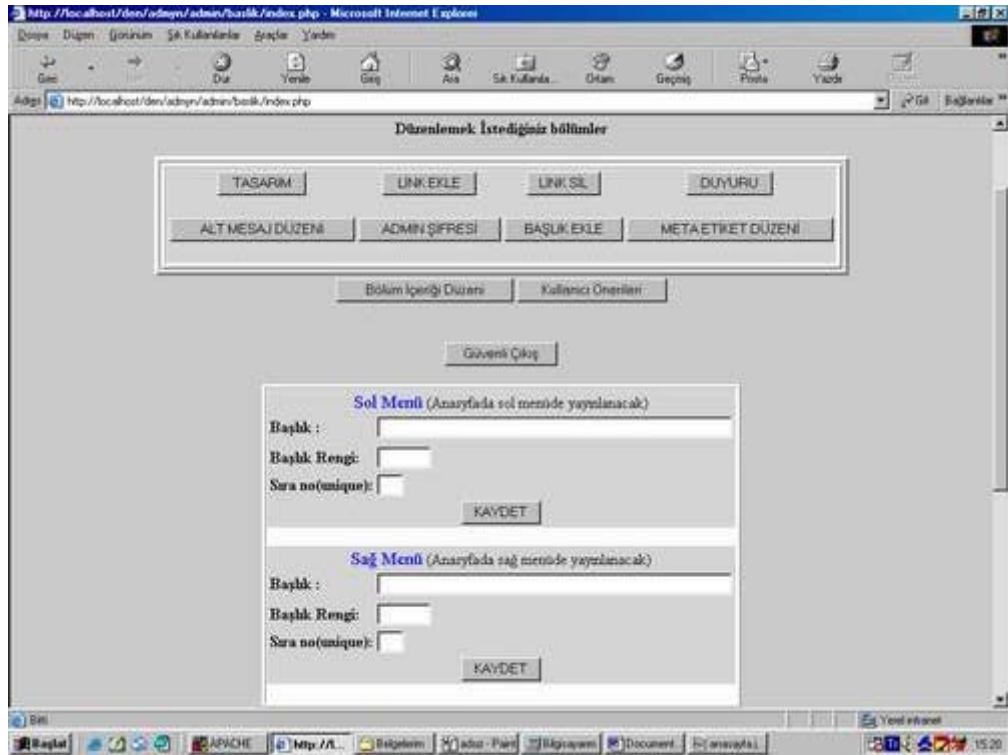


Şekil 9. Admin şifresi bölümü browser görüntüsü

#### 4.1.7. Başlık Ekle

Bu bölüm ana sayfada, ana tablonun sağ ve sol taraflarında bulunan link başlıklarının düzenlenmesi ile ilgilidir. Bu bölümde sadece ana sayfaya link başlığı ekleme işlemi yapılabilir.

Başlık ekleme işlemi şekil 10’da görüldüğü gibi “sağ menü” ve “sol menü” olmak üzere iki ayrı form yoluyla yapılabilir. “Sağ menü “ yazısının altındaki forma yapılan kayıtlar ana tablada sağ tarafta, “sol menü” yazısının altındakiler ise anabloda sol tarafta görüntüleneceklerdir. Kayıtlar alan isimlerine göre yapılır:



Şekil 10. başlık ekle bölümü browser görüntüsü

#### a- Başlık

Bu alana ana sayfaya eklenmek istenilen başlığın adı yazılır. Hergangi bir harf kısıtlaması yoktur.

#### b- Başlık rengi

Bu alana başlığın rengi yazılır.

#### c- Sıra no (unique)

Bu alan başlığa bir “id” vermek amacıyla yapılmıştır. Bilinmesi gereken husus, anasayfada başlıklar artan sırada olmak koşulu ile sıralanırlar. Dolayısıyla bu alana girilecek sıra no değeri aslında anasayfada başlığın kaçınıcı olarak görüntülenmek istendiği sorusunun cevabıdır. Bu değer mutlaka tek (unique) yani başka bir başlık tarafından kullanılmayan bir değer olmalıdır (diğer başlıkların kullandığı sıra no değerleri sayfanın altında bir tablo şeklinde görüntülenmektedir).

#### 4.1.8. Meta Etiket Düzeni

Bu bölüm Astronomy Bölümü İnternet Sitesi'nin “meta tag”larını veritabanı etkileşimli yapmak için oluşturulmuştur. “Meta tag”ları kısaca tarayıcıların ve arama motorlarının site hakkındaki içerik bilgisini aldığı yer olarak tanımlanabilir. Bu bilgiler çeşitli anahtar kelimeler aracılığıyla alınır. Örneğin; “content”, “keywords”, “refresh” gibi.

Bu bölümde değiştirilme yetkisi verilen meta etiketleri; “keywords”, “description”, ve “refresh” tir.

Bu bölümün tarayıcı görüntüsü şekil 11'deki gibidir.

Not: Sistemde varsayılan olarak kayıtlı meta etiket kodları aşağıdaki gibidir:

```
<META NAME="AUTHOR" CONTENT="Talip Kaycı">
```

```
<META HTTP-EQUIV="CHARSET" CONTENT="ISO-8859-9">
```

```
<META HTTP-EQUIV="CONTENT-LANGUAGE" CONTENT="Turkish">
```



Şekil 11. Meta etiket düzeni bölümü browser görüntüsü

Bu bölümde yapılmak istenilen değişiklikler şekil 11’de görünen form aracılığıyla yapılır. Sayfada o an için kayıtlı olan meta etiket değerleri görüntülenmektedir. Yeni yapılacak kayıt o değerlerin üzerine yazılacağından meta değerleri silinir. Eğer var olan meta etiketlerine birkaç değer daha eklemek isteniyorsa, forma ilk önce var olan değerler yazılıp hemen ardından yeni değer eklenerek kayıt yapılmalıdır.

Güncelleme form aracılığıyla başlık bilgilerine göre yapılmalıdır.

#### a- Keywords

Bu alana girilecek bilgiler site ile ilgili anahtar kelimeler olmalı ve her iki anahtar kelime arasına virgül(,) konulmalıdır. “Keywords” arama motorlarının arama yaparken dikkate aldığı ve anahtar değer okuduğu etikettir.

#### b- Description

Bu alana girilecek bilgiler site ile ilgili açıklamalar olmalıdır. Örneğin; Ankara Üniversitesi Astronomi ve Uzay Bilimleri Bölümü İnternet Sitesi gibi. Arama motorları site ile ilgili açıklamayı bu etiketten okurlar.

#### c- Refresh

Refresh etiketi ya sayfayı başka bir sayfaya yönlendirmek yada mevcut sayfayı belirli zaman aralıklarında yenilemek için kullanılır. Burada ki kullanım amacı sayfayı yenilemektir. Anasayfada bulunan “her dakika yenilenen resim programının işleme için düşünülmüş bir bölümdür. Bu alana sayısal bir değer girilmelidir(Refresh etiketi girilen değeri saniye biriminde algılar).

#### 4.1.9. Bölüm İçeriği Düzeni

Bu bölüm site içinde veritabanı ile etkileşimli olan sayfaların kontrolü için yapılmıştır. Site içerisinde veritabanı etkileşimli iki bölüm bulunmaktadır. Bunlar : çalışanlar ve astronomi programları sayfalarıdır. Admin bu bölümü kullanarak bahsi geçen sayfaları kontrol edebilme yetkisine sahiptir.

#### I- Çalışanlar

Bu bölüm Astronomi Bölümü çalışanlarının görüntülediği sayfanın kontrollerini içermektedir. Burada yapılabilecek işler; çalışanların düzenlenmesi ve yeni isimler eklemek şeklindedir. Bu bölümün tarayıcı görüntüsü şekil 12’deki gibidir.

Çalışanlar sayfasına kişi eklemek için aşağıdaki alanları doldurun.

Ad Soyad :

Telefon :

E Mail :

Web :

Sıra no :

Kayıtlarda Düzenleme Yaptıktan sonra kaydet butonuna basmalısınız:

ID(unique)(sıra no)	Ad Soyad	Telefon no	E_mail	Web adresi		
0					<input type="button" value="kaydet"/>	<input type="button" value="sil"/>
1	Prof. Cemal AYDIN (Fen)	2126729-1001	aydin@eps.science.nsk		<input type="button" value="kaydet"/>	<input type="button" value="sil"/>
2	Prof. Semra ENGIN (A)	2126729-1308	engin@astro1.science.tr		<input type="button" value="kaydet"/>	<input type="button" value="sil"/>
3	Prof. Zeki TUFEKÇİOĞULLI	2126729-1350	tufekci@astro1.science.c		<input type="button" value="kaydet"/>	<input type="button" value="sil"/>
4	Prof. Osman DEMİRCAN		demircan@comu.edu.tr		<input type="button" value="kaydet"/>	<input type="button" value="sil"/>
5	Prof. Nihal YILMAZ (Emel)				<input type="button" value="kaydet"/>	<input type="button" value="sil"/>
6	Prof. Eliften DERMAN	2126729-1319	derman@ankara.edu.tr		<input type="button" value="kaydet"/>	<input type="button" value="sil"/>

Şekil 12. Çalışanlar bölümü browser görüntüsü

Çalışanlar sayfasına yeni kişi eklemek için bu bölümde üstte yer alan form kullanılır. Ad, soyad, telefon, e\_mail ve web adresleri olduğu gibi kaydedilir. Buradaki önemli nokta “sıra no” alanının doldurulmasıdır. Bu bölüm yeni kayıt yapılan kişinin, çalışanlar sayfasındaki görüntüsünün nerede olacağını belirler. Sıra no tek(unique) olmalıdır.

Var olan kayıtlarda değişiklik yapmak veya silmek için altta yer alan büyük form kullanılır. Bu formda veritabanında kayıtlı bütün değerler mevcuttur ve güncelleme işlemleri, direkt bu değerler üzerinde değişiklik yapılarak o satırın yanındaki “kaydet” butonuna basılmak sureti ile yapılır. Silme işlemi ise, hangi veri silinmek isteniyorsa o satırın yanındaki “sil” butonuna basılarak gerçekleştirilir.

## II- Astronomi Programları

Bu bölüm astronomi programları sayfasının kontrollerini içermektedir. “Yeni kayıt”, “kayıt düzenle” ve “kayıt sil” olmak üzere toplam üç bölümden oluşur.

### II.a.- Yeni Kayıt

Yeni program ekleme işi bu bölümden yapılır. Tarayıcı görüntüsü şekil 13’teki gibidir.

The screenshot shows a web browser window with the URL [http://localhost/den/adyayn/bolum/ast\\_prog.php](http://localhost/den/adyayn/bolum/ast_prog.php). The page title is "Program Eklenecek İçin Aşağıdaki Alanları Doldurmalısınız." (You must fill in the following fields to add a program). The form is divided into two sections: "Dos Programları" and "Macintosh Programları". Each section contains the following fields: "Program url:", "Update Url:", "Program Adı:", "Puan:", and "Açıklama:". There is an "EKLE" button below each section. The browser's taskbar shows the system tray with the time 15:21 and the date 15.02.2007.

Şekil 13 Astronomi programları yeni kayıt bölümü browser görüntüsü

Eklenecek program için “dos-windows” veya “macintosh” olmak üzere iki ayrı form kullanılır. Ekleme işlemi form alan başlıklarına göre yapılır.

#### a- Prođram url

Buraya eklenen prođramın “localhost/astro/ast\_prog” klasörü temel alınmak sureti ile prođramın adresi yazılır. Örneđin; “localhost/astro/ast\_prog/dos” klasöründe bulunan astronomi.zip isimli prođramı sisteme kaydetmek için “prođram url” bölümüne “dos/astronomi.zip” yazılmalıdır.

#### b- Update url

Bu bölüme eklenen prođramın güncellenebileceđi bir web sitesi varsa onun adresi yazılır.

#### c- Prođram adı

Buraya prođramın görüntülenmesi istenilen adı yazılır.

#### d- Puan

Buraya prođramın kalitesini göstercek herhangi bir deđer yazılır(1,2,3,4,5 gibi sayısal deđerler olaması veritabanı işleyişı açısından tavsiye edilir).

#### e- Açıklama

Bu bölüme prođramın yanında prođram ile ilgili görüntülenmesi istenilen bilgiler(açıklamalar) yazılmalıdır.

### II.b. - Kayıt Düzenle

Bu bölümde yanlış giriş yapılmıř prođramların düzenlenmesi mümkündür. İlk önce düzenlenecek prođram çoktan seçmeli kutulardan seçilmeli ve daha sonra düzeltilmek istenilen deđerler yazılarak kaydet butonuna basılmalıdır.



## II.c.- Kayıt Sil

Bu bölümde veritabanında kayıtlı program bilgilerini silmek mümkündür. Silinmek istenilen kayıt çoktan seçmeli kutulardan seçilip sil butonuna basılmalıdır.

## III- Kullanıcı Önerileri

Bu bölümde site ziyaretçileri ile interaktif etkileşim halinde bulunan sayfaların kontrolü mümkün olmaktadır. Bu sayfalar temel anlamda site ziyaretçilerinden görüş, öneri ve bilgi almak amacını taşımaktadır. Bu bilgi alma alanları; link önerileri, program önerileri ve ziyaretçi görüşleri olmak üzere toplam üç adettir.

### III.a- Link önerileri

Link önerileri bölümünde site ziyaretçilerinden astronomi ile ilgili ilginç, değişik ve faydalı linklerin site yönetimine bildirilmesi amacı güdülmüştür. Bu sayfanın ziyaretçi yansıması “localhost/interaktif/index.php?k=link” adresinde bulunmaktadır. Admin için düzenleme ve kontrol bölümüne ise “Admin” sayfasından “kullanıcı önerileri” butonuna basılmak sureti ile ulaşılmaktadır. Bu butona basıldıktan sonra ortaya çıkan üç alt buton gözlenecektir. Bu üç butondan “link önerileri” ne basılmak sureti ile kullanıcılardan gelen link önerilerinin takibi yapılabilir.

Bu bölümde yapılabilecek şey; ziyaretçilerden gelen link önerilerini görebilmek ve silebilmektir. Bu görüntü ; ad, soyad, e\_mail, link ve gönderilen link ile ilgili açıklama yer aldığı bir tablo şeklindedir. Burada gerekli incelemeler yapıldıktan sonra kaydadeğer bulunmayan önerilerin silinmesi tavsiye edilir ( veritabanını boşuna dolu tutmamak için).

### III.b- Program önerileri

Bu bölümün amacı site ziyaretçilerinden, astronomi ile ilgili, internet üzerinde yayımlanan programlar hakkında bilgi almaktır. Ziyaretçiler için bu sayfa yansıması “localhost/interaktif/index.php?k=program” adresindedir. Admin içinse “kullanıcı önerileri” bölümünden “program önerileri” butonuna basmak sureti ile ulaşılabilmektedir.

Bu bölümde yapılacak iş aynen “link önerileri” bölümünde olduğu gibi tablo içerisinden inceleme yapmak ve gerektiğinde öneriyi veya önerileri silmektir.

### III.c- Kullanıcı görüşleri

Bu bölümün amacı ziyaretçilerden site veya Astronomi Bölümü ile ilgili yorum alabilmek ve bu yorumları site içerisinde “ziyaretçi görüşleri” bölümünde yayımlamaktır. Aslında bu bölüm birçok İnternet Sitesi’nde var olan ziyaretçi defteri bölümleri ile aynı işleve sahiptir.

Bölümün ziyaretçilerin önerisi için yansıması “localhost/interaktif/index.php?k=program” adresindedir. Ziyaretçi görüşlerinin görüntülediği adres ise “localhost/interaktif/ziyaretçi.php” dir.

Bu bölümün düzenlemeleri için “kullanıcı önerileri” bölümünden “kullanıcı görüşleri” butonuna basılması gerekir.

Butona basıldıktan sonra ortaya çıkan bir tablo içerisinde ad, soyad, e\_mail, tarih ve mesaj(yorum) bilgileri yer almaktadır. Bu bilgilerin hemen yanında ise “sil” ve “aktif yap” veya “pasif yap” linkleri belirecektir. Admin gerekli incelemeleri yaptıktan sonra gönderilen yorumu silme, pasif durumda bekletme ve aktif hale getirip görüntülenmesine izin verme yetkisine sahiptir.

“Ziyaretçi görüşleri” sayfasında yayımlanan mesajların yanında “sil” ve “pasif yap” linkleri belirecektir( görüşün yayımlanması için veritabanında aktif hale

gelmesi gerekir). Bu durumdaki bir mesaj “pasif yap” butonuna basmak sureti ile “ziyaretçi görüşleri” sayfasında görüntülenmesi engellenir. Ters işlemde ise görüntülenmeyen bir mesajın görüntülenmesi sağlanır. Bir diğer link olan “sil” linki ise o satırdaki mesajı tüm verileri ile birlikte veritabanından silecektir.

## 4.2. Site İçerisindeki Dinamik Sayfaların Kullanımı

Sitedeki admin bölümünden, sadece sitenin veritabanı ile etkileşimli bölümlerinin güncellenmesi ve kontrol edilmesi mümkün olmaktadır. Fakat site içerisinde veritabanı etkileşimli olmayıp dinamik olan sayfalar vardır. Bu sayfalar genel anlamda görüntülenmesi istenilen verileri kendi dosyasından değilde başka dosyadan okuma şeklinde yaratılmışlardır. Bundaki amaç; bir sayfanın ana yapısını sabit tutup ( resimler, animasyonlar, tablolar gibi), bu yapı içerisinde varolan verileri başka bir dosyanın içerisinde okutarak sisteme dahil etmektir. Bu işleme güzel bir örnek “Bölüm Yayınları”nın sayfasıdır. Bu sayfada anayapı yani yayın kategorileri linkleri, sayfa logo’su, arkaplan rengi gibi değerler “localhost/astro/yayin/index.php” dosyasında sabit olup, yayınların içeriği yani yayın bilgileri (kategorisine göre) başka bir “txt” uzantılı dosyadan okutularak sayfa içerisinde görüntülenmektedir. Bu sayede sayfaya yeni bir yayın eklenmek istendiğinde tasarımsal kodlar içeren esas dosya ile uğraşmak yerine yayınların girilmiş olduğu “txt” uzantılı dosya açılarak içerisine yeni veriler eklenmek sureti ile sayfa güncelleme yapılabilir.

Bu şekilde olan sayfaların listesi ve nasıl güncelleneceği aşağıdadır.

### 4.2.1. Yayınlar Klasörü

Bu klasördeki bütün sayfalar, verileri bir “txt” uzantılı dosyadan görüntüler. Klasör “localhost/astro/yayin” adresindedir. Bu klasörün “index”inde sadece sayfa ile ilgili tasarım ve link kodları bulunmaktadır. Bunun dışında yayın

verilerini beş ayrı “txt” uzantılı dosyadan okumaktadır. Bu dosyalar ve içerdiği bilgiler :

a- “interdergi.txt” dosyasında uluslararası dergilerde yayınlanan yayınların listesi bulunmaktadır. Yeni ekleme yapmak için dosyanın en altına eklenmek istenilen yayını yazmak yeterlidir.

b- “intertoplanti.txt” dosyasında uluslararası toplantılardaki yayınlar yer almaktadır. Yeni ekleme yapmak için dosyanın en altına eklenmek istenilen yayını yazmak yeterlidir.

c- “nationdergi.txt” dosyasında ulusal dergilerde yayınlanan yayınların listesi bulunmaktadır. Yeni ekleme yapmak için dosyanın en altına eklenmek istenilen yayını yazmak yeterlidir.

d- “nationtoplanti.txt” dosyasında ulusal toplantılardaki yayınlar yer almaktadır. Yeni ekleme yapmak için dosyanın en altına eklenmek istenilen yayını yazmak yeterlidir.

e- “kitap.txt” dosyasında kitap ve popüler yayınların listesi yer almaktadır. Yeni ekleme yapmak için dosyanın en altına eklenmek istenilen yayını yazmak yeterlidir.

Görüldüğü üzere bu işlem sayesinde beş ayrı internet sayfası yapılması yerine tek bir sayfaya beş tane mantıksal döngü yazarak, bir sayfadan beş ayrı internet sayfası görüntüsü elde edilebilmektedir.

#### 4.2.2. Önemli Linkler Klasörü

Bu bölümde sayfalar, “eğitim linkleri”, temel astronomi linkleri” ve “ilginç linkler” olmak üzere üç adet sayfa şeklinde katagorilere ayrılmıştır. Bu sayfaların içindedey ayrı katagorilerde linkler vardır. Bunlar aşağıdaki gibidir:

#### a- Eğitim linkleri

Bu sayfa “localhost/astro/link/egitim.php” adresindedir. Sayfa içerisinde “çocuklar için astronomi linkleri”, “astronomi tarihi linkleri” ve “öğrencilerimizin web sayfaları” olmak üzere üç ayrı link katagorisini barındırmaktadır. Bunlardan:

“Çocuklar için astronomi linkleri” bölümü “link” klasörü içerisinde bulunan “cocuk.txt” dosyasından okunmaktadır. Bu dosyadaki güncelleme işlemleri bir önceki bölümde anlatılan şekilde yapılabilmektedir.

“Astronomi tarihi linkleri” bölümü “link” klasörü içerisinde bulunan “tarih.txt” dosyasından okunmaktadır. Bu dosyadaki güncelleme işlemleri bir önceki bölümde anlatılan şekilde yapılabilmektedir.

“örencilerimizin web sayfaları” bölümü “link” klasörü içerisinde bulunan “ogrenciweb.txt” dosyasından okunmaktadır. Bu dosyadaki güncelleme işlemleri bir önceki bölümde anlatılan şekilde yapılabilmektedir.

#### b – Temel astronomi linkleri

Bu sayfa “localhost/astro/link/temellink.php” adresindedir. Sayfa içerisinde “veritabanları”, “temel astronomi linkleri”, “yeni linkler” ve “periodikler” olmak üzere dört ayrı link katagorisini barındırmaktadır. Bunlardan:

“Veritabanları” bölümü “link” klasörü içerisinde bulunan “veritabani.txt” dosyasından okunmaktadır. Bu dosyadaki güncelleme işlemleri bir önceki bölümde anlatılan şekilde yapılabilmektedir.

“Temel astronomi linkleri” bölümü “link” klasörü içerisinde bulunan “temellink.txt” dosyasından okunmaktadır. Bu dosyadaki güncelleme işlemleri bir önceki bölümde anlatılan şekilde yapılabilmektedir.

“Yeni linkler” bölümü “link” klasörü içerisinde bulunan “yenilink.txt” dosyasından okunmaktadır. Bu dosyadaki güncelleme işlemleri bir önceki bölümde anlatılan şekilde yapılabilmektedir.

“Periodikler” bölümü “link” klasörü içerisinde bulunan “periodik.txt” dosyasından okunmaktadır. Bu dosyadaki güncelleme işlemleri bir önceki bölümde anlatılan şekilde yapılabilmektedir.

#### c - İlginç linkler

Bu sayfa “localhost/astro/link/ilginc.php” adresindedir. Sayfa içerisinde “İlginç Astronomi Linkleri”, “Hava Durumu, Sözlük, Kütüphane, Yıldız Haritası ve Astronomik Hesap Makinası” ve “diğer linkler” olmak üzere üç ayrı link katagorisini barındırmaktadır. Bunlardan:

“İlginç astronomi linkleri” bölümü “link” klasörü içerisinde bulunan “ilginc.txt” dosyasından okunmaktadır. Bu dosyadaki güncelleme işlemleri bir önceki bölümde anlatılan şekilde yapılabilmektedir.

“Hava Durumu, Sözlük, Kütüphane, Yıldız Haritası ve Astronomik Hesap Makinası” ve “diğer linkler” bölümleri “link” klasörü içerisinde bulunan “diger.txt” dosyasından okunmaktadır. Bu dosyadaki güncelleme işlemleri bir önceki bölümde anlatılan şekilde yapılabilmektedir.

#### 4.2.3. Astronomik Görüntüler

Bu bölüm diğerlerinden biraz farklı olmak sureti ile tam dinamiktir. Aynı zamanda site içerisinde en kolay güncellenebilen bölümdür.

Bu bölüdeki sayfalar oluşturulurken PHP ‘nin klasör ve dizin okuyabilme yeteneğinden faydalanılmıştır. Resimler çeşitli katagorilere ayrılmış ve her katagori için ayrı proğram sayfaları yaratılmıştır. Bu proğram sayfaları içiçe bulunan karışık döngüler içermektedir. Döngülerin sonucunda istenilen bir klasör içerisindeki varlıklar (her formatta var olan herşey) okutularak, ekrana tablolar içerisinde ve belirli sayılarda( ilk on veya ilk yirmi resim gibi) gönderilmesi sağlanmaktadır. Klasörler içerisindeki varlık sayısına göre bu sayfaların işleyiş biçimi değişmektedir. Örneğin her yirmi varlık için sayfa kendi içerisinde otomatik olarak yeni bir sayfa daha açmaktadır.

Bu sistemde resimler ve videolar kendi sayfaları ile aynı isimde olan bir klasör ve bu klasörün önünde “tmp” uzantısı olan ikinci bir klasörden okutularak ekrana gönderilmektedir. Buna bir örnek olarak galaksi resimlerinin bulunduğu “galaxy.php” dosyasını ele almakta fayda vardır. Bu sayfada; dosyanın okuduğu bilgiler “ast\_goruntu” klasörü içerisinde bulunan ”galaxy” ve “tmpgalaxy” klasörlerinden okunmaktadır. “tmpgalaxy” klasöründen sayfa üzerindeki tablolar içerisinde görüntülenmek istenilen ve “harddisk”teki kapladığı alan küçültülmüş olan resimler. “galaxy” klasöründe ise orjinal boyuttaki resimler bulunmaktadır. Proğram bu iki klasörü değerlendirerek tablo içerisinde görüntülenecek resimleri “tmpgalaxy” klasöründen çeker. O resimin üzerine tıkladığında ise orjinal resimin bulunduğu “galaxy” klasöründen gerçek resim çağırılarak sayfada görüntülenmektedir.

Böyle bir sayfaya resim eklemek için; sayfanın okuma yaptığı klasörlere resimlerin konulması yeterlidir. Silme işlemide aynı şekilde iki klasöredende silmek sureti ile yapılabilir. Burada önemli bir kural vardır. Her iki klasöre eklenen resimde aynı isim ve aynı uzantıya sahip olmalıdır. Örneğin eklenilmek istenilen “earth.jpg” isimli bir resim için ilk önce o resimin küçültülmüş boyutu yaratılmalı ve aynı isimle saklanmalıdır(“earth.jpg”). Daha sonra küçültülmüş resim”tmp” klasörüne asıl resim ise gerçek resimin okunduğu klasöre konulmalıdır.

“Astronomik görüntüler” bölümü (localhost/astro/ast\_goruntu) içerisinde bu şekilde çalışan 7(yedi) adet PHP uzantılı dosya bulunmaktadır. Bunlar:

a- “Galaxy.php” bu dosya galaksi resimlerinin görüntülenmesini sağlar. Dosyanın okuduğu bilgiler “ast\_goruntu” klasörü içerisinde bulunan ”galaxy” ve “tmpgalaxy” klasörlerindedir.

b- “ilginc.php” bu dosya ilginç resimlerin görüntülenmesini sağlar. Dosyanın okuduğu bilgiler “ast\_goruntu” klasörü içerisinde bulunan ”ilginc” ve “tmpilginc” klasörlerindedir.

c- “nebula.php” bu dosya nebula(bulutsu) resimlerinin görüntülenmesini sağlar. Dosyanın okuduğu bilgiler “ast\_goruntu” klasörü içerisinde bulunan ”nebula” ve “tmpnebula” klasörlerindedir.

d- “solar.php” bu dosya güneş sistemi resimlerinin görüntülenmesini sağlar. Dosyanın okuduğu bilgiler “ast\_goruntu” klasörü içerisinde bulunan ”solar” ve “tmpsolar” klasörlerindedir.

e- “space.php” bu dosya çeşitli uzay resimlerinin görüntülenmesini sağlar. Dosyanın okuduğu bilgiler “ast\_goruntu” klasörü içerisinde bulunan ”space” ve “tmpspace” klasörlerindedir.

f- “telescope.php” bu dosya teleskop resimlerinin görüntülenmesini sağlar. Dosyanın okuduğu bilgiler “ast\_goruntu” klasörü içerisinde bulunan ”telescope” ve “tmptelescope” klasörlerindedir.

g- “video.php” bu dosya videoların görüntülenmesini sağlar. Dosyanın okuduğu bilgiler “ast\_goruntu” klasörü içerisinde bulunan ”video” ve “video-jpg” klasörlerindedir. “Video-jpg” klasöründe video ile ilgili resim yer almaktadır. Aynı diğer tmp klasörlerinde olduğu gibi küçültülmüş ve filmin



isimi ile aynı isimde olmalıdır. Bu resim tablo içerisinde filmin görüntüsünü verecek ve üzerine tıklandığında ise bir film oynatıcısı ile filmi oynatılmaya başlayacaktır.

## SONUÇ

Bu kitapçıkta, PHP ve MySql dillerinin kullanımından çok, elde edilen ürünün özelliklerine yer verilmesinin doğru olacağı düşünülmüştür. Bu nedenle PHP ve MySql bölümleri kısa olarak ve bu dillerin genel hatlarını içerecek şekilde ele alınmış, daha sonra PHP ve MySql dillerini bilmeyen bir kişinin anlayabileceği şekilde ürünün anayapısı ve içeriği anlatılmaya çalışılmıştır.

Temelde anlatılan şey, statik yapıdaki bir Web Sayfası'nın dinamik ve kısmen interaktif bir yapıya büründürülmüş olması ve bu yapının kullanıcı yönüdür. Bu anlatım yapısından dolayı ürünün sabit(statik) olduğu düşüncesi doğabilir (çünkü ürünün teknik ayrıntıları bu kitapçıkta ele alınmamıştır). Oysa elde edilen ürün işe başlama felsefesine uygun yani statik değil dinamiktir. Açık kaynak programların kullanılması nedeniyle elde edilen üründe açık kaynak olma özelliğini taşımaktadır. Ürün, orta derecede PHP ve MySql bilgisi ve biraz programcılık deneyimine sahip herkesin, kodlara girip program içerisinde neler olduğunun anlaşılabilir şekilde elde edilmeye çalışılmıştır. Bundan dolayı ürün kodlarına girilerek istenilen değişiklikler yapılabilir hatta istenilen ek programlar ürün ile uyum içerisinde çalıştırılabilir.

Sonuç olarak, değiştirilebilme ve eklenebilme özelliklerinden dolayı elde edilen ürün bir çekirdek yapıdır. Örnek olarak, daha sonradan yapılacak çeşitli veritabanı ve veritabanı sorgulama sayfaları çok kolay bir şekilde ve bir iki günlük kodlama çalışması ile ürüne eklenebilir. Burada ürüne eklenmesinden kasıt; ürünün yönetim bölümü olan "admyın" içerisine yeni yapılan program yönetimlerinin yerleştirilebilmesidir.

İşin sonunda elde edilen şey değişme özelliği olan bir çekirdek yapıdır. Yani çekirdek sabit kalıp üstüne başka şeyler ekleme zorunluluğu yoktur. Eklenecek şeyin ihtiyacına göre çekirdek programda değişiklik yapılabilmektedir. İşte bu özellik, yaptığım işe güven duymamı sağlamış ve bu güvenin rahatlığı ile ürünü savunabilecek duruma ulaştırmıştır.

## EKLER

### 1. SİSTEM GÜVENLİĞİ

#### 1.1. Giriş

İnteraktif site yapısına sahip olmak ne kadar çok kolaylık sağlasada, beraberinde güvenlik sorununu da getirir. Çünkü interaktif olmayan bir sitede, site, ziyaretçileri ile hiçbir etkileşimde(karşılıklı bilgi alışverişi) bulunmadığı için, kötü niyetli ziyaretçiler ancak ve ancak sitenin kurulu olduğu sunucuya “saldırmak” sureti ile o siteye zarar verebilirler. Bu durumda bütün güvenlik sunucunun sorumluluğundadır. Fakat interaktif bir site sözkonusu olduğunda durum biraz değişir. Böyle bir sistemde ziyaretçiler ile etkileşim var olduğundan; en azından etkileşim sırasında aktif olan “web programları”nın güvenliği sistemi yapana yani site sahibine aittir.

Astronomi Bölümü için yapılan bu sistemde, “hacker” denilen internet korsanlarından korunmak için çeşitli güvenlik tedbirleri alınmıştır.

#### 1.2. Yapısal Güvenlik

Buradaki kasıt veritabanındaki olası çökmelerin engellenmesidir. Bir veritabanının çökmesi iki sebepten olur. Birincisi yanlış veya yetersiz kurulum sonucu veri sorgulamada aksaklıklar çıkmak sureti ile veritabanının zorlanarak komutları yerine getirmesi ve sonuçta çökmesi. İkincisi özellikle web tabanlı sistemlerde aynı anda çoklu sorgulama olduğu için sorgu esnasında veritabanında yapılacak olası değişikliklerin veritabanını çökertmesi. Burada birinci sebep site sahibini değil sunucuyu ilgilendirmektedir ve dolayısıyla bu konuda sistem içerisinde herhangi bir önlem almak mümkün değildir. Esas önemli olan ikinci sebeptir.

Sistem içerisinde birçok yerde veritabanında aynı anda sorgulama olayı gerçekleşmektedir. Başka bir deyişle veritabanı aynı anda birçok kullanıcıya sahip olmaktadır. Buna en iyi örnek “anasayfa”dır. Daha önce anlatıldığı gibi ana sayfa görüntüsünün oluşması için; 13(on üç) veritabanı tablosu (table) ve 64 (altmışdört) tablo alanı (field) aktif halde çalışarak PHP tarafından yapılan sorguların sonuçlarını programa göndermektedir. Bu işlem her ziyaretçi için ayrı ayrı yapılmaktadır. Örneğin belirli bir anda “Astronomi Bölümü İnternet Sitesi”ni ziyaret eden on kişi varsa bu işlemlerin hepsi ayrı ayrı on kez tekrarlanacaktır. Buraya kadar olan olayların sorumluluğu sunucuya aittir. Ancak sitede var olan güncelleme bölümü (admin sayfası) veritabanına çeşitli kayıtların silinmesi veya düzenlenmesi konusunda komut göndermektedir. İşte bu noktadan itibaren güvenlik artık site sahibinin veya yapımcısının sorumluluğundadır. Bunu bir örnek ile açıklayalım: Bir önceki örnekte olduğu gibi sitede aktif on (mütevazi bir sayı) ziyaretçinin olduğunu varsayalım. Bunların altışar saniye aralıklarla siteye girdiklerini kabul edelim. Biliyoruz ki ana sayfa kendisini her altmış saniyede (bir dakikada) tekrar yenilemektedir (ortadaki “resim” bölümünün çalışması için). Bu bilgiler ışığında veritabanının her altı (6) saniyede bir sorgulama işlemi yapacağı açıktır. Şimdi hiçbir güvenlik tedbiri almadan, “admin sayfası”na girerek ana sayfa ile ilgili yani ana sayfayı ilgilendiren veritabanı tabloları ile ilgili bir değişiklik yapacak olursak. Hemen oturup ziyaretçilerden gelen sorgu ile bizim “admin sayfası”ndan yapmış olduğumuz sorgunun eşzamanlı olmaması için “dua” etmek gerekecektir. Çünkü ziyaretçiden gelen sorgu veritabanında kayıtlı verilerin çekilmesi işlemini, admin tarafından gerçekleştirilen sorgu ise veritabanında kayıtlı verinin değiştirilmesi işlemini gerçekleştirmektedir. İşte tam bu noktada iki işlemin eş zamanlı olduğu düşünülürse veritabanı, ziyaretçi sorgusunu gerçekleştirirken eline bir alandan iki sonuç geçecektir (biri varolan sonuç diğeri admin’in eşzamanlı değiştirdiği sonuç) bu durumda veri tabanı hiçbir işlem yapamayacak ve kendisini kitleyecektir.

Yukarıda anlatılan durumu engellemek amacıyla “admin sayfası”na giriş yapıldığında veritabanında belirli bir değer işlemekte ve bu değerın yansıması

olarak tüm sitenin veritabanı bağlantısı kesilmektedir. Böylece admin işlem yaptığı sürece hiçbir ziyaretçinin veritabanına sorgu göndermesi mümkün olmayacaktır. Bunun sonucunda veritabanı çökmesi engellenmiştir. Herşeyin bir bedeli olduğu gibi bu güvenlik önleminin bedelide, adminin sistemde kaldığı süre içerisinde ziyaretçilerin veritabanı etkileşimli sayfalara ulaşamamasıdır. İşte burada önemli olan unsur adminin “admin sayfasını” terkederken mutlaka ama mutlaka “güvenli çıkış” butonuna basmasıdır. Aksi takdirde sistemle veritabanı bağlantısı süresiz kesilmiş olur.

### 1.3. Admin Sayfası Güvenliği

Sistemin en canalıcı noktası bu sayfadır ve bundan dolayı bu sayfa/sayfalar için birçok güvenlik tedbiri alınmıştır.

#### 1.3.1 Şifreli Giriş

“Admin sayfası”na giriş veritabanında kayıtlı olan şifreyle birebir eş şifreyi bilmek ile mümkündür. Bu şifre bilinmeden sisteme girilmesi olanaksızdır. Şifre çözme programlarının prensibi; çok kısa bir süre içerisinde birçok olasılık deneyip şifreyi bulmak olduğu göz önüne alınarak şifre girme sayısı bir defa ile sınırlandırılmıştır. Eğer yanlış şifre girilirse sayfa kendisini otomatik olarak (anında) “http://www.ankara.edu.tr” adresine yönlendirmektedir. Dolayısıyla hiçbir programın ikinci bir deneme şansı olmayacaktır.

#### 1.3.2. Diğer Sayfaların Güvenliği

Diğer sayfalarda güvenlik, “admin sayfası” yönlendirmesi dışında ulaşılmamak sureti ile sağlanmıştır. Yani “admyrn” klasöründeki diğer sayfalara tarayıcı adres çubuğuna sayfa adresini yazarak ulaşmak mümkün değildir. Bu işlem iki kollu güvenlik ile yapılmıştır. Birincisi; alt sayfalar ( link ekle, başlık ekle, duyuru gibi), kendilerine admin tarafından gönderilecek gizli değerler olmadan içerisindeki PHP programının işlemesine izin vermezler. Dolayısıyla “admin

sayfası”na girmeden diğer sayfaların açılabilmesi mümkün değildir. İkinci olarak; gönderilen gizli değerin bulunması ihtimali göz önüne alınarak, adminin sisteme girmesi ile nasıl ki ana sayfa kitleniyorsa aynı şekilde bu sayfalarda adminin girişinden haberdar olurlar. Ancak ve ancak bu iki şart geçerli olduğunda sayfalar içerisindeki programlar çalışacak ve dolayısıyla sayfalar kullanımda olacaktır.

Sonuç olarak “Admin Sayfası”na giriş yapılmadan ( Admin şifresi bilinmeden) sistemde herhangi bir güncelleme yapılması veya güncelleme yapılabilecek alanlara ulaşılması mümkün değildir.

### 1.3.3. Olası Admin Güvenlik Hatası

Bu başlıktaki kasıt, adminin “güvenli çıkış” butonuna basmadan sistemi terk etmesi durumunda meydana gelecek güvenlik hatasıdır. Bu işlem sonucunda veritabanı ile sayfaların bağlantısı koptuğundan artık sistemde hiç bir işlem yapılamaz hale gelir. Buna “Admin Sayfası” na girmek de dahil. Yani böyle bir durumda sistem tamamen kapatılmış ve işe yaramaz hale gelmiş olur.

Bu durumu engellemek amacı ile “admyn” klasörü içerisine “alternatif.php” adında bir dosya konulmuştur. Bu dosya bir şekilde veritabanı ile bağlantı kurup yukarıda anlatılan bütün sorunları düzeltecektir. Bu dosyayı çalıştırmak için yapılması gereken tek şey tarayıcının adres çubuğuna “localhost/ admyn /alternatif.php” yazarak sayfayı çağırmasıdır.

Önemli bir nokta olarak bu sayfanın bir şekilde saklanması tavsiye edilir. Çünkü bu sayfanın adresini öğrenecek olan ziyaretçiler veya “hacker”lar bu sayfayı “çağırarak” sureti ile Admin’in işlerini yapmasını yani “admin sayfası”na girişi engelleyebilirler. Bu yüzden dosyanın (alternatif.php) sistem dışı bir yerde örneğin bir diskette saklanıp yukarıda anlatılan şekilde bir hata olduğunda sisteme dahil edilip çağırılmak sureti ile kullanılması tavsiye edilir.

## 2. SİTE İÇİ KLASÖR YAPISI

### 2.1. Anaklasör

Sitenin anaklasöründe 5 (beş) adet klasör, 1(bir) adet JPG uzantılı resim dosyası ve 4 (dört) adet PHP uzantılı dosya bulunmaktadır. Bunlar:

Klasörler;

“admyn”, “anasayfa”, ”astro”, “interaktif”, “logo”.

PHP dosyaları;

“index.php” (anasayfa bu klasörden okunur), ”main.php”(anasayfa içeriği bu klasördedir), “arama.php” (arama motoru programının kayıtlı olduğu dosya), ”cons.php”(Admin’in sistem girişi sırasında anasayfanın devre dışı kalır ve anasayfa yerine bu dosya okunur).

### 2.2. Admyn

Bu klasör içerisinde site yönetimi ile ilgili dosyalar yer almaktadır. Toplam 43 (kırküç) dosya ve 10 (on) klasörden oluşur.

### 2.3. Anasayfa

Bu klasör içerisinde iki adet dosya ve iki adet klasör vardır. Bunlar; anasayfanın ortasındaki resim programını ve veritabanı bağlantısını içeren dosyalar ve resim dosyasınınçağırıldığı resim klasörleri.

### 2.4. Astro

Tüm site içeriği bu klasör içerisinde yer almaktadır. Bu klasör içerisinde 3588 (üçbinbeşyüzseksensekiz) dosya ve 92 (doksaniki) klasör bulunmaktadır.

## 2.5.İnteraktif

Bu klasör içerisinde interaktif olan bölümlerin işlemleri gerçekleştirilmektedir (ziyaretçi görüşleri, ziyaretçi önerileri gibi).

## 2.6. Logo

Bu klasör içerisinde site logoları yer almaktadır.



## KAYNAKLAR

- 1- OTANER, Kayra, 2001. Niçin MySql. Sistem Yayıncılık, İstanbul
- 2- <http://www.php.org.tr>
- 3- <http://www.php.net>
- 4- <http://www.php.com>
- 5- <http://phpnuke.org>
- 6- <http://www.phpbuilder.com>
- 7- <http://www.mysql.com>
- 8- <http://php.weblogs.com>
- 9- <http://gtk.php.net>
- 10- <http://www.phpwizard.com>
- 11- <http://www.phplivesupport.com>
- 12- <http://www.php-accelerator.co.uk>
- 13- <http://bugs.php.net>
- 14- <http://www.zend.com>
- 15- <http://pear.php.net>
- 16- <http://www.php-award.net>
- 17- <http://hosts.php.net>
- 18- <http://www.phpworld.com>
- 19- <http://www.phpindex.com>
- 20- <http://news.php.net>

