

# Sosyal Paylaşım Ağlarında PHP ve MySQL Kullanarak Etkileşimli Arkadaş Ekleme-Çıkarma Yöntemi

\*<sup>1</sup>Ferhat ASLAN <sup>2</sup>Ali BULDU <sup>3</sup>Kazım YILDIZ <sup>4</sup>Serdar ARSLAN

<sup>1,2,3</sup>Marmara Üniversitesi Teknik Eğitim Fakültesi, Elektronik-Bilgisayar Eğitimi Bölümü, İstanbul, Türkiye

<sup>4</sup>Koç Sistem Bilgi ve İletişim Hizmetleri A.Ş., Üsküdar, İstanbul, Türkiye

## Özet

Gelişen bilgisayar dünyasına kuruluşlar veya kullanıcılar bireysel olarak geliştirdikleri yeni ve yaratıcı ürünler ile katkıda bulunmaktadır. Böylelikle internet üzerinden insanlar istediklerini yapabilmektedirler. Bunun yanı sıra internet dünyasında arkadaşlık siteleri ve sosyal paylaşım ağları da oldukça önemli bir yer tutmaktadır. Geliştirilen uygulamada, MySQL veritabanı ve PHP programlama dili ile etkileşimli arkadaş ekleme ve arkadaşlıktan çıkarma algoritması geliştirilmiştir. Uygulamanın temel yapısı itibarıyla bir sayfada sisteme üye olan kullanıcılar bulunmakta, başka bir sayfada ise aktif olan kullanıcının takip ettiği sistem üyeleri bulunmaktadır. Bu yapı da, aktif olan kullanıcının, takip ettiği üyelerin gönderilerini takip edebilme olanağı sağlamaktadır. Geliştirilen bu yapının yapılacak olan sosyal paylaşım ağlarında kolaylık sağlaması hedeflenmiştir.

**Anahtar Kelimeler:** Kullanıcı takip etme, Arkadaş olma, Arkadaş ekleme

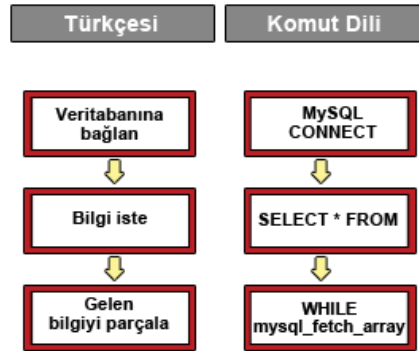
## 1.Giriş

Bilişim teknolojisi ve bu teknolojiadaki gelişmeler, çalışma ortamlarımızı, günlük yaşantımızı ve iletişim araçlarımızı değiştirmektedir [1]. Sosyal ağlar insanları paylaşılan bilgiler aracılığıyla etkileşim halinde olmalarını sağlar, onları birbirlerine bağlar ve bunu zaman içerisinde profesyonel bir destek haline getirir. Bu ağların amacı gelişmeyi sağlamak için kurumsal ya da bireysel düzeyde esnek etkileşim imkânı sunmaktır. Ayrıca, kullanıcıların anlık ileti paylaşımında bulunabilmeleri ve düşüncelerini diledikleri gibi yazabilecekleri bir ortam oluşturmaktır. Aynı zamanda gelişen olaylar hakkında paylaşımında bulunup toplumu haberdar edecek bir kitle iletişim aracı olmasını sağlamaktır. Bunun gibi birçok sosyal paylaşım ağı olmasına rağmen yerlilerine çok az rastlanmamaktadır. Bu çalışmanın amacı ilk olarak ulusal düzeyde, daha sonra uluslar arası düzeyde bir paylaşım ağı oluşturmaktır. Bu çalışmada kodlama dili olarak PHP programlama dili kullanılmıştır. PHP, sunucu tabanlı olarak çalışan geniş kullanımlı, genel amaçlı, HTML içerisine gömülebilen betik, açık kaynak kodlu bir yazılım ve web programlama dilidir [2,3].

Bu paylaşım ağının oluşturulma sürecinde karşılaşılan zorluklardan biri de arkadaş ekleme ve arkadaş çıkarma aşamasıdır. MySQL veritabanında uygun tablo tasarımı ve PHP ile doğru bir algoritma oluşturma sonucunda bu sorun giderilmeye çalışılmıştır. Uygulanan tasarım da her türlü kullanıcı düzeyine uygun olması dikkate alınarak oluşturuldu.

## 2. Sistemin Genel Yapısı

Şekil 1’de görüldüğü gibi sistem önce veritabanına bağlanmaktadır. Veritabanından bilgi çekildikten sonra, gelen bilgiyi parçalayarak istediğimiz şekilde kullanmamıza olanak sağlamaktadır. Veritabanına bağlantı kurulmasının sebebi, farklı bir program olmasıdır. Farklı bir bilgisayarda, hatta başka bir ülkede bile olabilir [4].



Şekil 1. PHP ile MYSQL Veritabanından Bilgi Okumak [4]

## 3. MySQL Veritabanında Uygun Tablo Tasarımı

Sistem yapısının kurulabilmesi için veritabanında iki tablodan faydalanıldı. Bunlardan biri sistem üyelerini bulduran üyeler tablosu, diğeri ise arkadaşlık tablosudur.

### 3.1. Üyeler Tablosunun Oluşturulması

Üyeler tablosu, Şekil 2’de görüldüğü gibi *id*, *adSoyad*, *email*, *sifre*, *kayıtTarihi* ve *foto* olmak üzere altı adet alandan oluşturuldu. Sisteme kayıt olan kullanıcılar ad-soyad, şifre ve daha önce sistemde olmayan bir e-mail girerek kayıtlarını tamamlarlar. Kayıt yapan bu kullanıcıların bilgileri veritabanında ‘*uyeler*’ tablosuna kaydedilir. Oluşturulurken kullanılan SQL kodları aşağıdaki gibidir [5-7] :

```
CREATE TABLE IF NOT EXISTS `uyeler` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `adSoyad` varchar(40) NOT NULL,
  `email` text NOT NULL,
  `sifre` text NOT NULL,
  `kayıtTarihi` date NOT NULL,
  `foto` varchar(200) NOT NULL,
  PRIMARY KEY (`id`) )
```

id	adSoyad	email	sifre	kayıtTarihi	foto
1	Ferhat Aslan	ferhataslan111@hotmail.com	4de35585e43001e7436de75dae44b67f	2012-09-26	3-profil/resimler/e83c13058219db985ddaf17e61a65189...
2	Fatih Aslan	fatyy@hotmail.com	343a595e43301e7136da95dae34b67f	2012-09-26	3-profil/resimler/182590_295936053852936_938612973...
3	Murat Aslan	murataslan@hotmail.com	9d59195d43301a6436da7adae37b66f	2012-09-27	3-profil/resimler/261865_1957462068583_1625995_n.j...
4	Ali Berat Aslan	aliaslan@hotmail.com	19f265d65365e1156ea12afe390bf195	2012-10-06	
5	Harun Cam	haruncam@hotmail.com	2fa232a15864e5126e612ade360bf325	2012-10-06	

Şekil 2. MYSQL Veritabanında Oluşturulan “*uyeler*” Tablosu

### 3.2. Arkadaşlık Tablosunun Oluşturulması

Arkadaşlık tablosu Şekil 3'te görüldüğü gibi *id*, *takipEden*, *takipEdilen* ve *aktiflik* olmak üzere dört adet alandan oluşmaktadır. Giriş yapmış olan kullanıcı Şekil 4'te görünen 'Arkadaşlar' menüsü altındaki 'Yeni Arkadaş Bul' alt menüsünde bulunan her kullanıcıya ait 'Takip Et' butonuna tıklayarak takip etmeye başlar ve bunların kaydı Şekil 3'te görüldüğü gibi, veritabanındaki 'arkadaslik' tablosuna yapılır. Bunun arka planında butona tıklayan, sistemdeki aktif kullanıcı, veritabanında *takipEden* alanına, takip etmek için tıkladığı kullanıcı bilgisi de *takipEdilen* alanına kaydedilir. Aynı zamanda aktiflik durumu '1' yapılır ve arkadaş takip edilir. Kullanıcı, arkadaşını takip etmeyi bıraktığında ise aktiflik durumu sıfır '0' yapılarak takip etmekten vazgeçilir. Oluşturulurken kullanılan SQL kodları aşağıda gösterilmektedir [5-7]:

```
CREATE TABLE IF NOT EXISTS `arkadaslik` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `takipEden` text NOT NULL,
  `takipEdilen` text NOT NULL,
  `aktiflik` int(11) NOT NULL,
  PRIMARY KEY (`id`)
)
```

id	takipEden	takipEdilen	aktiflik
34	ferhataslan111@hotmail.com	fatyy@hotmail.com	1
35	ferhataslan111@hotmail.com	murataslan@hotmail.com	1
36	ferhataslan111@hotmail.com	aliaslan@hotmail.com	0
39	ferhataslan111@hotmail.com	haruncam@hotmail.com	0
40	ferhataslan111@hotmail.com	deneme@hotmail.com	0

Şekil 3. MYSQL Veritabanında Oluşturulan "arkadaslik" Tablosu

## 4. PHP ile Uygun Algoritma Oluşturma

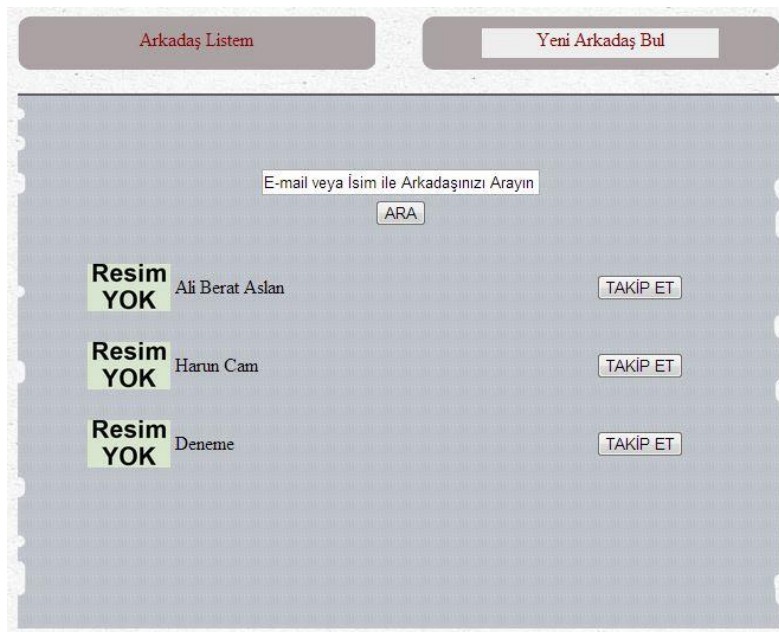
### 4.1. Yeni Arkadaş Bul Sayfasının PHP ile Oluşturulması

"Yeni Arkadaş Bul" PHP sayfası Şekil 4'te görüldüğü gibi oluşturulmuştur. SQL sorgusu ile 'üyeler' tablosundan oluşturulan şartlara göre kullanıcılar bu sayfaya çekilir. Fakat sisteme giriş yapmış olan kullanıcının takip ettiği kullanıcıları çekmez. Yalnızca takip edilmeyenler çekilir. Daha sonra çekilen bu kullanıcılar bir döngü içerisine alınmıştır. Döngü içinde her kullanıcının bilgileri için bir tablo oluşturularak, bilgiler tablo içerisine çekilir. Tablo içerisine çekilen bilgiler, Şekil 4'te de görüldüğü gibi kullanıcı resmi, adı-soyadı ve bunlarla beraber bir takip et butonudur. Kullanıcı, 'Takip Et' butonuna tıkladığında aktif kullanıcı bilgisi ve takip edilecek kullanıcı bilgisi veritabanında 'arkadaslik' tablosuna kaydedilir ve oradaki aktiflik durumu '1' yapılır. Böylelikle, artık "Yeni Arkadaş Bul" sayfasında görüntülenmez. Çünkü "Yeni Arkadaş Bul" sayfasında görüntülenen kullanıcılar, 'arkadaslik' veritabanında aktiflik durumu '1' olmayanlardır.

Yeni Arkadaş Bul sayfası PHP algoritması kod bloğu görülmektedir:

```
$sorgu = @mysql_query("SELECT adSoyad, email, foto FROM uyeler WHERE email NOT IN (SELECT takipEdilen FROM arkadaslik WHERE takipEden = '".$_SESSION['email']."' AND aktiflik = '1' AND email<>'".$_SESSION['email'].")")

while($row = mysql_fetch_assoc($sorgu))
{
    echo "<center>";
    echo "<table class='arkadasSecili'>";
    $value = "TAKİP EDİLİYOR";
    echo "<tr><form method='post' action='takipKaldir.php'>";
    echo "<td height='60' width='90'><img src='".$row['foto']."' width='70' height='40'></td>";
    echo "<td width='200'>".$row['adSoyad'].</td>";
    echo "<td width='100'><input type='hidden' name='takipEdilen' value='".$row['email']."'";
    /><input type='submit' value='".$value."' width='100' /></td></form>";
    echo "</table>";
}
```



Şekil 4. PHP İle Oluşturulan “Yeni Arkadaş Bul” Sayfası

#### 4.2. Arkadaş Listem Sayfasının PHP ile oluşturulması

“Arkadaş Listem” PHP sayfası Şekil 5’te görüldüğü gibi oluşturulmuştur. SQL sorgusu ile ‘arkadaslik’ tablosundan yalnızca sisteme giriş yapmış olan kullanıcının takip ettiği arkadaşlar listelenir. Sistemdeki aktif kullanıcı ‘Yeni Arkadaş Bul’ sayfasından kullanıcıları takip et butonuna tıkladığında bu kullanıcı bilgileri veritabanında ‘arkadaslik’ tablosuna eklenmekte. Döngü ile tablo oluşturularak kullanıcı bilgilerinden olan kullanıcı resmi, adı-soyadı ve bunlarla beraber bir ‘Takip Ediliyor’ butonu Şekil 5’te görüldüğü gibi tablo içinde gösterilir. Kullanıcı, ‘Takip Ediliyor’ butonuna tekrar tıkladığında, sistemdeki aktif kullanıcı bilgisi ve takip edilen kullanıcı bilgisi veritabanından ‘arkadaslik’ tablosuna gönderilerek ‘1’ olan aktiflik durumu ‘0’ yapılır. Bu durumda takip edilen kullanıcı artık takip edilmeyenler

arasına geçer. Tekrar takip edilmek istendiğinde aynı işlemler tekrarlanarak takip edilebilir duruma geçecek şekilde kodlanmıştır.

```
$sorgu = @mysql_query("SELECT adSoyad, email, foto FROM uyeler WHERE email IN (SELECT takipEdilen FROM arkadaslik WHERE takipEden = '$_SESSION[email]'. ' AND aktiflik = '1')")

while($row = mysql_fetch_assoc($sorgu))
{
    echo "<center>";
    echo "<table class='arkadasSecili'>";
    $value = "TAKİP EDİLİYOR";
    echo "<tr><form method='post' action='takipKaldir.php'>";
    echo "<td height='60' width='90'><img src='$_row[foto]'. ' width='70' height='40'></td>";
    echo "<td width='200'>".$_row['adSoyad']. "</td>";
    echo "<td width='100'><input type='hidden' name='takipEdilen' value='$_row[email]'. ' /><input type='submit' value='$_value'. ' width='100' /></td></form>";
    echo "</table>";
}
```

Arkadaş Listem sayfası PHP algoritmasının kod bloğu görülmektedir:



Şekil 5. PHP İle Oluşturulan “Arkadaş Listem” Sayfası

## Sonuç

Günümüzde zamansal ve mekânsal anlamda büyük kolaylıklar sağlayan internet ve ona bağlı olarak da sosyal paylaşım ağları, sosyalleşmenin önemli bir tamamlayıcı unsuru olarak ön plana çıkmaktadır. Sosyal paylaşım ağlarının amacı, sosyalleşme sürecinde toplumsal ilişkileri ve paylaşımları bir platforma taşıyabilmektedir. Gerçek hayatlarda paylaşılan anların bu ortamda paylaşımına devam edilebilir.

Örneğin oluşturulan bir etkinlik için bir araya gelip sosyalleşen bireyler, etkinlik sonrasında çekilen fotoğrafları sosyal paylaşım ağları üzerinden paylaşabilir ve bu paylaşım çevresinde sosyalleşme sürecini sürdürebilirler.

Sonuç olarak, bu çalışmada genel itibariyle her kullanıcının yalnızca arkadaşı olduğu sistem üyelerinin gönderilerini, paylaşımlarını görebilmesi amaçlanmıştır. Yapılan projede özgün bir arkadaş ekleme-çıkarma yöntemi oluşturuldu. Bu çalışmanın ileride yapılacak etkileşimli sitelerde kolaylık sağlaması amaçlanmıştır.

### **Teşekkür**

Bu çalışma KoçSistem Bilgi ve İletişim Hizmetleri A.Ş. tarafından desteklenmektedir.

### **Kaynakça**

- [1] Çakır R., Yükseltürk E., “Bilgi toplumu olma yolunda öğrenen organizasyonlar, bilgi yönetimi ve e-öğrenme üzerine teorik bir çözümleme,” Kastamonu Eğitim Dergisi, 18(2), pp. 501-512, Mayıs 2010.
- [2] <http://www.php.net/> Erişim Tarihi: Şubat 2013
- [3] <http://php.sitesi.web.tr/> Erişim Tarihi: Şubat 2013
- [4] <http://www.phpkodlari.com/mysql-dersleri/mysql-ders-2-veritabanından-bilgi-almak/> Erişim Tarihi: Mart 2013
- [5] <http://sql.sitesi.web.tr/tablo-olusturmak.html/> Erişim Tarihi: Mart 2013
- [6] <http://www.r10.net/database/> Erişim Tarihi: Mart 2013
- [7] <http://www.webteders.com/dersler/ders/Mysql-Veritabani-yaratma-ve-tablo-olusturma/18> Erişim Tarihi: Mart 2013