

**MOBİL İLETİŞİM TEKNOLOJİLERİNİN ÖĞRENCİ BİLGİ  
SİSTEMLERİNDE KULLANIMI  
VE BİR UYGULAMA**

Demet GÜLSEREN  
Yüksek Lisans Tezi

Fen Bilimleri Enstitüsü  
Bilgisayar Mühendisliği Anabilim Dalı  
Aralık-2006

## JÜRİ VE ENSTİTÜ ONAYI

Demet Gülseren'in "Mobil İletişim Teknolojilerinin Öğrenci Bilgi Sistemlerinde Kullanımı ve Bir Uygulama" başlıklı Bilgisayar Mühendisliği Anabilim Dalındaki, Yüksek Lisans Tezi ..... tarihinde, aşağıdaki jüri tarafından Anadolu Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

	Adı-Soyadı	İmza
Üye (Tez Danışmanı)	: Prof. Dr. YAŞAR HOŞCAN	.....
Üye	: Prof. Dr. ALİ GÜNEŞ	.....
Üye	: Yard. Doç. Dr. HAKAN KAĞNICIOĞLU	.....

Anadolu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun  
..... tarih ve .....sayılı kararıyla onaylanmıştır.

Enstitü Müdürü

## ÖZET

**Yüksek Lisans Tezi**

### **MOBİL İLETİŞİM TEKNOLOJİLERİNİN ÖĞRENCİ BİLGİ SİSTEMLERİNDE KULLANIMI VE BİR UYGULAMA**

**Demet GÜLSEREN**

**Anadolu Üniversitesi  
Fen Bilimleri Enstitüsü  
Bilgisayar Mühendisliği Anabilim Dalı**

**Danışman: Prof. Dr. Yaşar HOŞCAN  
2006, 139 sayfa**

Telekomünikasyon alanındaki en büyük gelişmeler mobil iletişim ve veri iletişimde yaşanmaktadır. Kablosuz bir ortamda istenilen anda, istenilen noktadan ihtiyaç duyulan bilgiye erişim mobil iletişim teknolojileri sayesinde gerçekleştirilir. Bu yüzden de gün geçtikçe sabit sistemler yerine taşınabilir sistemler tercih edilmektedir.

Yüksek lisans çalışması olarak sunulan bu çalışmada statik ve dinamik bilgilerden oluşan bir öğrenci bilgi sistemi hazırlanmış ve yüksek öğretim kurumlarında okuyan öğrencilerin sınav sonuçları ve ders programı gibi çeşitli bilgilere cep telefonu üzerinden istedikleri konum ve zamanda erişebilmesi amaçlanmıştır.

Öncelikle mobil iletişimin genel yapısı, gelişim evreleri ve mimari özellikleri, öğrenci bilgi sistemlerinin kullanım ve önemi, daha sonra da PhpMyAdmin ve MySQL özellikleri anlatılmıştır. Uygulama bölümünde ise hareketli ortamlarda cep telefonu kullanarak öğrencilerin not ve çeşitli bilgilere erişmesini sağlayan PhpMyAdmin’de hazırlanmış uygulama örneği sunulmuş ve geliştirilen bu uygulama tanıtılmıştır.

**Anahtar Kelimeler:** Mobil iletişim, Öğrenci Bilgi Sistemi, GSM, WAP, PhpMyAdmin, MySQL

## **ABSTRACT**

### **Master of Science Thesis**

## **THE USE OF MOBILE COMMUNICATION TECHNOLOGY IN STUDENT INFORMATION SYSTEMS AND AN APPLICATION ON IT**

**Demet GÜLSEREN**

**Anadolu University  
Graduate School of Sciences  
Computer Engineering Program**

**Supervisor: Prof. Dr. Yaşar HOŞCAN  
2006, 139 pages**

Greatest development in telecommunication field is in mobile and data communication. In a wireless environment, accessing the desired data at any time is managed by mobile communication technologies. Because of these reasons, mobile systems are preferred against stationary ones day by day.

In this work presented as a master's thesis a student information system consisting of static and dynamic information is prepared. And it is aimed at providing the university students with an opportunity to access their exam result or course schedule via mobile phone in any place and any time.

Especially the general structure of the mobile communication, its developmental phases architectural characteristics the usage and significance of student information systems then PhpMyAdmin and MySQL characteristics are mentioned. In the application part, PhpMyAdmin, which enables the students to access various information and their exam grades using mobile phone in mobile environments, was prepared, application sample was presented, and the developed material was introduced.

**Keywords:** Mobile communication, Student information systems, GSM, WAP, PhpMyAdmin, MySQL

# İÇİNDEKİLER

	<u>Sayfa</u>
<b>ÖZET</b> .....	<b>i</b>
<b>ABSTRACT</b> .....	<b>ii</b>
<b>İÇİNDEKİLER</b> .....	<b>iii</b>
<b>ŞEKİLLER DİZİNİ</b> .....	<b>v</b>
<b>ÇİZELGELER DİZİNİ</b> .....	<b>vii</b>
<b>SİMGELER ve KISALTMALAR DİZİNİ</b> .....	<b>viii</b>
<b>1. GİRİŞ</b>	<b>1</b>
<b>2. MOBİL İLETİŞİM</b>	<b>3</b>
2.1. Mobil İletişim Sistemlerinin Türkiye’de ve Dünyadaki Gelişimi .....	3
2.2. Mobil İletişim Teknolojileri ve Standartlar .....	10
2.2.1. Birinci nesil (Analog) sistemler .....	11
2.2.2. İkinci nesil (Sayısal) sistemler .....	12
2.2.3. 2.5 Nesil (2.5G) sayısal ses ve veri sistemleri .....	25
2.2.4. Üçüncü nesil (genişbant) sistemler .....	34
<b>3. ÖĞRENCİ BİLGİ SİSTEMİ VE KULLANILAN ARAÇLAR</b>	<b>48</b>
3.1. Öğrenci Bilgi Sistemi .....	48
3.2. Bilgiye Zamanında Erişim ve Öğrenci Bilgi Sistemi’nin Önemi .....	48
3.3. Veritabanı Kullanma Sebepleri .....	49
3.4. Web’de Veritabanı Kullanımı .....	49
3.5. PHP’nin Temelleri ve Avantajları .....	50
3.5.1. PHP’nin yazım kuralları.....	51
3.5.2. Değişkenler .....	51
3.5.3. Sabitler .....	52
3.5.4. PHP’de kontrol yapıları.....	53
3.5.5. PHP’de döngüler .....	55
3.5.6. Fonksiyonlar.....	56
3.6. PHP’nin Özellikleri ve PHP ile Yapabileceklerimiz.....	57
3.7. MySQL Özellikleri .....	58

3.8. MySQL ve PHP Kullanımı.....	58
3.8.1. MySQL'e bağlanması ve veritabanı seçilmesi.....	60
3.8.2. MySQL veritabanında sorgu yapılması .....	60
3.8.3. MySQL bağlantısının kapatılması.....	61
<b>4. UYGULAMA</b> .....	<b>62</b>
4.1. Problem.....	62
4.2. Amaç.....	62
4.3. Yöntem .....	62
4.4. Veritabanının Oluşturulması.....	63
4.5. Kontrol Panel .....	63
4.5.1. Programlar .....	66
4.5.2. Dersler .....	68
4.5.3. Öğrenciler.....	69
4.5.4. Haftalık ders programı .....	71
4.5.5. Sınav takvimi .....	72
4.5.6. Sınav sonuçları .....	73
4.5.7. Yönetim.....	74
4.5.8. Personel .....	75
4.5.9. Yurtlar .....	77
4.5.10. Duyurular .....	79
4.5.11. Sık sorulan sorular.....	80
4.5.12. Çoklu veri girişi.....	83
4.5.13. Statik sayfalar.....	84
4.6. Wap Sayfası.....	86
<b>5. SONUÇ</b> .....	<b>94</b>
<b>KAYNAKLAR .....</b>	<b>95</b>
<b>Ek-1 Kontrol Panel Modülünün Program Kodları .....</b>	<b>98</b>
<b>Ek-2 WAP Modülünün Program Kodları .....</b>	<b>127</b>

## ŞEKİLLER DİZİNİ

1.1. Devre Anahtarlama ve Paket Anahtarlama Sistemi.....	2
2.1. Ülkelerin 3G'ye Doğru Takip Ettiği Teknolojiler [5].....	10
2.2. GSM Ağının Yapısı [14].....	15
2.3. SS7 Networks [15].....	16
2.4. Dalga Boyu Modülasyonu Uygulanışı[22].....	22
2.5. TDMA'nın Kullandığı Farklı Modeller.....	23
2.6. TDMA [21].....	24
2.7. FDMA[21].....	24
2.8. CDMA[20].....	25
2.9. Hat Tahsisli Sistem.....	27
2.10. Paket Anahtarlama Sistemi [12].....	28
2.11. Wap ve İnternet Protokol Yapısı.....	33
2.12. Wap Network Protokol Yapısı[23].....	34
2.13. IMT-2000 (UMTS) Geçiş Adımları.....	36
2.14. Küresel Mobil Erişim.....	39
2.15. 1, 2 ve 3. Nesil Sistemlerinin Bir Arada Çalışması[11].....	41
2.16. UMTS'in Sağladığı Hizmetler [11].....	42
2.17. UMTS Geçiş Aşamaları.....	42
2.18. UMTS Şebeke Blok Gösterimi.....	43
2.19. UTRAN Şebeke Yapısı.....	45
3.1. PHP ve MySQL'in bir arada çalışması.....	59
4.1 Kontrol Panel Login Ekranı.....	64
4.2. Kontrol Panel Anasayfa.....	65
4.3. Programlar.....	66
4.4. Yeni Program Ekleme.....	67
4.5. Dersler Sayfası Listele.....	68
4.6. Yeni Ders Ekleme.....	69
4.7. Öğrenciler Sayfası.....	70
4.8. Yeni Öğrenci Ekleme Sayfası.....	71
4.9. Haftalık Ders Programı.....	72

4.10. Sınav Takvimi .....	72
4.11. Sınav Sonuçları Sayfası .....	73
4.12. Yönetim Sayfası .....	74
4.13. Yeni Yönetici Ekleme Sayfası .....	75
4.14. Personel Sayfası .....	76
4.15. Yeni Personel Ekleme Sayfası .....	77
4.16. Yurtlar Sayfası .....	78
4.17. Yeni Yurt Ekleme Sayfası.....	78
4.18. Duyurular Sayfası .....	79
4.19. Yeni Duyuru Ekleme Sayfası.....	80
4.20. Sık Sorulan Sorular Sayfası .....	81
4.21. Yeni SSS Ekleme Sayfası .....	82
4.22. Çoklu Veri Girişi Sayfası .....	83
4.23. Okulumuz Hakkında Sayfası .....	84
4.24. İletişim Bilgileri Sayfası .....	85
4.25. Tavşanlı Hakkında Sayfası.....	85
4.26. Akademik Takvim Sayfası.....	86
4.27. Wap Anasayfa .....	87
4.28. Okulumuz ve Tavşanlı Sayfası .....	88
4.29. Akademik Takvim Sayfası.....	88
4.30. Haftalık Ders Programı Sayfası .....	89
4.31. Haftalık Ders Programı Sayfası .....	89
4.32. Sınav Sonuçları .....	90
4.33. Sınav Sonuçları Şifre Girişi .....	90
4.34. Sınav Sonuçlarının Gösterilmesi.....	91
4.35. Duyurular .....	91
4.36. Sık Sorulan Sorular .....	92
4.37. Sık Sorulan Sorular Detayı .....	92
4.38. İletişim Bilgileri .....	93



## ÇİZELGELER DİZİNİ

2.1. 1, 2 ve 3. Nesil Sistemlerin Karşılaştırmalı Tablosu.....	6
2.2. Yıllara Göre NMT Abone Sayıları (1987-2005) [9].....	7
2.3. GSM Üye İstatistikleri [12].....	8
2.4. GSM 900 ile GSM 1800'ün temel özellikleri[19] .....	19

## SİMGELER ve KISALTMALAR DİZİNİ

1G	: Birinci jenerasyon. Mobil analog sistemler için
2G	: İkinci jenerasyon. Mobil dijital sistemler için ses ve düşük hızda veri iletimi
3G	: Üçüncü jenerasyon. Kablosuz, yüksek hızlı veri iletimi
3GPP	: Third Generation Partnership Project
AMPS	: Advanced Mobile Phone Service
BSC	: Base Station Controller
CDMA	: Code Division Multiple Access
CN	: Core Network
CS	: Circuit Switched
DCS 1800	: Digital Cellular System 1800
DECT	: European Digital Enhanced Cordless Telecommunications
EDGE	: Enhanced Data Rate for GSM Evolution
ETSI	: European Telecommunications and Standards Institute
FDD	: Frequency Division Duplex
HLR	: Home Location Register
HSCSD	: High Speed Circuit Switched Data
IMT-2000	: International Mobile Telecommunications-2000
ITU	: International Telecommunication Union
MS	: Mobile Station – Mobil istasyon
MSC	: Mobile Switching Center – Mobil anahtarlama merkezi veya santral
NMT	: Nordic Mobile Telephone
PDC	: Personal Digital Communications
PS	: Packet Switched
PSTN	: Public Switched Telephone Network
SIM	: Subscriber Identity Module
TACS	: Total Access Communication System
TDMA	: Time Division Multiple Access
UE	: User Equipment

UMTS	: Universal Mobile Telecommunications System
UTRAN	: UMTS Terrestrial Radio Access Network
W-CDMA	: Wideband Code Division Multiple Access
WAP	: Wireless Application Protocol – Kablosuz Uygulama Protokolü
WARC	: World Administrative Radio Conference
WRC	: World Radiocommunication Conference

## 1. GİRİŞ

Günümüzde hemen hemen her alanda kullanılan ve bilgiye erişimde büyük kolaylıklar sağlayan internetin mobil (ya da kablosuz) hale dönüşmesiyle yani cep telefonu gibi mobil cihazlar üzerinden kullanılmasıyla birlikte daha da esnek bir kullanım alanına sahip olunmuştur. Böylelikle istediğimiz anda, istediğimiz noktadan, sadece ihtiyaç duyduğumuz veriye kablo ya da herhangi bir bağlantı olmaksızın küçük mobil cihazlar ya da cep telefonları sayesinde erişebilmekteyiz.

Bu tez çalışmasında; yükseköğretim kurumunda okuyan öğrenciler için bir öğrenci bilgi sistemi hazırlanmış ve okulları tarafından öğrencilerle paylaşılacak üzere hazırlanmış olan statik ve dinamik bilgilere cep telefonları ile internet üzerinden ulaşmaları amaçlanmıştır. Böylece okul, ders, iletişim ile ilgili bilgilere ya da açıklanan sınav sonuçlarına konum ayırt etmeksizin anında ulaşmaları sağlanmaktadır.

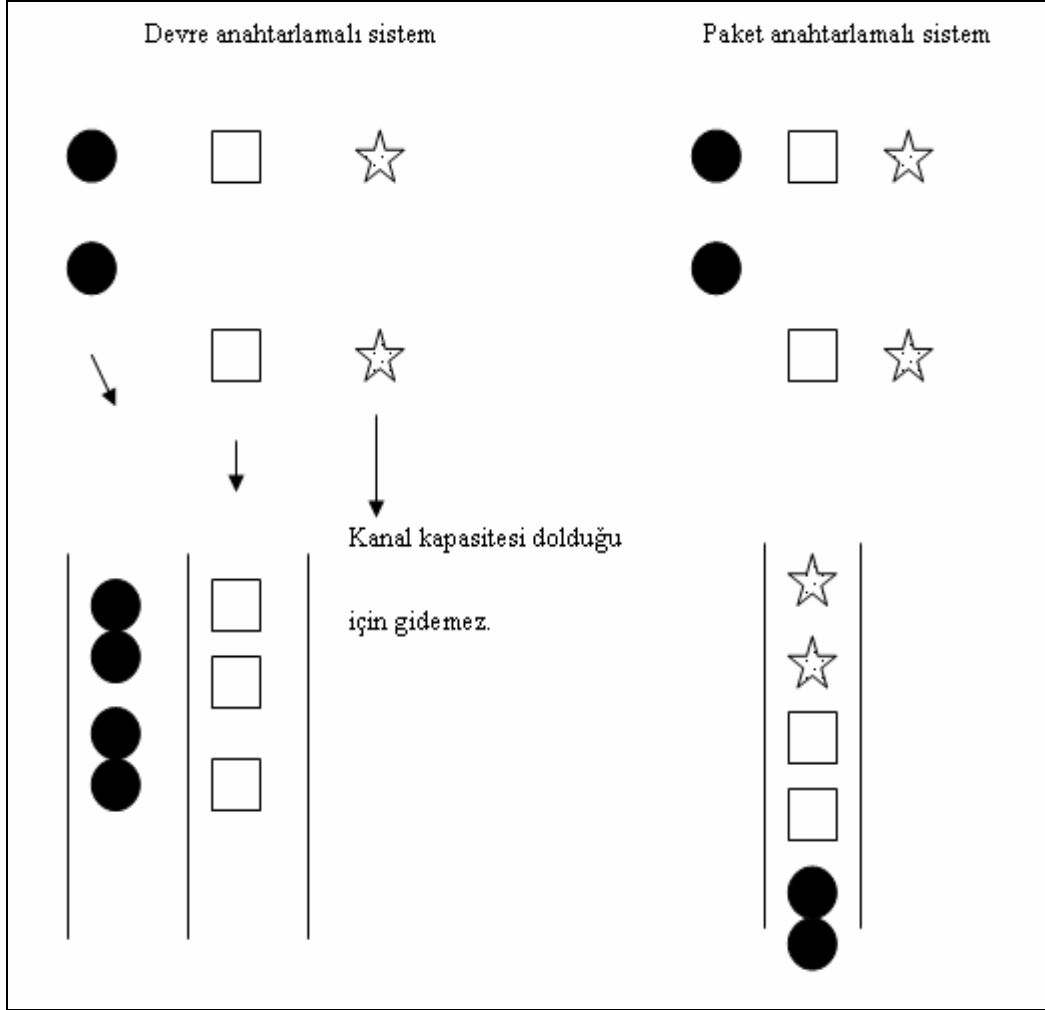
1980'li yıllarda analog sistemler sadece ses iletişiminin gerçekleşmesini sağlıyordu. Analog sistemlerden sayısal sistemlere geçiş 1990 yıllarında gerçekleşti. Böylece ses iletişiminin yanında veri iletişimi de gündeme geldi. Şimdi ise hızlı bir şekilde veri iletişimde büyük adımlar atılmaktadır, devre anahtarlama yapıları yerini paket anahtarlama sistemlerine bırakmış durumdadır.

Devre anahtarlama sistemleri veri transferinin gerçekleşmesi için önce iletişim yapacak düğümler arasında bir bağlantının kurulması ve bir yol belirlenmesi prensibine dayanır. Aktarım sırasında veri paketlerinin sırası bozulmaz, alıcıya gönderildikleri sırada ulaşırlar. Düğümler arasında bir bağlantı olduğu için veri paketlerinin içine alıcı ve gönderen adresleri konulmaz. Bu sistemde ses ve veri iletişimi yapılmak istendiğinde bir kanalın bu işleme ayrılması gerekir. Bu iletişim kanalını tamamen meşgul eder.

Paket anahtarlama ise bir veri gönderilirken belirli uzunluktaki parçalara ayrılır, her parça alıcı ve gönderici adreslerini içererek birbirinden bağımsız yollardan geçerek iletilir.

Paket anahtarlama sistemlerinde kanalın boşalmasını beklemeksizin kullanıcı verileri iletilebilir. Devre anahtarlama ve paket anahtarlama

sistemlerin işleme mantığı Şekil 1.1’de gösterilmiştir. Mobil telefon sistemlerinde paket anahtarlama sistemlerinin ilk uygulaması GPRS (General Packet Radio System) ile başlamıştır.



Şekil 1.1: Devre Anahtarlama ve Paket Anahtarlama Sistemi

## 2. MOBİL İLETİŞİM

Mobil haberleşme cep telefonlarıyla kurulan iletişimdir. Bunun için gereken alt yapı ise, yeterli sayıda baz istasyonunun oluşturduğu telsiz ağıdır. Bu istasyonlar alıcı ve verici görevi üstlenirler. Gelen konuşmaları mobil telefona gönderir, oradan geri gönderilen cevaplarıysa telsiz ağına geri gönderirler. Mobil telefonların kapsama alanı birkaç km (açık ideal alanda baz istasyonu-telefon haberleşmesi 30–70 km.ye kadar çıkabilir; fakat şehir içlerinde, bodrum-derin bina içi gibi yerlerde bu mesafe birkaç yüz metreye kadar düşebilir) ile sınırlı olduğundan alanı genişletebilmek için pek çok baz istasyonuna ihtiyaç vardır.

Mobil telekomünikasyon sistemlerinde iletilmek istenen mesaj (ses ve ya veri), taşıyıcı görevi yapan elektromanyetik bir dalga sayesinde iletilir. Belli bir frekansta gönderici ile yayınlanan bu dalga, bir anten vasıtasıyla alıcıya aktarılır. Alıcıda elde edilen mesaj, ses ise bir hoparlöre, veri ise başka bir dış birime (yazıcı, veri terminali v. b)gönderilir ve bu yolla iletişim gerçekleşir.[1]

Sonuç olarak kablosuz iletişim; haberleşmede esneklik, kampus içi haberleşme, kablolu masraflarının ve yer değişikliği maliyetinin azalması, LAN ortamlarının kablosuz olarak birbirine bağlanabilmesi, her yerde çalışma kolaylığı ve verimliliğin artırılmasını sağlar. Bu yüzden de gün geçtikçe sabit sistemler yerine taşınabilir sistemler tercih edilmektedir.

### 2.1. Mobil İletişim Sistemlerinin Türkiye’de ve Dünyadaki Gelişimi

Mobil iletişim sistemleri, günümüze gelinceye kadar önemli bir süreç yaşamıştır. Bu süreç, birinci nesil analog sistemlerden başlayarak çeşitli değişim ve gelişmeler göstererek günümüze kadar süren bir süreci ifade etmektedir.

Radyo iletişiminin temelleri “Hertz” ile atılmıştır fakat 19. yüzyılın sonlarında “Marconi” nin deneyleri radyo iletişiminin ilk başlangıcı kabul edilmiştir. [2]

Devamlı hareket halinde olan kişilerin telefon haberleşmesinde karşılaştıkları imkânsızlıklar, elektromanyetik dalgaların telekomünikasyon aracı olarak kullanılabilmesi fikrinin doğuşu ve bu yöndeki çalışmaların başlaması ile çözüme kavuşmaya başlamış ve bu gelişmeler ışığında mobil iletişimin ilk

meyveleri A.B.D’de 1940’lı yılların sonlarında, Avrupa’da ise 1950’li yılların başlarında tek hücreli analog araç telefonlarının kullanılmaya başlanması ile alınmıştır. Bunu izleyen adım, 1970’lerin sonlarında hücreli analog mobil telefonların kullanılmaya başlanmasıyla atılmıştır. Bu sistemler, birinci nesil (1G) analog teknolojiyi kullanmakta olup, kullanıcıların zamanla artan ses kalitesi, kapasite, kapsama alanı gibi ihtiyaçlarına cevap vermekte yetersiz kalması, ikinci nesil (2G) sayısal teknolojiye doğru yol alınmasını zorunlu kılmıştır.[3]

1928 yılında ilk karasal mobil radyo sistemi (Land Mobile Radio System) polis araçları arasında mesaj iletimini sağlamak için geliştirilmiştir. 1933’te Bayonne New Jersey polis ofisi tarafından iki yönlü mobil radyo ses sistemi geliştirildi. İlk mobil radyo alıcı vericileri çok basit, gürültülü, ağır ve hantaldı. Güç tüketimleri çok fazlaydı.

Bugünkü anlamda ilk hücreli cep telefonu fikri 1940’lı yılların ortalarında belirdi. 1947 yılında AT&T şirketi FCC’ye (Federal Communication Commission) mobil telefonlar için geniş bir radyo frekans aralığının tahsis edilmesini önerdi. Bu öneri kabul görmeye birlikte uygulaması yavaş oldu. Aynı yıl Bell laboratuvarları hücreli radyo çalışmalarına başladılar. Ancak uzun bir süre pratik uygulamaya geçemediler. Bu süre teknolojiye olacak gelişmeleri bekleyerek geçti. 1968’de frekanslardaki genişletme tamamlandı ve IMTS (Integrated Mobile Telephone System) olarak bilinen geliştirilmiş mobil telefon sistemi kuruldu. [2]

1978 yılında “Bell telephone” 2000 abonesi ile ilk hücreli telefonu test etti. 1980’li yılların gelişi ile mobil iletişim alanındaki gelişmeler baş döndürücü bir hız kazandı. İlk kurulan hücreli sistemler analog teknolojiyi kullanıyordu ve sadece ses iletimine imkân veriyordu.[4]

Seksenli yılların başında ilk hücreli erişim ağları oluşmaya başladığında mobil telefonlar (o zamanki adıyla araç telefonları) lüks sayılmaktaydı. “Analog haberleşme sistemi” adı verilen ilk kablosuz iletişim sistemi İskandinav ülkeleriyle İngiltere’de başlamış, daha sonra da Fransa ve Almanya’ya yayılmıştır. NMT (Nordic Mobile Telephony) standardında telsiz erişim sağlayan bu ürünlere sonradan “1. Nesil” adı verilmiştir. NMT analog bir sistem olup sadece ses iletimi için tasarlanmıştır. Bu türün diğer örnekleri olan TACS (Total

Access Communication System), E-TACS (Enhanced TACS), AMPS (Advanced Mobile Phone System) farklı standart ve frekans bantlarında dünyanın birçok köşesinde uygulama alanı bulmuştur.

Birinci nesil mobil sistemler analog oldukları için hem kapasiteleri (abone kapasitesi) çok düşük hem de güvenlik ve performans açısından zayıftırlar.[5]

2G mobil telefonlar ise 1991'in ortalarında piyasaya sürülmüş ve kullanımı büyük bir hızla yaygınlaşmıştır. Bugün kullandığımız "Global system for mobile communication" (GSM)<sup>1</sup> standartlarındaki cep telefonları, 2G sayısal teknolojiyi kullanan sistemlere bir örnek teşkil etmektedir. "Code Division Multiple Access" (CDMA) ve "Personal Digital Communication" (PDC), GSM dışındaki diğer 2G standartlarına örnek olarak gösterilebilir. Ancak GSM; GSM900, GSM-railway (GSM-R), GSM1800, GSM1900 ve GSM400 ile en yaygın kullanılan 2G hücreli sayısal mobil telekomünikasyon standardını teşkil etmektedir.[6]

GSM, sayısal olması ve kullandığı radyo erişim teknikleri açısından birçok yenilikler getirmesine karşılık erişilen son nokta olmayıp, daha gelişmiş ve küresel sistemlere geçişi sağlayacak bir aşama olarak kabul edilmektedir. İkinci nesil GSM sistemlerindeki gelişmeleri, her biri daha hızlı veri iletimine imkan sağlayan HSCSD (Yüksek Hızda Devre Anahtarlama Veri-High Speed Circuit Switched Data), GPRS (Genel Paket Telsiz Hizmeti-General Packet Radio Service) ve EDGE (Gsm Evrimi için Geliştirilmiş Veri Hızı-Enhanced Data Rate for GSM evolution) sistemleri izlemektedir. (Çizelge 2. 1) [7]

Bu teknolojiler, 3G'ye giden yolda son basamaklar olarak görülmekte ve topluca 2,5G teknolojisi olarak kabul edilmektedir.

---

<sup>1</sup> John Scourias'ın "A Brief Overview of GSM" dokümanının 2.sayfasında (<http://kbs.cs.tu-berlin.de/~jutta/gsm/js-intro.html>) GSM'in gerçekte, bu alanda faaliyet göstermek üzere CEPT tarafından kurulan çalışma grubunun adının (Groupe Special Mobile) kısaltılmış şekli olduğu belirtilmektedir.



Bilgi aktarım hızı yönünden bir karşılaştırma yapılacak olursa; GSM 1. safha ile 9.6 kbit/s (eğer şebekede uygun ayarlar yapılmışsa 14.4 kbit/s), HSCSD'de 4 kanal aynı anda kullanılabilir dolayısı ile  $9.6 \times 4 =$  yaklaşık 38kbit/s uygun ayarlar ile  $14.4 \times 4 =$  yaklaşık 57.6 kbit/s, GPRS ile 171.2 kbit/s, EDGE ile 384 kbit/s ve 3G ile ise 2Mbit/s hızlarında bilgi transferi yapılmaktadır.

**Çizelge 2.1: 1, 2 ve 3. Nesil Sistemlerin Karşılaştırmalı Tablosu<sup>2</sup>**

		Teknoloji	Bandgenişliği (Kbit/s)	Özellikler
1. Nesil Mobil	AMPS/ NMT	Gelişmiş Mobil Telefon Sistemi- (Advanced Mobile Phone System)	9.6	<ul style="list-style-type: none"> <li>Analog ses hizmeti</li> <li>Veri kapasitesi yok</li> </ul>
		Nordic Mobil Telefon- (Nordic Mobile Telephony)		
2. / 2.5. Nesil Mobil	GSM	Küresel Mobil İletişim Sistemleri (Global System for Mobile Communication)	9.6 → 14.4	<ul style="list-style-type: none"> <li>Sayısal ses hizmeti</li> <li>Gelişmiş mesaj gönderme hizmeti</li> <li>Evrensel dolaşım</li> <li>Devre anahtarlamalı veri</li> </ul>
		HSCSD Yüksek Hızda Devre Anahtarlamalı Veri (High-Speed Circuit Switched Data)	9.6 → 57.6	<ul style="list-style-type: none"> <li>Gelişmiş GSM</li> <li>Daha hızlı veri hızı</li> </ul>
		GPRS Genel Paket Telsiz Hizmeti (General Packet Radio Service)	9.6 → 115	<ul style="list-style-type: none"> <li>Gelişmiş GSM</li> <li>Her zaman bağlantı imkanı</li> <li>Paket anahtarlamalı veri</li> </ul>
		EDGE GSM Evrimi için Geliştirilmiş Veri Hızı (Enhanced Data Rate for GSM Evolution)	64 → 384	<ul style="list-style-type: none"> <li>Gelişmiş GSM</li> <li>GPRS'den daha hızlı</li> </ul>
3.Nesil Mobil	IMT-2000/ UMTS	Uluslararası Mobil İletişim 2000/ International Mobile/ Evrensel Mobil İletişim Sistemi (Telecommunications 2000 / Universal Mobile Telecommunications System)	64 → 2,048	<ul style="list-style-type: none"> <li>Her zaman bağlantı imkanı</li> <li>Küresel dolaşım</li> <li>IP-olanağı</li> </ul>

<sup>2</sup>Forrester research

Türkiye’de 1985 yılında mobil iletişim hizmetini sağlamak için çalışmalara başlanmıştır. Analog NMT (Nordic Mobile Telephone) sistemi, 1986 yılında Ankara ve İstanbul’da aktif hale geçmiş, 1994 yılında ise 93503 abone sayısına ulaşmıştır. Yıllara göre NMT abone sayısı Çizelge 2.2’de gösterilmektedir.

Çizelge 2.2: Yıllara Göre NMT Abone Sayıları (1987-2005) [8]



Kanal ve baz istasyonlarının artırılmasına rağmen Çizelge 2.2’deki değerlerden anlaşılacağı gibi abone sayısında azalmalar görülmektedir. Bu nedenle NMT sistemine yatırım yapılması durdurulmuş, mevcut hali ile işletilmeye devam edilmektedir.

Cep telefonları, iletim teknolojisinde yaşanan büyük değişimin sonucudur. Teknolojik yenilik asıl olarak kısıtlı bir kaynak olan frekans spektrumu’nun (frekans genişliği) en verimli biçimde kullanılabilmesi ihtiyacı tarafından yönlendirilmiştir. Mobil teknolojinin ilk uygulamaları olan birinci nesil mobil teknolojiler 450 MHz frekans civarında çalışırlar.[9]

Avrupa’da GSM sistemi kullanılır ancak bu sistem Amerika’daki GSM ile uyumlu değildir. Japonya’da kullanılan 2G mobil iletişim tekniği ise paket anahtarlama tekniğini kullanan PDC’dir (Personal Digital Communications).

Çizelge 2.3’de GSM üye istatistikleri gösterilmektedir.

**Çizelge 2.3: GSM Üye İstatistikleri [10]**

	Millions	2003				2004				2005				2006	
		Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2
World		1,187.9	1,239.4	1,302.2	1,382.9	1,453.8	1,525.4	1,612.3	1,714.1	1,820.6	1,920.6	2,027.2	2,177.1	2,245.8	2,405.8
GSM		851.5	894.7	945.4	1,012.0	1,070.9	1,131.6	1,207.2	1,296.0	1,378.5	1,467.6	1,561.7	1,709.2	1,796.8	1,941.6
3GSM		0.4	1.0	1.6	2.8	4.4	7.5	11.4	16.3	24.0	29.9	37.9	50.0	61.2	74.7
CDMA		102.3	99.8	99.3	98.9	96.6	93.6	90.6	87.4	72.5	68.5	63.8	62.4	42.5	37.0
CDMA 1X		45.7	56.5	67.5	80.1	93.5	106.6	118.9	131.9	167.6	182.9	197.2	213.1	213.3	225.0
CDMA 1X EV-DO		0.7	1.7	3.0	4.6	7.1	8.7	10.5	12.3	14.7	17.1	19.7	21.2	29.6	34.5
TDMA		100.7	99.8	99.8	100.1	98.1	95.6	92.8	90.0	82.8	79.0	71.9	48.5	31.9	26.1
PDC		57.5	58.2	58.5	58.1	57.7	56.7	55.7	54.2	51.6	49.5	46.5	46.3	42.6	38.5
IDEN		11.5	12.1	12.8	13.4	13.8	14.4	15.0	16.8	19.5	18.8	19.8	21.1	22.6	23.8
Analog		17.6	15.7	14.3	12.9	11.9	10.7	10.2	9.2	8.3	7.6	6.8	5.4	5.2	4.5

Aynı hattın birden çok kullanıcı tarafından paylaşıldığı 171. 2 kbit/s iletim hızını destekleyen paket anahtarlama tekniği olan GPRS 2000 tarihinde dünyada kullanılmaya başlanmış, İngiltere ve Almanya'dan sonra ise 3. ülke olarak Türkiye'de Ağustos 2000'de faaliyete geçmiştir.

HSCSD'de GSM şebekesi üzerine kurulabilecek bir teknolojidir ve hızlı anahtarlama tekniğine sahiptir. Hızlı veri iletişimi yapmak isteyenler birkaç abonenin kullandığı kanal genişliğini kullanarak 48 kbps hızlara ulaşabileceklerdir. HSCSD dünyada 1999 tarihinde kullanılmaya başlanmış Mart 2000'de Türkiye'de aktif hale gelmiştir.

3G sistemlerine doğru giden yolda en son adımı oluşturan EDGE (GSM Evrimi İçin Geliştirilmiş Veri Hızları – Enhanced Data Rates for GSM Evolution) ise GPRS'in gelişmiş bir hali olup GSM operatörlerine 3G'nin sunacağı hizmetleri sunan, ITU'nun 3G sistemler için seçtiği radyo arayüzü standartlarından biridir.

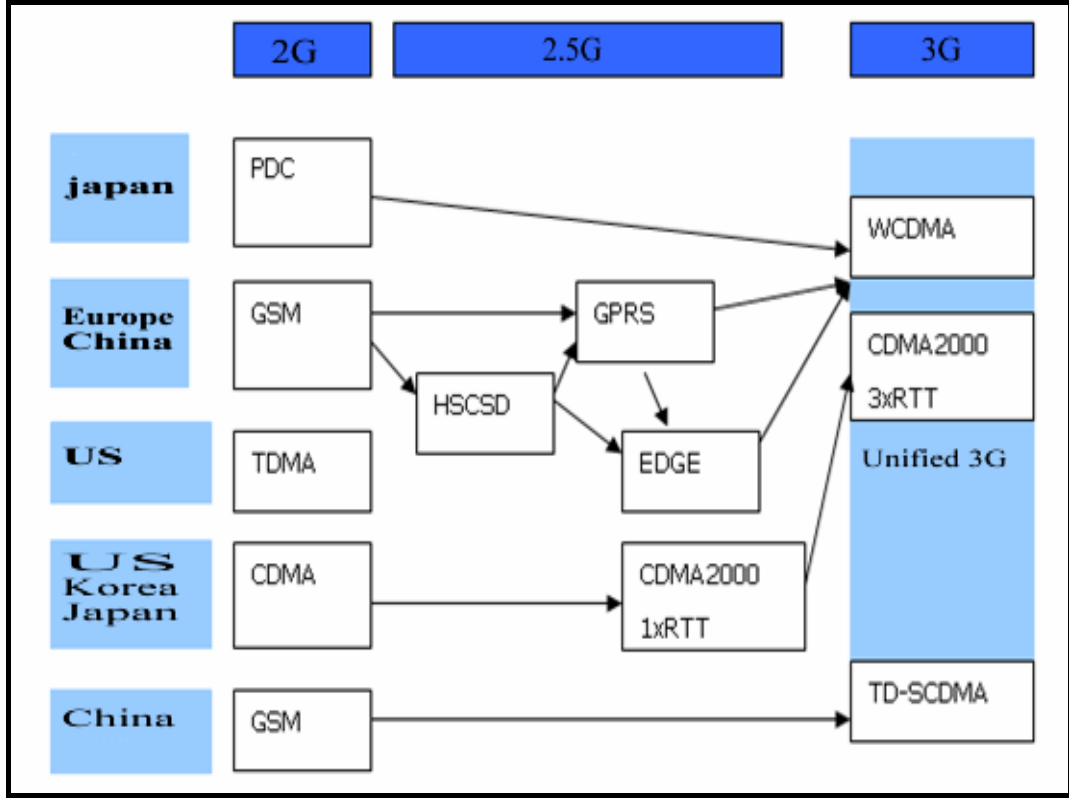
Farklı bölgelerin farklı teknolojiler tercih etmesinden kaynaklanan haberleşme sistemlerindeki uyumsuzluğu gidermek, sabit sistemlerin sağladığı kaliteli ve güvenilir iletişimi mobil iletişimle gerçekleştirebilmek, yeni gelir alanları açarak 2 Mbps hızında yüksek kalitede geniş bantta iletim ile küresel dolaşım platformu sağlayabilmek için 3. nesil sistemlere geçiş çalışmaları 1985 yılında başlatılmıştır.

1998 yılında ITU üçüncü nesil hücresel haberleşme standartlarının genel adı olarak IMT–2000 (International Mobile Telecommunications year 2000) kabul etmiş, aynı yıl Avrupa Haberleşme Standartları Enstitüsü (ETSI) Avrupa’da üçüncü nesil sistemler için kullanılacak standartları Universal Mobile Communications (UMTS) adı altında ITU’ye global standart önerisi olarak sunmuştur. USA’da ise, Kuzey Amerika’da kullanılmakta olan hücresel sistemlerden AMPS ve CDMA ile uyumlu olan CDMA-2000’ni 3G global standardı olarak önermiştir.[11]

Her bölge farklı bir 2G teknolojisi kullandığı için diğerlerinden farklı bir yoldan 3G’ye geçiş yapma yolunu izlemiştir. Dünyada en çok kullanım alanına sahip olan mobil iletişim standardı GSM ve TDMA, Uluslararası Telekomünikasyon Birliği (ITU) tarafından onaylanan üçüncü nesil telsiz ara yüz standardı olan EDGE ve WCDMA yolunu kullanarak 3. nesil hücresel mobil sistemlere geçişi benimsemiştir.

CDMA mobil iletişim standardını kullanan şebekeler ise CDMA2000 ile 3. nesile geçişi en uygun yol olarak bulmuşlardır. CDMA2000’de ITU tarafından 3. nesil standardı olarak tanımlanan CDMA şebekelerinin veri alış-veriş hızını ve ses kalitesini artıran bir teknolojidir.

Güney Kore’de kullanılan CDMA ve Japonya’da kullanılan PDC ise 3G’ye geçişte WCDMA ve CDMA2000 olmak üzere 2 farklı standardı kullanmıştır. Bu anlatılanların hepsi Şekil 2.1’de ülkelerin 3G’ye doğru takip ettiği teknolojilerde gösterilmiştir.[5]



Şekil 2.1: Ülkelerin 3G'ye Doğru Takip Ettiği Teknolojiler [5]

## 2.2. Mobil İletişim Teknolojileri ve Standartlar

Hüresel mobil sistemleri gelişim ve kullanım alanlarına göre 3 nesil olarak adlandırmak mümkündür. Buna göre; ilk olarak birinci nesil mobil iletişim sistemleri incelenebilir. Bunlar;

AMPS (Advanced Mobile Phone System)

TACS (Total Access Com. System)

NMT (Nordic Mobile Tlf.)

RC2000

CT-1

Net-C gibi analog sistemlerdir.

### **2.2.1. Birinci nesil (Analog) sistemler**

İlk kablosuz iletişim sistemi olan analog haberleşme sistemi İskandinav ülkeleriyle İngiltere’de kullanılmış, daha sonra Almanya ve Fransa’ya yayılmıştır. Yetersiz kapsama alanı, kararsız yapısı ve zayıf ses kalitesi ile telsiz iletişim sağlayan bu sisteme 1. nesil sistemler adı verilmiştir.

1947’de hücresel fikrinin ortaya atılışı ile birlikte sınırlı frekans spektrumu ile çok kullanıcı sistemler yapılabileceği görülmüştür. Gelişen teknoloji ile 1980’lerin başında analog, FM tam dubleks, kanallı -trunk- erişimli hücresel mobil haberleşme sistemleri kullanıma girmiştir. Birinci nesil hücresel şebekeler olarak adlandırılan AMPS (Advanced Mobile Phone System-Gelişmiş Mobil Telefon Sistemi) 850 Mhz’de, TACS (Total Access Control System-Bütünsel Erişimli Haberleşme Sistemi) 900 Mhz’de, NTT 900 Mhz’de FM modülasyonu kullanırken NMT450 450 Mhz’de FM türevi PM kullanıyordu.[12]

#### **Advanced Mobile Phone System (AMPS)**

AT&T ve MOTOROLA tarafından 1979 yılında Chicago’da (Illinois, ABD) ilk hücresel sistem olan AMPS (Advanced Mobile Phone Services) gerçekleştirilmiştir. AMPS 800 ila 900 Megahertz (MHz) frekans bandını kullanır. Kuzey ve Güney Amerika’da kullanılan AMPS, Güney Kore, Yeni Zelanda, Singapur, Tayvan, İsrail, Tayland, Hong Kong ve Avustralya gibi ülkelerin yanında Asya/Pasifik bölgesinde kullanılan yaygın bir sistemdir.

AMPS’in bir benzeri olan TACS (Total Access Communication System) İngiltere’de British Telecom tarafından 1985’te kullanıma sunulmuştur. AMPS ve TACS çok kaliteli ses iletişimi sağlarken, veri iletişimde çok kısıtlı işlemler gerçekleştiriyorlardı. Ayrıca sık sık gecikmeler, bağlantı kopuklukları yaşanmasına ek olarak ücretler de çok yüksekti.

Sayısal hücrelere geçinceye kadar problemleri biraz olsun azaltmak üzere Motorola tarafından NAMPS (Darbant-AMPS) geliştirilmiştir. [13]

### **Nordic Mobile Telephone (NMT)**

1970'lerin sonlarında kullanılmaya başlanan analog hücreli mobil telekomünikasyon sistemlerinin Avrupa'da kullanılan standardı NMT (NORDIC MOBILE TELEPHONE)'dir. Araç telefonu olarak bilinen bu sistem 1981'den itibaren Avusturya, Hollanda, Fransa, Macaristan, Belçika ve Türkiye'de uygulanmıştır.

Kuzey ülkelerinin coğrafi koşullarına göre tasarlanan NMT (NORDIC MOBILE TELEPHONE) sistemi analog bir sistemdir ve sadece ses iletiminin gerçekleştirilmesini sağlar.

### **2.2.2. İkinci nesil (Sayısal) sistemler**

Analog sistemlerden sonra, sayısal haberleşme teknolojisinin kullanıldığı dijital sistemlere ikinci nesil sistemler adı verilir. Sayısal sistemlerin gelişmesiyle analog sistemlerde oluşan birçok problem çözülmüş, ses kalitesi artmış veri iletimi ise daha yüksek seviyelere ulaşmıştır.

2.nesil sistemler (GSM, CDMA, TDMA, DCS-1800, PDC) sayısal bir teknoloji olduklarından 1.nesil sistemler ile kıyaslandığında kapasite ve performans açısından daha güçlü bir durumdadırlar. Ancak 2.nesil sistemler dar frekans bantları ile çalıştıklarından veri transferinde etkin değillerdir. Bunun yanında yetersiz kapasite ve yüksek hızda veri transferinde yeterli olmaması 2. nesilin başlıca zayıf noktalarıdır. 2.nesil sistemlerde iletişim devre anahtarlamalı olduğu için hızlı değildir.

Dünyada 2.nesil sistemler olarak bilinen 4 farklı teknoloji kullanılmaktadır. Bunlar GSM, CDMA, TDMA, PDC'dir. Bu sistemlerden dünyada en çok alanda kullanılanı GSM'dir. GSM TDMA tabanlıdır ve Avrupa'da GSM sistemi kullanılmaktadır.

2G Amerika Birleşik Devletlerinde PCS (Personal Communication Services) olarak geçer ve 3 farklı teknolojiden oluşur. IS95-CDMA (Code Division Multiple Access), IS136-TDMA (Time Division Multiple Access) ve GSM 1900 (Global System for Mobile Communication).

Japonya’da ise 2.nesil teknolojisi olarak paket anahtarlama tekniğini uygulayan PDC (Personal Digital Communication) kullanılmaktadır. Böylelikle devre anahtarlama tekniğinden çok daha hızlı iletişim sağlanmaktadır.

İkinci nesil teknolojiler; veri iletiminde çok düşük ve pahalıya mal olacak 9.6 kbps veri iletim hızını desteklemektedir.

### **Global system for mobile (GSM)**

80’li yıllarda kullanılan birbirinden farklı iletişim teknolojileri daha sonra tek bir mobil sistem olarak düşünülmüştür. Kullanıcının istediği anda güvenli bir şekilde iletişim kurmasını sağlayan sayısal hücreli haberleşme denilen bu sistemler o zaman kullanılan farklı tüm teknolojilerin bir model haline getirilmesiyle oluşmuştur.

Bunun için 1982 yılında Avrupa’da hücreli haberleşme standartlarını oluşturmak için Group Special Mobile (GSM) kurulmuş ve mobil haberleşmede evrensel sistem olarak bilinen (GSM) yeni bir sistem oluşturulmuştur.

Kablolu iletişime alternatif olarak hazırlanan ve ilk 1982 yılında çalışmalara başlanan GSM; mobil iletişim sistemi olarak bilinir. İlk olarak dağınık bir yerleşim ve coğrafi yapısı nedeniyle kablosuz ortamla daha rahat iletişimin sağlanabileceği Finlandiya’da kullanılmış,1990’lı yıllarda tüm dünyaya yayılmıştır. Amerika ve Japonya ise GSM ile uyumlu olmayan kendi teknolojilerini geliştirmişlerdir.

### **GSM’in avantajları**

GSM’de veri transferi sistem içinde gerçekleştirilir ve analog sistemlere göre ses kalitesi daha yüksektir. Radyo frekansını verimli bir şekilde kullanarak çalışan GSM’de kullanıcılar konuşmalarının şifrelenmesi ile güvenli bir şekilde konuşma yapabilirler. Dünyanın her yerinde GSM uluslararası dolaşım imkânı sağlayarak büyük bir avantaj sağlamaktadır.



## **GSM mimarisi**

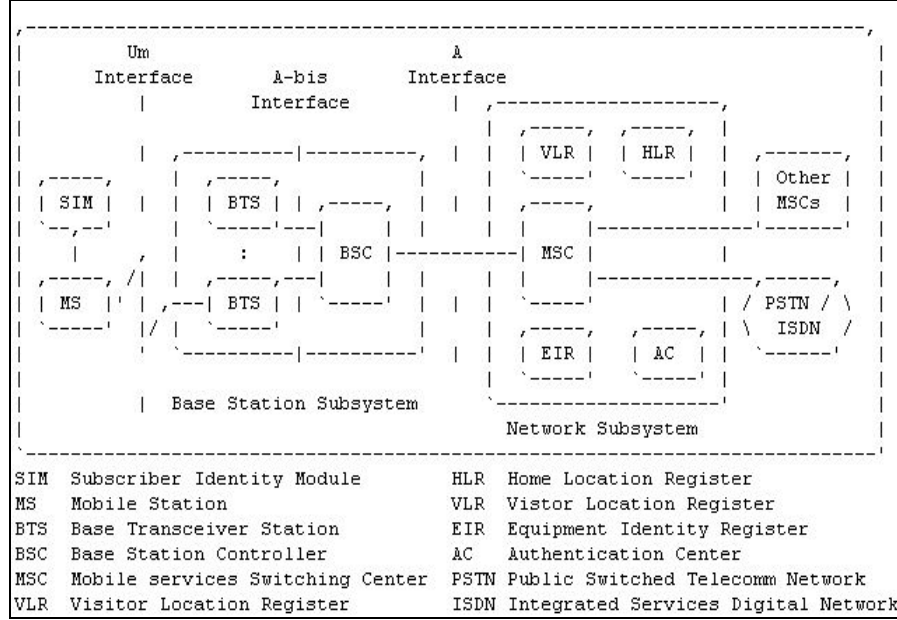
Üç ana bölümden oluşur: BSS, NMS ve OMS.

Hücresel sisteme özgü pek çok fonksiyon, mobil terminal cihazı ile kontak halinde olan baz istasyon alt sisteminde (BSS-Base Station Subsystem) gerçekleşir. Sistemin ekonomik olması için baz istasyonu alt sistemi iki fonksiyonel parçaya ayrılmıştır: Baz istasyonu (BTS-Base Transceiver Station) ve baz istasyonu kontrolörü (BSC-Base Station Controller). Yüksek kapasiteli tipik bir GSM şebekesi binlerce BTS içerir. Bu yüzden BTS'ler olabildiğince basit tutulmaya çalışılır ve şu fonksiyonları sağlar: Hava ara yüzünün işletimi, diğer şebeke elemanları ile iletişim için gerekli işaretleme ve iletim gibi. Bir BTS'e ait radyo kapsama alanı bir hücre olarak adlandırılır. Baz istasyonu alt sisteminin bağlama fonksiyonları ve radyo kanallarının kontrolü BSC tarafından sağlanır. BSC, bir çağırmanın başlangıcı esnasında radyo kanallarını tahsis eder ve çağırma sonlandığında kaynakları serbest bırakır. BSS servis alanındaki mobilite fonksiyonları BSC tarafından yürütülür. Bu görevleri yüzünden BSC, BTS'e göre daha pahalı ve karmaşıktır. BSS ile birlikte GSM'e özgü konuşma sinyalinin sabit telefon şebekelerinde kullanılan formata çeviren kod dönüştürücü birimi vardır. Bu birim aynı zamanda veri sinyalinin hız ayarlamasını da yapar.[12]

Aboneler arası bir görüşme bağlanması seçiciler tarafından yapılır. GSM'de bu birim MSC'dir. Bir MSC (Mobile Switching Center-Mobil Bağlama Merkezi) PSTN (Public Switched Telephone Network-Yerel Sabit Telefon Ağı), PSPDN (Packet Switched Public Data Network-Paket Bağlamalı Yerel Veri Şebekesi), ISDN (Integrated Services Digital Network-Birleşik Hizmetler Sayısal Şebekesi) gibi diğer şebekelerle haberleşir.[14]

Abone hareketliliğinden dolayı, şebeke alt sistemi (NSS-Network Subsystem) bir mobilite yönetim fonksiyonunu yerine getirmelidir. HLR (Home Location Register-Ev Yerleşim Kütüğü) lojik bir veri tabanı olup, her aboneye ait bilgileri ve her mobil istasyonun konum bilgilerini içerir. Gelen çağrımları yönlendirebilmek için, ziyaret edilen alanın adres bilgisi HLR'da depolanır. Yine lojik bir veri tabanı olan VLR (Visitor Location Register-Ziyaretçi Yerleşim Kütüğü), kendi alanına giren abonelerin bilgilerini HLR'dan alır, abone kendi alanını terk edinceye kadar tutar. Ayrıca, EIR (Equipment Identity Register-Cihaz

Kimlik Kütüğü) genelde MSC’de yer alır ve cihazın çalıntı olup olmadığını kontrol eder.[12]



Şekil 2.2: GSM Ağının Yapısı [14]

Bir GSM ağı, fonksiyonları ve ara yüzleri belirlenmiş olan birkaç fonksiyonel varlıktan oluşur. Şekil 2.2’de genel GSM ağının yapısı gösterilmiştir. GSM ağı 3 genel kısma ayrılabilir. Mobil istasyonuna abone tarafından ulaşılır, ana istasyon alt sistemi, mobil istasyon ile birlikte radyo bağlantılarını kontrol eder. Network alt sistemi, mobil servisi merkezinin ana parçasıdır. Bu sistem mobil telefonda sabit veya mobil telefonlara yapılan aramaları yönlendirme işini yapar, aynı zamanda mobil telefonun şebeke ile ilgili ayarlarını ve yetki işlemlerini de düzenler.[14]

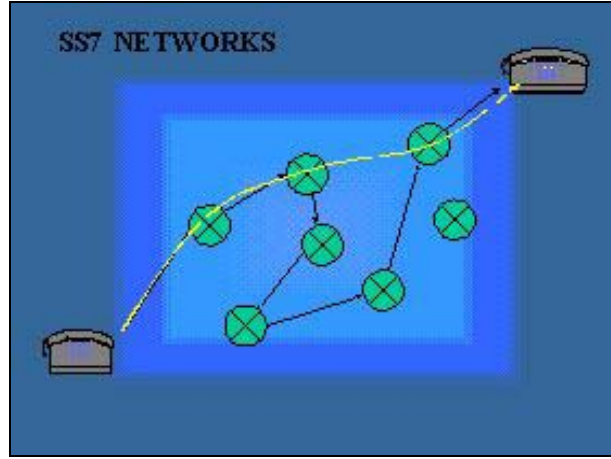
İşletim ve bakım sistemi (OMS- Operation and Maintenance System) şebekeden gelen hata mesajlarını tutar ve BSC ile BTS’in trafik yükünü kontrol eder.

Mobil terminale gelen bir çağırma, abonenin PLMN’inde (Public Land Mobile Network-Yerel Mobil Telefon Ağı) geçit MSC’ye yönlendirilir. HLR ve VLR’daki bilgiler kullanılarak, ziyaret edilen MSC üzerinden çağırma aboneye yönlendirilir. Geçit MSC ve ziyaret edilen MSC, abone kendi PLMN’inde ise

aynıdır. Şebeke elemanları arasındaki işaretleşme SS7 şebekesi (Ortak Kanal İşaretleşme Sistemi No:7) ile taşınır.[12]

### SS7 GSM network yapısı

GSM sistemi SS7 (Signalling System No7) prensibine bağlı olarak çalışmaktadır. Bu prensipte konuşmanın gerçekleşebilmesi için; bir x numarası y numarasını aradığında aranılan numara her seferinde bir diğer santrale sorularak değerlendirilir ve arama yapılır. Bu esnada santraller arasında sanal bir yol oluşur. Sonuç olarak y numarasının boş olması durumunda konuşma kanalı ayrılır ve konuşma için platform kurulur. SS7 yapısı Şekil 2.3’de gösterilmiştir.



Şekil 2.3: SS7 Networks [15]

### Hücresel sistem

Hücresel bir sistem üzerine kurulan GSM’de CELL (hücre), bu sistemdeki en küçük parçayı ifade eder. Her bir hücre 200 m-30 km arasındaki bir alana hizmet verebilir.

Kullanılan, hizmet verilen alana göre çeşitli hücre tipleri bulunur. Bunlar outdoor mekanlar da hizmet veren Mikrocell, geniş alanlarda Makrocell, konferans salonu, lobiler gibi özel mekanlarda kullanılan ise Picocell’dir. Bu hücrelerin birleşmesi ile ise Location Area (LA) bölgelerini oluşturur. Bir LA bölgesinde 1 MSC ve 1 de VLR bulunur. LA içerisinde bulunan abonelerin bilgileri VLR database’inde geçici olarak tutulur.

Konuşma esnasında gerçekleşen hareketlilik yer değişimi ise handover olayını gerektirir.

Handover; süregelen bir konuşmanın farklı kanallarda ya da hücrelerde bağlantısının kopmadan sürdürebilmesi için bağlantısı olayıdır. Ya da kısaca; aynı location area içinde bir hücreden diğerine devir olma durumudur. 2 farklı handover olayı bulunmaktadır;

- 1) Aynı cell içerisinde handover
- 2) Farklı celler arasında gerçekleştirilen handover

Aynı cell içerisinde handover BSC tarafından, farklı celler arasında handover MSC tarafından yapılır. Handover yapılabilmesi için, gidilen hücrede boş frekans tahsisinin yapılması gerekir. MSC önce gidilecek hücrede boş frekans olduğuna bakıp ondan sonra devreder.

Location update (yer güncelleme); cep telefonu BCCH (broadcast channel) kanalını devamlı olarak dinler. BCCH kanalı, baz istasyonu kimlik bilgilerini, tahsis edilmiş yada boşta olan frekans bilgilerini tutan bir kanaldır.

Sim kart içinde bulunan LA identity ile BCCH kanalından gelen ID tanımları birbirini tutmadığı zaman MS harekete geçer ve MSC ile bağlantı kurarak Location Update talebi yapar ve kendini tanıtır. Yeni Location bilgileri VLR ve HLR'a gönderilir. HLR ve ardından VLR bu bilgileri update eder. [15]

### **GSM'de veri servisleri**

Ses iletimi gibi GSM'de veri de devre bağlaşmalı olarak iletilir. İleri yönde hata düzeltmeden sonra 12 kbit/sn'lik bir hava ara yüzü hızı mümkündür (ses için 13 kbit/sn). Mobil istasyon terminal adaptasyon fonksiyonu ve MSC ara çalışma fonksiyonu arasındaki lojik link tabakasında; otomatik tekrar iletim (ARQ) olsun ya da olmasın maksimum 9.6 kbit/sn'lik kullanıcı hızı mümkündür.[12]

GSM standartları; PSTN, ISDN, PSPDN, CSPDN (Circuit Switched Public Data Network-Devre Bağlaşmalı Yerel Veri Şebekesi) ve direk erişim şebekeleri ile ara çalışma fonksiyonunu tanımlar. PSTN ile ara çalışma MSC ara çalışma fonksiyonuna yerleştirilmiş modemler tarafından sağlanır. Ayrıca bir

X.25 paket veri şebekesi ile ara bağlantı mümkündür. Bu, PAD (Packet Assembler and Disassembler-Paket ayrıştırıcı ve birleştirici) lere bir modem bağlantısı ile yürütülür. Asenkron terminal cihazın imkanı, asenkron servislerin kullanımını artırmış ve senkron servisleri kullanılmayan duruma getirmiştir. Bu servislere ek olarak, dar bantlı ISDN erişimi GSM’de önemli bir veri servisi olmaktadır. Bir GSM-ISDN veri çağırmasının avantajı uçtan uca bağlantı kurma süresinin kısılmasıdır. Tam bir sayısal ISDN bağlantısına çağırma kurma süresini önemli ölçüde azaltan modem el sıkışmasına gerek yoktur. GSM işaretlemesi ile birlikte çağırma kurma süresi bir PSTN bağlantısında 22 sn iken, bir GSM-ISDN bağlantısında 10 sn’dir. Ayrıca ISDN veri veya konuşma servisleri arasındaki farkı tek bir abone numarası kullanarak olanaklı kılar.

Saf veri servislerine ek olarak veri servisleri ile aynı taşıyıcıyı kullanan faksimile grup 3 tele servisleri de sağlanmaktadır. ITU (International Telecommunications Union-Uluslararası Telekomünikasyon Birliği) tavsiyesi T.30 bağlantı kurulması ve iletim fazı sırasında güvenilir bir uçtan uca bağlantı sağlamak üzere kullanılır.

İkinci nesil sistemler tarafından sağlanan bir veri servisi de kısa mesaj servisi (SMS-Short Message Service). 140 oktete kodlanmış 160 karakterlik metin mesajlarının iletimi ve alınmasını sağlayan bir tip paket servisi. Mesaj, bir mobil istasyon ile bir kısa mesaj servisi merkezi arasında iletilir (sakla-ilet tekniğine uygun olarak). Boş bir mobil istasyona doğru olan veya istasyondan gelen mesaj iletimi hızlı ortak kontrol kanallı üzerinden yürütülür. MS, bir çağırma kurulmasına sahipse mesaj bir yavaş ortak kontrol kanalı (SACCH-Slow Associated Control Channel) üzerinden iletilir. Kapsamlı işleme gereksinimlerine bağlı olarak, iletim birkaç saniye sürer. [12]

### **GSM 900 ve GSM 1800**

GSM servislerine olan talebin birçok ülkede artmasıyla 900 MHz’lik bant yeterli gelmemiş ve 1800 MHz bandında da şebekeler kurulmuştur.

Bunların bir kısmı GSM1800 şebekeleri, diğer bir kısmını da dual band şebekeleridir.

Dual band şebekeler yeni operatörlere ait olabildiği gibi, mevcut GSM 900 veya GSM 1800 operatörlerine diğer bant'dan frekans tahsis edilerek de kurulabilmektedir.

Dünyada dual band GSM işletmecileri için 900 MHz bandından tahsis edilen frekans aralığının 5 MHz' in altına çok fazla düşmemesi dikkat çekicidir. (GSM sisteminde bir hücredeki şebeke yayınının diğerleri ile karışmaması için gereken en düşük band genişliği 2,5 MHz' dir.) Standart GSM bandında (Primary GSM, P-GSM) yer bulunmayan ülkelerde, dual band şebekeler için, genişletilmiş GSM bandından (Extended GSM, E-GSM) frekans tahsisi yapılabilmektedir. (E-GSM bandı P-GSM bandına ek olarak 880-890 MHz ve 925-935 MHz aralıklarından oluşan 2\*10 MHz'i kapsar.) GSM 900 ile GSM 1800 sistemlerinde şebeke mimarisi, çoklu erişim yöntemi, çerçeve yapısı, modülasyon tekniği, hız, konuşma kodlaması, kanal kodlaması, sinyalleşme gibi konularda hiçbir fark bulunmamaktadır. GSM 900 ile GSM 1800'ün temel özellikleri çizelge 2.4'de verilmiştir. [16]

**Çizelge 2.4: GSM 900 ile GSM 1800'ün temel özellikleri[16]**

	GSM 900	GSM 1800
Frekans bandı	890-915 935-960	1710-1785 1805-1880
Bir TDMA çerçevesindeki time slot sayısı	8	8
Kanal aralığı	200 kHz	200 kHz
Kanal kapasitesi	124 kanal	374 kanal
Modülasyon tekniği	GMSK	GMSK
Modülasyon hızı	271 kbps	271 kbps
Konuşma kodlama hızı	13 kbps (6,5 kbps half rate)	13 kbps (6,5 kbps half rate)
En yüksek veri hızı	>150 kbps (GPRS ile)	>150 kbps (GPRS ile)
Çoklu erişim yöntemi	TDMA/FDMA	TDMA/FDMA
Hücre yarıçapı	<0,3 - 35 km	0,1 - 15 km
Mobil İstasyon çıkış gücü	2 - 8 W	0,25 - 4 W
Baz İstasyon çıkış gücü	2,5 - 640 W	2,5 - 40 W
Mobil İstasyon için duyarlılık eşiği	-102 / -104 dBm	-100 /-102 dBm
Baz İstasyon için duyarlılık eşiği	-104 dBm	-102 dBm

## **Ağ alt sistemi**

MSC (Mobile Switching Center) PSTN'deki Class 5 Central Ofisin görevini yapar. Ayrıca bunun dışında gerekli anahtarlamalar, mobil cihaz için çağrı işleme fonksiyonları gibi görevleri de vardır. Ayrıca MSC çağrı kuruluşu için omurga ağ işletme arabirimini de sağlar.

MSC'deki HLR (Home Location Register) bölümünde belirli ağ sistemindeki kayıtlı kullanıcıların veritabanı bilgileri tutulur. HLR, belli ağ operatörlerinin her abonesi için SIM (Subscriber Identity Module) içerir. Kullanıcının bir sistemden başka bir sisteme geçmesi VLR'yi (Visitor Location Register) ilgilendirir. VLR'de bütün kullanıcılar için geçici bir veritabanı tutulur. VLR, HLR'ye ilgili kullanıcının geçici olarak bir bölgede bulunduğunu ve bu kullanıcıya ait tüm isteklerin VLR tarafından yönetileceğini belirtir. Mantıksal olarak her GSM ağında HLR'lar bulunur ve ancak çok fazla bölgeye dağıtılmış da olabilirler. Ayrıca VLR, HLR üzerinden kullanıcı için seçilmiş bilgilerini ve tüm çağrılarının kontrollerini de üzerine alabilir.

MSC belli aygıt bilgileri üzerinde oynama yapamaz. Çünkü bilgi, anahtarlama sistemine ait değil veritabanına aittir. Kullanılan bir diğer donanım kontrolün artırılmasına yönelik olan AuC (Authentication Center) ve EIR'dır (Equipment Inventory Register). AuC'nin görevi kullanıcılara ait belirli bilgilerin doğrulanmasını kontrol eder ve bu bilgilerin içinde şifre de olabilir.

Bu özel bilgiler hava arabirimi üzerinden kullanıldığı için AuC sonucu, bu bilgilerin onaylanmasını ve korunmasını gerçekleştirir. Bir diğer veritabanı olan EIR ise tüm radyo setleri için satıcı bilgilerini tutar. Bu bilgiler üretici ve IMEI olarak tutulur. Eğer kullanılmakta olan herhangi bir set çalınmış ise veya hile şüphesi taşıyorsa veritabanındaki IMEI bayrağı geçersiz yapılır. Böylece bu set üzerinden ağ kullanmak isteyen kullanıcı reddedilmiş olur. [17]

## **Hava ve bağlantı arabirimi**

Uplink (yukarı bağlantı) olarak tanımlanan mobil birimden ana istasyona 890-915 MHz arasındaki radyo frekanslarında, downlink (aşağı bağlantı) olarak tanımlanan ana istasyon ile mobil cihaz ise 935-960 MHz bandında çalışacak şekilde tanımlanmıştır.

### **Eriřim teknikleri**

GSM'ler havadaki arabirim (air interface) olarak TDMA'yı kullanırlar. Gerçekte ise GSM, TDMA ve FDMA'nın kombinasyonunu kullanmaktadır. GSM için ayrılan 25 MHz'lik alanı FDMA 200kHz'lik alanlarda toplam 124 taşıyıcı frekansı ile kullanır. Bu taşıyıcı frekanslardan bir veya daha fazlası ana istasyon için atanmıştır. Daha sonra her bir taşıyıcı frekans TDMA kullanılarak alt bölümlere ayrılır. TDMA'nın en temel bileşeni veri 0,577 ms'lik veri yayılım periyodu (data burst period) boyunca yayılımıdır. Bundan sonra 7 yayılım periyodu bir çerçeve içinde gruplanır ve yaklaşık 4,615 ms'lik bir TDMA çerçevesi elde edilir. Bu çerçeve formatı mantıksal kanalların temelini oluşturur. Bir fiziksel kanal, TDMA çerçevesindeki bir yayılım periyodudur. Yayılım periyoduna karşılık gelen pozisyon ve sayı kanalı tanımlamakta kullanılır. Bu tanımlama işlemleri her 3 saatte bir yapılır.[17]

### **Kodlama/Modülasyon teknikleri**

Radyo haberleşmesinde mesajlar belli biçimlerde gönderilmelidir. Bu işlem için de modülasyon teknikleri kullanılarak gidecek mesajlarda gerekli değişiklikler yapılmaktadır. Alıcı tarafta gelen mesajlar demodüle edilerek bilgilere ulaşılır. Bu teknikte 4 temel form kullanılır:

Amplitude Modulation: AM ve ya ASK

Frequency Modulation: FM veya FSK

Phase Modulation: PM veya PSK

Quadrature and Amplitude Modulation: QAM veya QPSK

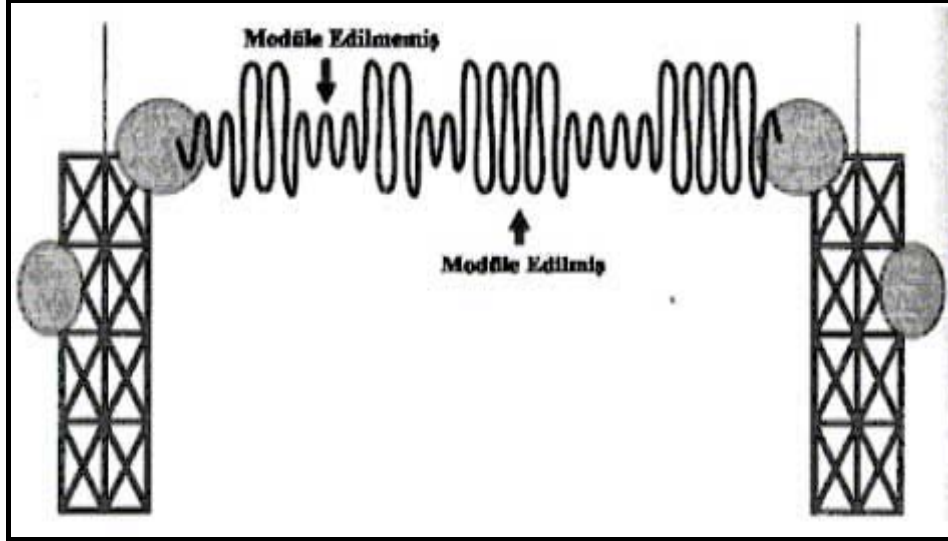
### **Modüle edilmiş sinyal zarfları**

Ses, veri ve video aktarımlarında taşıyıcı olarak, modüle edilmemiş bir taşıyıcı olan radyo sistemleri kullanılır. Bu iletim noktadan noktaya sabit taşıyıcı tonları ile gerçekleştirilir. Modülatör aracılığı ile, eklemek istediğimiz bilgiler sinyallere modüle edilir ve modüle edilmiş zarflar alıcı istasyona gönderilerek iletilmek istenen veri aktarımını sağlamış olur.



## Dalga boyu (Applitude) modülasyonu

Radyo frekans aktarımında ilk kullanılan analog sistemler Dalga Boyu Modülasyon (DBM) sistemleri idi. DBM’de, bilgiler sinyalin dalga boyunu değiştirerek sinyaller üzerine modüle edilirdi. Bu uygulamada frekans sabit tutulurdu. Şekil 2.4’de bu modülasyon tekniği görünmektedir.



Şekil 2.4: Dalga Boyu Modülasyonu Uygulanışı[22]

## Çoklu erişim ve çoğullama yöntemleri

Radyo frekansı spektrumu sonlu bir kaynaktır; bu nedenle aynı anda aktarım yapmak isteyen farklı uç sistemlerin kaçınılmaz olarak belirli frekans aralıklarını paylaşmaları gerekir. Frekans spektrumun bölünmesi ve birçok kullanıcı arasında paylaşımın birçok yolu/yöntemi vardır.[18]

Bir elektriksel iletim hattı sadece iki kişi arasındaki konuşmayı iletmek üzere ayrıldığı zaman önemli bir israf yapılmış olur. Çoğullama teknikleri kullanılarak bir hat üzerinden birden fazla bilgi simultane ya da sırayla iletilebilir. Analog işaretle çoğullama yöntemi frekans bölmeli çoğullama - Frequency Division Multiplexing - FDM; dijital işaretle ise zaman bölmeli çoğullama - Time Division Multiplexing - TDM' dir.[19]

### a) TDMA (Time division multiple access)

Spektrumun zaman aralıklarına bölünmesi prensibi olarak tanımlanan TDMA zaman bölümlenmeli çoklu erişim yöntemidir.

Bu uygulamada iletişim çerçeveler içine biçimlenir ve çerçeveler özel zaman dilimlerine bölünürler. Her çağırma/mesaj özel zaman dilimine atanır ve sadece bu zaman dilimini kullanmasına izin verilir. Her frekans pek çok zaman dilimine ayrıldığından (Şekil 2.5'de görüldüğü gibi) pek çok zaman dilimi yaratılabilir ve pek çok kullanıcı aynı frekansı ve fakat farklı zamanı aynı anda kullanırlar.

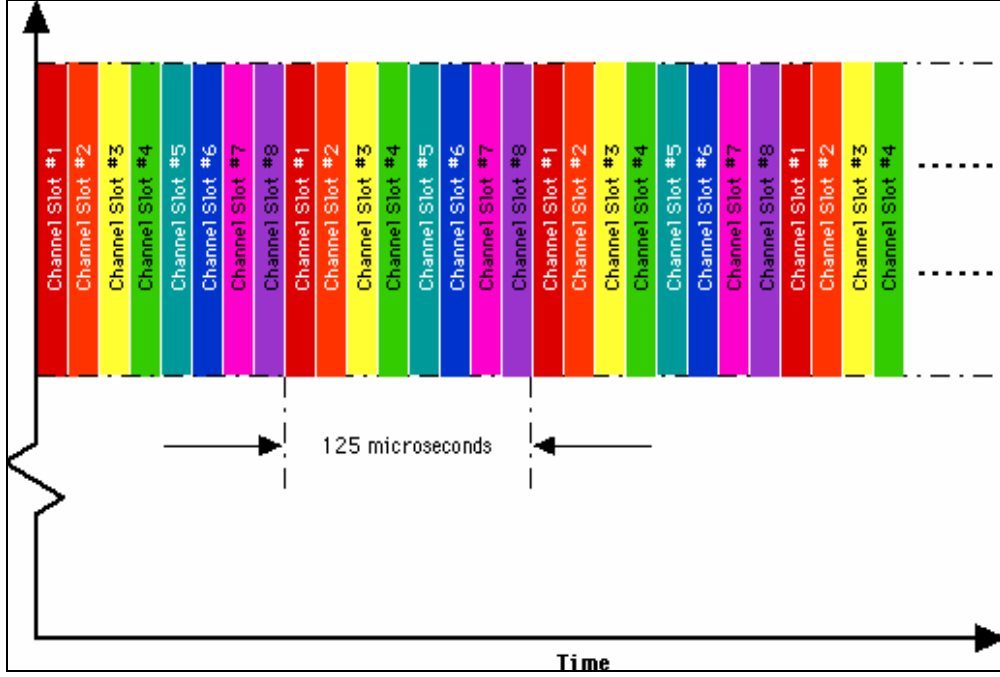
Zaman Dilimleri (Time Slots)					
1	1	1	1	1	1
2	2	2	2	2	2
3	3	3	3	3	3
4	4	4	4	4	4
5	5	5	5	5	5
6	6	6	6	6	6
7	7	7	7	7	7
8	8	8	8	8	8
F1	F2	F3	F1'	F2'	F3'

Şekil 2.5: TDMA'nın Kullandığı Farklı Modeller

DAMPS ve GSM, TDMA'yı kullanırlar, ancak zaman dilimleri farklı biçimlenmiştir. IS-54 DAMPS standartları, sadece konuşma kanalları üzerinde çoklu konuşmayı gerçekleştirmek üzere TDMA'yı kullanır. Buna karşılık DAMPS en son gelişmiş şekli ise IS-136'dır. Bu standart zaman dilimlerinin çoklanmasının hem ses hem de diğer kanallar üzerine uygulanmasını olanaklı kılmaktadır.

GSM ise aynı kanallar üzerinde hem ses hem de setup dilimlerini çoklamak üzere TDMA'yı kullanır. Küçük sistemlerde çağırma setup'ı için tek bir zaman dilimi kullanılır. Büyük sistemlerde bu amaçla 8 zaman dilimi kullanılmaktadır.[20]

Şekil 2.6'da TDMA görülmektedir.

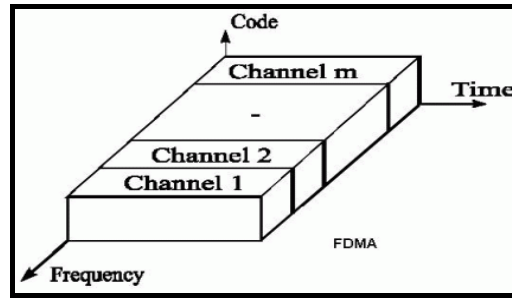


Şekil 2.6: TDMA [21]

#### b) FDMA (Frequency division multiple access)

FDMA olarak adlandırılan frekans bölümlenmeli çoklu erişim yöntemidir. FDMA’da frekans spektrumu, frekans alanında birbiri üzerine taşmayan bölmelere ayrılır; bu bölmeler uç sistemlerin belirli bir çağrısı için sistemlere atanırlar.[18]

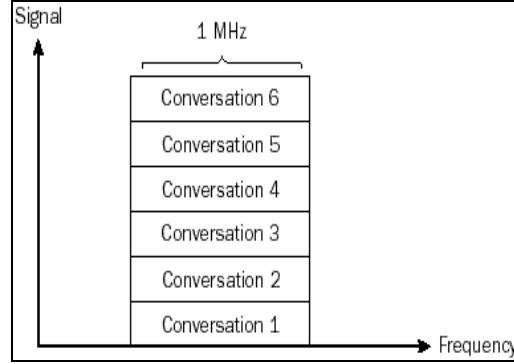
FDMA, analog sistemlerde kullanılan, radyo vericilerinin aynı spektrumda veri göndermesini sağlayan bir sistemdir. Frekans 30 KHz bölünür ve her bir bölüm ses ve veri gönderme özelliğine sahiptir. Kanal bir kullanıcıya tahsis edilir ve konuşma tamamlandıktan sonra diğer kullanıcıya atanır. Şekil 2.7’de FDMA görülmektedir.



Şekil 2.7: FDMA[21]

### c) CDMA (Code division multiple access)

Dağılmış spektrum kavramına dayanan, hem bir modülasyon ve hem de bir erişim yöntemi olan CDMA kod bölümlenmeli çoklu erişim'dir. Bu sistemde çağrılar kodlamayla birbirinden ayrılır.Şekil 2.8'de CDMA görülmektedir.



Şekil 2.8: CDMA[20]

### 2.2.3. 2.5 Nesil (2.5G) sayısal ses ve veri sistemleri

3G'ye giden yolda 2.5G teknolojisi olarak adlandırılan bu sistemde amaç sesin yanında düşük aktarım hızıyla veri trafiğini sağlamaktır. Bu teknolojinin en çok kullanılanları 171.2 kbps'a kadar iletim hızı sağlayan GSM üzerine geliştirilen GPRS, 384 Kbps'a kadar olan EDGE ve 64 Kbps'a kadar destek veren HSCSD'dir.

GSM standartlarının belirlenme çalışmasının 1.safhası, Avrupa Telekomünikasyon Standartları Enstitüsü (ETSI) tarafından 1990 yılında tamamlanmıştır. GSM standartları bundan sonra da, değişen teknolojik ilerleme ve ihtiyaçlar doğrultusunda 2. safha ve 2. safha + olarak adlandırılan evrelerle, 1990'lı yılların ortalarında daha da geliştirilmiştir. Söz konusu geliştirilmiş 2G teknolojileri, "High Speed Circuit Switched Data" (HSCSD), "General Packet Radio Service (GPRS) ve "Enhanced Data Rates in a GSM Environment" (EDGE) olarak adlandırılmaktadır. Anılan teknolojiler, 3G'ye doğru giden yolda son basamaklar olarak görülmekte ve topluca 2,5G teknolojisi olarak kabul edilmektedir.

### **HSCSD (High speed circuit switched data)**

HSCSD kablosuz verilerin 38.4 kbps ve daha yüksek hızlarda GSM sistemine transferini sağlayan bir teknolojidir.

HSCSD, 64Kbps'e kadar iletişim hızı sunmaktadır. HSCSD bugüne dek yalnızca ses kanalları için kullanılan radyolink protokolünde modifikasyon yapılmasına ihtiyaç duymaktadır. HSCSD iki büyük yenilik getirmektedir:

Hızlı anahtarlama sayesinde bir zaman bölmesinin iletişim hızı 9.6 Kbps'dan 14.4 Kbps'a çıkmaktadır. Ancak operatörlerin verdiği servise göre 9.6 Kbps hızının kullanılmaya devam edilmesi de mümkündür.

HSCSD sekiz bölmenin bir kaçını kullanarak 64 Kbps hıza ulaşacaktır.

Bir başka ifade ile, hızlı data iletişimi yapmak isteyen aboneler bir kaç abonenin kullandığı kanal genişliğini tek başlarına kullanacaklardır. Başta internet olmak üzere sabit telefon hatlarıyla yapılabilen her türlü data iletişimi HSCSD sayesinde GSM üzerinden yapılabilmektedir.

Bu yaklaşımın temel kazancı mevcut GSM şebekesi üzerinde tesis edilebilmesidir. GSM servis sağlayıcıları, HSCSD'yi kolayca ve etkili bir biçimde şebekelerine ekleyebilirler. [8]

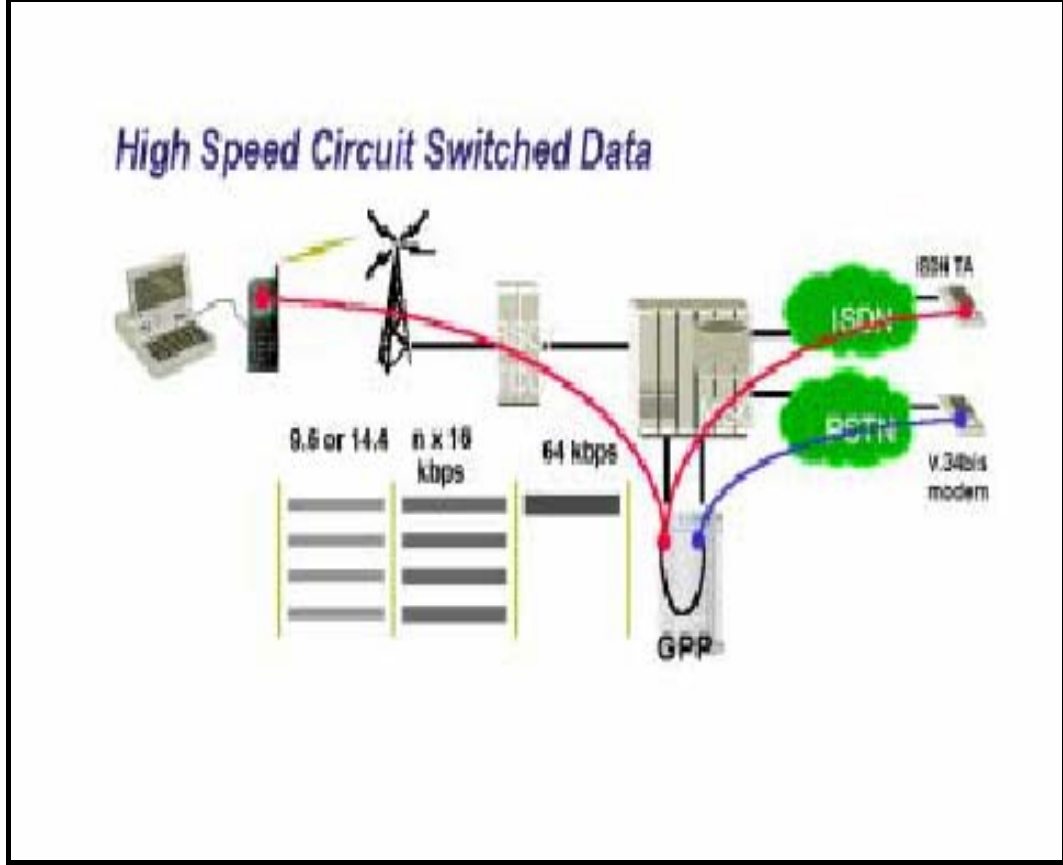
HSCSD birden fazla konuşma kanalını sürekli tuttuğu için operatör içinde, kullanıcı açısından da pahalı bir iletişim yoludur. Bu sebeple daha etkili ve verimli çalışan GPRS paket anahtarlama altyapılara geçilmiştir.

### **GPRS (General packet radio service)**

Mobil telefon teknolojisindeki en büyük adımlardan birisi GPRS'tir. GPRS devre anahtarlama (circuit-switched) yerine paket anahtarlama (packet-switched) bir yapıyı kullanan ve iletişime yüksek hızlar getiren bir teknolojidir. Paket anahtarlama bir yapıya geçişle iletişim hızı 115 Kbps'e çıkmış, GPRS gelecek nesil mobil iletişimde kilit noktasını oluşturmuştur.

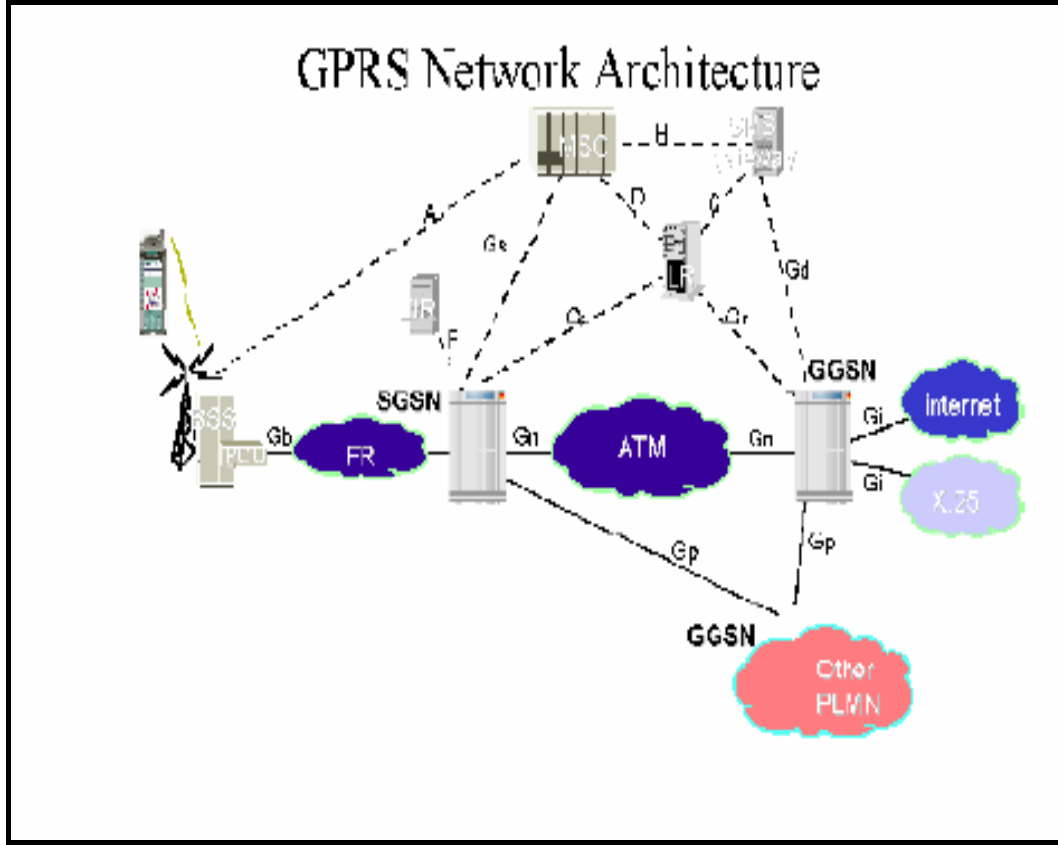
Devre anahtarlama sistemde ses ve ya veri iletişimi yapılırken kullanıcılara bir hat ayrılmaktadır. Bu hat kullanıcılara tahsis edildiğinden hattın yoğun olmadığı, hatta boş olduğu zamanlarda bu hat meşgul durumda bulunmakta hat kapasitesi boşa harcanmaktadır. Bu yüzden bu sistemin adı hat tahsisli sistem (Şekil 2.9) olarak ta kullanılmaktadır. GPRS sisteminde ise kullanıcılara hat

tahsisi yapılmamakta, iletişimde veri ilgili adreslere paketler halinde gönderilmektedir.



Şekil 2.9: Hat Tahsisli Sistem

Aşağıdaki Şekil 2.10'da ise paket anahtarlama sistemine dayanan hattın meşgul edilmediği GPRS ağ mimari yapısı gösterilmektedir.



Şekil 2.10: Paket Anahtarlama Sistemi [12]

GPRS, bağlantı süresine göre değil gerçekleştirilen veri alış-veriş miktarına göre ücretlendirmenin yapıldığı yüksek hızlı erişimi sağlar. Bu teknoloji; patlamalı tür, küçük veri miktarlarının sık, büyük veri miktarlarının sık olmayan iletimlerinde ise şebeke kaynaklarının verimli ve ekonomik kullanımını sağlar.

GPRS'te iki kullanıcı arasında bir veya daha fazla paketin iletimini sağlayan PTP (point to point-noktadan noktaya) servis tipi tanımlıdır. PTP bağlantısız şebeke servisi ve PTP bağlantı yönelimli şebeke servisi olmak üzere 2 çeşit PTP servisten oluşur.

### **Paket anahtarlama sisteminin sağladığı özellikler**

Paket anahtarlama sistemine geçilmesiyle birlikte sadece veri gönderileceği zamanların ücretlendirilmenin yapılması, boş geçen sürelerin ücretlendirmeye katılmaması, hattın sadece tek bir kullanıcı tarafından işgal

edilmeyip aynı anda birçok kullanıcıya da hizmet vermesi ve iletişim yapmak isteyenlerin hat alma zorunluluğunun ortadan kalkıp sürekli sisteme bağlı duruma gelmesi gibi kolaylıklar sağlanmıştır.

### **GPRS sistem özellikleri**

Mobil istasyonu bağlantı cihazı olarak sunan GPRS, GSM sistemine ek bir servistir. GSM telefonu olan bir mobil terminali (MT-Mobile Terminal) ve buna bağlı bir bilgisayar olan terminal cihazını (TE-Terminal Equipment)'nın birleşimi bir MS'i oluşturur.

GPRS veri transferi internet protokolüne dayanır. Bu yüzden paket veri iletimi hava ara yüzünü içeren uçtan uca kaidesine dayanarak yürütülür.

GSM sistemi telefon için devre bağlaşmalı hava ara yüzünü kullanırken GPRS sistemi paket bağlaşmalı hava ara yüzünü kullanır ve her ikisi de GSM standartlarına göredir. Bir GPRS şebekesi, bir GSM şebekesinin bir uzantısı olarak görülebilir ve kendine özgü bazı eklemelere ihtiyaç duyar. GPRS sistemini GSM sistemine tanıtmakla, abone ve terminal verisini koordine etmek, bağlamak, idare etmek ve güvenliklerini sağlamak hem devre bağlaşmalı hem de paket bağlaşmalı haberleşme için mümkün olacaktır.[12]

### **Kullanıcı özellikleri**

Devre anahtarlama sistemini eksiklerini kapatan ses ve veri transferini sağlayan GPRS aynı anda 8 zaman dilimini kullanarak GSM veri servislerinden 10 kat daha fazla hızda çalışmasını sağlayarak maksimum 171.2 kbit/s hızına ulaşabilir.

### **GPRS sınırlamaları**

GPRS 171.2 kbit/sn'lik hızı destekler ancak bu 8 tane zaman diliminin kullanıldığı zaman gerçekleşmektedir. Bir operatör bütün zaman dilimlerini sadece GPRS kullanıcılarına ayırmayacaktır. Bu da yüksek hızda işlem yapılamayacağını göstermektedir. İletim esnasında ise paketler aynı noktaya farklı yolları kullanarak gidebilirler ve iletim sırasında da bazı verilerin kaybolmasına sebep olabilir. Bu da iletimde aksaklıklara yol açar. GPRS'teki bir başka sınırlama ise depolama mekanizmasının olmamasıdır.



### **EDGE (Enhanced data rates for global evolution)**

2002 yılından sonra kullanıma hazır olan EDGE, operatörlere mevcut olan 800, 900, 1800 ve 1900 frekans spektrumlarını daha etkili kullanmalarını sağlayan, radyo protokollerinde de yenilikler içeren, yeni bir modülasyon tekniği sunmaktadır. EDGE, 384 Kbit/s'e ve daha fazlasına kadar veri, multimedya servisleri ve uygulamalarını desteklemektedir.[22]

EDGE için gerçek veri hızları abone başına 4 time-slot ile max 230kbit/s dur. Şebekelerdeki ortalama abone hızı (4 ts ile) yaklaşık 130-170kbit/s gerçekleşmektedir. EDGE teknolojisi;

1. Farklı bir modülasyon yöntemi kullanarak data iletişim hızınının 384 Kbps'e çıkarılması,
2. Hem paket anahtarlama hem de tahsisli hat sistemine uygun olarak kurulabilmesi
3. Yüksek data iletişim hızı sayesinde, pek çok multimedia uygulamasının GSM üzerinden gerçekleştirilebilecek olması,
4. Yüksek data iletişim hızı sayesinde, hizmet verilebilecek kullanıcı sayısının artması gibi yenilikleri beraberinde getirmiştir.[8]

Bu önemli yeniliklerle beraber EDGE teknolojisinin bir takım dezavantajları da bulunmaktadır.

Bunlar; Baz istasyonlarındaki GSM alıcı verici ünitelerinin EDGE desteğinin olmamasından dolayı sadece yoğun abone sayılarının olduğu alanlar EDGE'nin kapsama alanına gireceklerdir. 200KHz bant genişliği yine korunmakta ve iletişim hızı 384Kbps'a çıkarılmaktadır. EDGE teknolojisinden yararlanabilmek için yüksek hızlarda data iletişimi yapmak isteyenler dual-mode cep telefonları kullanmak zorunda kalacaklardır.

### **Kablosuz uygulama protokolu (wireless application protocol-wap)**

Cep telefonu ile sağlanan veri akış sistemine WAP (kablosuz uygulama protokolü) ismi verilmiştir. WAP (Wireless Application Protocol - Kablosuz Uygulama Protokolü), sunucu uygulamaları ve istemciler arasındaki iletişim, mobil cihazdaki özel fonksiyonların istemcinin mevcut konumunda kullanılması yöntemi ve kablosuz uygulamalar için genişletilmiş bir işaretleme dili (Wireless

Markup Language - WML) için bir protokol tanımlayan spesifikasyonlar paketidir. Bu spesifikasyonlar şu amaçlar doğrultusunda belirlenmiştir:

Özel standartlar gibi sınırlı olmayan global bir standart tanımlama

Dar bantlı iletişim için optimizasyon

Özellikle kullanıcı arabirimi yönünden sınırlı özelliklere sahip olan cep telefonları ve benzer kablosuz mobil cihazlar için optimizasyon

Microbrowsing ile entegre telefon servisleri

Kablosuz uygulama protokolü, 1997 yılı Aralık ayında Ericsson, Motorola, Nokia ve Unwired tarafından kurulan bir endüstri konsorsiyumu (WAP Forum) tarafından belirlenmiştir. Forum, çalışma grupları ve komiteler olmak üzere iki ana faaliyet grubuna ayrılmıştır. Komiteler yürütülen çalışmaları izlerler ve spesifikasyonların birbirlerinden ayrı olmasını önlemeye çalışırlar. WAP spesifikasyonları kısaca şunları kapsar:

Microbrowser

Scripting

Kablosuz telefon uygulamaları (wireless telephony applications - WTA) ve mobil cihazlar için bir WTA arabirimi

Uygulamalar için içerik formatı

Uygulama, oturum, geçiş ve güvenlik katmanlarını içeren bir katman yığını.[1]

WAP; CDPD, PDC, TDMA, CDMA, TETRA, GSM, PHS, FLEX, ReFLEX, iDEN, DECT, Mobitex ve DataTAC gibi kablosuz iletişim teknolojileri ile çalışarak cep telefonları, avuç içi PC' ler, çift yönlü telsiz cihazları ve akıllı telefonlar gibi elde taşınabilen, dijital, kablosuz cihazlarda kullanılarak mobil kullanıcılara bilginin ve servislerin hızlı, kolay ve güvenilir bir şekilde sunulmasını sağlar.

### **WAP'ın genel yapısı**

WAP; kablosuz uygulamaları geliştirme teknolojisidir. WAP uyumlu telefonlar ile internet'e cep telefonundan ulaşmayı sağlar.

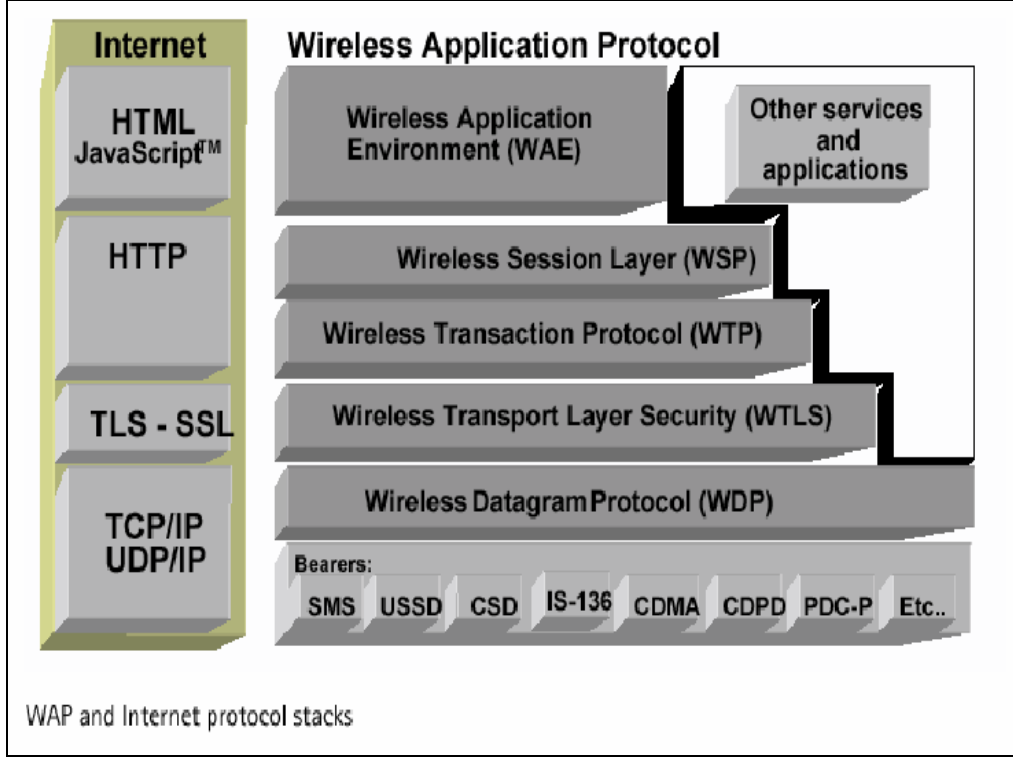
WML dili internet sayfalarının cep telefonundan görüntülenebilmesini sağlar. WML HTML diline çok benzer fakat http'deki gibi çok fazla script'in engellenmesini sağlar.

Mobil telefonlar gibi kablosuz cihazların sınırlı işlem kapasitesi, HTML sayfaları gibi yüklü bilgilerin ekranlarına taşınmaması, kablosuz iletişim servis ağlarının kapsama alanı problemleri ve sürekli bağlantı sağlanamaması, sınırlı dalga boyunun getirdiği kısıtlamalar gibi problemler bugüne kadar kablosuz iletişimde internet servislerini mümkün kılmamıştır. Bütün bu problemlerin çözümü olarak geliştirilen WAP, kesintili kapsama alanlarında, çeşitli kablosuz iletişim standartları ile mümkün olduğu yerde IP, yoksa diğer protokolleri kullanarak, kullanıcılara sanal kesintisiz bağlantı sağlar. WAP içerikleri için geliştirilen WML programlama dili, kablosuz iletişim terminallerinin ufak ekranlarına gerekli bilgiyi en verimli şekilde sığdırabilmekte ve klavye olmadan tek el ile bilgi taramasını mümkün kılmaktadır.[22]

WAP için; kablosuz uygulama ortamı (WAE) tanımlıdır. Cep telefonlarının internette browser görevi yapabilmesi için kullanıcıların uygulama geliştirebilmesini sağlayan programlama dili WML ve WMLscript ile bu uygulamaların daha gelişmişinin hazırlanmasına imkan tanır.

Kablosuz uygulama protokolü farklı işlerin farklı yerlerde yapılması için düşünülen katmanlı bir yapıya sahiptir.

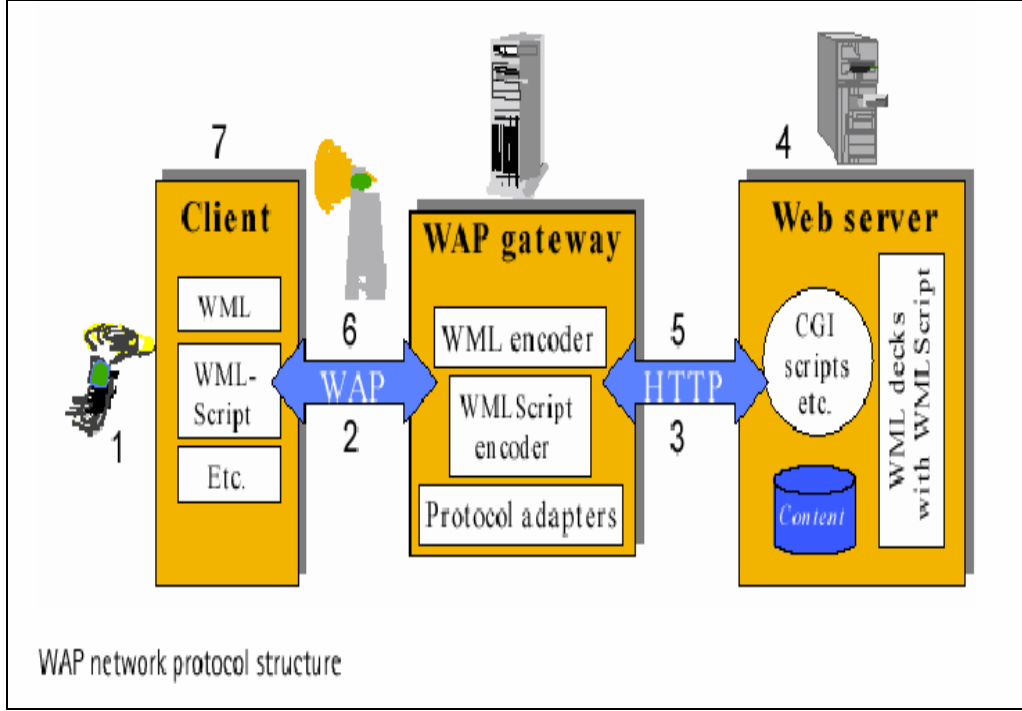
Protokolün mimarisi aşağıdaki Şekil 2.11'de daha net görülebilir.[23]



Şekil 2.11: Wap ve İnternet Protokol Yapısı

### WAP işleyiş yapısı

WAP'ın genel işleyiş yapısı ise Şekil 2.12'deki gibidir. Cep telefonundan görünen wap uyumlu internet sayfaları web sunucusu üzerinde durmaktadır. Cep telefonundan bir WAP sayfası çağrıldığında, bu istek öncelikle kablosuz GSM ağlarından geçip WAP gateway' e gelir. WAP gateway' in görevi GSM ağları ile yerel bilgisayar ağları arasında kapı görevi görmektir. Cep telefonundan gelen isteği WAP protokolünden TCP/IP protokolüne çevirerek istenilen internet adresini web sunucusundan alır ve sonra aynı yoldan cep telefonuna cevap gönderir. WAP uyumlu uygulamalar web sunucusu üzerinde durur ve WAP gateway sayesinde istek yapan cep telefonuna gönderilir.



Şekil 2.12: Wap Network Protokol Yapısı[23]

### WAP güvenliği

WAP'ın geçerlilik kazanabilmesi ve uzun ömürlü olabilmesi için güvenli aktarımlara ortam sağlaması gerekir. Bunun için kullanıcı tanımı ve şifresi gibi önlemler sunan WTLS tanımlaması yapılmıştır. İnternet aktarımlarını sayısal imza ve şifreleme algoritması ile güvenli hale getiren WIM (Wireless Identity Modules) ve JAVA, WAP'da kullanılarak daha güvenli hale getirilmesi düşünülmüştür.

### 2.2.4. Üçüncü nesil (genişbant) sistemler

İkinci nesil haberleşme sistemleri, ses servisinin yanında faks, veri haberleşmesi, mesaj işlemleri vs. gibi hizmetleri de vermesine rağmen, gelişen teknolojiye bağlı olarak artan daha hızlı veri haberleşme talebi ve çoklu ortam uygulamalarına olan eğilim ile birlikte, kullanıcı isteklerini yerine getirmede yetersiz kalmaya başlamanın yanı sıra hiçbirinin haberleşmede bütünüyle küreselleşmeyi sağlayamamış olması üçüncü nesil haberleşme sistemini ortaya çıkarmıştır. [24]

“3G” diye tabir edilen Üçüncü Nesil Haberleşme standartlarını geliştirmek için 1985 yılında Uluslararası Telekomünikasyon Birliği (ITU) çalışmalarına başlamıştır. 3G sistemleri, aşağıdaki özellikleri desteklemektedir:

Her türlü telsiz ortamında 144 kbit/sn ve düşük kapsama alanı veya dahili ortamlarda 2 Mbit/sn iletim hızları

Simetrik ve asimetrik veri iletimi,

Şebeke bağlantılı veya paket bağlantılı internet protokolü (IP) trafiği gibi paket bağlantılı hizmetler ve gerçek zamanlı görüntü,

Daha iyi ses kalitesi,

Daha büyük kapasiteli ve geliştirilmiş spektrum verimliliği,

Çoklu ortam hizmetleri için son kullanıcı veya terminallere aynı anda birden fazla hizmet,

2G mobil sistemi ile kesintisiz bağlanabilme ve farklı 3G işletim ortamlarında küresel dolaşım, ölçek ekonomisi ve kitle pazarı gereksinimlerini karşılaması beklenen açık uluslararası standart. [25]

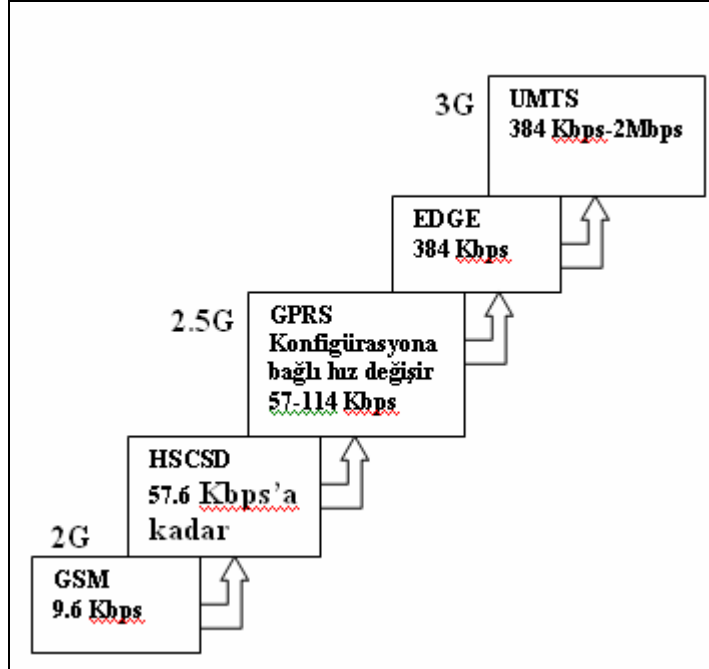
## **2G-3G İlişkisi**

2G, 2,5G ve 3G, analog 1G teknolojisinden tamamen farklı teknolojileri içermektedir. Ancak, 3G altyapılarının 2G altyapıları ile benzer yanları bulunacağı ve belirli bir geçiş dönemi boyunca 2G ve 3G ile benzer hizmetlerin sunulabileceği öngörülmektedir. Esasen, 3G şebekelerinin geliştirilmesinde 2,5G GSM GPRS standartları dayanak teşkil etmekte olup, bu da GSM işletmecilerinin GPRS teknolojisinin veya 3G ara yüzlerinin kullanılması yoluyla 3G hizmetlerini verebilmelerini temin edecektir.[26]

İlginç olan 2G'den 2,5G'ye geçişin, 2,5G'den 3G'ye geçişe göre daha zor olacaktır. Operatörlere GPRS'den UMTS'e geçiş esnasında değil GSM'den GPRS'e geçiş esnasında büyük görevler düşecektir. Çünkü zaman temelli ücretlendirmeden, hacim tabanlı ücretlendirmeye ve ses tabanlı servisler yerine internet temelli geçişte mevcut iş modellerini tamamen değiştirmeleri gerekecektir. 2,5G' den 3G' ye geçişin yumuşak olacağı umulmakla birlikte, 2G' den 2,5G' ye geçiş esnasında yaşanan WAP deneyiminin kullanıcıları tam olarak

memnun etmediği görülmüştür. Bu durum pazardaki aşırı beklentilerin sonucundan kaynaklanmaktadır.[27]

Şekil 2.13 IMT-2000 (UMTS) geçiş adımları gösterilmektedir.



Şekil 2.13: IMT-2000 (UMTS) Geçiş Adımları

### 3G Spektrumu

1992 yılında Dünya Yönetimsel Telsiz Konferansı ("World Administrative Radio Conference", WARC) 3G mobil sistemler için ilk spektrumu belirlemiştir. Bu spektruma göre 1885-2025 MHz ile 2110-2200 MHz frekans bantları 3G mobil sistemler için tahsis edilmiştir.

Bu kararların ardından 3G sistemlerindeki yoğunluk yeni spektrum ihtiyacını doğurmuştur. Bu nedenle Dünya Telsiz Konferansı 2000'de (İstanbul'da yapıldı) ("World Radio Conference", WRC) 806-960 MHz, 1710-1885 MHz ve 2500-2690 MHz olan 3 frekans bandını 3G uygulamalarına ilave etmişlerdir.

UMTS Forum tarafından yapılan tahmin ve pazar araştırmalarına göre, 2010 yılı itibariyle 3G hizmetlerinin verilebilmesi için fazladan bir 185 MHz' lik frekans bandı daha gerekeceği tahmin edilmektedir. [28]

### **3GPP 3.Nesil ortaklık projesi**

Üçüncü nesil gezgin haberleşme sisteminin mevcut GSM alt yapısı ile uyumlu olmasını sağlayacak teknik özellik ve standartları belirlemek amacı ile Avrupa' dan ETSI, Japonya' dan ARIB ve TTC, Birleşik Devletlerden ANSI ve Kore' den TTA gibi dünyanın önde gelen standart enstitüleri 1998 yılı Aralık ayında bir araya gelerek; Üçüncü Nesil Ortaklık Projesi 3GPP (3G Partnership Project) oluşturmuşlardır. IMT-2000 gerçekleştirilen çalışmalarla bir dünya standardı haline gelmiştir.

### **IMT-2000**

IMT-2000 olarak bilinen 3G mobil telekomünikasyon teknolojisine ait standartlar, Uluslararası Telekomünikasyon Birliği (ITU) tarafından geliştirilmektedir. IMT-2000 tanımlamasında IMT uluslararası mobil telekomünikasyonu, 2000 ise sistemlerin çalışacağı 2000 MHz frekans bandını temsil etmektedir.

IMT-2000 hizmetlerinin, evrensel hizmetler olması beklenmektedir. Söz konusu evrensellik, karasal ve uydu telsiz ara yüzleri konusundaki spesifikasyonların ayrıntılı bir biçimde belirlenmesi ve benimsenmesi ile sağlanabilecektir. Bu maksatla, ITU'nun IMT-2000 konusunda faaliyet gösteren komitesine, çok fazla sayıda standart önerilmiş, söz konusu standartların benimsenmesi konusunda uyum sağlanabilmesi amacıyla uzun süren bir çalışma sürecinin neticesinde, 1999'un Kasım ayında, IMT-2000 telsiz ara yüz standartları belirlenmiştir.[29]

Fakat gene de başarılı olunamamış WCDMA/UMTS (Japonya'da da kullanılmakla beraber Avrupa'daki UMTS ile tam uyumlu değildir) , CDMA2000 ve TD-CDMA gibi farklı standartlar oluşmuştur.

IMT-2000, ITU tarafından; hem yer hem de uydu haberleşme sistemlerini kullanacak, sabit ve ikinci nesil mobil haberleşme sistemlerinin her ikisi ile uyumluluk sağlayarak bütün dünyada kesintisiz haberleşme sağlayacak ve dünyanın ilgisinin yöneldiği başta internet olmak üzere veri haberleşmesine ve çoklu ortam hizmetlerine kolay ve hızlı erişim sağlayacak bir sistem olarak tanımlanmıştır.



2 Mbps' a kadar yüksek hızı destekleyen, hem paket anahtarlama hem de devre anahtarlama veri iletişimi sağlayarak bütün dünyada tüm mobil uygulamalarda kullanım imkânı IMT-2000 tarafından sağlanmaktadır.

IMT-2000, mega/makro/mikro ve piko-hücreli karasal mobil sistemleri, kablosuz sistemleri, uydu sistemlerini bir araya getirerek küresel hizmet sağlayacaktır. [7]

### **Mega (uydu) hücreler**

Geniş alanları kaplamada ve özellikle trafik yoğunluğunun düşük olduğu uzak alanlarda kullanılır. Hücre boyutları itibariyle, mega hücreler uzak alanlardan şehir merkezlerine hatta karasal iletişim şebekelerine ve erişim yapılamayan alanlar gibi çok çeşitli ortamlar için kullanılabilir. Gelişmekte olan ülkelerde, tek uygulanabilen hücre tipi olarak, şehir merkezlerinde de kullanmak mümkündür.

Mega hücreler uydular tarafından sağlandığı için bazı zamanlarda mega hücreler yerine uydu hücreleri ifadesi de kullanılmaktadır. Bununla birlikte makro hücre kaplamasının gelecekte uydular ile de sağlanabilecek olması nedeniyle mega hücre kavramını kullanmak daha uygun olacaktır.[7]

Hücre boyutları uydu yüksekliğine, çıkış gücüne ve anten yüksekliğine bağlıdır ve çok büyüktür. Non-GSO yörüngelerinde hücreyi yaratan uydunun dünya etrafında dönüşünün dünyanın dönüşü ile ters olduğuna dikkat etmek gerekir. Bu durum bazı şartlarda istasyon durgun olsa dahi kesinti yaratabilmektedir.[25]

### **Makro hücreler**

Hücre yarıçapları 35 km'ye kadar ulaşabilen hücrelerdir. Yönlü anten kullanım gibi bazı yöntemlerle bu mesafeyi arttırmak mümkündür.

Makro hücreler, trafik yoğunluğunun fazla olmayıp orta düzeyde olduğu ve mobil istasyon hızlarının çok olmadığı dar band hizmetlerinin sunulduğu ortamlarda tercih edilir. Genelde kırsal ve banliyö bölgelerinde kullanılır. En büyük makro hücre şebekesi 900 MHz frekans bandında çalışan GSM' dir. GSM,

şehir merkezlerinde 1800 MHz frekansı çevresinde artan kullanımla birlikte mikro hücre ortamında da kullanılabilir.[30]

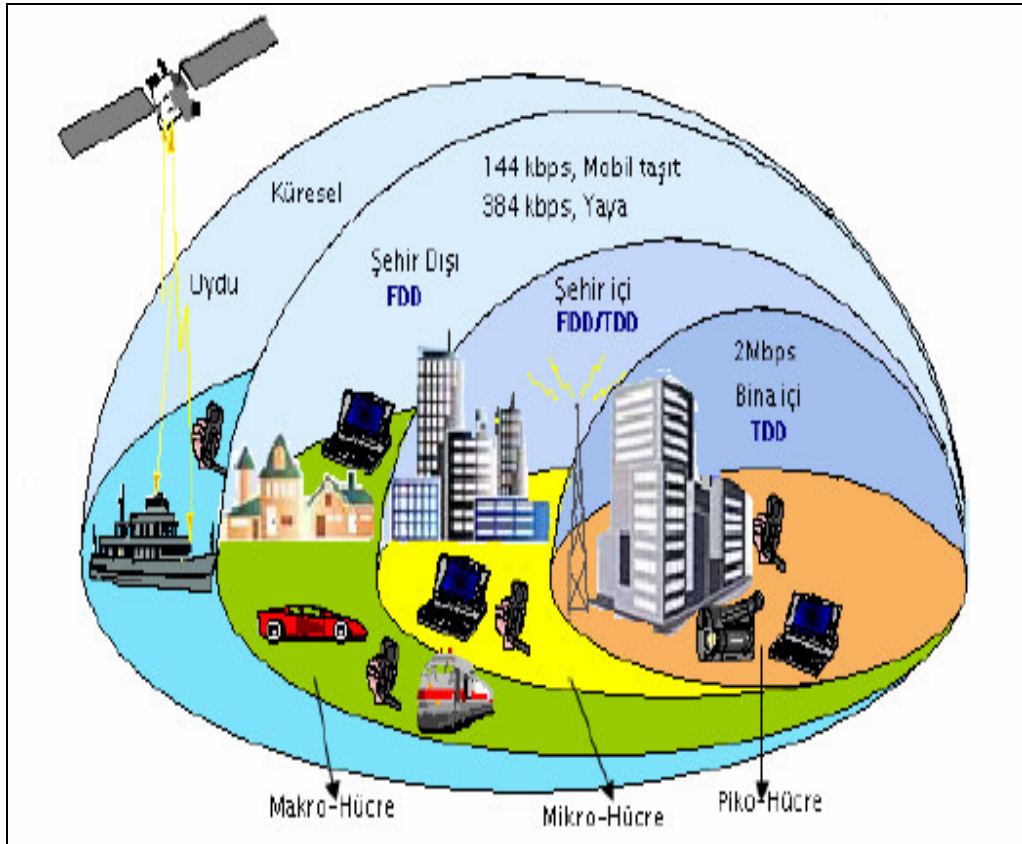
### Mikro hücreler

Çoğunlukla kentlerde hücre yarıçapının 1 km'ye kadar ulaştığı hücre tipidir. Trafik yoğunluğunun fazla ve mobil istasyon hızlarının az olduğu dar band uygulamalarında kullanılır. Mikro hücre ortamında insan yapımı yapılar önemli engel oluşturabilir.

### Piko hücreler

Hücre yarıçapları 50 m'den küçük çok yüksek trafik kapasitesi sunabilen, bina içi kullanımlara uygun hücrelerdir. Piko hücreler, trafik yoğunluğunun fazla, mobil istasyon hızlarının düşük olduğu geniş band uygulamalarında kullanılırlar.[7]

Şekil 2.14'de küresel mobil erişim görülmektedir.



Şekil 2.14: Küresel Mobil Erişim

Etkili bir kapsama alanı için piko, mikro ve makro hücre yapıları ile uydu gezgin şebekelerinin kullanılması düşünülmektedir. Bu hücre ve şebeke yapıları hiyerarşik hücre yapıları olarak adlandırılır ve şebeke işleticisinin ihtiyaçlarına göre düzenlenirler.

Örneğin devre anahtarlama gerektiren işlemler veya veri iletimi piko hücreler tarafından yürütülürken şehir içinde yaya hızındaki yüksek hızlı veri trafiği mikro hücre üzerinden yüksek hızla seyahat halindeki düşük veri trafiği de makro hücre tarafından yürütülür.[11]

### **IMT-2000 Telsiz arayüz standartları**

Tanımlanan IMT-2000 telsiz arayüz standartları aşağıdaki gibidir:

a) “UMTS Terrestrial Radio Access (UTRA)” [Evrensel Mobil Telekomünikasyon Sistemi (Universal Mobile Telecommunication System”, UMTS) Karasal Telsiz Erişimi (UTRA)]

b) “Code Division Multiple Access (CDMA) 2000” [Kod Bölmeli Çoklu Erişim 2000]

c) “Time Division-Synchronous Code Division Multiple Access (TD-SCDMA)” [Zaman Bölmeli-Senkron Kod Bölmeli Çoklu Erişim (TD-SCDMA)]

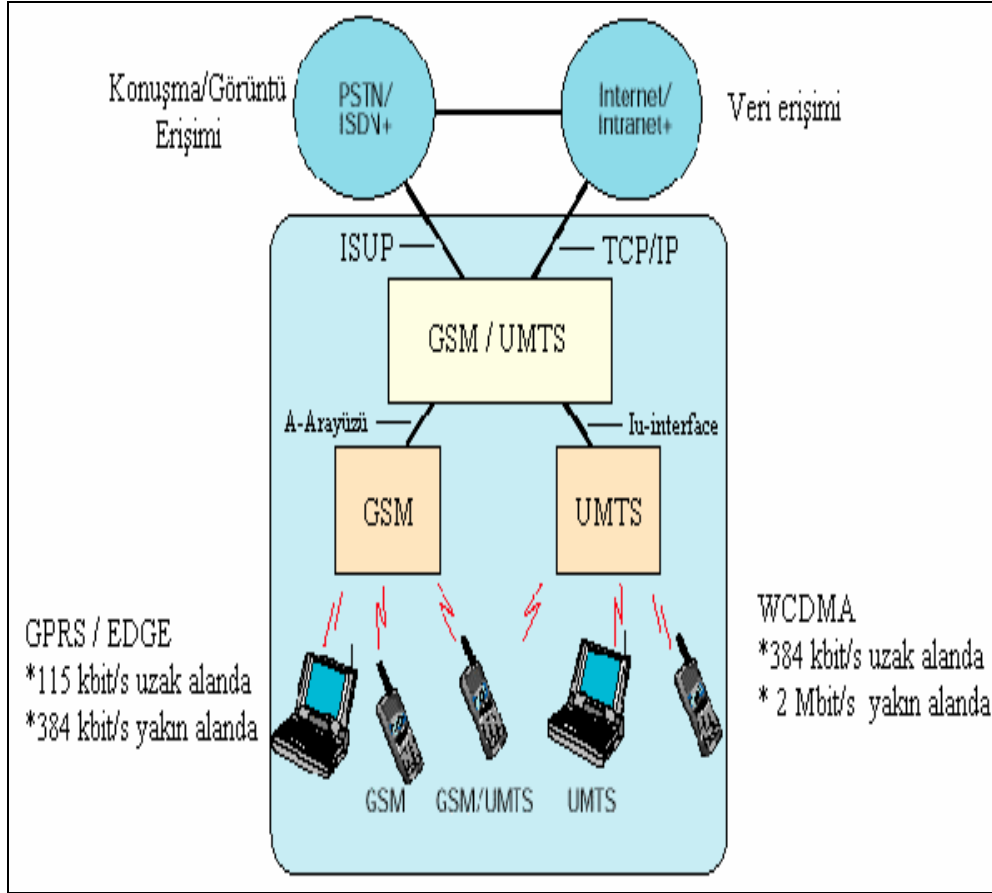
d) “Digital Enhanced Cordless Telecommunications (DECT)”[Sayısal Gelişmiş Kablosuz Telekomünikasyon (DECT)]

### **a) UMTS (Universal mobile telecommunication system)**

UMTS, IMT-2000 olarak tanımlanan, yüksek hızlı veri iletimini sağlayan üçüncü nesil haberleşme sistemidir.

UMTS, geniş band çoklu ortam servislerinin kullanımına olanak sağlayan yüksek veri hızlarını desteklemektedir. Hem simetrik hem de asimetrik veri transferine olanak tanıyarak kaynakların verimli kullanılmasını sağlayan, devre ve paket anahtarlama servislerin aynı anda kullanımını mümkün kılan ve IP (Internet Protokolü- Internet Protocol) protokolünü destekleyen bir hizmettir. Ses kalitesi telli şebekelerle karşılaştırılabilir derecede yüksek olacaktır. Ayrıca, ikinci nesil mobil haberleşme sistemleri ile kesintisiz geçiş yapabilecek, küresel olarak belirsiz geçişi ve kesintisiz dolaşım sağlayacaktır.[24]

Yeni UMTS şebekesinin mevcut GSM işleticilerinin kullandıkları şebeke alt yapısı üzerine kurulması işleticiler tarafından tercih edilmektedir. 2. nesil ve 3. nesil sistemlerinin bir arada çalışma düşüncesi Şekil 2.15’ de verilmiştir.



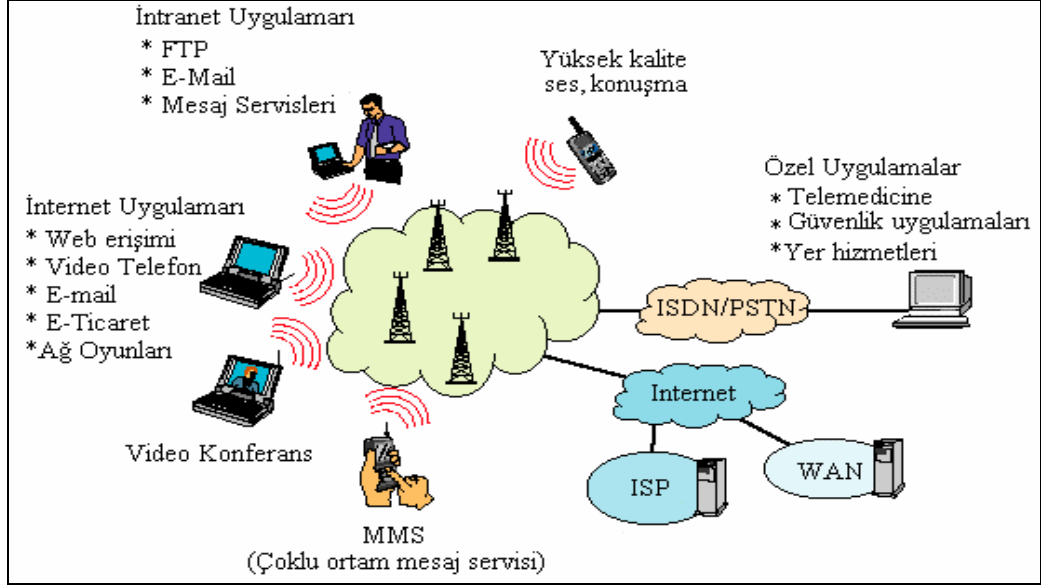
Şekil 2.15: 1, 2 ve 3. Nesil Sistemlerinin Bir Arada Çalışması[11]

### UMTS'nin sağladığı avantajlar

Ses kalitesinde artış, bireye özel uygulamalar ve güvenlik artışı, kaplama alanının genişlemesi ve hücre sayısında azalma, sistem planlama, kurulum ve işletme masraflarında azalma, diğer elektronik cihazlarla olan elektromanyetik etkide azalma, aktarmadan kaynaklanan çağrı kayıplarında azalma, kablosuz güvenli veri transferi, eski teknolojilerle uyumluluk olarak sayılabilir.[11]

## UMTS'nin sağladığı hizmetler ve uygulamalar

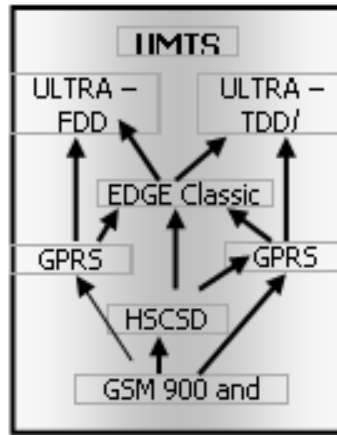
UMTS'in sağladığı hizmetler Şekil 2.16'da gösterilmiştir.



Şekil 2.16: UMTS 'in Sağladığı Hizmetler [11]

## UMTS'e geçiş aşamaları

UMTS' e geçişte izlenecek çeşitli yollar kullanılmaktadır. Temel olarak 3 aşamadan oluşan geçişte ilk safha GPRS, ikinci safha EDGE ve son safha ise UMTS olarak tanımlanmıştır. UMTS'e geçiş aşamaları Şekil 2.17'de gösterilmiştir.[11]



Şekil 2.17: UMTS Geçiş Aşamaları

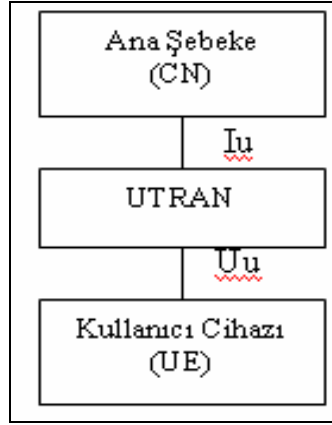
## UMTS şebeke mimarisi

UMTS şebeke yapısı üç bölümden oluşmaktadır: (Şekil 2.18)

Ana Şebeke (Core Network, CN)

UMTS Karasal Telsiz Erişim Şebekesi (UMTS Terrestrial Radio Access Network, UTRAN)

Kullanıcı Cihazı (User Equipment, UE) [7]



Şekil 2.18: UMTS Şebeke Blok Gösterimi

Şekil 2.18’de görülen blok şemada kullanıcı cihaz bölümü son kullanıcının yeni nesil şebekeden faydalanmasını sağlayan kısımdır. UTRAN bölümü telsiz haberleşmesi ile ilgili işlemlerin yürütülmesini sağlar Çekirdek şebeke ise çağrılarının yönlendirilmesi, IP şebekelerine veri bağlantılarının sağlanması gibi işlemleri yürütür (ETSI TS 125 4011). UMTS’de mevcut dört ara yüz vardır:

Uu: Kullanıcı cihazı ile Node B arasındaki ara yüzdür,

Iu: Ana şebeke ile Telsiz Şebeke Kontrol Edicisi (Radio Network Controller, RNC) arasındaki arayüzdür

Iub: RNC ile Node B arasındaki ara yüzdür,

Iur: RNC’leri birbirine bağlayan ara yüzdür. [6]

## Kullanıcı cihaz birimi

Kullanıcının UMTS hava arabağı (Uu) üzerinden ses ve veri iletişimi yapmasını sağlayan telsiz terminali ile abone bilgilerinin ve abone bilgilerini

doğrulama algoritmalarının, şifreleme anahtarlarının ve bazı abonelik bilgilerinin saklandığı Abone Tanıtım Modülü (USIM)' den oluşur.[31]

### **UTRAN şebeke yapısı**

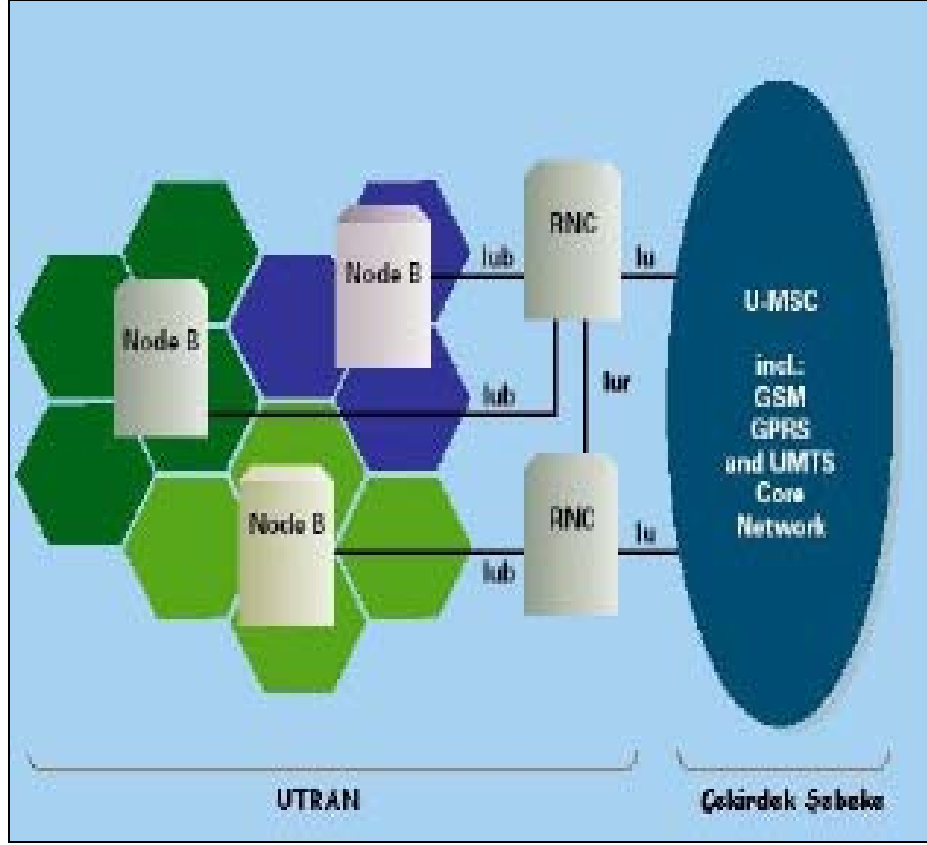
Kullanıcının tek bir hava ara yüzü üzerinden devre anahtarlama ya da paket anahtarlama harici şebekelere ulaşmasını sağlayan birimdir. Bu ara yüzü üzerinden ana şebekeye bağlanan Telsiz Şebeke Altbirimleri (Radio Network Subsystem, RNS'nden oluşur. Bir RNS ise Telsiz Şebeke Kontrol Edicisi (Radio Network Controller, RNC) ve bir ya da daha fazla baz istasyonundan (Node B) meydana gelir. Node B RNC'ye Iub arayüzü ile bağlıdır. Node B, FDD (Frequency Division Duplex- Frekans Bölmeli Eşleşim) modu, TDD (Time Division Duplex- Zaman Bölmeli Eşleşim) modu ya da ikili modu destekleyebilir. GSM sistemlerinde kullanılan baz istasyonları aynı işlevi yerine getirirler. UTRAN şebeke yapısı Şekil 2.19'de gösterilmektedir.[7]

### **UTRAN (UMTS karasal telsiz erişim şebekesi)**

Node-B ve RNC (Radio Network Controller)' den oluşmaktadır. Node-B GSM' deki baz istasyonlarına (Base Transceiver Station)'e eşdeğerdir. RNC ise GSM' deki BSC (Base Station Controller)' e eşdeğerdir.

Çekirdek Şebeke (Core Network); GSM'deki NSS (Network Switching Subsystem)' e eşdeğerdir.

UMTS Karasal Telsiz Erişim Şebekesi Şekil 2.19'da gösterilmektedir.[32]



Şekil 2.19: UTRAN Şebeke Yapısı

## UMTS erişim yöntemi

### Hava arayüzü ve özellikleri

1992 yılında Dünya İdari Radyo Konferansında (World Administrative Radio conference- WRAC) yeni radyo spektrumunun 230 MHz lik bölümü, karasal ve uydu servislerine ayrılmıştır. Bunun 2 GHz bandındaki 155 MHz kısmı ITU tarafından karasal 3G servislerine tahsis edilmiştir. Ancak UMTS' in kalıcı olmasını ve ileride doğabilecek yüksek bant genişliği taleplerini karşılayabilmek amacıyla Mayıs 2000 de İstanbul'da WRC 2000 görüşmeleri sonucunda 2500-2690 MHz frekans aralığı genişleme frekans aralığı olarak tahsis edilmiştir.[11]

### Erişim yöntemi

Bilindiği gibi GSM de 200 KHz band genişliği olan kanallar TDMA yoluyla sekiz zaman dilimine ayrılmaktadır. Her bir zaman dilimi ayrı bir konuşmayı taşımaktadır. UMTS de tamamen farklı bir mantıkla birden çok kanal



CDMA tekniđi sayesinde aynı geniş bandlı (WCDMA için 5 MHz) Kanal üzerinden çođullanmaktadır.[11]

### **Frekans kullanımı**

UMTS' de her BS'de ve her BS in her sektöründe aynı yukarı bađlantı (uplink) ve ařađı bađlantı (downlink) frekansı kullanmaktadır. Bu GSM'de olduđu gibi aynı kanal giriřime neden olmaktadır. Ancak kullanılan kod fonksiyonları sayesinde bu problemin üstesinden gelinmektedir. Diđer BS'den veya aynı BS'in farklı sektörlerinden gelecek olan aynı kanal giriřimi, geniş bant gürültüsü denen toplam gürültünün artmasına sebep olacaktır. Toplam gürültü de alıcının istenilen iřareti çözmesini zorlařtıracaktır. Dolayısıyla CDMA hava ara yüzünde tek bir gezgin istasyondan gelecek aynı kanal giriřimi deđil tüm gezgin birimlerden kaynaklanan toplam gürültü önemlidir.[11]

### **UMTS' de güvenlik**

Bařlıca UMTS güvenlik özellikleri; kimlik dođrulama (Authentication), gizlilik (Confidentiality), gerçek ismini saklama (Anonymity), eriřim kontrolü (Access Control), bütünlük (Integrity), inkar edememe (Non-repudiation), bütünlüyici (Supplementary) olarak sıralanabilir.

### **b) Code division multiple access CDMA2000 (Kod bölmeli çoklu eriřim 2000)**

IMT-2000 için 3GPP2 grubunun önerisidir. Qualcomm'un orijinal CDMA standardının (cdmaOne adıyla bilinen) geniş band uygulamasıdır ve mevcut CDMA operatörlerinin 3G'ye geçiřinin kolay bir şekilde gerçekleştirilebilmesi için dizayn edilmiřtir. W-CDMA'da olduđu gibi aynı ortamda eř zamanlı iletiřim (duplexing) için FDD kullanılır. Farklı olarak CDMA2000'de 3.75 MHz band genişliđi kullanılıp, chip hızı da 3,6864 Mcps dır. Cdma2000 ve W-CDMA birbiriyle uyumsuz sistemlerdir.

CDMA2000 taşıyıcısı, daha hızlı chip hızına sahip tek bir 3.75 MHz taşıyıcısı gibi kullanılmak üzere bir araya getirilmiř üç standart 1,25 MHz taşıyıcıların toplamından oluşur. Üç taşıyıcının kullanılma nedeni, toplamlarından oluşan band genişliđinin IMT-2000 için istenen 2 Mbps hız ihtiyacını karřılıyor

olmasıdır. Bununla birlikte, kullanıcılardan gelen talepler doğrultusunda 1'den 12'ye kadar sayısı artırılabilen 1,25 MHz'lik kanalların kullanımı ile kanal genişliğini 15 MHz'e kadar çıkarmak mümkündür.[33]

**c) Time division-synchronous code division multiple access TD-SCDMA  
(Zaman bölmeli-senkron kod bölmeli çoklu erişim)**

TD-SCDMA, Çin'de Çin Telekomünikasyon Teknoloji Akademisi (CATT) ve Siemens tarafından hem TDMA hem de CDMA tabanlı bir teknoloji olarak geliştirilmiş ve RTT önerisi olarak ITU' ye sunulmuştur. TD-SCDMA'ın telsiz iletim yapısı temel erişim tekniklerinden FDMA, TDMA ve CDMA'ın esnek bir kombinasyonundan oluşmaktadır. UTRA TDD ise Avrupa tarafından önerilmiştir ve ETSI' nin RTT önerisinin ikinci işlemsel moduna dayanmaktadır. IMT-TC, 3GPP tarafından da desteklenmektedir. Hem UTRA TDD hem de TD-SCDMA, TDD moda çalışır, yani tekli bant (unpaired band) kullanır. Tekli bant, internet gibi asimetrik servislerin sunulması için uygun görülmektedir. Kamu mikro (kentsel) ve piko (bina içi) alanlarda (özellikle yüksek yoğunluk içeren şehir merkezleri, iş merkezleri, hava alanları ve fuarlar gibi bölgelerde) ve düşük hareketlilik (low mobility) gerektiren uygulamalarda kullanılır ve 2 Mbps hız sağlamaktadır. Çünkü bu bölgelerde trafik yüksek oranda asimetrik olma eğilimindedir ve yüksek bant genişliği gerektirir. [5]

**d) Digital enhanced cordless telecommunications DECT (Sayısal gelişmiş kablosuz telekomünikasyon)**

Avrupa'da kablosuz kamu telefon hizmetinde kullanılan sayısal gelişmiş kablosuz iletişim standardı (European Digital Enhanced Cordless Telecommunications- DECT) temelli bir standarttır. DECT+, 2.88 Mbps bit hızına ulaşabilme ve mobil terminal cihazlarına daimi bağlantı hizmeti sunmaktadır. TDD iletişimi ile TDMA tabanlı telsiz ara yüzü kullanılmaktadır. Bu standart Avrupa dışında fazla dikkat çekmemesinin yanı sıra 3G standart yarışında da zayıf bir adaydır.[7]

### **3. ÖĞRENCİ BİLGİ SİSTEMİ VE KULLANILAN ARAÇLAR**

#### **3.1. Öğrenci Bilgi Sistemi**

Bilgi sistemi en yaygın anlamıyla, enformasyonun, farklı gereksinimleri karşılamak üzere düzenlenmesi, işlenmesi, depolanması ve istenildiği zaman üretilmesi için organize edilmiş bir kurallar bütünüdür.[34]

Öğrenci bilgi sistemi ise; öğrencilerle ilgili bilgileri saklayan, sorgulayan ve kullanan bir bilgi sistemidir. Ve bu bilgi sistemiyle amaçlanan; ilgili yükseköğretim kurumu ile ilgili işlerin internet üzerinden yürütülebileceği bir sistem olarak kullanılmasını sağlamaktır. İçeriği daha farklı işlemleri yapmak için sonradan genişletilebilir olmakla beraber genel hatlarıyla yükseköğretim kurumuna dahil programlar, öğrenciler, yönetim, personel, dersler ve yurtlar ile ilgili bilgiler, haftalık ders programı, sınav takvimi, sınav sonuçları, duyurular ve sık sorulan sorular gibi alanları içerir.

Bu ortak bilgi sistemi sayesinde, ihtiyaç duyulan tüm bilgiler bir veritabanında saklanarak hizmete sunulur.

#### **3.2. Bilgiye Zamanında Erişim ve Öğrenci Bilgi Sistemi'nin Önemi**

Bilgilerden fayda sağlanabilmesi için; ihtiyaç duyanların bu bilgilere ihtiyaç duydukları zamanlarda ulaşılabilmesi gereklidir. Bilgilerin iletişim teknolojileri kullanılmadan sunulması bilgilere ihtiyaç duyanların erişiminde güçlükler yaşanmasını sağlar. Bu da bilgi talep edenlerin bilginin bulunduğu yerde bulunmalarını zorunlu kılar.

Günümüzde iletişim araçları ve özellikle internet çok gelişmiş ve yaygınlaşmıştır. Bu teknolojiler sayesinde insanlar buldukları yerden uzaktaki kişilerle, ses ve görüntünün yüksek hızda iletilmesi yani video konferans sayesinde aynı ortamdaymış gibi toplantı yapabilmektedir. Bu tür internet uygulamalarında bilgi istendiği zaman elde edilmektedir.

Benzer şekilde GSM hizmetleri de yaygınlık kazanmakta ve internet hizmetleri ile bütünleşik uygulamalar geliştirilmektedir. Bu çalışmada internet ve

mobil iletişim teknolojileri kullanılarak öğrencilerin bilgiye zamanında erişiminin sağlanması hedeflenmiştir.

Öğrenciler bu bilgi sistemini kullanarak istedikleri yer ve zamanda gerekli olan tüm bilgilere (haftalık ders programı, sınav sonuçları, iletişim, duyurular, vb.) ulaşabileceklerdir.

### **3.3. Veritabanı Kullanma Sebepleri**

Bir veritabanı bilgilerin depolandığı hiyerarşik bir yapıdır. Aynı bilgiler bir dosyaya yazılıp gerektiğinde dosya açılıp okunabilir ama bu işlem hem daha karmaşık ve zahmetli hem de daha yavaştır.

Veritabanları çok fazla miktarda veriyi düzenli bir şekilde depolayabilen ve gerektiğinde bir programlama dili ile hızlı bir şekilde verilere ulaşabilmemizi sağlayan araçlardır.

Veri'yi yönetmek, bu verilerin boyutları büyüdükçe daha zorlaşır. Birkaç parça veriyi yönetmek problem olmayabilir ama bunların boyutları arttıkça bunları etkili ve başarılı bir şekilde yönetebilmek için bir sisteme, yani veritabanına ihtiyaç duyulur.

Veritabanı aynı amaç altında bir araya getirilmiş farklı özelliklere ya da niceliklere sahip verilerden oluşur.[35]

### **3.4. Web'de Veritabanı Kullanımı**

Web sayfaları artık statik sayfalar olmaktan çıkıp dinamik ve etkileşimli sayfalar halinde yayınlanmaktadır. İçeriği sabit olan ve ziyaretçileri ile etkileşimde bulunmayan web sayfalarına statik, tasarımcının dışında değiştirilip, içeriği yenilenebilen sayfalara ise dinamik web sayfaları denir.

Bir web sayfasının başarısı geniş içeriği, kullanıcı ile etkileşimi ve sürekli yenilenmiş yapısı ile mümkündür. Bu da web sayfalarının statik HTML dosyalar yerine veritabanı üzerinden çalışabilen siteler olmasıyla sağlanabilir. Web sayfasında veritabanı kullanılabilmesi için veritabanı yönetim sistemine sahip olunması gerekir. Bu tez çalışmasında veri tabanı yönetim sistemi olarak MySQL kullanılmıştır.

### 3.5. PHP'nin Temelleri ve Avantajları

PHP bir script dilidir ve PHP ile yazılan kodlar bir editörde yazılıp PHP veya (kullanılan sürüme göre) PHP2, PHP3 gibi uzantılı dosya olarak kaydedilir. PHP ile yazılan dosyalar derlenmezler (compile edilmezler). Sadece web server'da bu dilde yazılmış scriptleri yorumlayabilecek bir PHP yorumlayıcı program mevcuttur. Bu yorumlayıcı yazılmış PHP scriptlerini web server'ın anlayabileceği bir biçime dönüştürür ve yollar.

PHP ile yazılmış bir web sayfasına bağlanıldığında temel olarak aşağıdaki işlemler yapılmaktadır:

1. İstemci (client) tarafından PHP dosyası bir tarayıcı (browser) ile çağrılır.
2. Web Server (web sunucu) dosya uzantısından bu isteğin bir PHP dosyası olduğunu algılar ve PHP yorumlayıcıya yollar.
3. PHP yorumlayıcısı ilgili dosya içindeki scriptleri çalıştırarak geriye döndürdüğü sonucu web server' a tekrar gönderir.
4. Web Server' a ulaştırılan sonuç istemci (client) tarafına HTML dosya olarak yollanır.

PHP oyun programı yazmaya kadar birçok amaca hitap edebilecek nitelikte bir dildir ve HTML dilinin yapamadığı işlemleri yapabilmek, HTML olarak daha işlevsel sayfalar yapabilmek, web server'a bir takım işler yaptırmak gibi yararlar sağlamak PHP'nin temellerindedir.

PHP'de yazdığımız kodların çalışabilmesi için yazdığımız kodların aşağıdaki komut ayrıçaları arasında olması gerekir.

```
<?
    "echo php scripti";
?>
```

PHP'nin avantajlarına gelince; her şeyi kendi yapı düzeninde çalıştırırken COM nesnelere ile haberleşmediği için yüksek bir hıza sahiptir. PHP ile web uygulamaları hızlı ve rahat kodlanabilir ve kod derlenmeden HTML içine yerleştirilerek çalışır. PHP ile SQL veritabanı sunucularına hızlı bağlantılar gerçekleştirilir ve istenen platform ve tarayıcıda rahatlıkla çalışılabilir.

### 3.5.1. PHP'nin yazım kuralları

PHP komutları ; (noktalı virgül) işaretiyle sonlandırılır. Echo, Print ve Printf: Ekrana bilgi yazdırma komutudur.

// : Tek satırlık açıklamalar için kullanılır

/\* ve \*/ : Birden çok açıklama satırlarında kullanılır. Bu komutlar arasına yazılan tüm PHP komutlarını dikkate alınmaz.

•\n Yeni satıra geçiş yapar (New Line)

•\r Satır başı (Return)

•\t Sekme (Tab) karakteri

<?

```
echo("merhaba \ PHP \ "dünyası");
```

?>

ekrana merhaba "PHP" dünyası şeklinde yazılmasını sağlar.

### 3.5.2. Değişkenler

Değişken bir verinin saklanmasında kullanılan hafıza alanıdır. PHP içerisinde bir değer hafızada tutulması için değişkene bir isim verip başına "\$" işareti getirmek gerekir. Değişkenler, harf, rakam veya alt çizgi (\_) ile başlayabilirler; bu karakterleri içerebilirler; ama içinde boşluk veya diğer işaretler bulundurmazlar.

<?

```
$adi;
```

```
$soyadi;
```

?>

Değişkenlerin harflerle başlaması ve içinde sayı ve harfin dışında karakter barındırmaması gerekir. PHP ile kullanılan değişkenlerde herhangi bir karakter sayısı limiti yoktur. En çok kullanılan değişken tipleri integer ve string tipleridir. Integer sayılar için, string ise karakter dizileri için kullanılmaktadır. PHP'nin asıl güzel yönü ise değişken tiplerinin belirtilmesine gerek olmamasıdır.

PHP’de deęişkenler sayıdan karaktere ya da tersine otomatik olarak geiş yaparlar; yani bir integer deęişken string’e, bir string deęişken de integer’a rahatlıkla dnüşebilir.

```
<?
$sayi = "1";
$sayi++;
echo $sayi;
?>
```

Örnekte \$sayi deęişkeni string deęişken olarak tanıtılmış ancak daha sonraki satırda bu deęişkene ++ işaretlerini ekleyerek \$sayi deęişkeninin deęerinin bir artırılması sağlanmıştır. Yani \$sayi deęişkeninin deęeri 2 olur ve deęişken tipide integer’a dnüşür.

Bir deęişkenin deęerini dięer bir deęişkene atamak için ise "=" den sonra ikinci deęişken yazılır.

```
<?
$adi = "Ahu";
$sisim = $adi;
print $sisim;
?>
```

Önce \$adi deęişkenine "Ahu" deęeri atanmakta ardından \$sisim deęişkenine \$adi deęişkeninin deęeri atanmaktadır. Sonuç olarak ekrana "Ahu" yazılacaktır.

### 3.5.3. Sabitler

Sabitler program boyunca deęişmeyen deęerlerdir. Bir sabit tanımlandıktan sonra atanan ilk deęerini korur, yeni bir deęer ataması yapılamaz. Sabitler define() komutu ile tanımlanırlar.

```
define (sabit_adi,deger)
<?
define (pi_sayisi , 3.14);
define (dis_acilar, 360);
$a1 = 130;
```

```
$a2 = 100;
$a3 = dis_acilar - ($a1 + $a2);
print ($a3); // 130 değerini döndürür.
?>
```

### 3.5.4. PHP’de kontrol yapıları

#### If Kontrol Yapısı

```
if (koşullar) {
    Komutlar
}
elseif (diğer koşullar) {
    Komutlar
}
else {
    Komutlar
}
```

If kontrol yapısı PHP dilindeki en temel kontrol yapılarından biridir. Bir koşulun olması veya olmaması durumunda yapılacak işlemleri ayırmak için kullanılır. Bir koşul doğru olduğu takdirde bir komut çalıştırılırken, şartın sağlanmadığı koşullarda ise farklı bir komutun çalıştırılması sağlanır. Bu da if yapısından sonra kullanacağımız else ile mümkündür.

```
<?
if ($a>$b)
{
    print ("a b'den büyük");
}
else if ($a==$b) {
    print ("a b'ye eşittir");
}
else
{
    print ("a b'den küçük");
}
```



```
}  
?>
```

Yukarıdaki örnekte a ve b değişkenlerinin birbirlerine göre küçük ve büyük olma durumları karşılaştırılmıştır. İlk if karşılaştırmasında a'nın b'den büyük olması durumunda ekrana a b'den büyük yazdırılması, büyük değilse eşitliğinin karşılaştırılması, eşitlik şartının da sağlanamaması durumunda a b'den küçük yazdırılması sağlanmıştır.

### Switch Kontrol Yapısı

Switch deyimi, verilen bir değişkenin değerinin belirttiğimiz koşullardan hangisine uygun olduğunu sınırlar ve o koşula ilişkin komutların çalıştırılmasını sağlar. PHP'nin yaptığı işi bitirdikten sonra switch deyiminin dışına çıkmasını sağlayan, break komutu vardır:

```
switch ($değişken)  
{  
    case koşul1;  
        komut_1 doğru ise yapılacak işlere ilişkin komutlar  
        break;  
    case koşul2;  
        komut_2 doğru ise yapılacak işlere ilişkin komutlar  
        break;  
    case koşul3;  
        komut_3 doğru ise yapılacak işlere ilişkin komutlar  
        break;  
    .....  
    .....  
    default:  
        komut ifadesi;  
}
```

Switch deyimini yazarken mutlaka break ifadesini kullanmamız gerekmez; switch yapısında break kullanılmazsa bile, program ilk şartın

gerçekleşmesi durumunda dahi diğer şartları da sınamaya devam eder ve en sondaki default deyimine ulaşarak, bu bölümdeki komutları çalıştırır.

### 3.5.5. PHP’de döngüler

#### While Döngüsü

Programlama dilinde bir komut ifadesinin şart sağlandığı sürece tekrar tekrar çalıştırılması istenebilir. Programın çalışırken while bloğuna gelindiğinde koşul kontrol edilir ve koşul sağlanıyorsa while döngüsü içerisine girilir.

```
while (şart) {  
    komut ifadesi  
}  
  
<?  
$cnt = 0;  
while ($cnt <= 100)  
{  
    $cnt ++;  
    $toplam = $toplam + $cnt;  
}  
print ($toplam);  
?>
```

Yukarıdaki örnek 1'den 100'e kadar olan sayıların toplamını bulan bir program örneğidir. İlk olarak \$cnt değişkeni tanımlanmış ve bu değişkene 0 değeri atanmıştır. While döngüsünde \$cnt<= 100 şartı konulmuş yani \$cnt değişkeninin 100'e eşit ve 100'den küçük olduğu müddetçe çalıştırılması sağlanmıştır. \$cnt++ ile değişkenin değeri bir artırılmış ve \$toplam değişkeninde \$cnt 100 değerini alıncaya kadarki toplamları gerçekleştirilmiştir.

#### Do – While Döngüsü

While döngüsünden farklı olarak programın çalıştırılması sırasında do - while bloğuna gelindiğinde şart kontrol edilmez. do-while bloğuna girilir ve koşul ikinci döngünün başlayacağı sırada kontrol edilir

```
do {  
    Şart sağlanıyorsa yapılacaklar  
}  
while (şart);
```

### **For Döngüsü**

Döngünün belirli bir sayıda çalıştırılması gereken durumlarda for döngüsü kullanılır. Döngüden çıkılması istenirse break komutu kullanılır. for bloğunun herhangi bir yerinde bir sonraki tekrara geçmek için ise continue komutu kullanılmalıdır.

Ayrıca for döngüsü şartın yerine gelmesini sağlayacak arttırma veya eksiltme işlemlerini kendisi yaparak sonsuz döngüye girme durumunu ortadan kaldırmış olur.

for döngüsünde ilk öge, sayac 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, sayac değişkeninin artış hızı verilir.

```
for ( $degisken = ilk_deger ; kosul ; adim ) {  
    Koşul doğru ise yapılacak işlere ilişkin komutlar  
}
```

### **3.5.6. Fonksiyonlar**

Fonksiyonlar, bir kez tanımlanır ve birçok kez ihtiyaç olduğu zaman yeniden çağrılırlar. Fonksiyonlar genellikle bir iş veya hesaplama yaparlar ve bunun sonucu olarak ortaya yeni bir değer çıkartırlar.

PHP'de fonksiyonlar function komutu ile oluşturulur. Tanımladığımız fonksiyon, kendisini göreve çağırarak komuttan, yapacağı işlemde kullanmak üzere değer alacaksa, bu değerlere vereceğimiz değişken isimleri fonksiyon adının yanında parantez içinde gösterilir. Fonksiyon birden fazla değer bekleyecekse, bunların değişken adlarının arasına virgül konulur. Fonksiyonun dışardan aldığı değerlere argüman adı verilir. Bir fonksiyon birden fazla argüman kullanabilir. Fakat geriye tek değer döndürür. Bir fonksiyon dışardan aldığı değerlere karşılık

bir deęer döndürme mantığıyla çalışır. Bir deęer geri döndürölmek istendięinde ‘return’ komutu kullanılmalıdır.

Buna göre PHP scripti içerisinde bir fonksiyon şöyle tanımlanır:

```
function fonksiyon_adi ($Sargüman_1, $Sargüman_2, ... $Sargüman_n)
{
fonksiyonun yapacağı işe ilişkin komutlar
return $geriye_donen_sonuc;
}
```

Deęişken adlarında kullanılan kurallar fonksiyon adlarında da geçerlidir. Bir fonksiyon içerisinde başka bir fonksiyon çağrılabilir. PHP’de fonksiyonlar dięer programlama dillerinde olduęu gibi iki türdedir. İlki PHP içerisindeki hazır fonksiyonlar ve ikincisi programcı tarafından oluşturulan fonksiyonlardır. Kendi yazdığımız fonksiyonlarımız da PHP içerisinde kullanabilir fakat öncelikle fonksiyonlarımızın tanımlanması gerekir.

Bir PHP scriptinde fonksiyonun beklenen işi yapamaması durumunda ise die () komutu fonksiyonun durdurulmasını sağlar. Fonksiyon durdurulduęu takdirde ekrana gelmesini istediğimiz metinde gösterilebilir.

```
$liste=@mysql_query("select * from dersler order by kodu") or
die("Dersler tablosuna ulaşamadı!");
```

### **3.6. PHP’nin Özellikleri ve PHP ile Yapabileceklerimiz**

PHP dilinin ilk amacı, web geliştiricilerin dinamik sayfaları hızlı ve kolay hazırlayabilmesini sağlayarak kullanıcı ile interaktif iletişim kurabilmektir.

Bunun yanında PHP ile;

- veritabanı bağlantılı uygulamalar
- dinamik olarak oluşan grafikler
- kullanıcıya, tarayıcıya ve tarihe göre özel durumlar veya içerikler
- tartışma forumları
- elektronik ticaret uygulamaları
- web tabanlı e-posta uygulamaları
- XML verilerini okuma ve oluşturma
- WAP uygulamaları yapılabilir.[36]

PHP'nin en önemli özelliklerinden biri de çok geniş bir veritabanı kullanımını desteğini sağlamasıdır. PHP sayesinde ileri seviyedeki veritabanlarının kullanımını rahatlıkla gerçekleştirilebilir.

### **3.7. MySQL Özellikleri**

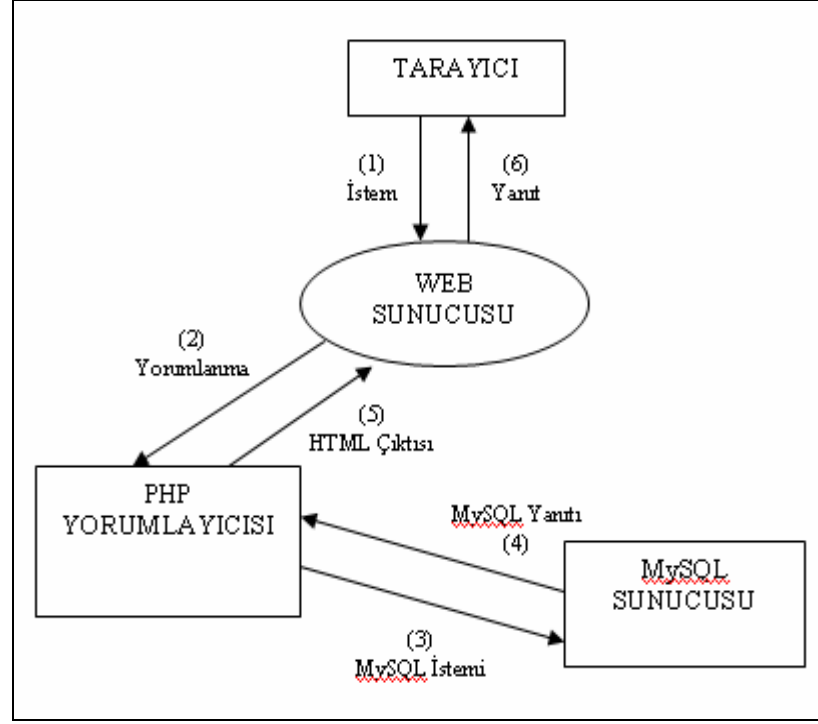
MySQL yüksek performanslı, güvenilir ve kolay kullanılabilen bir açık kaynak SQL veritabanı sistemidir. Çok büyük miktarlarda verileri depolayabilir ve aynı bir web sunucusu gibi arka planda çalışan ve herhangi bir zamanda bir bilgi isteği olduğu zaman buna cevap veren bir sunucu olarak düşünülebilir.

MySQL; hem standart client/server yapısıyla hem de embedded veritabanı olarak kullanılır. Tamamıyla C/C++ kullanılarak yazılmıştır ve gcc araçlarının mevcut olduğu tüm platformlarda çalışır (Windows, Linux, OpenBSD v.s.) [37]

MySQL; tüm verileri tek bir ambara yığmak yerine farklı tablolarda ve veritabanlarında düzenli bir biçimde saklar. SQL adı verilen, veritabanlarına erişmek için kullanılan en yaygın ve standart dili kullanır. Apache ve PHP ile beraber web-veritabanı uygulamalarında çok yaygın olarak kullanılır. Apache-PHP-MySQL üçlüsü için hazırlanmış çok geniş bir yazılım yelpazesi bulunmaktadır. Özellikle internet ortamında önem kazanan, çok esnek ve güçlü bir kullanıcı erişim kısıtlama/yetkilendirme sistemine sahiptir. [38]

### **3.8. MySQL ve PHP Kullanımı**

Tarayıcımız ile PHP ve MySQL kullanılan bir web sayfasına erişildiğinde Şekil 3.1'deki işlemler sırasıyla gerçekleşir.



Şekil 3.1. PHP ve MySQL'in bir arada çalışması

1. Tarayıcımız web sunucusu ile bağlantı kuracak, görmek istediğimiz web sayfasını web sunucusundan isteyecektir.

2. Web sunucusu, istemi yapılan sayfanın bir PHP scripti olduğunu anlayacak, dolayısı ile bu sayfanın önce yorumlanıp daha sonra bize gönderilmesine karar verecek ve sayfayı yorumlaması için PHP yorumlayıcısını devreye sokacaktır.

3. PHP yorumlayıcısı sayfayı yorumlarken karşılaşıacağı bazı komutlar olması durumunda MySQL veritabanı ile iletişim kurması gerektiğini anlayacaktır ve gerekli istemleri MySQL veritabanı sunucusuna yapacaktır.

4. MySQL veritabanı sunucusu, PHP yorumlayıcısı tarafından isteme göre veritabanında gerekli işlemleri yapacaktır. Bu işlemler sonucu döndürülmesi gereken verileri de PHP yorumlayıcısına yanıt olarak iletacaktır.

5. PHP yorumlayıcısı MySQL veritabanı sunucusundan aldığı yanıtı göre gerekli işlemleri yapacak, daha sonra da bizim hazırladığımız şekilde bir HTML çıktısını web sunucusuna iletacaktır.

6. Web sunucusu da PHP yorumlayıcısından bu dinamik olarak hazırlanmış sayfayı alıp basit bir web sayfasıymış gibi tarayıcıımıza geri gönderir.

[35]

### 3.8.1. MySQL'e bağlanması ve veritabanı seçilmesi

PHP kullanarak MySQL'e bağlanmak için;

```
Mysql_connect("localhost","root","sifre");
```

```
Mysql_select_db("baslangicdb");
```

komut satırları kullanılır.

Bir PHP dosyasında MySQL veritabanı kullanmak istendiğinde mutlaka veritabanı sunucusuna bağlanmalı ve bir veritabanı seçilmelidir. Mysql\_connect komutuyla veritabanı sunucusuna IP, kullanıcı adı ve şifre verilerek bağlantı sağlanır daha sonrada; Mysql\_select\_db komutuyla da dosyadaki tüm MySQL işlemlerinde kullanılacak olan veritabanı tercihi belirtilmiş olur. PHP dosyalarında tüm veritabanı işlemlerinde önce veritabanı sunucusuyla bağlantı kurulmalıdır.

### 3.8.2. MySQL veritabanında sorgu yapılması

PHP'de MySQL'e sorgu göndermek için mysql\_query() komutu kullanılır. Parantezler arasına çift tırnak işareti kullanılarak komutlar gönderilir ve PHP değişkenine aktarılır.

```
mysql_query()
```

```
$isimler = mysql_query("select * from öğrenciler");
```

```
mysql_result()
```

```
$ad=mysql_result($isimler,$i,"ad");
```

```
mysql_numrows()
```

```
$sayi=mysql_numrows($isimler)
```

```
mysql_fetch_array()
```

```
$gelen = mysql_fetch_array($result)
```

Bu örnekte öğrenciler adındaki tablodan tüm kayıtların çekilmesi SQL ile istenmiştir. MySQL sorgulanmasında geriye dönen sonuç tablosundaki veriler

mysql\_result() alınır. mysql\_numrows sonuç tablosundaki satır sayısını öğrenmek için kullanılır. mysql\_fetch\_array sonuçları dizi deęişkeni olarak alır. [39]

### **3.8.3. MySQL bağlantısının kapatılması**

PHP scriptimizde MySQL ile işimiz bittiğinde bağlantıyı sonlandırmak için;

```
Mysql_close(baglantinumarasi);
```

fonksiyonu kullanılır. Bu komut satırı MySQL bağlantısının kapatılmasını sağlar.



## **4. UYGULAMA**

### **4.1. Problem**

Öğrencilerin okulları tarafından açıklanan bilgilere erişmek için sabit sistemleri kullanması gerekliliği onlara sınırlayıcı bir durum yaratmaktadır.

Bu sebepten dolayı istedikleri anda ve buldukları konumdan bu bilgilere ulaşmaları mümkün olmayabilir. Bilgiye zamanında erişimin önemi düşünüldüğünde bu durumun yaratacağı sonuçların problemlere yol açacağı anlaşılabilir.

### **4.2. Amaç**

Öğrencilerin ihtiyaç duydukları bilgiye istedikleri anda ve istedikleri yerden ulaşabilmeleri için öğrenci bilgi sistemini oluşturmak.

İhtiyaç duyabilecekleri tüm verilerden oluşan bilgi sistemini bir veritabanı altında toplamak ve bunları internet üzerinden ulaşılabilir hale getirmek.

Öğrencilerin bu bilgilere sabit kalma zorunluluğu olmadan yalnızca cep telefonlarını kullanarak mobil sistemler üzerinden ulaşmalarını sağlamak.

### **4.3. Yöntem**

Bu uygulamanın gerçekleştirilebilmesi için aşağıdaki işlemler sırasıyla hazırlanmıştır.

Veritabanı oluşturuldu.

Veri girişinin sağlanabilmesi için kontrol panel hazırlandı.

Bu verilerin öğrenciler tarafından görüntülenebilmesi için wap sitesi oluşturuldu.

Uygulama Apache Web Server üzerinde geliştirildi. Php programlama dili ve MySQL veri tabanı kullanıldı. PhpMyAdmin arabirimi kullanılarak tablolar oluşturuldu. Macromedia Dreamweaver ile kontrol panel ve wap sitesi hazırlandı.

#### 4.4. Veritabanının Oluşturulması

Öğrencilerin, ders programı, akademik takvim, sınav takvimi, sınav sonuçları, personel, yönetim, duyurular, yurtlar, sıkça sorulan sorular gibi bilgilere ulaşmasını sağlamak amacıyla MySQL veritabanıyla 12 adet tablo hazırlandı. Bu tablolar;

Programlar tablosu; kodu, adi, detay

Öğrenciler tablosu; ogrencino, adi, soyadi, programkodu, sifre

Yönetim tablosu; sicilno, unvan, adi, soyadi, gorevi, detay

Personel tablosu; sicilno, adi, soyadi, unvani, gorevi, detay

Statiksayfalar tablosu; sayfano, metin, sayfabasligi

Yurtlar tablosu; kodu, adi, metin

Dersler tablosu; kodu, adi, programkodu

Haftaliktakvim tablosu; derskodu, gun, baslangic, bitis

Sinavtakimi tablosu; derskodu, tarih, saat

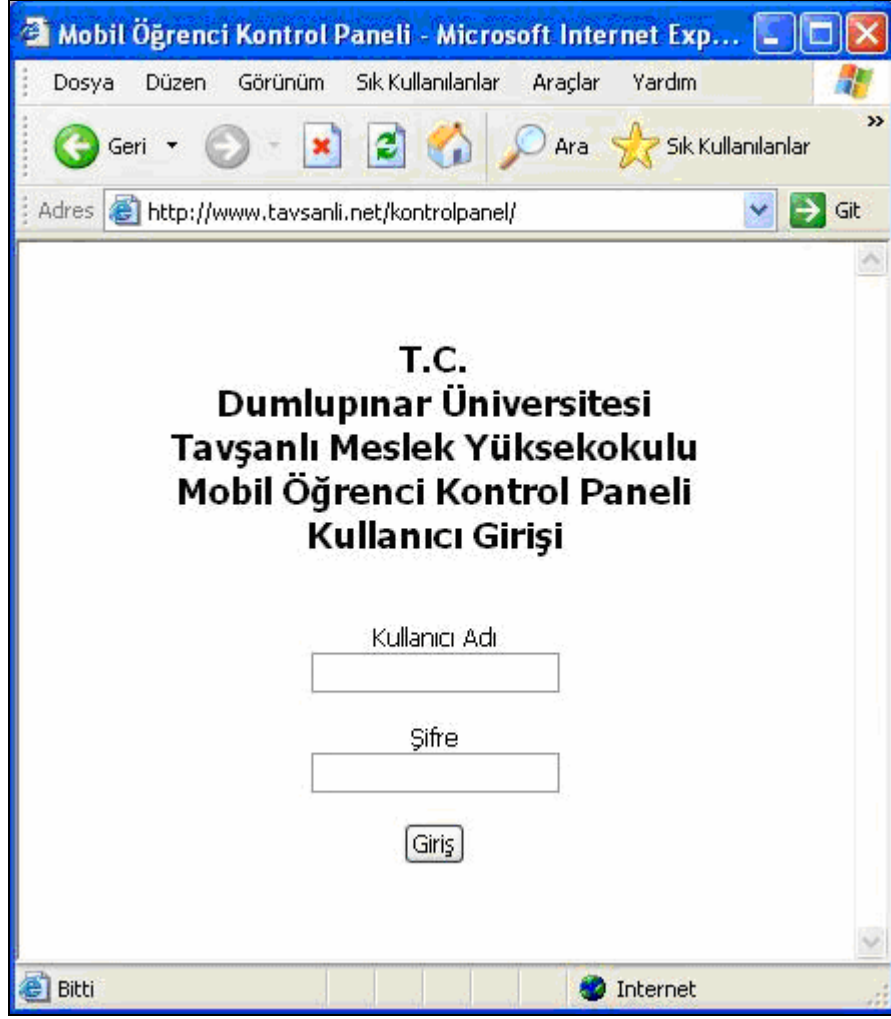
Sinavsonuclari tablosu; ogrencino, derskodu, vize1, vize2, final, ortalama, harfnotu

Duyurular tablosu; kodu, konu, metin

Sss tablosu; kodu, konu, metin alanlarından oluşacak şekilde hazırlandı.

#### 4.5. Kontrol Panel

Veri girişinin sağlanabilmesi için oluşturulan kontrol panel Macromedia Dreamweaver programı ile tasarlandı. Yalnızca tanımlanan kullanıcı adı ve atanan şifre ile girişin sağlanabileceği ana sayfa aşağıdaki Şekil 4.1'deki gibi görüntülenmektedir. Kullanıcı adı ve şifrenin doğru olması durumunda veri girişi gerçekleştirilebilir. Eğer kullanıcı adı ve şifreden herhangi biri yanlış ise sürekli Şekil 4.1'deki login ekranı açılacaktır.



**Şekil 4.1 Kontrol Panel Login Ekranı**

Login ekranında girilen kullanıcı adı ve şifrenin doğru olması durumunda ise ekran görüntüsü Şekil 4.2'ye dönüşecektir.

### Menü

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)

### Statik Sayfalar

- [Okulumuz Hakkında](#)
- [İletişim Bilgileri](#)
- [Tavşanlı Hakkında](#)
- [Akademik Takvim](#)

### Programlar

| [Ekle](#) | [Listele](#) |

Sıra No	Kodu	Adı		
1	btp	Bilgisayar Teknolojisi ve Programlama	<a href="#">Değiştir</a>	<a href="#">Sil</a>
2	buro	Büro Yönetimi ve Sekreterlik	<a href="#">Değiştir</a>	<a href="#">Sil</a>
3	eo	Endüstriyel Otomasyon	<a href="#">Değiştir</a>	<a href="#">Sil</a>
4	hmo	Harita ve Maden Ölçme	<a href="#">Değiştir</a>	<a href="#">Sil</a>
5	ith-ıhr	İthalat-İhracat	<a href="#">Değiştir</a>	<a href="#">Sil</a>
6	mak	Makine	<a href="#">Değiştir</a>	<a href="#">Sil</a>
7	muh	Muhasebe	<a href="#">Değiştir</a>	<a href="#">Sil</a>
8	oto	Otomotiv	<a href="#">Değiştir</a>	<a href="#">Sil</a>

Liste sonu...

Şekil 4.2: Kontrol Panel Anasayfa

2 panelden oluşan programda sol taraftaki ana menü alanı programda ulaşabileceğimiz verileri, sağ taraftaki alan ise solda yapılan seçimin sonucundaki içeriği göstermektedir.

Ana menü’de ilk kısım dinamik sayfalara işaret eder. Bunlar programlar, dersler, öğrenciler, haftalık ders programı, sınav takvimi, sınav sonuçları, yönetim, personel, yurtlar, duyurular, sss ve çoklu veri girişi’dir.

Menü’nün alt kısmında yer alan statik sayfalar ise okulumuz hakkında, iletişim bilgileri, tavşanlı hakkında ve akademik takvim’dir.

Her biri bir bağlantıya sahip olan bu alanlarda seçim yapılan seçeneğin içeriği sağ tarafta görüntülenir.

#### 4.5.1. Programlar

Ana menü'deki programlar bağlantısına tıkladığımızda Şekil 4.3'deki ekran görüntüsü karşımıza gelir. Bu alanda sıranın, kodu ve adı'yla birlikte okuldaki programlar listelenir. Değiştir bağlantısına tıkladığımızda seçili programa ait bilgilerin değiştirilebileceği ekranla karşılaşırız. Sil bağlantısı ile ise seçili programın listeden çıkartılmasını sağlar.

Ekle bağlantısına tıkladığımızda yeni program eklememizi sağlayan Şekil 4.4'deki ekran görüntüsü açılır.

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

### Menü

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)
  
- Statik Sayfalar
  - [Okulumuz Hakkında](#)
  - [İletişim Bilgileri](#)
  - [Tavşanlı Hakkında](#)
  - [Akademik Takvim](#)

### Programlar

[Ekle](#) | [Listele](#) |

Sıra No	Kodu	Adı		
1	btp	Bilgisayar Teknolojisi ve Programlama	<a href="#">Değiştir</a>	<a href="#">Sil</a>
2	buro	Büro Yönetimi ve Sekreterlik	<a href="#">Değiştir</a>	<a href="#">Sil</a>
3	eo	Endüstriyel Otomasyon	<a href="#">Değiştir</a>	<a href="#">Sil</a>
4	hmo	Harita ve Maden Ölçme	<a href="#">Değiştir</a>	<a href="#">Sil</a>
5	ith-ıhr	İthalat-İhracat	<a href="#">Değiştir</a>	<a href="#">Sil</a>
6	mak	Makine	<a href="#">Değiştir</a>	<a href="#">Sil</a>
7	muh	Muhasebe	<a href="#">Değiştir</a>	<a href="#">Sil</a>
8	oto	Otomotiv	<a href="#">Değiştir</a>	<a href="#">Sil</a>

Liste sonu...

Şekil 4.3: Programlar

## Menü

[Programlar](#)  
[Dersler](#)  
[Öğrenciler](#)  
[Haftalık Ders Programı](#)  
[Sınav Takvimi](#)  
[Sınav Sonuçları](#)  
[Yönetim](#)  
[Personel](#)  
[Yurtlar](#)  
[Duyurular](#)  
[SSS](#)  
[Çoklu Veri Girişi](#)

### Statik Sayfalar

[Okulumuz Hakkında](#)  
[İletişim Bilgileri](#)  
[Tavşanlı Hakkında](#)  
[Akademik Takvim](#)

## Programlar

| [Ekle](#) | [Listele](#) |

### Yeni Program Ekleme

Program Kodu

Program Adı

Program Hakkında  
Detaylı Bilgi

Kaydet

Şekil 4.4: Yeni Program Ekleme

Yeni bir program eklemek istediğimiz takdirde program kodu, program adı ve program hakkında detaylı bilgi alanları doldurularak kaydet butonuna basılır ve yeni program listeye eklenir. Sayfanın üst kısmında bulunan listele bağlantısına tıkladığımızda ise son eklediğimiz program ile birlikte okuldaki tüm programlar görüntülenir.

## 4.5.2. Dersler

Ana menüdeki dersler bağlantısına tıkladığımızda; her bir dersin ders kodu, adı, hangi programa dahil olduğunu gösteren program kodu ile birlikte eklenen tüm dersler görüntülenir. (Şekil 4.5)

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

Menü		Dersler			
<a href="#">Programlar</a> <a href="#">Dersler</a> <a href="#">Öğrenciler</a> <a href="#">Haftalık Ders Programı</a> <a href="#">Sınav Takvimi</a> <a href="#">Sınav Sonuçları</a> <a href="#">Yönetim</a> <a href="#">Personel</a> <a href="#">Yurtlar</a> <a href="#">Duyurular</a> <a href="#">SSS</a> <a href="#">Çoklu Veri Girişi</a>  <a href="#">Statik Sayfalar</a> <a href="#">Okulumuz Hakkında</a> <a href="#">İletişim Bilgileri</a> <a href="#">Tavşanlı Hakkında</a> <a href="#">Akademik Takvim</a>		<a href="#">Ekle</a>   <a href="#">Listele</a>			
Sıra No	Kodu	Adı	Program Kodu		
1	331111001	A.İ.İ.T. - I	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
2	331111003	Türk Dili - I	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
3	331111005	İngilizce - I	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
4	331111319	Bilgisayara Giriş	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
5	331111324	Büro Yönetimi ve İletişim Teknikleri	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
6	331111359	Genel Hukuk Bilgisi	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
7	331111361	Genel İşletme	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
8	331111364	Genel Muhasebe	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
9	331111386	İktisada Giriş	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
10	331111430	Matematik	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
11	331111701	Bedensel Eğitimi - I (Seçmeli)	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
12	331112002	A.İ.İ.T. - II	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>

Şekil 4.5: Dersler Sayfası Listele

Değiştir bağlantısına tıkladığımızda ilişkili derse ait bilgilerin değiştirileceği ekran görüntüsü karşımıza gelir. Sil komutu ise ilişkili dersin listeden çıkartılmasını sağlar. Ekle bağlantısına tıkladığımızda ise istenilen dersin listeye eklenmesi işlemini sağlayan yeni ders ekleme sayfası açılır. Yeni ders

eklemek için dersin kodu, dersin adı ve ait olduğu programın adı bilgilerinin girilmesi ve kaydet butonuna basılması yeterlidir. (Şekil 4.6)

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

**Menü**

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)

**Statik Sayfalar**

- [Okulumuz Hakkında](#)
- [İletişim Bilgileri](#)
- [Tavşanlı Hakkında](#)
- [Akademik Takvim](#)

**Dersler**

| [Ekle](#) | [Listele](#) |

**Yeni Ders Ekleme**

Ders Kodu

Ders Adı

Program

Şekil 4.6: Yeni Ders Ekleme

Listele bağlantısıyla ise son eklenen ders ile birlikte tüm programlara ait tüm derslerin listelendiği ekran görüntüsü karşımıza çıkar.

#### 4.5.3. Öğrenciler

Ana menüdeki öğrenciler bağlantısına tıkladığımızda ekran görüntüsü Şekil 4.7'dekine dönüşür. Bu ekranda okul numaraları küçükten büyüğe artacak şekilde sıralanmış olmak üzere tüm bölümlere ait tüm öğrencilerin kayıtları görüntülenir. Sıra numarası eklediğimiz her öğrenciye göre otomatik olarak



artacak şekilde düzenlenmiştir. Listedeki program kodu öğrencinin dahil olduğu programı işaret eder. Değiştir bağlantısı seçili öğrenciye ait bilgilerin değiştirilebileceği ekranın açılmasını, sil ise öğrencinin listeden çıkartılarak kayıtlardan silinmesini sağlar.

T.C. Dumlupınar Üniversitesi Tavşanlı Meslek Yüksekokulu Mobil Öğrenci Kontrol Paneli						
Menü		Öğrenciler				
<a href="#">Programlar</a> <a href="#">Dersler</a> <a href="#">Öğrenciler</a> <a href="#">Haftalık Ders Programı</a> <a href="#">Sınav Takvimi</a> <a href="#">Sınav Sonuçları</a> <a href="#">Yönetim</a> <a href="#">Personel</a> <a href="#">Yurtlar</a> <a href="#">Duyurular</a> <a href="#">SSS</a> <a href="#">Çoklu Veri Girişi</a>  <a href="#">Statik Sayfalar</a> <a href="#">Okulumuz Hakkında</a> <a href="#">İletişim Bilgileri</a> <a href="#">Tavşanlı Hakkında</a> <a href="#">Akademik Takvim</a>		<a href="#">Ekle</a>   <a href="#">Listele</a>				
Sıra No	Öğrenci No	Adı	Soyadı	Program Kodu		
1	199633131013	Cahit	GÖĞÜSLÜ	ith-ih	<a href="#">Değiştir</a>	<a href="#">Sil</a>
2	199733121037	İbrahim Halil	PARLAK	btp	<a href="#">Değiştir</a>	<a href="#">Sil</a>
3	199733141706	Mehmet bahadır	AYDINLI	btp	<a href="#">Değiştir</a>	<a href="#">Sil</a>
4	199833141007	Kenan burak	ELİBOL	btp	<a href="#">Değiştir</a>	<a href="#">Sil</a>
5	199933131043	Ersin	KEŞEN	ith-ih	<a href="#">Değiştir</a>	<a href="#">Sil</a>
6	199933132009	Murat	ÇELİK	ith-ih	<a href="#">Değiştir</a>	<a href="#">Sil</a>
7	199933151014	Levent	BÜYÜKCİVELEK	hmo	<a href="#">Değiştir</a>	<a href="#">Sil</a>
8	199933161003	Mustafa	ALTINSOY	mak	<a href="#">Değiştir</a>	<a href="#">Sil</a>
9	200233111013	Esra	ÇALIŞ	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
10	200233111021	Kenan	DEMİREL	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>
11	200233111024	Hacer	DURMUŞ	muh	<a href="#">Değiştir</a>	<a href="#">Sil</a>

Şekil 4.7: Öğrenciler Sayfası

Şekil 4.7'deki öğrenciler sayfasında ekle bağlantısına tıkladığımızda yeni öğrenci ekleme ekran görüntüsü karşımıza çıkar. (Şekil 4.8) Listeye eklenecek öğrencinin numarası, adı, soyadı ve ait olduğu program bilgileriyle birlikte, wap sitesine eriştiğinde kişisel bilgilerine erişebilmesi için gerekli olacak şifre tanımlaması da bu ekranda yapılır. Kaydet butonuyla bilgileri girilen öğrencinin

kayıt işlemleri tamamlanır ve listele bağlantısıyla görüntülediğinde kayıtlı olan tüm öğrencilerle birlikte yeni eklediğimiz öğrenci bilgileri de görüntülenir.

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

**Menü**  
[Programlar](#)  
[Dersler](#)  
[Öğrenciler](#)  
[Haftalık Ders Programı](#)  
[Sınav Takvimi](#)  
[Sınav Sonuçları](#)  
[Yönetim](#)  
[Personel](#)  
[Yurtlar](#)  
[Duyurular](#)  
[SSS](#)  
[Çoklu Veri Girişi](#)  
  
**Statik Sayfalar**  
[Okulumuz Hakkında](#)  
[İletişim Bilgileri](#)  
[Tavşanlı Hakkında](#)  
[Akademik Takvim](#)

**Öğrenciler**  
[Ekle](#) | [Listele](#) |  
  
**Yeni Öğrenci Ekleme**  
Öğrenci No   
Adı   
Soyadı   
Program   
Şifre

Şekil 4.8: Yeni Öğrenci Ekleme Sayfası

#### 4.5.4. Haftalık ders programı

Ana menüdeki haftalık ders programı bağlantısıyla adı belirtilen dersin yapılacağı gün ve saat bilgilerinin girilmesini sağlayan haftalık ders programı sayfası açılır.(Şekil 4.9)

Menü	Haftalık Ders Programı				
	Ders Adı	Gün	Başlangıç Saati	Bitiş Saati	
<a href="#">Programlar</a> <a href="#">Dersler</a> <a href="#">Öğrenciler</a> <a href="#">Haftalık Ders Programı</a> <a href="#">Sınav Takvimi</a> <a href="#">Sınav Sonuçları</a> <a href="#">Yönetim</a> <a href="#">Personel</a> <a href="#">Yurtlar</a> <a href="#">Duyurular</a> <a href="#">SSS</a> <a href="#">Çoklu Veri Girişi</a>  Statik Sayfalar <a href="#">Okulumuz Hakkında</a> <a href="#">İletişim Bilgileri</a> <a href="#">Tavsaneli Hakkında</a> <a href="#">Akademik Takvim</a>	Bilgisayar Teknolojisi ve Programlama - A.İ.İ.T. - I	Pazartesi	00:00	00:00	Ekle
	Muhasebe - A.İ.İ.T. - II	Çarşamba	19:00	21:00	
	Muhasebe - Türk Dili - II	Salı	17:00	19:00	
	Muhasebe - İngilizce - II	Çarşamba	13:00	17:00	
	Muhasebe - Bilgisayar Büro Programları	Cuma	18:00	22:00	
	Muhasebe - Dönem Sonu Muhasebe İşlemleri	Cuma	10:00	14:00	
	Muhasebe - İstatistik	Perşembe	17:00	19:00	
	Muhasebe - İşletme Becerileri Grup Çalışması	Pazartesi	15:00	17:00	
	Muhasebe - Makro İktisat	Salı	19:00	21:00	
	Muhasebe - Ticaret Hukuk Bilgisi	Cuma	16:00	18:00	

Şekil 4.9: Haftalık Ders Programı

Bu sayfada öncelikle ders programına eklenecek olan ders ve dersin hangi gün yapılacağı açılır liste kutusundan seçilir ve ardından dersin başlangıç ve bitiş saatleri girilerek ekle butonuyla dersin ders programına kayıt işlemi gerçekleştirilir. Ders programından çıkartılmak istenen bir ders ise sil butonuna basarak listeden çıkartılabilir.

#### 4.5.5. Sınav takvimi

Menü	Sınav Takvimi			
	Ders Adı	Tarih	Saat	
<a href="#">Programlar</a> <a href="#">Dersler</a> <a href="#">Öğrenciler</a> <a href="#">Haftalık Ders Programı</a> <a href="#">Sınav Takvimi</a> <a href="#">Sınav Sonuçları</a> <a href="#">Yönetim</a> <a href="#">Personel</a> <a href="#">Yurtlar</a> <a href="#">Duyurular</a> <a href="#">SSS</a> <a href="#">Çoklu Veri Girişi</a>  Statik Sayfalar <a href="#">Okulumuz Hakkında</a> <a href="#">İletişim Bilgileri</a> <a href="#">Tavsaneli Hakkında</a> <a href="#">Akademik Takvim</a>	Bilgisayar Teknolojisi ve Programlama - A.İ.İ.T. - I	00.00.0000	00:00	Ekle
	Muhasebe - A.İ.İ.T. - II	31.05.2006	09:30	
	Muhasebe - Türk Dili - II	05.06.2006	09:30	
	Muhasebe - İngilizce - II	06.06.2006	13:30	
	Muhasebe - Bilgisayar Büro Programları	08.06.2006	15:00	
	Muhasebe - Dönem Sonu Muhasebe İşlemleri	07.06.2006	13:30	
	Muhasebe - İstatistik	09.06.2006	14:00	
	Muhasebe - İşletme Becerileri Grup Çalışması	08.06.2006	16:30	
	Muhasebe - Makro İktisat	02.06.2006	14:00	
	Muhasebe - Ticaret Hukuk Bilgisi	02.06.2006	16:30	
	Muhasebe - Ticari Matematik	01.06.2006	13:30	

Şekil 4.10: Sınav Takvimi

Ana menüdeki sınav takvimi bağlantısı ise bizi hangi sınavın hangi tarih ve saatte yapılacağını girebileceğimiz sayfaya ulaştırır. Ders adı açılır liste kutusundan seçilir ve tarih, saat bilgisinin girilmesinin ardından ekle butonuyla listeye dahil edilir.

Sınav takviminden çıkartılmak istenen ders ise; ders adı bulunduğundan sonra yanındaki sil butonuna basılarak listeden çıkartılır.

#### 4.5.6. Sınav sonuçları

Sınav sonuçlarını girmek için ana menüdeki sınav sonuçları bağlantısı kullanılır. Şekil 4.11'deki ekran ile açılır liste kutusundan numarası ve adını seçtiğimiz öğrencinin yine ders alanındaki açılır liste kutusundan hangi derse ait bilgilerinin girileceği seçilir. Vize1, vize2, final, ortalama ve harf notu'nun girilmesiyle ilişkili öğrencinin belirtilen dersine ait sınav sonuçları girilmiş olur.

T.C. Dumlupınar Üniversitesi Tavşanlı Meslek Yüksekokulu Mobil Öğrenci Kontrol Paneli					
Sınav Sonuçları					
Öğrenci	Ders	Vize1	Vize2	Final	
199633131013 - Cahit GÖĞÜSLÜ	Bilgisayar Teknolojisi ve Programlama - A.İ.İ.T. - I				
200233112002 - Aysun AKSAYER	Muhasebe - Bilgisayar Büro Programları	44	0	60	
200233142004 - Besime Betül ÇAVUŞOĞLU	Bilgisayar Teknolojisi ve Programlama - Veri Tabanı Yönetim Sistemleri - I	80	0	41	
200333111020 - Fatih KARAKAYA	Muhasebe - Bilgisayar Büro Programları	48	0	65	
200333111026 - Halil İbrahim OVALI	Muhasebe - Bilgisayar Büro Programları	52	0	50	
200333181016 - Esra ÇELEBİ	Bilgisayar Teknolojisi ve Programlama - Veri Tabanı Yönetim Sistemleri - I	80	0	97	
200333181044 - Meliha ŞUMNULU	Bilgisayar Teknolojisi ve Programlama - Veri Tabanı Yönetim Sistemleri - I	90	0	49	

Şekil 4.11: Sınav Sonuçları Sayfası

#### 4.5.7. Yönetim

Ana menüdeki yönetim bağlantısı okul yönetiminin görüntülenebileceği sayfanın açılmasını sağlar. Sicil numarası, ünvanı, adı, soyadı, görevi bilgileri ve otomatik sıra numarası bu sayfada görüntülenecek bilgilerdendir. (Şekil 4.12)

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

**Menü**  
[Programlar](#)  
[Dersler](#)  
[Öğrenciler](#)  
[Haftalık Ders Programı](#)  
[Sınav Takvimi](#)  
[Sınav Sonuçları](#)  
[Yönetim](#)  
[Personel](#)  
[Yurtlar](#)  
[Duyurular](#)  
[SSS](#)  
[Çoklu Veri Girişi](#)  
  
**Statik Sayfalar**  
[Okulumuz Hakkında](#)  
[İletişim Bilgileri](#)  
[Tavşanlı Hakkında](#)  
[Akademik Takvim](#)

**Yönetim**  
[Ekle](#) | [Listele](#) |  

Sıra No	Sicil No	Ünvanı	Adı	Soyadı	Görevi		
1	DA0524	Öğr.Gr.	M.Çağatay	BALI	Müdür Yardımcısı	<a href="#">Değiştir</a>	<a href="#">Sil</a>
2	DA0968	Öğr.Gr.	Bülent	CIRIK	Müdür Yardımcısı	<a href="#">Değiştir</a>	<a href="#">Sil</a>
3	DA1029	Prof.Dr.	Biröl	ELEVLİ	Müdür	<a href="#">Değiştir</a>	<a href="#">Sil</a>

Liste sonu...

Şekil 4.12: Yönetim Sayfası

Şekil 4.12’de gösterilen yönetim sayfasındaki seçili yöneticinin yanında tıklayacağımız değiştir bağlantısı ilişkili yöneticinin bilgilerinin değiştirilebileceği sayfanın açılmasını, sil ise kayıtlı yöneticinin listeden silinmesini sağlayacaktır.

Ekle’ye tıkladığımızda Şekil 4.13’deki ekran ile sicil numarası, ünvanı, adı, soyadı, görevi ve detaylı bilginin girilebileceği yeni yönetici ekleme sayfası açılır. Bilgilerin girilmesinin ardından kaydet butonuyla bilgileri girilen yönetici listeye dahil edilir.

## Menü

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)

### Statik Sayfalar

- [Okulumuz Hakkında](#)
- [İletişim Bilgileri](#)
- [Tavşanlı Hakkında](#)
- [Akademik Takvim](#)

## Yönetim

[Ekle](#) | [Listele](#) |

### Yeni Yönetici Ekleme

Sicil No

Ünvanı

Adı

Soyadı

Görevi

Detaylı Bilgi

Şekil 4.13: Yeni Yönetici Ekleme Sayfası

#### 4.5.8. Personel

Ana menüdeki personel bağlantısı okulda görevli personelin bilgilerinin düzenlenebileceği sayfanın açılmasını sağlar. Bu sayfada personelin sicil numarası, ünvanı, adı, soyadı ve görevi gösterilmektedir. Her bir personelin yanında bulunan değiştir butonu ilişkili personelin bilgilerinin değiştirilmesini sağlayan ekranın görüntülenmesini, sil ise personelin listeden silinmesini sağlar. (Şekil 4.14)

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

## Menü

[Programlar](#)  
[Dersler](#)  
[Öğrenciler](#)  
[Haftalık Ders Programı](#)  
[Sınav Takvimi](#)  
[Sınav Sonuçları](#)  
[Yönetim](#)  
[Personel](#)  
[Yurtlar](#)  
[Duyurular](#)  
[SSS](#)  
[Çoklu Veri Girişi](#)

Statik Sayfalar  
[Okulumuz Hakkında](#)  
[İletişim Bilgileri](#)  
[Tavşanlı Hakkında](#)  
[Akademik Takvim](#)

## Personel

| [Ekle](#) | [Listele](#) |

Sıra No	Sicil No	Ünvanı	Adı	Soyadı	Görevi		
1	AB0218	Öğr.Grv.	Mehmet	ÖZKAYA		<a href="#">Değiştir</a>	<a href="#">Sil</a>
2	DA0275	Öğr.Grv.	Nedret	KORKMAZ		<a href="#">Değiştir</a>	<a href="#">Sil</a>
3	DA0444	Öğr.Grv.	Eyüp	IRGAT		<a href="#">Değiştir</a>	<a href="#">Sil</a>
4	DA0524	Öğr.Grv.	M.Çağatay	BALI	Müdür Yardımcısı	<a href="#">Değiştir</a>	<a href="#">Sil</a>
5	DA0553	Öğr.Grv.	Yavuz	SOYKAN		<a href="#">Değiştir</a>	<a href="#">Sil</a>
6	DA0568	Öğr.Grv.	Mustafa	KAYIK		<a href="#">Değiştir</a>	<a href="#">Sil</a>
7	DA0640	Öğr.Grv.	Sibel	MANİSALI		<a href="#">Değiştir</a>	<a href="#">Sil</a>
8	DA0730	Öğr.Grv.	Demet	GÜLSEREN		<a href="#">Değiştir</a>	<a href="#">Sil</a>
9	DA0767	Öğr.Grv.	Emre	GEDİK		<a href="#">Değiştir</a>	<a href="#">Sil</a>
10	DA0905	Öğr.Grv.	Serkan	AKAN		<a href="#">Değiştir</a>	<a href="#">Sil</a>
11	DA0968	Öğr.Grv.	Bülent	CIRIK	Müdür Yardımcısı	<a href="#">Değiştir</a>	<a href="#">Sil</a>

Şekil 4.14: Personel Sayfası

Şekil 4.14'deki ekle bağlantısına tıkladığımızda ise yeni personel ekleme sayfası (Şekil 4.15) görüntülenir. Bu sayfada eklenecek personelin sicil numarası, ünvanı, adı, soyadı, görevi ve görüntülenmesi istenen detaylı bilginin girilebileceği alanlar bulunur. Bilgiler girildikten sonra listele ile eklenen son personel ile birlikte kayıtlı tüm personel bilgileri görüntülenir.

**Menü**

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)

**Statik Sayfalar**

- [Okulumuz Hakkında](#)
- [İletişim Bilgileri](#)
- [Tavsaneli Hakkında](#)
- [Akademik Takvim](#)

**Personel**

| [Ekle](#) | [Listele](#) |

### Yeni Personel Ekleme

Sicil No

Ünvanı

Adı

Soyadı

Görevi

Detaylı Bilgi

Şekil 4.15: Yeni Personel Ekleme Sayfası

#### 4.5.9. Yurtlar

Ana menüdeki yurtlar seçeneği ile öğrencilerin kalabileceği civardaki yurt bilgilerinin görüntülenmesi ve bilgilerinin değiştirilmesi sağlanır. Yurtlar bağlantısına tıklandığında Şekil 4.16’da gösterilen ekran görüntüsü açılır. Açılan sayfada eklenen yurtların kodu, adı ve detay bilgilerine ulaşılabilir. Her bir yurdun yanında bulunan değiştir bağlantısı ile ilişkili yurdun bilgilerinin değiştirilmesi, sil ile ise kayıtlı yurdun silinmesi işlemleri gerçekleştirilir.

Şekil 4.16’daki sayfada bulunan ekle bağlantısı ise bizi yeni yurt ekleme sayfasına ulaştırır.(Şekil 4.17) Bu sayfada otomatik olarak verilen yurt kodu, eklenecek yurdun adı ve yurtla ilgili açıklayıcı bilgilerin girilmesinin ardından kaydet butonuyla yeni yurdun eklenmesi sağlanır. Listele butonu son eklenen yurt dahil tüm yurtların listelenmesini sağlar.



T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

### Menü

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)

Statik Sayfalar

- [Okulumuz Hakkında](#)
- [İletişim Bilgileri](#)
- [Tavşanlı Hakkında](#)
- [Akademik Takvim](#)

### Yurtlar

[Ekle](#) | [Listele](#) |

Sıra No	Kodu	Adı		
1	7	Bimpaş	<a href="#">Değiştir</a>	<a href="#">Sil</a>
2	8	Taşçılar kız yurdu	<a href="#">Değiştir</a>	<a href="#">Sil</a>

Liste sonu...

Şekil 4.16: Yurtlar Sayfası

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

### Menü

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)

Statik Sayfalar

- [Okulumuz Hakkında](#)
- [İletişim Bilgileri](#)
- [Tavşanlı Hakkında](#)
- [Akademik Takvim](#)

### Yurtlar

[Ekle](#) | [Listele](#) |

#### Yeni Yurt Ekleme

Yurt Kodu

Yurt Adı

Yurt Hakkında Detaylı Bilgi

Şekil 4.17: Yeni Yurt Ekleme Sayfası

#### 4.5.10. Duyurular

Ana menüdeki duyurular bağlantısına tıkladığımızda Şekil 4.18'deki ekran görüntüsü karşımıza çıkar. Duyurular öğrencilerin bilgilendirilmesi amacıyla okul tarafından girilen bilgileri içerir. Her bir duyurunun otomatik olarak atanan sıra numarası, kodu ve konusu bulunur.

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

### Menü

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)

Statik Sayfalar

- [Okulumuz Hakkında](#)
- [İletişim Bilgileri](#)
- [Tavşanlı Hakkında](#)
- [Akademik Takvim](#)

### Duyurular

| [Ekle](#) | [Listele](#) |

Sıra No	Kodu	Konu		
1	5	Mezuniyet	<a href="#">Değiştir</a>	<a href="#">Sil</a>

Liste sonu...

Şekil 4.18: Duyurular Sayfası

İlgili konunun yanında bulunan değiştir bağlantısı konuyla ilgili açıklamanın değiştirilmesini sağlarken, sil bağlantısı ilgili duyurunun listeden çıkarılmasını sağlayacaktır.

Yeni bir duyuru eklenmek istendiğinde ekle bağlantısına tıklanır ve Şekil 4.19'daki yeni duyuru ekleme sayfasının açılmasını sağlar.

Konunun ve duyuru metninin girilmesinin ardından kaydet butonuna basarak duyuru kaydedilmiş olur.

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

### Menü

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)

Statik Sayfalar

- [Okulumuz Hakkında](#)
- [İletişim Bilgileri](#)
- [Tavşanlı Hakkında](#)
- [Akademik Takvim](#)

### Duyurular

[Ekle](#) | [Listele](#) |

#### Yeni Duyuru Ekleme

Duyuru Kodu

Konu

Duyuru Metni

Şekil 4.19: Yeni Duyuru Ekleme Sayfası

#### 4.5.11. Sık sorulan sorular

Ana menüdeki SSS bizi sıkça sorulan sorular sayfasına yönlendirir. Bu sayfa öğrencilerin sıklıkla karşılaştıkları ve öğrenmek istedikleri soruların cevaplarının yazılacağı alandır.

Her bir sorunun otomatik olarak atanan sıra numarası, kodu ve konusu bulunur. Her bir sorunun yanında bulunan deęiřtir baęlantısı ilgili soruyla ilgili açıklamalarda deęiřiklik yapılmasını saęlar.

řekil 4.19'daki ekran görüntüsünde ekle'ye tıkladığımızda ise yeni bir açıklama eklenmesini saęlayan řekil 4.21'deki sayfa görüntülenir.

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

### Menü

---

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)

Statik Sayfalar

- [Okulumuz Hakkında](#)
- [İletişim Bilgileri](#)
- [Tavşanlı Hakkında](#)
- [Akademik Takvim](#)

### Sık Sorulan Sorular

---

| [Ekle](#) | [Listele](#) |

Sıra No	Kodu	Konu		
1	6	Kayıt için gerekli belgeler	<a href="#">Deęiřtir</a>	<a href="#">Sil</a>
2	4	Yaz okulu	<a href="#">Deęiřtir</a>	<a href="#">Sil</a>

Liste sonu...

řekil 4.20: Sık Sorulan Sorular Sayfası

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

### Menü

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)

Statik Sayfalar

- [Okulumuz Hakkında](#)
- [İletişim Bilgileri](#)
- [Tavşanlı Hakkında](#)
- [Akademik Takvim](#)

### Sık Sorulan Sorular

| [Ekle](#) | [Listele](#) |

### Yeni SSS Ekleme

SSS Kodu      Yeni

Konu     

SSS Cevap Metni

Şekil 4.21: Yeni SSS Ekleme Sayfası

Konu başlıklı metin kutusu açıklanması istenen konunun başlığını, SSS cevap metni ise soruya ait cevabın yazılacağı alanı ifade eder. Soru ve cevabı yazıldıktan sonra kaydet butonuyla yeni sss'nin eklenmesi sağlanmış olur. Listele butonununa basılarak tüm sss'ler görüntülenebilir.

#### 4.5.12. Çoklu veri girişi

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

**Menü**  
[Programlar](#)  
[Dersler](#)  
[Öğrenciler](#)  
[Haftalık Ders Programı](#)  
[Sınav Takvimi](#)  
[Sınav Sonuçları](#)  
[Yönetim](#)  
[Personel](#)  
[Yurtlar](#)  
[Duyurular](#)  
[SSS](#)  
[Çoklu Veri Girişi](#)  
  
**Statik Sayfalar**  
[Okulumuz Hakkında](#)  
[İletişim Bilgileri](#)  
[Tavşanlı Hakkında](#)  
[Akademik Takvim](#)

### Çoklu Veri Girişi

---

[Programlar](#) | [Dersler](#) | [Öğrenciler](#) | [Haftalık Ders Programı](#) | [Sınav Takvimi](#) | [Sınav Sonuçları](#) | [Yönetim](#) | [Personel](#) | [Yurtlar](#) | [Duyurular](#) | [SSS](#) |

---

#### Programlar

**CSV Formatı:**  
programkodu;programadi;detaybilgi

Aşağıdaki alana yukarıdaki formatta verileri giriniz. Kayıt bittiğinde enter'a basarak sonrakine geçiniz.

Şekil 4.22: Çoklu Veri Girişi Sayfası

Ana menüdeki çoklu veri girişi ise Şekil 4.22’de gösterilen ekran görüntüsünü karşımıza getirir. Daha önceki sayfalarda gösterilen ana menü seçeneklerindeki veri girişi her bir kayıt için tek tek gerçekleştirilmektedir. Bu alanda CSV formatı kullanılarak verilerin toplu olarak girilmesi sağlanır. Üstte bağlantılarla belirtilen programlar, dersler, öğrenciler, haftalık ders programı, sınav takvimi, sınav sonuçları, yönetim, personel, yurtlar, duyurular ve sss’a ait bağlantılara tıklanıldığında seçili alana toplu verilerin girilebilmesi için gerekli olan csv formatı görüntülenir. Şekil 4.22’deki örnekte programlara ait kayıtların girilebilmesi için “programkodu; program adı; detaybilgi” alanlarının bu formatta ve her bir satıra bir kayıt olacak şekilde girilmesi gerekmektedir.

Ayrıca bu alana girilecek verileri diğer programlardan csv formatına uyacak şekilde kaydederek buraya kopyalamak kaydıyla da gerçekleştirebiliriz. Böylelikle bir adımda birden çok verinin girişi kolaylıkla gerçekleştirilmiş olur.

#### 4.5.13. Statik sayfalar

Ana menünün alt kısmında yer alan statik sayfalar ise sabit verilerin tutulduğu alanlara işaret eder ve buradaki bağlantılara tıklayarak ilgili başlıklara ait açıklamaları gösteren sayfalara ulaşılır. Bunlardan okulumuz hakkında Şekil 4.23, iletişim bilgileri Şekil 4.24, tavşanlı hakkında Şekil 4.25 ve akademik takvim Şekil 4.26’da gösterilmektedir.

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

### Menü

- [Programlar](#)
- [Dersler](#)
- [Öğrenciler](#)
- [Haftalık Ders Programı](#)
- [Sınav Takvimi](#)
- [Sınav Sonuçları](#)
- [Yönetim](#)
- [Personel](#)
- [Yurtlar](#)
- [Duyurular](#)
- [SSS](#)
- [Çoklu Veri Girişi](#)
- Statik Sayfalar
  - [Okulumuz Hakkında](#)
  - [İletişim Bilgileri](#)
  - [Tavşanlı Hakkında](#)
  - [Akademik Takvim](#)

### Statik Sayfalar

#### Okulumuz Hakkında

Sayfa Kodu okulumuzhakkında

Sayfa Başlığı

Sayfa Metni

Şekil 4.23: Okulumuz Hakkında Sayfası

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

**Menü**

---

[Programlar](#)  
[Dersler](#)  
[Öğrenciler](#)  
[Haftalık Ders Programı](#)  
[Sınav Takvimi](#)  
[Sınav Sonuçları](#)  
[Yönetim](#)  
[Personel](#)  
[Yurtlar](#)  
[Duyurular](#)  
[SSS](#)  
[Çoklu Veri Girişi](#)

Statik Sayfalar

[Okulumuz Hakkında](#)  
[İletişim Bilgileri](#)  
[Tavşanlı Hakkında](#)  
[Akademik Takvim](#)

**Statik Sayfalar**

---

**İletişim Bilgileri**

Sayfa Kodu: iletisimbilgileri

Sayfa Başlığı:

Sayfa Metni: 

Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Tavşanlı/KÜTAHYA  
  
Web: http://tavmyo.dumlupinar.edu.tr  
  
Telefonlar: (0274) 614 8671 - 614 8672  
  
Faks: (0274) 614 8673

Şekil 4.24: İletişim Bilgileri Sayfası

T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

**Menü**

---

[Programlar](#)  
[Dersler](#)  
[Öğrenciler](#)  
[Haftalık Ders Programı](#)  
[Sınav Takvimi](#)  
[Sınav Sonuçları](#)  
[Yönetim](#)  
[Personel](#)  
[Yurtlar](#)  
[Duyurular](#)  
[SSS](#)  
[Çoklu Veri Girişi](#)

Statik Sayfalar

[Okulumuz Hakkında](#)  
[İletişim Bilgileri](#)  
[Tavşanlı Hakkında](#)  
[Akademik Takvim](#)

**Statik Sayfalar**

---

**Tavşanlı Hakkında**

Sayfa Kodu: tavsanlihakkında

Sayfa Başlığı:

Sayfa Metni: 

Tavşanlı tarihi, Anadolu'nun tarihi kadar uzun bir geçmişe dayanmaktadır. İlçemiz tarihinin iz ve kalıntılarına İlçemizin birçok yöresinde rastlamak mümkündür. 1071 Malazgirt Meydan Savaşından sonra Anadolu'nun fethi başlamıştır. İlçemiz de, 1078 yılında Süleyman Şah Komutasındaki birliklerce, Türk topraklarına katılmıştır. 1097 yılındaki Birinci Haçlı seferleri ile İlçemiz

Şekil 4.25: Tavşanlı Hakkında Sayfası



T.C.  
Dumlupınar Üniversitesi  
Tavşanlı Meslek Yüksekokulu  
Mobil Öğrenci Kontrol Paneli

**Menü**  
[Programlar](#)  
[Dersler](#)  
[Öğrenciler](#)  
[Haftalık Ders Programı](#)  
[Sınav Takvimi](#)  
[Sınav Sonuçları](#)  
[Yönetim](#)  
[Personel](#)  
[Yurtlar](#)  
[Duyurular](#)  
[SSS](#)  
[Çoklu Veri Girişi](#)  
  
**Statik Sayfalar**  
[Okulumuz Hakkında](#)  
[İletişim Bilgileri](#)  
[Tavşanlı Hakkında](#)  
[Akademik Takvim](#)

**Statik Sayfalar**  
  
**Akademik Takvim**  
  
Sayfa Kodu akademiktakvim  
Sayfa Başlığı Akademik Takvim  
Sayfa Metni  
2006-2007 AKADEMİK YILI  
AKADEMİK TAKVİMİ  
Güz Yarıyılı  
Yeni Kayıtlar 05-11 Eylül 2006  
Kayıt Yenilemeler 12-15 Eylül 2006  
Derslerin Başlaması 18 Eylül 2006  
İngilizce Seviye Tespit Sınavı 23 Eylül 2006  
Ekle-Sil Son Gün 29 Eylül 2006  
Tek Ders Sınavları (25. ve Geçici 4. Madde) 8

Şekil 4.26: Akademik Takvim Sayfası

Statik sayfaların her birinde bulunan sayfa başlığı ve sayfa metni bilgileri girilerek kaydet butonuyla öğrencilerin ulaşabileceği şekle dönüşür.

#### 4.6. Wap Sayfası

Kontrol panel bizim verileri girmemizi, gerekli değişiklikleri yapmamızı ve gerektiğinde silmemizi sağlayan bir arabirimdir. Bu alana yalnızca kullanıcı adı ve şifresi daha önceden tanımlanmış kullanıcılar erişebilir ve veri girişinde bulunabilirler. Öğrencilerin bu bilgilere ulaşmasını sağlamak için ise bir wap sitesi hazırlanmış ve bunun için Macromedia Dreamweaver programı kullanılmıştır.

Cep telefonundan okulumuz bilgilerinin tutulduđu wap sitesi olan <http://wap.tavsanli.net> adresine girildiğinde karřımıza kontrol panel aracılıđıyla girdiđimiz veriler çıkar.

Bu bilgilerin cep telefonundan nasıl görüntüleneceđini görmek amacıyla ise bilgisayar ortamında bir emülatör kullanabiliriz. Biz bu örnekte Microsoft Mobile Explorer ile sayfalara eriştik.

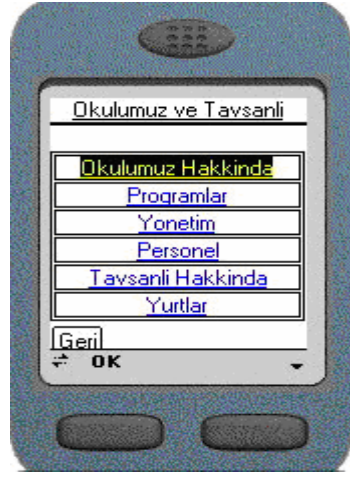
İlk aşamada emülatörde URL kısmına <http://wap.tavsanli.net> yazarak ana sayfanın görüntülenmesini sağlarız. (Şekil 4.27)



Şekil 4.27: Wap Anasayfa

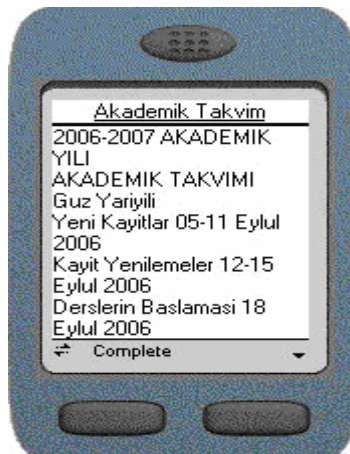
Ana sayfada öğrenciler ana menü ile karşılaşılır. Bu sayfada yön tuşlarını kullanarak menü seçenekleri arasında dolaşabilir ve istenen menü seçeneğini seçmek için OK tuşuna basarak ilgili menüye giriş yapabilirler.

İlk seçenek olan okulumuz ve tavşanlı seçildiğinde Şekil 4.28'deki ekran görüntüsü karşımıza gelir. Alt menü seçenekleri yine bağlantılarla belirtilmiştir. Bunlar okuldaki programlar, yönetim, personel, Tavşanlı hakkında ve yurtlar seçenekleridir.



Şekil 4.28: Okulumuz ve Tavşanlı Sayfası

Şekil 4.27'deki ana menüdeki akademik takvim seçeneğine tıklanıldığında ekran görüntüsü Şekil 4.29'dakine dönüşür.



Şekil 4.29: Akademik Takvim Sayfası

Haftalık ders programı menü seçeneğinin seçilmesi durumunda ise ilk aşamada öğrencinin görmek istediği programa ait ders programını seçebileceği açılır liste kutusu ekrana gelir. (Şekil 4.30)



Şekil 4.30: Haftalık Ders Programı Sayfası

Bu listede istenen program seçildikten sonra göster butonu seçilerek seçilen programa ilişkin ders programı görüntülenir. (Şekil 4.31)



Şekil 4.31: Haftalık Ders Programı Sayfası

Ana menüdeki sınav sonuçları ise öğrencinin numarasını ve şifresini yazması sonucunda gösterilir.(Şekil 4.32) Yazılan şifre ekranda \*'lar şeklinde gösterilir.(Şekil 4.33)



Şekil 4.32: Sınav Sonuçları



Şekil 4.33: Sınav Sonuçları Şifre Girişi

Örnekte numarası ve şifresi girilen öğrencinin göster butonuna basılarak sisteme girilmiş olan notları Şekil 4.34'de olduğu gibi gösterilir.



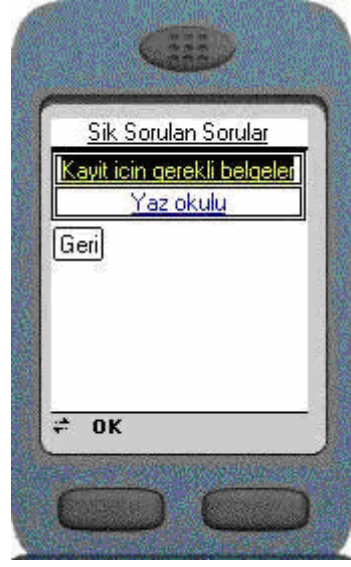
**Şekil 4.34: Sınav Sonuçlarının Gösterilmesi**

Öğrenci duyurulara ulaşmak için ana menüdeki duyuruları seçer. Ekran görüntüsü Şekil 4.35'e dönüşür.



**Şekil 4.35: Duyurular**

Öncelikle duyurulara ait ana başlıklarla karşılaşır. Öğrenmek istediği duyuruya ait konu başlığını seçmesi durumunda ise konuyla ilgili detay bilgileri görüntülenir.



Şekil 4.36: Sık Sorulan Sorular



Şekil 4.37: Sık Sorulan Sorular Detayı

Şekil 4.36 ana menüdeki SSS alanının seçilmesi sonucunun ekran görüntüsünü göstermektedir. Seçenekler halinde gösterilen başlıklardan birinin seçilmesi durumunda ise Şekil 4.37’deki SSS detayı görüntülenir.

Son olarak öğrencinin okula ait iletişim bilgilerini görebilmesi için ana menüdeki iletişimi seçmesi gerekir. Adres, web adresi ve telefona ulaşabileceği ekran görüntüsü Şekil 6.38’de verilmiştir.



**Şekil 4.38: İletişim Bilgileri**



## 5. SONUÇ

BT (Bilgi Teknolojileri) ve TK (Telekomünikasyon) teknolojileri globalleşmedeki en önemli adımlardan biridir. Dünyada internet ve cep telefonu uygulamalarına verilen önem gittikçe artmakta, kullanıcı sayıları büyük bir hızla katlanmaktadır. Bugünkü duruma bakarak gelecek yıllarda her iki kişiden birinin internet kullanıcısı olacağı, bunun yanında telli telefonların talep görmeyeceği açıkça gözükmemektedir. Cep telefonu sayısındaki artış da bilgi teknolojilerindeki ilerlemelerle birlikte küçük boyutlu, iletim hızları yüksek, kesintisiz iletişim yaratmaları nedeniyle hızla artmaktadır. Günümüzde hayatımızı kolaylaştıracağı sinyallerini vermeye başlayan ileri teknolojilerle hazırlanan bilgisayar cep telefonu karışımı sistemler yerini almaya başlamıştır.

Yapılan bu tez çalışmasında yükseköğretim kurumlarında okuyan öğrencilerin okul ile ilgili bilgilere mobil iletişim aracılığıyla ulaşması sağlanmıştır. Yükseköğretim kurumlarının öğrencilere duyurmak istediği bilgiler öğrenci bilgi sistemi adı altında tasarlanmış ve bilgilerin kurum tarafından girilebilmesi için bir kontrol panel, öğrencilerin bu bilgilere erişmesi için ise bir wap sayfası hazırlanmıştır. Ders programı, sınav sonuçları, duyurular, iletişim bilgileri gibi dinamik ve statik alanlardan oluşan bilgi sistemine erişmek için öğrencilerin mobil bir cihaza ve internet bağlantısına sahip olması yeterli kılınmıştır.

Sonuç olarak, hangi meslekten ve hangi yaş grubundan olursa olsun insanların cep telefonu ve uygulamalarına karşı ilgisi ve eğilimi her geçen gün artmaktadır. Bu uygulamanın amacı öğrencilerin kablosuz ortamlarda ve hareketli durumda kendilerine gerekli bilgilere ulaşabilmesini sağlamaktır. Yapılan bu çalışma da sabit olma zorunluluğu olmaksızın istenilen noktadan, istenilen zamanda bilgi sistemlerine ulaşımı sağlamıştır.

## KAYNAKLAR

- [1] Ertekin, Ö. , Özbay, A. , Mobil&Wap İletişim Teknolojileri, Hayat yayınları, Ankara, (Kasım 2000)
- [2] Erdoğan, K. , Mobil İletişim Teknolojisindeki Gelişmeler, yayımlanmamış Ankara Üniversitesi Yüksek Lisans Tezi (2001)
- [3] Yanık, D. Üçüncü Nesil (3G) Mobil Telekomünikasyon Sistemleri ve 3G Lisanslarının Verilmesi Konusunda Dünyadaki Uygulamalar ile Türkiye Analizi, Telekomünikasyon Kurumu (Nisan 2001)
- [4] <http://www.crea-world.com/features/102205-27082001.htm> (04.05.2005)
- [5] Darıcı, A. , 3.Nesil Mobil Haberleşme Sistemleri, Telekomünikasyon kurumu, (Aralık 2002)
- [6] International Engineering Resources-web proforum, ([www.iec.org/tutorials/umts/topic01.html](http://www.iec.org/tutorials/umts/topic01.html)),sayfa2 (25.03.2004)
- [7] Candan, M. 3. Nesil Mobil Haberleşme Sistemleri için Türkiye’de Uygulanacak Frekans Bandı, Lisans, Servisler, Uygulamalar ve Ülkemizdeki Durumu, Ankara (2002)
- [8] 8. Beş yıllık kalkınma planı, haberleşme özel ihtisas komisyonu raporu, <http://ekutup.dpt.gov.tr/bilisim/oik581.pdf> (20.01.2005)
- [9] Başaran, F. , Neo-Liberal Telekomünikasyon Politikalarının Deney Alanı Cep Telefonları, <http://dergi.emo.org.tr/index.php?sayi=416> (20.04.2005)
- [10] [http://www.gsacom.com/gsm\\_3g/market\\_update.php4](http://www.gsacom.com/gsm_3g/market_update.php4) (21.05.2006)
- [11] Özdemir, E. , Dinçer, H. , Burdurlu, H. , Üçüncü Nesil Gezgin Haberleşme Sistemi, Elektrik Mühendisliği Dergisi, cilt:41, sayı:416, s.38 - 44, (Kasım 2002)
- [12] Efe, E. , <http://members.tripod.com/enginefe/linkler/gprs.htm> (10.01.2006)
- [13] Manas,0.Kablosuz Bilgisayar Ağları (Wireless Network), Haziran 2002
- [14] Anonim, GSM Architecture, [http://www.m-indya.com/mwap/gsm/gsm\\_architecture.htm](http://www.m-indya.com/mwap/gsm/gsm_architecture.htm) (03.06.2005)
- [15] Atasoy, K. , Mobil Sistemde Frekans Ayırma Tipleri, <http://antrak.org.tr/gazete/042004/yazi12.html> (05.05.2005)

- [16] Orlu, Ş. , GSM 900 ve GSM 1800, <http://www.geocities.com/alicelebioglu/yy/gsm.htm> (15.04.2006)
- [17] Manas, O. , Gsm, Hareket için Küresel Sistemler, (Mayıs 2002)
- [18] Çölkesen, R. , Örencik, B. , Bilgisayar Haberleşmesi ve Ağ Teknolojileri, Papatya yayıncılık, (1999)
- [19] Seçkiner, M. ,WAN Teknolojileri I, <http://teknohaber.net/makale.php?İd=30217> (30.04.2005)
- [20] Büber, Ş. , Kablosuz Ağlar için İletişim Protokolleri 2-<http://mutasyon.net/makaleoku.asp?id=50> (30.04.2005)
- [21] Atasoy, K. , Mobil Sistemde Frekans Ayırma Tipleri, <http://antrak.org.tr/gazete/042004/yazi12.html> (05.05.2005)
- [22] [http://www1.ericsson.com.tr/mobilityworld/articles/features/102802-2708200\\_1.htm](http://www1.ericsson.com.tr/mobilityworld/articles/features/102802-2708200_1.htm) (04.05.2005)
- [23] <http://oguz.ieee.metu.edu.tr/docs/WAPyazi.pdf> (17.05.2005)
- [24] Bayız, Y. ve Eken, S. K. , GSM'in geleceği: UMTS, İletişim Teknolojileri Bildiriler Kitabı, (2001)
- [25] ITU-R:IMT - 2000, 3rd Generation Mobil Services and Applications ,International Telecommunication Union, <http://www.itu.int/imt/whatis/3rdgen/index.html> (05.06.2006)
- [26] Fransa Telekomünikasyon Regülasyon Otoritesi'nin (ART) Şubat 1999 tarih 'Public Consultation on the Introduction of UMTS in France' isimli raporu, s.6
- [27] Selian, A. , 3G Mobile Licensing Policy: From GSM to IMT - 2000-A Comparative Analysis, GSM Case-Study, ITU, 2001, www.itu.int/3g, s20.
- [28] UMTS Forum, Report No.1:A Regulatory Framework for UMTS, S6
- [29] Office the Director of Telecommunications Regulation (İrlanda), Doc. No. ODTR00/52, s. 6
- [30] Hanniaken, M. , Hamalainen, D. , Niemi, M. , Saarinen, J, Trends in personel wireless data communications, Computer Communications, (2002)
- [31] Holma, H. And Toksala, A, W-CDMA for UMTS, Jhon Wiley and Sons, New York, (2000)

- [32] 3G Dünya Tecrübeleri, UMTS Ulusal Koordinasyon Kurulu Alt Çalışma Grubu - 1, (Aralık 2002)
- [33] Sung, L. , Sweeney, D. , Fossa, C. , D'Souza, M. , 3G Migration in North America-Final Paper, Virginia Tech, (2001)
- [34] [www.anadolubil.edu.tr/biltek/ybs/calisma\\_sorulari.doc](http://www.anadolubil.edu.tr/biltek/ybs/calisma_sorulari.doc), 2006
- [35] Çaycı, Ö. , PHP ve MySQL, Seçkin yayıncılık, 2003
- [36] Doğan, H. , PHP Dokumanları, 2006, <http://www.php.web.tr/index.php?page=1-2>
- [37] Karaduman, C. , Mysql, PhpMyAdmin, PHP, Dinamik Web Sayfası, PHP NukeKullanımı, 2005, [http://www.inonu.edu.tr/dokuman/seminer\\_son.ppt](http://www.inonu.edu.tr/dokuman/seminer_son.ppt)
- [38] <http://www.mjturkiye.net/forum/Mysql-Nedir-Neden-Kullaniriz-Ozellikleri-Nelerdir-t30040.html>, 2006
- [39] Ütebay, G. , PHP ve MySQL, 2006, [www1.gantep.edu.tr/~sonercan/docs/php1.ppt](http://www1.gantep.edu.tr/~sonercan/docs/php1.ppt)

## Ek-1 Kontrol Panel Modülünün Program Kodları

```
index
<?
session_start();
include "config.inc.php";
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-9" />
<title>Mobil Öğrenci Kontrol Paneli</title>
<link href="mobilogrenci.css" rel="stylesheet" type="text/css" />
</head>
<body leftmargin="0" topmargin="0" bottommargin="0" rightmargin="0">
<?
if (!( @$_SESSION['kullaniciok']==1))
{
if (!isset($_POST['kullanici']))
{
//formgoster
echo "<table align=center>
<tr><td align=center><h3><br><br>T.C.<br>Dumlupınar Üniversitesi<br>
Tavşanlı Meslek Yüksekokulu<br>Mobil Öğrenci Kontrol Paneli<br>Kullanıcı Girişi</h3><br>
<form method='post'>
Kullanıcı Adı<br><input type='text' name='kullanici'><br><br>
Şifre<br><input type='password' name='sifre'><br><br>
<input type='submit' name='submit' value='Giriş'>
</form></td></tr></table>";
}
else
{
$kullanici=$_POST['kullanici']; $sifre=$_POST['sifre'];
if (($cpuser==$kullanici) and ($cpass==$sifre))
{
//şifre doğruysa
$_SESSION['kullaniciok']=1;
echo "<script language='javascript'> document.location='/kontrolpanel/' </script>";
}
else
{
//şifre yanlışsa
$kullaniciok=0;
session_unregister('kullaniciok');
echo "<table align=center>
<tr><td align=center><h3><br><br>T.C.<br>Dumlupınar Üniversitesi<br>
Tavşanlı Meslek Yüksekokulu<br>Mobil Öğrenci Kontrol Paneli<br>Kullanıcı Girişi</h3><br>
Kullanıcı adı ve/veya şifre yanlış girildi!<br>
Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar giriniz.
</td></tr></table>";
}
}
}
else
{
?>
<table width="100%" border="1" cellpadding="10" cellspacing="5">
<tr>
<td colspan="2" align="center" bgcolor="#FFCC66"><h5><br />
T.C. <br />
Dumlupınar Üniversitesi<br />
Tavşanlı Meslek Yüksekokulu<br />
```

```

Mobil Öğrenci Kontrol Paneli </h3></td>
</tr>
<tr>
<td align="left" valign="top" bgcolor="#CCFFCC"><h3>Menü</h3>
<hr>
<table width="100%" border="0" cellpadding="0" cellspacing="5">
<tr>
<td><a href="?sec=programlar">Programlar</a></td>
</tr>
<tr>
<td><a href="?sec=dersler">Dersler</a></td>
</tr>
<tr>
<td><a href="?sec=ogrenciler">Öğrenciler</a></td>
</tr>
<tr>
<td><a href="?sec=haftalikdersprogrami">Haftalık Ders Programı </a></td>
</tr>
<tr>
<td><a href="?sec=sinavtakvimi">Sınav Takvimi </a></td>
</tr>
<tr>
<td><a href="?sec=sinavsonuclari">Sınav Sonuçları </a></td>
</tr>
<tr>
<td><a href="?sec=yonetim">Yönetim</a></td>
</tr>
<tr>
<td><a href="?sec=personel">Personel</a></td>
</tr>
<tr>
<td><a href="?sec=yurtlar">Yurtlar</a></td>
</tr>
<tr>
<td><a href="?sec=duyurular">Duyurular</a></td>
</tr>
<tr>
<td><a href="?sec=sss">SSS</a></td>
</tr>
<tr>
<td><a href="?sec=cokluverigirisi">Çoklu Veri Girişi</a> </td>
</tr>
<tr>
<td>&nbsp;</td>
</tr>
<tr>
<td>Statik Sayfalar </td>
</tr>
<tr>
<td>&nbsp;&nbsp;&nbsp;<a href="?sec=ss&sayfa=okulumuzhakkinda">Okulumuz
Hakkında</a> </td>
</tr>
<tr>
<td>&nbsp;&nbsp;&nbsp;<a href="?sec=ss&sayfa=iletisimbilgileri">İletişim Bilgileri</a></td>
</tr>
<tr>
<td>&nbsp;&nbsp;&nbsp;<a href="?sec=ss&sayfa=tavsanlihakkinda">Tavsanlı Hakkında</a>
</td>
</tr>
<tr>
<td>&nbsp;&nbsp;&nbsp;<a href="?sec=ss&sayfa=akademiktakvim">Akademik Takvim</a>
</td>
</tr>
</table>

```

```

</td>
<td width="78%" align="left" valign="top" bgcolor="#ECE9D8">
<!--içerik -->
<?
    if (!isset($_GET["sec"])) $sec="programlar"; else $sec=$_GET["sec"];
        switch ($sec) {
            case "programlar" : include "programlar.php"; break;
            case "yurtlar" : include "yurtlar.php"; break;
            case "sss" : include "sss.php"; break;
            case "duyurular" : include "duyurular.php"; break;
            case "dersler" : include "dersler.php"; break;
            case "ogrenciler" : include "ogrenciler.php"; break;
            case "yonetim" : include "yonetim.php"; break;
            case "personel" : include "personel.php"; break;
            case "ss" : include "statiksayfalar.php"; break;
            case "haftalikdersprogrami" : include "haftalikdersprogrami.php"; break;
            case "sinavtakvimi" : include "sinavtakvimi.php"; break;
            case "sinavsonuclari" : include "sinavsonuclari.php"; break;
            case "cokluverigirisi" : include "cokluverigirisi.php"; break;
        }
    ?>
<!--içerik --> </td>
</tr>
</table>
<? } ?>
</body>
</html>
Çoklu Veri Girişi
<h3>Çoklu Veri Girişi</h3>
<hr>
| <a href="?sec=cokluverigirisi&sayfa=programlar">Programlar</a> | <a
href="?sec=cokluverigirisi&sayfa=dersler">Dersler</a> | <a
href="?sec=cokluverigirisi&sayfa=ogrenciler">Öğrenciler</a> | <a
href="?sec=cokluverigirisi&sayfa=haftalikdersprogrami">Haftalık Ders Programı</a> | <a
href="?sec=cokluverigirisi&sayfa=sinavtakvimi">Sinav Takvimi</a> | <a
href="?sec=cokluverigirisi&sayfa=sinavsonuclari">Sinav Sonuçları</a> | <a
href="?sec=cokluverigirisi&sayfa=yonetim">Yönetim</a> | <a
href="?sec=cokluverigirisi&sayfa=personel">Personel</a> | <a
href="?sec=cokluverigirisi&sayfa=yurtlar">Yurtlar</a> | <a
href="?sec=cokluverigirisi&sayfa=duyurular">Duyurular</a> | <a
href="?sec=cokluverigirisi&sayfa=sss">SSS</a> |
<hr>
<?
if (!isset($_GET["sayfa"])) $sayfa="programlar"; else $sayfa=$_GET["sayfa"];
switch ($sayfa) {
    case "programlar" :
        $scsvformati="programkodu;programadı;detaybilgi";
        $salarlar="programlar` (kodu`, `adi`, `detay)`";
        $sayi=3;
        $baslik="Programlar"; break;
    case "dersler" :
        $scsvformati="derskodu;dersadı;programkodu";
        $salarlar="dersler` (kodu`, `adi`, `programkodu)`";
        $sayi=3;
        $baslik="Dersler"; break;
    case "ogrenciler" :
        $scsvformati="öğrencino;adı;soyadı;programkodu;şifre";
        $salarlar="ogrenciler` (ogrencino`, `adi`, `soyadi`, `programkodu`, `sifre)`";
        $sayi=5;
        $baslik="Öğrenciler"; break;
    case "haftalikdersprogrami" :
        $scsvformati="derskodu;başlangıçsaati(ss:dd);bitişsaati(ss:dd)";
        $salarlar="haftaliktakvim` (derskodu`, `baslangic`, `bitis)`";
        $sayi=3;

```

```

        $baslik="Haftalık Ders Programı"; break;
    case "sinavtakvimi"
        :
        $csvformati="derskodu;tarih(yyyy-aa-gg);saat(ss:dd)";
        $alanlar="sinavtakvimi`(`derskodu`,`tarih`,`saat`)";
        $sayi=3;
        $baslik="Sınav Takvimi"; break;
    case "sinavsonuclari"
        :
        $csvformati="öğrencino;derskodu;vize1;vize2;final;ortalama;harfnotu";
        $alanlar="sinavsonuclari`(`ogrencino`,`derskodu`,`vize1`,`vize2`,`final`,`ortalama`,`harfnotu`)";
        $sayi=7;
        $baslik="Sınav Sonuçları"; break;
    case "yonetim"
        :
        $csvformati="sicilno;ünvan;adi;soyadi;gorevi;detaybilgi";
        $alanlar="yonetim`(`sicilno`,`ünvan`,`adi`,`soyadi`,`gorevi`,`detay`)";
        $sayi=6;
        $baslik="Yönetim"; break;
    case "personel"
        :
        $csvformati="sicilno;ünvan;adi;soyadi;gorevi;detaybilgi";
        $alanlar="personel`(`sicilno`,`ünvan`,`adi`,`soyadi`,`gorevi`,`detay`)";
        $sayi=6;
        $baslik="Personel"; break;
    case "yurtlar"
        :
        $csvformati="adi;detaybilgi";
        $alanlar="yurtlar`(`adi`,`metin`)";
        $sayi=2;
        $baslik="Yurtlar"; break;
    case "duyurular"
        :
        $csvformati="konu;metin";
        $alanlar="duyurular`(`konu`,`metin`)";
        $sayi=2;
        $baslik="Duyurular"; break;
    case "sss"
        :
        $csvformati="konu;metin";
        $alanlar="sss`(`konu`,`metin`)";
        $sayi=2;
        $baslik="Sık Sorulan Sorular"; break;
}
echo "<h5>$baslik</h5><br>";
if (!isset($_GET["islem"])) $islem="form"; else $islem=$_GET["islem"];
switch ($islem) {
    case "form" : form(); break;
    case "kaydet" : kaydet(); break;
}
function form() {
    global $sayfa, $csvformati;
    echo "<b><u>CSV Formatı:</u></b><br><font color='red'>$csvformati</font><br><br>";
    Aşağıdaki alana yukarıdaki formatta verileri giriniz. Kayıt bittiğinde enter'a basarak sonrakine geçiniz.";
    echo "<form name='form1' method='post'";
    action="?sec=cokluverigirisi&sayfa=$sayfa&islem=kaydet">
    <textarea name='cokluveri' cols='105' rows='20'></textarea>
    <br><input type='submit' name='submit' value='Kaydet'></form>";
} // func form
function kaydet() {
    global $sayfa, $alanlar, $sayi;
    if (!isset($_POST["submit"])) { form(); exit; }
    $cokluveri=@$_POST["cokluveri"];
    if (empty($cokluveri))
    { echo "Lütfen <a href='javascript:history.back()'>geri</a> dönerek çoklu veri eklemek için
    bilgileri giriniz!";
        exit;
    }
    $satirlar=explode(chr(10), $cokluveri);
    if (count($satirlar)==0)

```





```

        <td><a
href='?sec=dersler&islem=sil&kodu=$kodu'>Sil</a></td></tr>";
    }
    echo "</table> Liste sonu... ";
}
} // func listele
function formgoster() {
if (!isset($_GET['kodu'])) $kodu="0"; else $kodu=$_GET['kodu'];
vtablanvesec();
if ($kodu=="0") {
    echo "<h4>Yeni Ders Ekleme</h4>";
    $adi=""; $programkodu=""; $islem="yenikaydet";
}
else
{
    echo "<h4>Ders Bilgi Değişikliği</h4>";
    $kayitlar=@mysql_query("select * from dersler where kodu='$kodu'") or die("Ders bilgisine
ulaşlamadı...");
    if (mysql_num_rows($kayitlar)==0) die("Ders bilgisine ulaşlamadı...");
    $kayit=mysql_fetch_array($kayitlar);
    $adi=$kayit[1]; $programkodu=$kayit[2]; $islem="degisiklikkaydet";
}
echo "<form name='form1' method='post' action='?sec=dersler&islem=$islem&kodu=$kodu'>
<table border='0' cellspacing='0' cellpadding='5'><tr>
<td>Ders Kodu </td><td>";
if ($kodu=="0") echo "<input name='kodu' type='text' id='kodu' maxlength='10'>"; else echo "$kodu";
echo "</td></tr><tr><td>Ders Adı </td>
<td><input name='adi' type='text' id='adi' size='50' maxlength='50' value='$adi'></td></tr>
<tr><td valign='top'>Program</td><td><select name='programkodu'>";
$programlar=@mysql_query("SELECT * FROM `programlar`") or die("Program bilgilerine
ulaşlamadı...");
while ($program=mysql_fetch_array($programlar)) {
    $prgkodu=$program[0]; $prgadi=$program[1];
    echo "<option value='$prgkodu'";
    if ($prgkodu==$programkodu) echo " selected";
    echo ">$prgadi</option>\n";
}
echo "</select></td></tr><tr><td>&nbsp;</td>
<td><br><input type='submit' name='submit' value='Kaydet'></td></tr></table></form>";
} // func formgoster
function yenikaydet() {
if (!isset($_POST['submit'])) { formgoster(); exit; }
$kodu=$_POST['kodu']; $adi=$_POST['adi']; $programkodu=$_POST['programkodu'];
vtablanvesec();
$s=@mysql_query("INSERT INTO `dersler` ( `kodu` , `adi` , `programkodu` )
VALUES ('$kodu', '$adi', '$programkodu')");
if ($s) echo "$kodu kodlu ders eklendi...";
else echo "$kodu kodlu ders eklenemedi!<br>
Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func yenikaydet
function degisiklikkaydet() {
if (!isset($_POST['submit'])) { formgoster(); exit; }
$kodu=$_GET['kodu']; $adi=$_POST['adi']; $programkodu=$_POST['programkodu'];
vtablanvesec();
$s=@mysql_query("UPDATE `dersler` SET `adi` = '$adi', `programkodu` = '$programkodu' WHERE `kodu`
= '$kodu' ");
if ($s) echo "$kodu kodlu ders bilgileri değiştirildi...";
else echo "$kodu kodlu ders bilgileri değiştirilemedi!<br>
Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func degisiklikkaydet
function sil() {
if (!isset($_GET['kodu'])) { listele(); exit; }
$kodu=$_GET['kodu'];
vtablanvesec();

```

```

$s=@mysql_query("DELETE FROM `dersler` WHERE `kodu` = '$kodu' ");
if ($s) echo "$kodu kodlu ders silindi...";
else echo "$kodu kodlu ders silinemedi!<br>Lütfen <a href='javascript:history.back()>geri</a> dönerek
tekrar deneyiniz...";
} // func sil
?>

```

### Duyurular

```

<h3>Duyurular</h3>
<hr>
| <a href='?sec=duyurular&islem=ekle'>Ekle</a> | <a
href='?sec=duyurular&islem=listele'>Listele</a> |
<hr>
<p>
<?
if (!isset($_GET["islem"])) $islem="listele"; else $islem=$_GET["islem"];

switch ($islem) {
    case "listele"           : listele(); break;
    case "ekle"             : formgoster(); break;
    case "degistir"        : formgoster(); break;
    case "yenikaydet"       : yenikaydet(); break;
    case "degisiklikkaydet" : degisiklikkaydet(); break;
    case "sil"              : sil(); break;
}
function listele() {
    vtbaglanvesec();
    $liste=@mysql_query("select * from duyurular order by konu") or die("Duyurular tablosuna ulařilamadı!");
    if (mysql_num_rows($liste)==0)
        echo "Kayıtlı duyuru yok...";
    else
        {
            echo "<table border='1' cellspacing='0' cellpadding='5'><tr>
                <th>Sıra
No</th><th>Kodu</th><th>Konu</th><th>&nbsp;</th><th>&nbsp;</th></tr>";
            $x=0;
            while ($prg=mysql_fetch_array($liste))
                {
                    $x++;
                    $kodu=$prg[0]; $konu=$prg[1];
                    echo "<tr><td>$x</td><td>$kodu</td><td>$konu</td>
                        <td><a
href='?sec=duyurular&islem=degistir&kodu=$kodu'>Deęiřtir</a></td>
                        <td><a
href='?sec=duyurular&islem=sil&kodu=$kodu'>Sil</a></td></tr>";
                }
            echo "</table> Liste sonu... ";
        }
} // func listele
function formgoster() {
if (!isset($_GET["kodu"])) $kodu="0"; else $kodu=$_GET["kodu"];
if ($kodu=="0") {
    echo "<h4>Yeni Duyuru Ekleme</h4>";
    $konu=""; $metin=""; $islem="yenikaydet";
}
else
{
    echo "<h4>Duyuru Deęiřiklięi</h4>";
    vtbaglanvesec();
    $kayitlar=@mysql_query("select * from duyurular where kodu='$kodu'") or die("Duyuru bilgisine
ulařilamadı...");
    if (mysql_num_rows($kayitlar)==0) die("Duyuru bilgisine ulařilamadı...");
    $kayit=mysql_fetch_array($kayitlar);
    $konu=$kayit[1]; $metin=$kayit[2]; $islem="degisiklikkaydet";
}
}

```

```

    }
    echo "<form name='form1' method='post' action='?sec=duyurular&islem=$islem&kodu=$kodu'>
        <table border='0' cellspacing='0' cellpadding='5'><tr>
            <td>Duyuru Kodu </td><td>";
    if ($kodu=="0") echo "Yeni"; else echo "$kodu";
    echo "</td></tr><tr><td>Konu</td>
        <td><input name='konu' type='text' id='konu' size='50' maxlength='50'
    value='$konu'></td></tr>
        <tr><td valign='top'>Duyuru Metni</td>
            <td><textarea name='metin' cols='51' rows='10'>$metin</textarea></td>
        </tr><tr><td>&nbsp;</td><td><br><input type='submit' name='submit'
    value='Kaydet'></td></tr></table></form>";
} // func formgoster
function yenikaydet() {
    if (!isset($_POST['submit'])) { formgoster(); exit; }
    $konu=$_POST['konu']; $metin=$_POST['metin'];
    vtbaglanvesec();
    $s=@mysql_query("INSERT INTO `duyurular` ( `kodu` , `konu` , `metin` )
        VALUES ( , '$konu', '$metin' );");
    if ($s) echo "$konu konulu duyuru eklendi...";
    else echo "$konu konulu duyuru eklenemedi!<br>Lütfen <a href='javascript:history.back()'>geri</a>
    dönerek tekrar deneyiniz...";
} // func yenikaydet
function degisiklikkaydet() {
    if (!isset($_POST['submit'])) { formgoster(); exit; }
    $kodu=$_GET['kodu']; $konu=$_POST['konu']; $metin=$_POST['metin'];
    vtbaglanvesec();
    $s=@mysql_query("UPDATE `duyurular` SET `konu` = '$konu', `metin` = '$metin' WHERE `kodu` =
    '$kodu' ");
    if ($s) echo "$kodu kodlu duyuru değiştirildi...";
    else echo "$kodu kodlu duyuru değiştirilemedi!<br>
        Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func degisiklikkaydet
function sil() {
    if (!isset($_GET['kodu'])) { listele(); exit; }
    $kodu=$_GET['kodu'];
    vtbaglanvesec();
    $s=@mysql_query("DELETE FROM `duyurular` WHERE `kodu` = '$kodu' ");
    if ($s) echo "$kodu kodlu duyuru silindi...";
    else echo "$kodu kodlu duyuru silinemedi!<br>
        Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func sil
?>

Haftalikdersprogrami
<h3>Haftalık Ders Programı</h3>
<hr>
<?
if (!isset($_GET["islem"])) $islem="listele"; else $islem=$_GET["islem"];

switch ($islem) {
    case "listele" : listele(); break;
    case "yenikaydet" : yenikaydet(); break;
    case "degisiklikkaydet" : degisiklikkaydet(); break;
    case "sil" : sil(); break;
}
function listele() {
    vtbaglanvesec();
    $liste=@mysql_query("select * from haftaliktakvim order by derskodu") or die("Ders programi
    tablosuna ulaşılamadı!");
    echo "<table border='1' cellspacing='0' cellpadding='5'><tr>
        <th>Ders Adı</th><th>Gün</th><th>Başlangıç
    Saati</th><th>Bitiş Saati</th><th>&nbsp;</th><th>&nbsp;</th></tr>";
}

```

```

                                <form method='post'
action='?sec=haftalikdersprogrami&islem=yenikaydet'><tr><td>";
                                derslerlistekutusu("");
                                echo "</td><td>";
                                gunlerlistekutusu("");
                                echo "</td><td><input type='text' name='baslangic' size='5' maxlength='5'
value='00:00'></td>
                                <td><input type='text' name='bitis' size='5' maxlength='5' value='00:00'></td>
                                <td colspan='2' align='center'><input type='submit' name='submit' value=' Ekle
'></td></tr></form>";
                                while ($prg=mysql_fetch_array($liste))
                                {
                                $derskodu=$prg[0]; $gun=$prg[1]; $baslangic=substr($prg[2],0,5);
                                $bitis=substr($prg[3],0,5);
                                echo "<form method='post'
action='?sec=haftalikdersprogrami&islem=degisiklikkaydet&derskodu=$derskodu'><tr><td>";
                                derslerlistekutusu($derskodu);
                                echo "</td><td>";
                                gunlerlistekutusu($gun);
                                echo "</td><td><input type='text' name='baslangic' size='5'
maxlength='5' value='$baslangic'></td>
                                <td><input type='text' name='bitis' size='5' maxlength='5'
value='$bitis'></td>
                                <td><input type='image' name='submit'
src='b_edit.png'></td>
                                <td><a
href='?sec=haftalikdersprogrami&islem=sil&derskodu=$derskodu'>Sil</a></td></tr></form>";
                                }
                                echo "</table> Liste sonu... ";
} // func listele
function yenikaydet() {
if (!isset($_POST['submit'])) { listele(); exit; }
$derskodu=$_POST['derskodu']; $gun=$_POST['gun']; $baslangic=$_POST['baslangic'];
$bitis=$_POST['bitis'];
vtabaglanvesec();
$s=@mysql_query("select * from `haftaliktakvim` where derskodu='$derskodu'");
if (@mysql_num_rows($s)>0)
{
echo "$derskodu kodlu ders daha önceden kaydedildi!";
listele();
exit;
}
$s=@mysql_query("INSERT INTO `haftaliktakvim` ( `derskodu`, `gun`, `baslangic`, `bitis` )
VALUES ('$derskodu', '$gun', '$baslangic', '$bitis')");
if (!$s) echo "$derskodu kodlu ders eklenemedi!";
listele();
} // func yenikaydet
function degisiklikkaydet() {
$derskodu=$_GET['derskodu']; $dk=$_POST['derskodu'];
$gun=$_POST['gun']; $baslangic=$_POST['baslangic']; $bitis=$_POST['bitis'];
vtabaglanvesec();
$s=mysql_query("select * from `haftaliktakvim` where derskodu='$dk'");
if ((@mysql_num_rows($s)==1) and ($dk<>$derskodu))
{
echo "$derskodu kodlu ders daha önceden kaydedildi!";
listele();
exit;
}
$s=@mysql_query("UPDATE `haftaliktakvim` SET `derskodu` = '$dk', `gun` = '$gun', `baslangic` =
'$baslangic', `bitis` = '$bitis'
WHERE `derskodu` = '$derskodu' ");
if (!$s) echo "$derskodu kodlu ders bilgileri değiştirilemedi!";
listele();
} // func degisiklikkaydet

```

```

function sil() {
if (!isset($_GET['derskodu'])) { listele(); exit; }
$derskodu=$_GET['derskodu'];
vtabaglanvesec();
$s=@mysql_query("DELETE FROM `haftaliktakvim` WHERE `derskodu` = '$derskodu' ");
if ($s) echo "$derskodu kodlu ders silindi..."; else echo "$derskodu kodlu ders silinemedi!";
listele();
} //func sil
?>

Öğrenciler
<h3>Öğrenciler</h3>
<hr>
| <a href="?sec=ogrenciler&islem=ekle">Ekle</a> | <a
href="?sec=ogrenciler&islem=listele">Listele</a> |
<hr>
<p>
<?
if (!isset($_GET["islem"])) $islem="listele"; else $islem=$_GET["islem"];

switch ($islem) {
case "listele" : listele(); break;
case "ekle" : formgoster(); break;
case "degistir" : formgoster(); break;
case "yenikaydet" : yenikaydet(); break;
case "degisiklikkaydet" : degisiklikkaydet(); break;
case "sil" : sil(); break;
}
function listele() {
vtabaglanvesec();
$liste=@mysql_query("select * from ogrenciler order by ogrencino") or die("Öğrenciler tablosuna
ulaşilamadı!");
if (mysql_num_rows($liste)==0)
echo "Kayıtlı öğrenci yok...";
else
{
echo "<table border='1' cellspacing='0' cellpadding='5'><tr>
<th>Sıra No</th><th>Öğrenci
No</th><th>Adı</th><th>Soyadı</th>
<th>Program
Kodu</th><th>&nbsp;</th><th>&nbsp;</th><th>&nbsp;</th></tr>";
$x=0;
while ($prg=mysql_fetch_array($liste))
{
$x++;
$ogrencino=$prg[0]; $adi=$prg[1]; $soyadi=$prg[2];
$programkodu=$prg[3];
echo
"<tr><td>$x</td><td>$ogrencino</td><td>$adi</td><td>$soyadi</td><td>$programkodu</td>
<td><a
href=?sec=ogrenciler&islem=degistir&ogrencino=$ogrencino'>Değiştir</a></td>
<td><a
href=?sec=ogrenciler&islem=sil&ogrencino=$ogrencino'>Sil</a></td></tr>";
}
echo "</table> Liste sonu... ";
}
} //func listele
function formgoster() {
if (!isset($_GET["ogrencino"])) $ogrencino="0"; else $ogrencino=$_GET["ogrencino"];
vtabaglanvesec();
if ($ogrencino=="0") {
echo "<h4>Yeni Öğrenci Ekleme</h4>";
$adi=""; $soyadi=""; $programkodu=""; $sifre=""; $islem="yenikaydet";
}
else

```

```

    {
        echo "<h4>Öğrenci Bilgi Değişikliği</h4>";
        $kayitlar=@mysql_query("select * from ogrenciler where ogrencino='$ogrencino'") or
die("Öğrenci bilgisine ulaşamadı...");
        if (mysql_num_rows($kayitlar)==0) die("Öğrenci bilgisine ulaşamadı...");
        $kayit=mysql_fetch_array($kayitlar);
        $adi=$kayit[1]; $soyadi=$kayit[2]; $programkodu=$kayit[3];
        $sifre=$kayit[4]; $islem="degisiklikkaydet";
    }
echo "<form name='form1' method='post'
action='?sec=ogrenciler&islem=$islem&ogrencino=$ogrencino'>
    <table border='0' cellspacing='0' cellpadding='5'><tr>
        <td>Öğrenci No</td><td>";
if ($ogrencino=="0") echo "<input name='ogrencino' type='text' id='ogrencino' maxlength='12'>"; else
echo "$ogrencino";
echo "</td></tr><tr><td>Adı </td>
        <td><input name='adi' type='text' id='adi' size='50' maxlength='50' value='$adi'></td></tr>
        <tr><td>Soyadı </td>
        <td><input name='soyadi' type='text' id='soyadi' size='50' maxlength='50'
value='$soyadi'></td></tr>
        <tr><td valign='top'>Program</td><td><select name='programkodu'>";
        $programlar=@mysql_query("SELECT * FROM `programlar`") or die("Program bilgilerine
ulaşamadı...");
        while ($program=mysql_fetch_array($programlar)) {
            $prgkodu=$program[0]; $prgadi=$program[1];
            echo "<option value='$prgkodu'";
            if ($prgkodu==$programkodu) echo " selected";
            echo ">$prgadi</option>\n";
        }
        echo "</select></td></tr>
        <tr><td>Şifre </td>
        <td><input name='sifre' type='password' id='sifre' maxlength='10' value='$sifre'></td></tr>
        <tr><td>&nbsp;</td>
            <td><br><input type='submit' name='submit'
value='Kaydet'></td></tr></table></form>";
} // func formgoster
function yenikaydet() {
if (!isset($_POST['submit'])) { formgoster(); exit; }
$ogrencino=$_POST['ogrencino']; $adi=$_POST['adi']; $soyadi=$_POST['soyadi'];
$sifre=$_POST['sifre']; $programkodu=$_POST['programkodu'];
vtabaglanvesec();
$$=@mysql_query("INSERT INTO `ogrenciler` (`ogrencino`, `adi`, `soyadi`, `programkodu`, `sifre`)
VALUES ('$ogrencino', '$adi', '$soyadi', '$programkodu', '$sifre')");
if ($$) echo "$ogrencino numaralı öğrenci eklendi...";
else echo "$ogrencino numaralı öğrenci eklenemedi!<br>
        Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar
deneyiniz...";
} // func yenikaydet
function degisiklikkaydet() {
if (!isset($_POST['submit'])) { formgoster(); exit; }
$ogrencino=$_GET['ogrencino']; $adi=$_POST['adi']; $soyadi=$_POST['soyadi'];
$sifre=$_POST['sifre']; $programkodu=$_POST['programkodu'];
vtabaglanvesec();
$$=@mysql_query("UPDATE `ogrenciler` SET `adi` = '$adi', `soyadi` = '$soyadi', `programkodu` =
'$programkodu',
`sifre` = '$sifre' WHERE `ogrencino` = '$ogrencino' ");
if ($$) echo "$ogrencino numaralı öğrenci bilgileri değiştirildi...";
else echo "$ogrencino numaralı öğrenci bilgileri değiştirilemedi!<br>
        Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar
deneyiniz...";
} // func degisiklikkaydet
function sil() {
if (!isset($_GET['ogrencino'])) { listele(); exit; }
$ogrencino=$_GET['ogrencino'];

```

```

vbaglanvesec();
$s=@mysql_query("DELETE FROM `ogrenciler` WHERE `ogrencino` = '$ogrencino' ");
if ($s) echo "Sogrencino numaralı öğrenci silindi..";
else echo "Sogrencino numaralı öğrenci silinemedi!<br>Lütfen <a
href='javascript:history.back()>geri</a> dönerek tekrar deneyiniz...";
} //func sil
?>

Personel
<h3>Personel</h3>
<hr>
| <a href='?sec=personel&islem=ekle'>Ekle</a> | <a href='?sec=personel&islem=listele'>>Listele</a> |
<hr>
<p>
<?
if (!isset($_GET["islem"])) $islem="listele"; else $islem=$_GET["islem"];

switch ($islem) {
    case "listele" : listele(); break;
    case "ekle" : formgoster(); break;
    case "degistir" : formgoster(); break;
    case "yenikaydet" : yenikaydet(); break;
    case "degisiklikkaydet" : degisiklikkaydet(); break;
    case "sil" : sil(); break;
}
function listele() {
    vbaglanvesec();
    $liste=@mysql_query("select * from personel order by sicilno") or die("Personel tablosuna
ulaşlamadı!");
    if (mysql_num_rows($liste)==0)
        echo "Kayıtlı personel yok...";
    else
        {
            echo "<table border='1' cellspacing='0' cellpadding='5'><tr>
                <th>Sıra No</th><th>Sicil
No</th><th>Ünvanı</th><th>Adı</th><th>Soyadı</th>
                <th>Görevi</th><th>&nbsp;</th><th>&nbsp;</th></tr>";
            $x=0;
            while ($prg=mysql_fetch_array($liste))
                {
                    $x++;
                    $sicilno=$prg[0]; $unvan=$prg[1]; $adi=$prg[2];
                    $soyadi=$prg[3]; $gorevi=$prg[4];
                    echo
                    "<tr><td>$x</td><td>$sicilno</td><td>$unvan</td><td>$adi</td>
                    <td>$soyadi</td><td>$gorevi</td>
                    <td><a
href='?sec=personel&islem=degistir&sicilno=$sicilno'>Değiştir</a></td>
                    <td><a
href='?sec=personel&islem=sil&sicilno=$sicilno'>Sil</a></td></tr>";
                }
            echo "</table> Liste sonu... ";
        }
} //func listele
function formgoster() {
    if (!isset($_GET["sicilno"])) $sicilno="0"; else $sicilno=$_GET["sicilno"];
    vbaglanvesec();
    if ($sicilno=="0") {
        echo "<h4>Yeni Personel Ekleme</h4>";
        $unvan=""; $adi=""; $soyadi=""; $gorevi=""; $detay=""; $islem="yenikaydet";
    }
    else
        {
            echo "<h4>Personel Bilgi Değişikliği</h4>";

```



```

        $kayitlar=@mysql_query("select * from personel where sicilno='$sicilno'") or die("Personel
bilgisine ulaşamadı...");
        if (mysql_num_rows($kayitlar)==0) die("Personel bilgisine ulaşamadı...");
        $kayit=mysql_fetch_array($kayitlar);
        $unvan=$kayit[1]; $adi=$kayit[2]; $soyadi=$kayit[3]; $gorevi=$kayit[4]; $detay=$kayit[5];
        $islem="degisiklikkaydet";
    }
    echo "<form name='form1' method='post' action='?sec=personel&islem=$islem&sicilno=$sicilno'>
    <table border='0' cellspacing='0' cellpadding='5'><tr>
    <td>Sicil No</td><td>";
    if ($sicilno=="0") echo "<input name='sicilno' type='text' id='sicilno' maxlength='10'>"; else echo
"$sicilno";
    echo "</td></tr><tr><td>Ünvanı </td>
    <td><input name='unvan' type='text' id='unvan' size='25' maxlength='25'
value='$unvan'></td></tr>
    <tr><td>Adı </td>
    <td><input name='adi' type='text' id='adi' size='50' maxlength='50' value='$adi'></td></tr>
    <tr><td>Soyadı </td>
    <td><input name='soyadi' type='text' id='soyadi' size='50' maxlength='50'
value='$soyadi'></td></tr>
    <tr><td>Görevi</td>
    <td><input name='gorevi' type='text' id='gorevi' size='50' maxlength='50'
value='$gorevi'></td></tr>
    <tr><td valign='top'>Detaylı Bilgi</td>
    <td><textarea name='detay' cols='51' rows='10'>$detay</textarea></td></tr>
    <tr><td>&nbsp;</td><td><br><input type='submit' name='submit'
value='Kaydet'></td></tr>
    </table></form>";
} // func formgoster
function yenikaydet() {
    if (!isset($_POST['submit'])) { formgoster(); exit; }
    $sicilno=$_POST['sicilno']; $unvan=$_POST['unvan']; $adi=$_POST['adi']; $soyadi=$_POST['soyadi'];
    $gorevi=$_POST['gorevi']; $detay=$_POST['detay'];
    vtbaglanvesec();
    $s=@mysql_query("INSERT INTO `personel` ( `sicilno` , `unvan` , `adi` , `soyadi` , `gorevi` , `detay` )
VALUES ('$sicilno', '$unvan', '$adi', '$soyadi', '$gorevi', '$detay')");
    if ($s) echo "$sicilno sicil numaralı personel eklendi...";
    else echo "$sicilno sicil numaralı personel eklenemedi!<br>
    Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func yenikaydet
function degisiklikkaydet() {
    if (!isset($_POST['submit'])) { formgoster(); exit; }
    $sicilno=$_GET['sicilno']; $unvan=$_POST['unvan']; $adi=$_POST['adi']; $soyadi=$_POST['soyadi'];
    $gorevi=$_POST['gorevi']; $detay=$_POST['detay'];
    vtbaglanvesec();
    $s=@mysql_query("UPDATE `personel` SET `unvan` = '$unvan', `adi` = '$adi', `soyadi` = '$soyadi',
`gorevi` = '$gorevi',
`detay` = '$detay' WHERE `sicilno` = '$sicilno' ");
    if ($s) echo "$sicilno sicil numaralı personel bilgileri değiştirildi...";
    else echo "$sicilno sicil numaralı personel bilgileri değiştirilemedi!<br>
    Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func degisiklikkaydet
function sil() {
    if (!isset($_GET['sicilno'])) { listele(); exit; }
    $sicilno=$_GET['sicilno'];
    vtbaglanvesec();
    $s=@mysql_query("DELETE FROM `personel` WHERE `sicilno` = '$sicilno' ");
    if ($s) echo "$sicilno sicil numaralı personel silindi...";
    else echo "$sicilno sicil numaralı personel silinemedi!<br>Lütfen <a
href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func sil
?>
Programlar
<h3>Programlar</h3>

```

```

<hr>
| <a href="?sec=programlar&islem=ekle">Ekle</a> | <a
href="?sec=programlar&islem=listele">Listele</a> |
<hr>
<p>
<?
if (!isset($_GET["islem"])) $islem="listele"; else $islem=$_GET["islem"];
switch ($islem) {
    case "listele"          : listele(); break;
    case "ekle"            : formgoster(); break;
    case "degistir"       : formgoster(); break;
    case "yenikaydet"     : yenikaydet(); break;
    case "degisiklikkaydet" : degisiklikkaydet(); break;
    case "sil"            : sil(); break;
}
function listele() {
    vtbaglanvesec();
    $liste=@mysql_query("select * from programlar order by adi") or die("Programlar tablosuna
ulaşilamadi!");
    if (mysql_num_rows($liste)==0)
        echo "Kayıtlı program yok...";
    else
        {
            echo "<table border='1' cellspacing='0' cellpadding='5'><tr>
                <th>Sıra
No</th><th>Kodu</th><th>Adı</th><th>&nbsp;</th><th>&nbsp;</th><th>&nbsp;</th></tr>";
            $x=0;
            while ($prg=mysql_fetch_array($liste))
                {
                    $x++;
                    $kodu=$prg[0]; $adi=$prg[1];
                    echo "<tr><td>$x</td><td>$kodu</td><td>$adi</td>
                        <td><a
href=?sec=programlar&islem=degistir&kodu=$kodu>Değiştir</a></td>
                        <td><a
href=?sec=programlar&islem=sil&kodu=$kodu>Sil</a></td></tr>";
                }
            echo "</table> Liste sonu... ";
        }
} // func listele
function formgoster() {
    if (!isset($_GET["kodu"])) $kodu="0"; else $kodu=$_GET["kodu"];
    if ($kodu=="0") {
        echo "<h4>Yeni Program Ekleme</h4>";
        $adi=""; $detay=""; $islem="yenikaydet";
    }
    else
        {
            echo "<h4>Program Bilgi Değişikliği</h4>";
            vtbaglanvesec();
            $kayitlar=@mysql_query("select * from programlar where kodu='$kodu'") or die("Program
bilgisine ulaşilamadi...");
            if (mysql_num_rows($kayitlar)==0) die("Program bilgisine ulaşilamadi...");
            $kayit=mysql_fetch_array($kayitlar);
            $adi=$kayit[1]; $detay=$kayit[2]; $islem="degisiklikkaydet";
        }
    echo "<form name='form1' method='post' action=?sec=programlar&islem=$islem&kodu=$kodu'>
        <table border='0' cellspacing='0' cellpadding='5'><tr>
            <td>Program Kodu </td><td>";
    if ($kodu=="0") echo "<input name='kodu' type='text' id='kodu' maxlength='10'>"; else echo "$kodu";
    echo "</td></tr><tr><td>Program Adı </td>
        <td><input name='adi' type='text' id='adi' size='50' maxlength='50' value='$adi'></td></tr>
        <tr><td valign='top'>Program Hakkında <br>Detaylı Bilgi </td>
        <td><textarea name='detay' cols='51' rows='10'>$detay</textarea></td>

```

```

</tr><tr><td>&nbsp;</td><td><br><input type='submit' name='submit'
value='Kaydet'></td></tr></table></form>";
} // func formgoster
function yenikaydet() {
if (!isset($_POST['submit'])) { formgoster(); exit; }
$kodu=$_POST['kodu']; $adi=$_POST['adi']; $detay=$_POST['detay'];
vtabaglanvesec();
$s=@mysql_query("INSERT INTO `programlar` ( `kodu` , `adi` , `detay` )
VALUES ('$kodu', '$adi', '$detay')");
if ($s) echo "$kodu kodlu program eklendi...";
else echo "$kodu kodlu program eklenemedi!<br>
Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func yenikaydet
function degisiklikkaydet() {
if (!isset($_POST['submit'])) { formgoster(); exit; }
$kodu=$_GET['kodu']; $adi=$_POST['adi']; $detay=$_POST['detay'];
vtabaglanvesec();
$s=@mysql_query("UPDATE `programlar` SET `adi` = '$adi', `detay` = '$detay' WHERE `kodu` = '$kodu'
");
if ($s) echo "$kodu kodlu program bilgileri değiştirildi...";
else echo "$kodu kodlu program bilgileri değiştirilemedi!<br>
Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func degisiklikkaydet
function sil() {
if (!isset($_GET['kodu'])) { listele(); exit; }
$kodu=$_GET['kodu'];
vtabaglanvesec();
$s=@mysql_query("DELETE FROM `programlar` WHERE `kodu` = '$kodu' ");
if ($s) echo "$kodu kodlu program silindi...";
else echo "$kodu kodlu program silinemedi!<br>Lütfen <a href='javascript:history.back()'>geri</a>
dönerek tekrar deneyiniz...";
} // func sil

?>
Sinav sonuçları
<h3>Sinav Sonuçları</h3>
<hr>
<?
if (!isset($_GET["islem"])) $islem="listele"; else $islem=$_GET["islem"];

switch ($islem) {
case "listele" : listele(); break;
case "yenikaydet" : yenikaydet(); break;
case "degisiklikkaydet" : degisiklikkaydet(); break;
case "sil" : sil(); break;
}

function listele() {
vtabaglanvesec();
$liste=@mysql_query("select * from sinavsonuclari order by ogrencino, derskodu") or
die("Sinav sonuçları tablosuna ulaşılamadı!");
echo "<table border='1' cellspacing='0' cellpadding='5'><tr>

<th>Öğrenci</th><th>Ders</th><th>Vize1</th><th>Vize2</th><th>Final</th><th>Ortalama
a</th><th>Harf Notu</th>

<th>&nbsp;</th><th>&nbsp;</th><th>&nbsp;</th></tr>
<form method='post'
action='?sec=sinavsonuclari&islem=yenikaydet'><tr><td>";
ogrencilerlistekutusu("");
echo "</td><td>";
derslerlistekutusu("");
echo "</td><td><input type='text' name='vize1' size='3' maxlength='3'></td>
<td><input type='text' name='vize2' size='3' maxlength='3'></td>
<td><input type='text' name='final' size='3' maxlength='3'></td>

```

```

<td><input type='text' name='ortalama' size='3' maxlength='3'></td>
<td><input type='text' name='harfnotu' size='3' maxlength='3'></td>
<td colspan='2' align='center'><input type='submit' name='submit' value=' Ekle
'></td></tr></form>";

while ($prg=mysql_fetch_array($liste))
{
    $sid=$prg[0]; $sogrencino=$prg[1]; $derskodu=$prg[2];
    $vize1=$prg[3]; $vize2=$prg[4];
    $final=$prg[5]; $ortalama=$prg[6]; $harfnotu=$prg[7];

    echo "<form method='post'
action='?sec=sinavsonuclari&islem=degisiklikkaydet&id=$sid'><tr><td>";
    ogrencilerlistekutusu($sogrencino);
    echo "</td><td>";
    derslerlistekutusu($derskodu);
    echo "</td><td><input type='text' name='vize1' size='3'
maxlength='3' value='$vize1'></td>
<td><input type='text' name='vize2' size='3' maxlength='3'
value='$vize2'></td>
<td><input type='text' name='final' size='3' maxlength='3'
value='$final'></td>
<td><input type='text' name='ortalama' size='3'
maxlength='3' value='$ortalama'></td>
<td><input type='text' name='harfnotu' size='3'
maxlength='3' value='$harfnotu'></td>
<td><input type='image' name='submit'
src='b_edit.png'></td>
<td><a
href='?sec=sinavsonuclari&islem=sil&id=$sid'>Sil</a></td></tr></form>";
}
echo "</table> Liste sonu... ";

} // func listele
function yenikaydet() {
if (!isset($_POST['submit'])) { listele(); exit; }
$sogrencino=$_POST['ogrencino']; $derskodu=$_POST['derskodu'];
$vize1=$_POST['vize1']; $vize2=$_POST['vize2'];
$final=$_POST['final']; $ortalama=$_POST['ortalama']; $harfnotu=$_POST['harfnotu'];
vbaglanvesec();
$s=@mysql_query("select * from `sinavsonuclari` where ogrencino='$sogrencino' and
derskodu='$derskodu'");
if (@mysql_num_rows($s)>0)
{
    echo "$sogrencino numaralı öğrencinin $derskodu kodlu dersi daha önceden kaydedildi!";
    listele();
    exit;
}
$s=@mysql_query("INSERT INTO `sinavsonuclari` ( `id`, `ogrencino`, `derskodu`, `vize1`, `vize2`, `final`
, `ortalama`, `harfnotu` )
VALUES (", '$sogrencino', '$derskodu', '$vize1', '$vize2', '$final',
'$ortalama', '$harfnotu'"));
if (!$s) echo "$sogrencino numaralı öğrencinin $derskodu kodlu dersi eklenemedi!";
listele();
} // func yenikaydet
function degisiklikkaydet() {
$Sid=$_GET['id'];
$sogrencino=$_POST['ogrencino']; $derskodu=$_POST['derskodu'];
$vize1=$_POST['vize1']; $vize2=$_POST['vize2'];
$final=$_POST['final']; $ortalama=$_POST['ortalama']; $harfnotu=$_POST['harfnotu'];
vbaglanvesec();
$sonuc=mysql_query("select * from `sinavsonuclari` where id='$Sid'");
$s=@mysql_fetch_array($sonuc);
$sono=$s['ogrencino']; $dk=$s['derskodu'];

```

```

$sonuc=mysql_query("select * from `sinavsonuclari` where ogrencino='$ogrencino' and
derskodu='$derskodu'");
if ((@mysql_num_rows($sonuc)==1) and (($ono<>$ogrencino) or ($dk<>$derskodu)))
{
    echo "$ogrencino numaralı öğrenci için $derskodu kodlu ders daha önceden kaydedildi!";
    listele();
    exit;
}
$s=@mysql_query("UPDATE `sinavsonuclari` SET `ogrencino` = '$ogrencino', `derskodu` = '$derskodu',
`vize1` = '$vize1', `vize2` = '$vize2', `final` = '$final', `ortalama` =
`Sortalama`, `harfnotu` = '$harfnotu'
WHERE `id` = '$id' ");
if (!$s) echo "$ono numaralı öğrencinin $dk kodlu ders bilgileri değiştirilemedi!";
listele();
} // func degisiklikkaydet
function sil() {
if (!isset($_GET['id'])) { listele(); exit; }
$id=$_GET['id'];
vbaglanvesec();
$sonuc=mysql_query("select * from `sinavsonuclari` where id='$id'");
$s=@mysql_fetch_array($sonuc);
$ono=$s['ogrencino']; $dk=$s['derskodu'];
$s=@mysql_query("DELETE FROM `sinavsonuclari` WHERE `id` = '$id' ");
if ($s) echo "$ono numaralı öğrencinin $dk kodlu dersine ait sonuç silindi..."; else echo "$ono numaralı
öğrencinin $dk kodlu dersine ait sonuç silinemedi!";
listele();
} // func sil

?>
Sinav takvimi
<h3>Sinav Takvimi</h3>
<hr>
<?
if (!isset($_GET["islem"])) $islem="listele"; else $islem=$_GET["islem"];

switch ($islem) {
    case "listele" : listele(); break;
    case "yenikaydet" : yenikaydet(); break;
    case "degisiklikkaydet" : degisiklikkaydet(); break;
    case "sil" : sil(); break;
}
function listele() {
    vbaglanvesec();
    $liste=@mysql_query("select * from sinavtakvimi order by derskodu") or die("Sinav takvimi
tablosuna ulaşamadı!");
    echo "<table border='1' cellspacing='0' cellpadding='5'><tr>
        <th>Ders
    Adı</th><th>Tarih</th><th>Saat</th><th>&nbsp;</th><th>&nbsp;</th></tr>
        <form method='post'
    action='?sec=sinavtakvimi&islem=yenikaydet'><tr><td>";
        derslerlistekutusu("");
        echo "</td><td><input type='text' name='tarih' size='10' maxlength='10'
    value='00.00.0000'></td>
        <td><input type='text' name='saat' size='5' maxlength='5' value='00:00'></td>
        <td colspan='2' align='center'><input type='submit' name='submit' value=' Ekle
    '></td></tr></form>";

        while ($prg=mysql_fetch_array($liste))
        {
            $derskodu=$prg[0]; $tarih=$prg[1]; $saat=substr($prg[2],0,5);
            $tarih=substr($tarih,8,2).".".substr($tarih,5,2).".".substr($tarih,0,4);
            echo "<form method='post'
        action='?sec=sinavtakvimi&islem=degisiklikkaydet&derskodu=$derskodu'><tr><td>";
            derslerlistekutusu($derskodu);
            echo "</td><td><input type='text' name='tarih' size='10'
        maxlength='10' value='$tarih'></td>

```

```

<td><input type='text' name='saat' size='5' maxlength='5' value='$saat'></td>
<td><input type='image' name='submit' src='b_edit.png'></td>
<td><a href='?sec=sinavtakvimi&islem=sil&derskodu=$derskodu'>Sil</a></td></tr></form>";
    }
    echo "</table> Liste sonu... ";
} // func listele
function yenikaydet() {
if (!isset($_POST['submit'])) { listele(); exit; }
$derskodu=$_POST['derskodu']; $tarih=$_POST['tarih']; $saat=$_POST['saat'];
$tarih=substr($tarih,6,4)."-".substr($tarih,3,2)."-".substr($tarih,0,2);
vtablanvesec();
$s=@mysql_query("select * from `sinavtakvimi` where derskodu='$derskodu'");
if (@mysql_num_rows($s)>0)
{
    echo "$derskodu kodlu ders daha önceden kaydedildi!";
    listele();
    exit;
}
$s=@mysql_query("INSERT INTO `sinavtakvimi` ( `derskodu`, `tarih`, `saat` )
VALUES ('$derskodu', '$tarih', '$saat')");
if (!$s) echo "$derskodu kodlu ders eklenemedi!";
listele();
} // func yenikaydet

function degisiklikkaydet() {
$derskodu=$_GET['derskodu']; $sdk=$_POST['derskodu'];
$tarih=$_POST['tarih']; $saat=$_POST['saat'];
$tarih=substr($tarih,6,4)."-".substr($tarih,3,2)."-".substr($tarih,0,2);
vtablanvesec();
$s=mysql_query("select * from `sinavtakvimi` where derskodu='$sdk'");
if (@mysql_num_rows($s)==1 and ($sdk<>$derskodu))
{
    echo "$derskodu kodlu ders daha önceden kaydedildi!";
    listele();
    exit;
}
$s=@mysql_query("UPDATE `sinavtakvimi` SET `derskodu` = '$sdk', `tarih` = '$tarih', `saat` = '$saat'
WHERE `derskodu` = '$derskodu' ");
if (!$s) echo "$derskodu kodlu ders bilgileri değiştirilemedi!";
listele();
} // func degisiklikkaydet
function sil() {
if (!isset($_GET['derskodu'])) { listele(); exit; }
$derskodu=$_GET['derskodu'];
vtablanvesec();
$s=@mysql_query("DELETE FROM `sinavtakvimi` WHERE `derskodu` = '$derskodu' ");
if ($s) echo "$derskodu kodlu ders silindi..."; else echo "$derskodu kodlu ders silinemedi!";
listele();
} // func sil

?>
Sık sorulan sorular
<h3>Sık Sorulan Sorular</h3>
<hr>
| <a href='?sec=sss&islem=ekle'>Ekle</a> | <a href='?sec=sss&islem=listele'>Listele</a> |
<hr>
<p>
<?
if (!isset($_GET["islem"])) $islem="listele"; else $islem=$_GET["islem"];

switch ($islem) {
    case "listele" : listele(); break;
    case "ekle" : formgoster(); break;
    case "degistir" : formgoster(); break;
}

```

```

        case "yenikaydet"      : yenikaydet(); break;
        case "degisiklikkaydet" : degisiklikkaydet(); break;
        case "sil"            : sil(); break;
    }

function listele() {
    vtbaglanvesec();
    $liste=@mysql_query("select * from sss order by konu") or die("sss tablosuna ulařilamadı!");
    if (mysql_num_rows($liste)==0)
        echo "Kayıtlı SSS yok...";
    else
        {
            echo "<table border='1' cellspacing='0' cellpadding='5'><tr>
                <th>Sıra
No</th><th>Kodu</th><th>Konu</th><th>&nbsp;</th><th>&nbsp;</th></tr>";
            $x=0;
            while ($prg=mysql_fetch_array($liste))
                {
                    $x++;
                    $kodu=$prg[0]; $konu=$prg[1];
                    echo "<tr><td>$x</td><td>$kodu</td><td>$konu</td>
                        <td><a
href='?sec=sss&islem=degistir&kodu=$kodu'>Deđiřtir</a></td>
                        <td><a
href='?sec=sss&islem=sil&kodu=$kodu'>Sil</a></td></tr>";
                }
            echo "</table> Liste sonu... ";
        }
} // func listele

function formgoster() {
    if (!isset($_GET["kodu"])) $kodu="0"; else $kodu=$_GET["kodu"];
    if ($kodu=="0") {
        echo "<h4>Yeni SSS Ekleme</h4>";
        $konu=""; $metin=""; $islem="yenikaydet";
    }
    else
        {
            echo "<h4>SSS Deđiřikliđi</h4>";
            vtbaglanvesec();
            $kayitlar=@mysql_query("select * from sss where kodu='$kodu'") or die("SSS bilgisine
ulařilamadı...");
            if (mysql_num_rows($kayitlar)==0) die("SSS bilgisine ulařilamadı...");
            $kayit=mysql_fetch_array($kayitlar);
            $konu=$kayit[1]; $metin=$kayit[2]; $islem="degisiklikkaydet";
        }
    echo "<form name='form1' method='post' action='?sec=sss&islem=$islem&kodu=$kodu'>
        <table border='0' cellspacing='0' cellpadding='5'><tr>
        <td>SSS Kodu </td><td>";
    if ($kodu=="0") echo "Yeni"; else echo "$kodu";
    echo "</td></tr><tr><td>Konu</td>
        <td><input name='konu' type='text' id='konu' size='50' maxlength='50'
value='$konu'></td></tr>
        <tr><td valign='top'>SSS Cevap Metni</td>
        <td><textarea name='metin' cols='51' rows='10'>$metin</textarea></td>
        </tr><tr><td>&nbsp;</td><td><br><input type='submit' name='submit'
value='Kaydet'></td></tr></table></form>";
} // func formgoster

function yenikaydet() {
    if (!isset($_POST['submit'])) { formgoster(); exit; }
    $konu=$_POST['konu']; $metin=$_POST['metin'];
    vtbaglanvesec();
    $s=@mysql_query("INSERT INTO `sss` ( `kodu`, `konu`, `metin` )
VALUES ( '$konu', '$metin' );");
    if ($s) echo "$konu konulu SSS eklendi...";
}

```





```

} // func form
function kaydet() {
global $sayfa;
if (!isset($_POST['submit'])) { form(); exit; }
$sayfabasligi=$_POST['sayfabasligi']; $metin=$_POST['metin'];
vbaglanvesec();
$s=@mysql_query("UPDATE `statiksayfalar` SET `sayfabasligi` = '$sayfabasligi', `metin` = '$metin'
WHERE `sayfa` = '$sayfa'");
if ($s) echo "$sayfabasligi sayfası kaydedildi...";
else echo "$sayfabasligi sayfası kaydedilemedi!<br>
Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func kaydet
?>

```

Yönetim

<h3>Yönetim</h3>

<hr>

| <a href="?sec=yonetim&islem=ekle">Ekle</a> | <a href="?sec=yonetim&islem=listele">>Listele</a> |

<hr>

<p>

<?>

```

if (!isset($_GET["islem"])) $islem="listele"; else $islem=$_GET["islem"];

```

```

switch ($islem) {
case "listele" : listele(); break;
case "ekle" : formgoster(); break;
case "degistir" : formgoster(); break;
case "yenikaydet" : yenikaydet(); break;
case "degisiklikkaydet" : degisiklikkaydet(); break;
case "sil" : sil(); break;
}

```

```

function listele() {
vbaglanvesec();
$liste=@mysql_query("select * from yonetim order by sicilno") or die("Yönetim tablosuna
ulaşlamadı!");
if (mysql_num_rows($liste)==0)
echo "Kayıtlı yönetici yok...";
else
{
echo "<table border='1' cellspacing='0' cellpadding='5'><tr>
<th>Sıra No</th><th>Sicil
No</th><th>Ünvanı</th><th>Adı</th><th>Soyadı</th>
<th>Görevi</th><th>&nbsp;</th><th>&nbsp;</th></tr>";
$x=0;
while ($prg=mysql_fetch_array($liste))
{
$xicilno=$prg[0]; $sunvan=$prg[1]; $sadi=$prg[2];
$soyadi=$prg[3]; $gorevi=$prg[4];
echo
"<tr><td>$x</td><td>$sicilno</td><td>$sunvan</td><td>$sadi</td>
<td>$soyadi</td><td>$gorevi</td>
<td><a
href=?sec=yonetim&islem=degistir&sicilno=$sicilno>Değiştir</a></td>
<td><a
href=?sec=yonetim&islem=sil&sicilno=$sicilno>Sil</a></td></tr>";
}
echo "</table> Liste sonu... ";
}
} // func listele
function formgoster() {
if (!isset($_GET["sicilno"])) $sicilno="0"; else $sicilno=$_GET["sicilno"];
vbaglanvesec();
if ($sicilno=="0") {

```

```

echo "<h4>Yeni Yönetici Ekleme</h4>";
$unvan=""; $sadi=""; $soyadi=""; $gorevi=""; $detay=""; $islem="yenikaydet";
}
else
{
echo "<h4>Yönetici Bilgi Değişikliği</h4>";
$skayitlar=@mysql_query("select * from yonetim where sicilno='$sicilno'") or die("Yönetici
bilgisine ulaşamadı...");
if (mysql_num_rows($skayitlar)==0) die("Yönetici bilgisine ulaşamadı...");
$skayit=mysql_fetch_array($skayitlar);
$unvan=$skayit[1]; $sadi=$skayit[2]; $soyadi=$skayit[3]; $gorevi=$skayit[4]; $detay=$skayit[5];
$islem="degisiklikkaydet";
}
echo "<form name='form1' method='post' action='?sec=yonetim&islem=$islem&sicilno=$sicilno'>
<table border='0' cellspacing='0' cellpadding='5'><tr>
<td>Sicil No</td><td>";
if ($sicilno=="0") echo "<input name='sicilno' type='text' id='sicilno' maxlength='10'>"; else echo
"$sicilno";
echo "</td></tr><tr><td>Ünvanı </td>
<td><input name='unvan' type='text' id='unvan' size='25' maxlength='25'
value='$unvan'></td></tr>
<tr><td>Adı </td>
<td><input name='adi' type='text' id='adi' size='50' maxlength='50' value='$sadi'></td></tr>
<tr><td>Soyadı </td>
<td><input name='soyadi' type='text' id='soyadi' size='50' maxlength='50'
value='$soyadi'></td></tr>
<tr><td>Görevi</td>
<td><input name='gorevi' type='text' id='gorevi' size='50' maxlength='50'
value='$gorevi'></td></tr>
<tr><td valign='top'>Detaylı Bilgi</td>
<td><textarea name='detay' cols='51' rows='10'>$detay</textarea></td></tr>
<tr><td>&nbsp;</td><td><br><input type='submit' name='submit'
value='Kaydet'></td></tr>
</table></form>";
} // func formgoster
function yenikaydet() {
if (!isset($_POST['submit'])) { formgoster(); exit; }
$sicilno=$_POST['sicilno']; $unvan=$_POST['unvan']; $sadi=$_POST['adi']; $soyadi=$_POST['soyadi'];
$gorevi=$_POST['gorevi']; $detay=$_POST['detay'];
vtabaglanvesec();
$s=@mysql_query("INSERT INTO `yonetim` ( `sicilno`, `unvan`, `adi`, `soyadi`, `gorevi`, `detay` )
VALUES ('$sicilno', '$unvan', '$sadi', '$soyadi', '$gorevi', '$detay')");
if ($s) echo "$sicilno sicil numaralı yönetici eklendi...";
else echo "$sicilno sicil numaralı yönetici eklenemedi!<br>
Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func yenikaydet
function degisiklikkaydet() {
if (!isset($_POST['submit'])) { formgoster(); exit; }
$sicilno=$_GET['sicilno']; $unvan=$_POST['unvan']; $sadi=$_POST['adi']; $soyadi=$_POST['soyadi'];
$gorevi=$_POST['gorevi']; $detay=$_POST['detay'];
vtabaglanvesec();
$s=@mysql_query("UPDATE `yonetim` SET `unvan` = '$unvan', `adi` = '$sadi', `soyadi` = '$soyadi',
`gorevi` = '$gorevi',
`detay` = '$detay' WHERE `sicilno` = '$sicilno' ");
if ($s) echo "$sicilno sicil numaralı yönetici bilgileri değiştirildi...";
else echo "$sicilno sicil numaralı yönetici bilgileri değiştirilemedi!<br>
Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func degisiklikkaydet
function sil() {
if (!isset($_GET['sicilno'])) { listele(); exit; }
$sicilno=$_GET['sicilno'];
vtabaglanvesec();
$s=@mysql_query("DELETE FROM `yonetim` WHERE `sicilno` = '$sicilno' ");
if ($s) echo "$sicilno sicil numaralı yönetici silindi...";
}

```

```

else echo "$sicilno sicil numaralı yönetici silinemedi!<br>Lütfen <a
href=javascript:history.back()>geri</a> dönerek tekrar deneyiniz...";
} // func sil
?>

Yurtlar
<h3>Yurtlar</h3>
<hr>
| <a href="?sec=yurtlar&islem=ekle">Ekle</a> | <a href="?sec=yurtlar&islem=listele">Listele</a> |
<hr>
<p>
<?
if (!isset($_GET["islem"])) $islem="listele"; else $islem=$_GET["islem"];

switch ($islem) {
    case "listele"           : listele(); break;
    case "ekle"             : formgoster(); break;
    case "degistir"        : formgoster(); break;
    case "yenikaydet"      : yenikaydet(); break;
    case "degisiklikkaydet" : degisiklikkaydet(); break;
    case "sil"             : sil(); break;
}

function listele() {
    vtbaglanvesec();
    $liste=@mysql_query("select * from yurtlar order by adi") or die("Yurtlar tablosuna
ulaşılamadı!");
    if (mysql_num_rows($liste)==0)
        echo "Kayıtlı yurt yok...";
    else
        {
            echo "<table border='1' cellspacing='0' cellpadding='5'><tr>
                <th>Sıra
No</th><th>Kodu</th><th>Adı</th><th>&nbsp;</th><th>&nbsp;</th><th>&nbsp;</th></tr>";
            $x=0;
            while ($prg=mysql_fetch_array($liste))
                {
                    $x++;
                    $skodu=$prg[0]; $sadi=$prg[1];
                    echo "<tr><td>$x</td><td>$skodu</td><td>$sadi</td>
                        <td><a
href=?sec=yurtlar&islem=degistir&kodu=$skodu'>Değiştir</a></td>
                        <td><a
href=?sec=yurtlar&islem=sil&kodu=$skodu'>Sil</a></td></tr>";
                }
            echo "</table> Liste sonu... ";
        }
} // func listele
function formgoster() {
    if (!isset($_GET["kodu"])) $skodu="0"; else $skodu=$_GET["kodu"];
    if ($skodu=="0") {
        echo "<h4>Yeni Yurt Ekleme</h4>";
        $sadi=""; $metin=""; $islem="yenikaydet";
    }
    else
        {
            echo "<h4>Yurt Bilgi Değişikliği</h4>";
            vtbaglanvesec();
            $skayitlar=@mysql_query("select * from yurtlar where kodu='$skodu'") or die("Yurt bilgisine
ulaşılamadı...");
            if (mysql_num_rows($skayitlar)==0) die("Yurt bilgisine ulaşılamadı...");
            $skayit=mysql_fetch_array($skayitlar);
            $sadi=$skayit[1]; $metin=$skayit[2]; $islem="degisiklikkaydet";
        }
}

```

```

echo "<form name='form1' method='post' action='?sec=yurtlar&islem=$islem&kodu=$kodu'>
  <table border='0' cellspacing='0' cellpadding='5'><tr>
    <td>Yurt Kodu </td><td>";
if ($kodu=="0") echo "Yeni"; else echo "$kodu";
echo "</td></tr><tr><td>Yurt Adı </td>
      <td><input name='adi' type='text' id='adi' size='50' maxlength='50' value='Sadi'></td></tr>
      <tr><td valign='top'>Yurt Hakkında <br>Detaylı Bilgi </td>
      <td><textarea name='metin' cols='51' rows='10'>$metin</textarea></td>
    </tr><tr><td>&nbsp;</td><td><br><input type='submit' name='submit'
value='Kaydet'></td></tr></table></form>";
} // func formgoster
function yenikaydet() {
if (!isset($_POST['submit'])) { formgoster(); exit; }
$adi=$_POST['adi']; $metin=$_POST['metin'];
vtabaglanvesec();
$$=@mysql_query("INSERT INTO `yurtlar` ( `kodu` , `adi` , `metin` )
                VALUES (", '$adi', '$metin')");
if ($$) echo "$adi adlı yurt eklendi...";
else echo "$adi adlı yurt eklenemedi!<br>Lütfen <a href='javascript:history.back()'>geri</a> dönerek
tekrar deneyiniz...";
} // func yenikaydet
function degisiklikkaydet() {
if (!isset($_POST['submit'])) { formgoster(); exit; }
$kodu=$_GET['kodu']; $adi=$_POST['adi']; $metin=$_POST['metin'];
vtabaglanvesec();
$$=@mysql_query("UPDATE `yurtlar` SET `adi` = '$adi', `metin` = '$metin' WHERE `kodu` = '$kodu' ");
if ($$) echo "$kodu kodlu yurt bilgileri değiştirildi...";
else echo "$kodu kodlu yurt bilgileri değiştirilemedi!<br>
Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func degisiklikkaydet
function sil() {
if (!isset($_GET['kodu'])) { listele(); exit; }
$kodu=$_GET['kodu'];
vtabaglanvesec();
$$=@mysql_query("DELETE FROM `yurtlar` WHERE `kodu` = '$kodu' ");
if ($$) echo "$kodu kodlu yurt silindi...";
else echo "$kodu kodlu yurt silinemedi!<br>
Lütfen <a href='javascript:history.back()'>geri</a> dönerek tekrar deneyiniz...";
} // func sil
?>

```

Config  
<?

```

$dbserver      = "localhost";
$dbname        = "mobilogrenci";
$dbuser        = "";
$dbpass        = "";

$cpuser        = "dg";
$cppass        = "keçi";

```

```

function vtabaglanvesec() {
global $dbserver, $dbname, $dbuser, $dbpass;
$c=@mysql_connect($dbserver, $dbuser, $dbpass) or die("Veritabanı sunucusuna bağlanılamıyor!");
@mysql_select_db($dbname) or die("Veritabanı açılmadı!");
return $c;
}
function derslerlistekutusu($dk) {
vtabaglanvesec();
$liste=@mysql_query("select d.kodu, d.adi, d.programkodu, p.adi from dersler d, programlar p where
p.kodu=d.programkodu order by p.adi, d.adi") or die("Dersler tablosuna ulaşılamadı!");
$$="<select name='derskodu'>";
while ($ders=@mysql_fetch_array($liste)) {

```



```

INSERT INTO `dersler` VALUES ('222', '33333aaaa', 'btp');
INSERT INTO `dersler` VALUES ('ders2', 'ders2adi', 'oto');
INSERT INTO `dersler` VALUES ('ders3', 'ders3adi', 'btp');
# -----

#
# Tablo için tablo yapısı `duyurular`
#
# Creation: Haziran 14, 2006 at 12:42 PM
# Last update: Haziran 16, 2006 at 04:59 PM
#

CREATE TABLE `duyurular` (
  `kodu` int(11) NOT NULL auto_increment,
  `konu` varchar(50) NOT NULL default "",
  `metin` text NOT NULL,
  PRIMARY KEY (`kodu`)
) TYPE=MyISAM AUTO_INCREMENT=5 ;
#
# Tablo döküm verisi `duyurular`
#
INSERT INTO `duyurular` VALUES (1, 'denemeeee', 'akjlsdjklssda\r\nndsf\r\nnsdf\r\nnsadf');
INSERT INTO `duyurular` VALUES (3, 'duy1', 'duy1metin');
INSERT INTO `duyurular` VALUES (4, 'duy2', 'duy2metin');
# -----

#
# Tablo için tablo yapısı `haftaliktakvim`
#
# Creation: Haziran 15, 2006 at 10:22 PM
# Last update: Haziran 18, 2006 at 11:41 AM
#

CREATE TABLE `haftaliktakvim` (
  `derskodu` varchar(10) NOT NULL default "",
  `gun` varchar(10) NOT NULL default "",
  `baslangic` time NOT NULL default '00:00:00',
  `bitis` time NOT NULL default '00:00:00'
) TYPE=MyISAM;
#
# Tablo döküm verisi `haftaliktakvim`
#
INSERT INTO `haftaliktakvim` VALUES ('331814102', 'Salı', '11:11:00', '22:22:00');
INSERT INTO `haftaliktakvim` VALUES ('331814101', 'Çarşamba', '21:22:00', '11:22:00');
INSERT INTO `haftaliktakvim` VALUES ('222', 'Pazar', '00:22:00', '00:00:00');
INSERT INTO `haftaliktakvim` VALUES ('ders1', 'Cuma', '11:00:00', '15:00:00');
INSERT INTO `haftaliktakvim` VALUES ('ders2', 'Perşembe', '12:00:00', '16:00:00');
# -----

#
# Tablo için tablo yapısı `ogrenciler`
#
# Creation: Haziran 14, 2006 at 10:41 PM
# Last update: Haziran 18, 2006 at 12:41 PM
#

CREATE TABLE `ogrenciler` (
  `ogrencino` varchar(12) NOT NULL default "",
  `adi` varchar(50) NOT NULL default "",
  `soyadi` varchar(50) NOT NULL default "",
  `programkodu` varchar(10) NOT NULL default "",
  `sifre` varchar(10) NOT NULL default "",
  PRIMARY KEY (`ogrencino`)
) TYPE=MyISAM;
#
# Tablo döküm verisi `ogrenciler`
#

```

```

INSERT INTO `ogrenciler` VALUES ('200533181001', 'Demet', 'Gülseren', 'btp', '456');
INSERT INTO `ogrenciler` VALUES ('200533181002', 'Mustafa', 'Kayık', 'btp', '123');
INSERT INTO `ogrenciler` VALUES ('ogr1', 'ogr1adi', 'ogr1soyadi', 'btp', '1234');
INSERT INTO `ogrenciler` VALUES ('ogr2', 'ogr2adi', 'ogr2soyadi', 'oto', '2345');
# -----
#
# Tablo için tablo yapısı `personel`
#
# Creation: Haziran 14, 2006 at 12:36 PM
# Last update: Haziran 16, 2006 at 04:59 PM
#
CREATE TABLE `personel` (
  `sicilno` varchar(10) NOT NULL default "",
  `unvan` varchar(25) default NULL,
  `adi` varchar(50) NOT NULL default "",
  `soyadi` varchar(50) NOT NULL default "",
  `gorevi` varchar(50) default NULL,
  `detay` text NOT NULL,
  PRIMARY KEY (`sicilno`)
) TYPE=MyISAM;
#
# Tablo döküm verisi `personel`
#
INSERT INTO `personel` VALUES ('aaaaaaaaa', '11bbbbbbbbbbbb', '11cccccccccccccccc',
'11dddddddddddd', '11eeeeeeeeeee', 'fffffffr\nfffffr\n111111111r\n1');
INSERT INTO `personel` VALUES ('a123', 'asda', 'ad', 'soyad', 'görev', 'detay');
# -----
#
# Tablo için tablo yapısı `programlar`
#
# Creation: Haziran 14, 2006 at 12:30 PM
# Last update: Haziran 16, 2006 at 04:59 PM
#
CREATE TABLE `programlar` (
  `kodu` varchar(10) NOT NULL default "",
  `adi` varchar(50) NOT NULL default "",
  `detay` text NOT NULL,
  PRIMARY KEY (`kodu`)
) TYPE=MyISAM;
#
# Tablo döküm verisi `programlar`
#
INSERT INTO `programlar` VALUES ('btp', 'Bilgisayar Teknolojisi ve Programlama', 'btp detay');
INSERT INTO `programlar` VALUES ('muh', 'Muhasebe', 'muh detay');
INSERT INTO `programlar` VALUES ('ith-ih', 'İthalat-İhracat', 'ith detay');
INSERT INTO `programlar` VALUES ('oto', 'Otomotiv', 'oto detay');
INSERT INTO `programlar` VALUES ('mak', 'Makine', 'Mak detay');
INSERT INTO `programlar` VALUES ('buro', 'Büro Yönetimi ve Sekreterlik', 'büro detay');
INSERT INTO `programlar` VALUES ('hmo', 'Harita ve Maden Ölçme', 'harita detay');
INSERT INTO `programlar` VALUES ('eo', 'Endüstriyel Otomasyon', 'eo detay');
INSERT INTO `programlar` VALUES ('prg2', 'prg2adı', 'prg2detay');
INSERT INTO `programlar` VALUES ('prg1', 'prg1adı', 'prg1detay');
# -----
#
# Tablo için tablo yapısı `sinavsonuclari`
#
# Creation: Haziran 16, 2006 at 12:24 AM
# Last update: Haziran 16, 2006 at 04:29 PM
#
CREATE TABLE `sinavsonuclari` (
  `id` int(11) NOT NULL auto_increment,
  `ogrencino` varchar(12) NOT NULL default "",
  `derskodu` varchar(10) NOT NULL default "",
  `vize1` int(11) default NULL,

```

```

`vize2` int(11) default NULL,
`final` int(11) default NULL,
`ortalama` int(11) default NULL,
`harfnotu` char(2) default NULL,
PRIMARY KEY (`id`)
) TYPE=MyISAM AUTO_INCREMENT=9 ;
#
# Tablo döküm verisi `sinavsonuclari`
#
INSERT INTO `sinavsonuclari` VALUES (2, '200533181001', '331814101', 11, 22, 33, 44, '55');
INSERT INTO `sinavsonuclari` VALUES (3, '200533181001', '331814102', 111, 222, 333, 444, '55');
INSERT INTO `sinavsonuclari` VALUES (4, '200533181002', '331814101', 44, 55, 66, 77, '88');
INSERT INTO `sinavsonuclari` VALUES (5, '200533181002', '331814102', 222, 333, 444, 555, '66');
INSERT INTO `sinavsonuclari` VALUES (6, '200533181001', '222', 1, 2, 3, 4, '5');
INSERT INTO `sinavsonuclari` VALUES (7, '200533181002', '222', 1, 2, 3, 4, '5');
INSERT INTO `sinavsonuclari` VALUES (8, 'ogr1', 'ders2', 11, 22, 33, 44, 'dd');
# -----
#
# Tablo için tablo yapısı `sinavtakvimi`
#
# Creation: Haziran 14, 2006 at 12:40 PM
# Last update: Haziran 16, 2006 at 04:29 PM
#
CREATE TABLE `sinavtakvimi` (
  `derskodu` varchar(10) NOT NULL default "",
  `tarih` date NOT NULL default '0000-00-00',
  `saat` time NOT NULL default '00:00:00'
) TYPE=MyISAM;
#
# Tablo döküm verisi `sinavtakvimi`
#
INSERT INTO `sinavtakvimi` VALUES ('331814102', '2005-07-02', '11:22:00');
INSERT INTO `sinavtakvimi` VALUES ('331814101', '2006-06-01', '13:00:00');
INSERT INTO `sinavtakvimi` VALUES ('ders2', '2007-06-03', '16:00:00');
INSERT INTO `sinavtakvimi` VALUES ('ders1', '2007-06-02', '15:00:00');
# -----
#
# Tablo için tablo yapısı `sss`
#
# Creation: Haziran 14, 2006 at 12:43 PM
# Last update: Haziran 16, 2006 at 04:59 PM
#
CREATE TABLE `sss` (
  `kodu` int(11) NOT NULL auto_increment,
  `konu` varchar(50) NOT NULL default "",
  `metin` text NOT NULL,
  PRIMARY KEY (`kodu`)
) TYPE=MyISAM AUTO_INCREMENT=6 ;
#
# Tablo döküm verisi `sss`
#
INSERT INTO `sss` VALUES (1, 'deneme sssssssssssss', 'dkalsdjksd\r\nad\r\nas\r\nndssssssssssssss');
INSERT INTO `sss` VALUES (3, 'asdfsdfasdfsdf', 'asdfsdf\r\nsdf\r\nasd\r\nfnsdasdfsdf');
INSERT INTO `sss` VALUES (4, 'sss1', 'sss1 metin');
INSERT INTO `sss` VALUES (5, 'sss2', 'sss2 metin');
# -----
#
# Tablo için tablo yapısı `statiksayfalar`
#
# Creation: Haziran 15, 2006 at 10:09 PM
# Last update: Haziran 16, 2006 at 02:59 PM
#
CREATE TABLE `statiksayfalar` (
  `sayfa` varchar(20) NOT NULL default "",

```



```

`sayfabasligi` varchar(50) default NULL,
`metin` text NOT NULL,
PRIMARY KEY (`sayfa`)
) TYPE=MyISAM;
#
# Tablo döküm verisi `statiksayfalar`
#
INSERT INTO `statiksayfalar` VALUES ('okulumuzhakkinda', 'Okulumuz Hakkında',
'o\r\nh\r\nng\r\nb\r\nnaaaa');
INSERT INTO `statiksayfalar` VALUES ('iletisimbilgileri', 'İletişim Bilgileri', 'i\r\nb\r\n');
INSERT INTO `statiksayfalar` VALUES ('tavsanlihakkinda', 'Tavşanlı Hakkında', 't\r\nh\r\nnaaaa');
INSERT INTO `statiksayfalar` VALUES ('akademiktakvim', 'Akademik Takvim', 'aaa\r\nbb\r\nccc');
# -----
#
# Tablo için tablo yapısı `yonetim`
#
# Creation: Haziran 14, 2006 at 12:34 PM
# Last update: Haziran 16, 2006 at 04:33 PM
#
CREATE TABLE `yonetim` (
`sicilno` varchar(10) NOT NULL default "",
`unvan` varchar(25) default NULL,
`adi` varchar(50) NOT NULL default "",
`soyadi` varchar(50) NOT NULL default "",
`gorevi` varchar(50) default NULL,
`detay` text NOT NULL,
PRIMARY KEY (`sicilno`)
) TYPE=MyISAM;
#
# Tablo döküm verisi `yonetim`
#
INSERT INTO `yonetim` VALUES ('da0568', 'cco', 'mustafa', 'kayık', 'xx', 'ajklslş
sdf\r\nadsfjklasdşklf\r\nsad\r\naaaaaaaaaaa');
INSERT INTO `yonetim` VALUES ('asd', 'asd', 'asdsa', 'asdasdasd', 'asdas', 'asdas');
INSERT INTO `yonetim` VALUES ('da001', 'prof.dr.', 'birol', 'evli', 'müdür', 'maden mühendisi');
INSERT INTO `yonetim` VALUES ('da002', 'okt.', 'bülent', 'cıvık', 'mdr. yrd.', 'ti');
# -----
#
# Tablo için tablo yapısı `yurtlar`
#
# Creation: Haziran 14, 2006 at 12:38 PM
# Last update: Haziran 16, 2006 at 04:59 PM
#
CREATE TABLE `yurtlar` (
`kodu` int(11) NOT NULL auto_increment,
`adi` varchar(50) NOT NULL default "",
`metin` text NOT NULL,
PRIMARY KEY (`kodu`)
) TYPE=MyISAM AUTO_INCREMENT=7 ;
#
# Tablo döküm verisi `yurtlar`
#
INSERT INTO `yurtlar` VALUES (1, 'bimpaş', '');
INSERT INTO `yurtlar` VALUES (2, 'bbbbbbbbb', 'bbbbbbbbbbbbbbbbbb\r\nnadaslkidş');
INSERT INTO `yurtlar` VALUES (5, 'yurt1', 'yurt1 hakkında detaylı bilgi');
INSERT INTO `yurtlar` VALUES (6, 'yurt2', 'yurt2 hakkında detaylı bilgi');

```

## Ek-2 WAP Modülünün Program Kodları

```
//Akademik takvim//
<?
vtabaglanvesec();
$s=@mysql_query("select * from statiksayfalar where sayfa='akademiktakvim'");
if (!$s) {
    $title="Hata!";
    $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
}
else
{
    $kayit=mysql_fetch_array($s);
    $title=$kayit['sayfabasligi'];
    $content="<p>" . nl2br($kayit['metin']) . "<br /></p><do type='prev'
label='Geri'><prev/></do>";
}
?>
//Anasayfa//
<?
$title="DPÜ Tavřanlı MYO";
$content="
<p mode='wrap' align='center'>
<img src='images/logo.wbmp' alt='logo' /><br />
DPÜ Tavřanlı MYO <br />
Wap Sitesine <br />
Hořgeldiniz <br />
</p>
<p> &nbsp; </p>
<p>
<table align='center' columns='1'>
<tr><td><a href='?p=tanitim'>Okulumuz ve Tavřanlı</a></td></tr>
<tr><td><a href='?p=akademiktakvim'>Akademik Takvim</a></td></tr>
<tr><td><a href='?p=haftalik'>Haftalık Ders Programı</a></td></tr>
<tr><td><a href='?p=takvim'>Sınav Takvimi</a></td></tr>
<tr><td><a href='?p=sonuc'>Sınav Sonuçları</a></td></tr>
<tr><td><a href='?p=duyurular'>Duyurular</a></td></tr>
<tr><td><a href='?p=sss'>SSS</a></td></tr>
<tr><td><a href='?p=iletisim'>İletişim</a></td></tr>
</table>
</p>
<p> &nbsp; </p>
<p mode='wrap' align='center'>
(c) 2006
</p>
";
?>
//Duyurular//
<?
vtabaglanvesec();
$s=@mysql_query("select * from duyurular order by kodu DESC");
if (!$s) {
    $title="Hata!";
    $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
    exit;
}
$title="Duyurular";
$content="<p><table align='center' columns='1'>";
while ($kayit=mysql_fetch_array($s))
    $content.="<tr><td><a href='duyurulardetay.php?kod=".$kayit['kodu']."'>" .
    $kayit['konu'] . "</a></td></tr>";
$content.="</table><br /></p><do type='prev' label='Geri'><prev/></do>";
```

```

?>
//Duyurular detay//
<?
// send wml headers
header("Content-type: text/vnd.wap.wml");
echo "<?xml version='1.0'?'>\n";
echo '<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3/TR". "\n\t";
echo "'http://www.wapforum.org/DTD/wml13.dtd">';
?>
<wml>
<?
$title="";
$content="";
include "library.inc.php";
include "..\kontrolpanel\config.inc.php";
$kod=@$_GET['kod'];
if (!isset($kod))
    {
        $title="Hata!";
        $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
    }
else
    {
        vtbaglanvesec();
        $s=@mysql_query("select * from duyurular where kodu='$kod'");
        if (!$s) {
            $title="Hata!";
            $content="<p>Hata oluřtu!<br /></p><do type='prev'
label='Geri'><prev/></do>";
        }
        else {
            $kayit=@mysql_fetch_array($s);
            $title="Duyuru Detayı";
            $content = "<p>". $kayit['konu'] . "</p><p>&nbsp;</p><p>";
nl2br($kayit['metin']);
            $content .= "<br /></p><do type='prev' label='Geri'><prev/></do>";
        }
    }
}
$W_TITLE=TurkceHarfleriAt($title);
$W_ENCODINGTYPE=0;
?>
<card id="main" title="<? echo $W_TITLE ?>">
<?
print TurkceHarfleriAt($content);
?>
</card>
</wml>

//Haftalık ders programı//
<?
$title="Haftalık Ders Programı";
$content="<p>Ders Programını Görmek İstedığınız Programı Seçiniz:<br/>";
$content.=programlarlistekutusu_wml();
$content.="</p>";
<do type='accept' label='Göster'>
    <go href='haftalikgoster.php?kod=$(kod)'/>
</do>
<do type='prev' label='Geri'><prev/></do>";
?>

//Haftalık göster//
<?
// send wml headers
header("Content-type: text/vnd.wap.wml");

```

```

echo "<?xml version='1.0'?'>\n";
echo '<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3/TR". "\n\t";
echo "'http://www.wapforum.org/DTD/wml13.dtd">';
?>
<wml>
<?
$title="";
$content="";

include "library.inc.php";
include "..\kontrolpanel\config.inc.php";

$skod=@$_GET['kod'];
if (!isset($skod))
{
    $title="Hata!";
    $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
}
else
{
    vtbaglanvesec();
    $s=@mysql_query("SELECT h.derskodu, d.adi, h.gun, h.baslangic, h.bitis FROM dersler
d, haftaliktakvim h
WHERE d.kodu = h.derskodu AND d.programkodu = '$skod'");
    if (!$s) {
        $title="Hata!";
        $content="<p>Hata oluřtu!<br /></p><do type='prev'
label='Geri'><prev/></do>";
    }
    else {
        $title="Haftalık Ders Programı";
        $content="<p><table align='center' columns='2'>";
        while ($kayit=@mysql_fetch_array($s))
            $content .= "<tr><td>" . $kayit[0] . "<br />" . $kayit[1] .
"</td><td>" .
            $kayit[2] . "<br />" . substr($kayit[3],0,5) . "<br />" .
substr($kayit[4],0,5) . "</td></tr>";
        $content .= "</table></p><do type='prev' label='Geri'><prev/></do>";
        if (@mysql_num_rows($s)==0) $content = "<p>Kayıt bulunamadı...<br
/></p><do type='prev' label='Geri'><prev/></do>";
    }
}
$W_TITLE=TurkceHarfleriAt($title);
$W_ENCODINGTYPE=0;
?>
<card id="main" title="<? echo $W_TITLE ?>">
<?
print TurkceHarfleriAt($content);
?>
</card>
</wml>
//İletişim//
<?
vtbaglanvesec();
$s=@mysql_query("select * from statiksayfalar where sayfa='iletisimbilgileri'");
if (!$s) {
    $title="Hata!";
    $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
}
else
{
    $kayit=mysql_fetch_array($s);
    $title=$kayit['sayfabasligi'];
}
}

```

```

        $content="<p>" . nl2br($kayit['metin']) . "<br /></p><do type='prev'
label='Geri'><prev/></do>";
    }

?>
//İndex//
<?
// send wml headers
header("Content-type: text/vnd.wap.wml");
echo "<?xml version='1.0'?'>\n";
echo '<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3/TR". "\n\t";
echo "'http://www.wapforum.org/DTD/wml13.dtd">';
?>
<wml>
<?
$title="";
$content="";

include "library.inc.php";
include "..\kontrolpanel\config.inc.php";

$sp=@$_GET['p'];
if (!isset($sp)) $sp="anasayfa";

switch ($sp) {
    case "anasayfa": include "anasayfa.php"; break; //ok
    case "tanitim": include "tanitim.php"; break; //ok
    case "akademiktakvim": include "akademiktakvim.php"; break; //ok
    case "haftalik" : include "haftalik.php"; break; //
    case "takvim" : include "takvim.php"; break; //
    case "sonuc" : include "sonuc.php"; break; //
    case "duyurular" : include "duyurular.php"; break; //ok
    case "sss": include "sss.php"; break; //ok
    case "iletisim" : include "iletisim.php"; break; //ok
    case "okulumuzhakkinda" : include "okulumuzhakkinda.php"; break; //ok
    case "programlar" : include "programlar.php"; break; //ok
    case "yonetim" : include "yonetim.php"; break; //ok
    case "personel" : include "personel.php"; break; //ok
    case "tavsanlihakkinda" : include "tavsanlihakkinda.php"; break; //ok
    case "yurtlar" : include "yurtlar.php"; break; //ok
}
$W_TITLE=TurkceHarfleriAt($title);
$W_ENCODINGTYPE=0;
?>
<card id="main" title="<? echo $W_TITLE ?>">
<?
print TurkceHarfleriAt($content);
?>
</card>
</wml>
//Library.inc//
<?php

function TurkceHarfleriAt($s)
{
    $trchars = "ÜĞİŞÇÖüğşçöi";
    $enchars = "UGISCOugscoi";
    $r = "";
    for ($x=0; $x<=(strlen($s)-1); $x++)
    {
        $c = $s{$x};
        $p = strpos($trchars, $c);
        if (!$p ===false) $c = $enchars{$p};
        $r .= $c;
    }
}

```

```

    }
    return $r;
}
function programlarlistekutusu_wml() {
vtbodybaglanvesec();
$liste=@mysql_query("select * from programlar order by adi");
$$s='<select name="kod" title="Program Seçiniz">';
while ($prg=@mysql_fetch_array($liste))
    $$s .= '<option value="'. $prg['kodu'] ."'>'. $prg['adi'] . '</option>';
$$s .= '</select>';
return $$s;
}

?>
//Okulumuz hakkında//
<?
vtbodybaglanvesec();
$$s=@mysql_query("select * from statiksayfalar where sayfa='okulumuzhakkinda");
if (!$s) {
    $title="Hata!";
    $content="<p>Hata oluştu!<br /></p><do type='prev' label='Geri'><prev/></do>";
}
else
    {
    $kayit=mysql_fetch_array($s);
    $title=$kayit['sayfabasligi'];
    $content="<p>" . nl2br($kayit['metin']) . "<br /></p><do type='prev'
label='Geri'><prev/></do>";
    }

?>
//Personel//
<?
vtbodybaglanvesec();
$$s=@mysql_query("select * from personel");
if (!$s) {
    $title="Hata!";
    $content="<p>Hata oluştu!<br /></p><do type='prev' label='Geri'><prev/></do>";
    exit;
}

$title="Personel";

$content="<p><table align='center' columns='1'>";
while ($kayit=mysql_fetch_array($s))
    $content .= "<tr><td><a href='personeldetay.php?kod=". $kayit['sicilno'] ."'>".
    $kayit['unvan'] . " " . $kayit['adi'] . " " . $kayit['soyadi'] .
"</a></td></tr>";
$content .= "</table><br /></p><do type='prev' label='Geri'><prev/></do>";

?>
//Personel detay//
<?
vtbodybaglanvesec();
$$s=@mysql_query("select * from personel");
if (!$s) {
    $title="Hata!";
    $content="<p>Hata oluştu!<br /></p><do type='prev' label='Geri'><prev/></do>";
    exit;
}

$title="Personel";

$content="<p><table align='center' columns='1'>";
while ($kayit=mysql_fetch_array($s))

```

```

                $content .= "<tr><td><a href='personeldetay.php?kod=".$Skayit['sicilno']. ">".
                    $kayit['unvan'] . " " . $kayit['adi'] . " " . $kayit['soyadi'] .
"</a></td></tr>";
                $content .= "</table><br /></p><do type='prev' label='Geri'><prev/></do>";
?>
//Programlar//
<?
    vtbaglanvesec();
    $s=@mysql_query("select * from programlar order by adi");
    if (!$s) {
        $title="Hata!";
        $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
        exit;
    }

    $title="Programlar";

    $content="<p><table align='center' columns='1'>";
    while ($kayit=mysql_fetch_array($s))
        $content .= "<tr><td><a href='programlardetay.php?kod=".$kayit['kodu']. ">".
$Skayit['adi'] . "</a></td></tr>";
    $content .= "</table><br /></p><do type='prev' label='Geri'><prev/></do>";
?>
//Programlar detay//
<?
// send wml headers
header("Content-type: text/vnd.wap.wml");
echo "<?xml version='1.0'>\n";
echo '<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3/TR". "\n\t";
echo "'http://www.wapforum.org/DTD/wml13.dtd">';
?>
<wml>
<?
$title="";
$content="";

include "library.inc.php";
include "..\kontrolpanel\config.inc.php";

$skod=@$_GET['kod'];
if (!isset($skod))
    {
        $title="Hata!";
        $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
    }
else
    {
        vtbaglanvesec();
        $s=@mysql_query("select * from programlar where kodu='$skod'");
        if (!$s) {
            $title="Hata!";
            $content="<p>Hata oluřtu!<br /></p><do type='prev'
label='Geri'><prev/></do>";
        }
        else {
            $kayit=@mysql_fetch_array($s);
            $title="Program Detay Bilgileri";
            $content="<p>". $kayit['adi'] . "</p><p>&nbsp;</p><p>".
nl2br($kayit['detay']);

            $content .= "<br /></p><do type='prev' label='Geri'><prev/></do>";
        }
    }
}
$W_TITLE=TurkceHarfleriAt($title);
$W_ENCODINGTYPE=0;

```

```

?>
<card id="main" title="<? echo $W_TITLE ?>">
<?
print TurkceHarfleriAt($content);
?>
</card>
</wml>

//Sonuç//
<?
$title="Sınav Sonuçları";
$content="<p>Numaranızı ve Şifrenizi Giriniz<br/></p><p>&nbsp;</p>
<p>Numaranız:<br />
<input type='text' name='ogrencino' maxlength='12' format='NNNNNNNNNNNN' /><br/>
Şifreniz:<br />
<input type='password' name='sifre' maxlength='10' /><br/></p>

<do type='accept' label='Göster'>
<go href='sonucgoster.php' method='post'>
    <postfield name='ogrencino' value='$(ogrencino)' />
    <postfield name='sifre' value='$(sifre)' />
</go>
</do>
<do type='prev' label='Geri'><prev/></do>";
?>
//Sonuçgöster//
<?
// send wml headers
header("Content-type: text/vnd.wap.wml");
echo "<?xml version='1.0'?'>\n";
echo '<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3/TR". " \n\t";
echo "'http://www.wapforum.org/DTD/wml13.dtd">';
?>
<wml>
<?
$ogrencino=@$_POST['ogrencino'];
$sifre=@$_POST['sifre'];
$title="Sınav Sonuçları";
$content="";
include "library.inc.php";
include "..\kontrolpanel\config.inc.php";
if ((!isset($ogrencino)) or (!isset($sifre)))
    {
        $title="Hata!";
        $content="<p>Hata oluştu!<br /></p><do type='prev' label='Geri'><prev/></do>";
    }
else
    {
        vtbaglanvesec();
        $s=@mysql_query("SELECT * from ogrenciler WHERE ogrencino='$ogrencino' and
sifre='$sifre'");
        if ((!$s) or (@mysql_num_rows($s)==0)) {
            $title="Hata!";
            $content="<p>Hata oluştu! Numara ve şifrenizi kontrol edin...<br /></p><do
type='prev' label='Geri'><prev/></do>";
        }
        else {
            $title="Sınav Sonuçları";
            $ogrenci=@mysql_fetch_array($s);
            $content="<p>". $ogrenci['ogrencino']. "<br/>". $ogrenci['adi']. "
". $ogrenci['soyadi']. "<br/>";
            $s=@mysql_query("SELECT d.adi, ss.vize1, ss.vize2, ss.final, ss.ortalama,
ss.harfnotu, ss.derskodu
from sinavsonuclari ss, dersler d

```



```

WHERE ss.ogrencino='$ogrencino' and ss.derskodu=d.kodu
                                order by d.adi");
if (!(!$s) or (@mysql_num_rows($s)==0)) {
    $stitle="Hata!";
    $scontent="<p>Hata oluřtu!<br /></p><do type='prev'
label='Geri'><prev/></do>";
    }
else {
    $scontent .="<table align='center' columns='2'>
<tr><td>Ders Kodu<br/>Ders Adı</td>
<td>Vize1<br/>Vize2<br/>Final<br/>Ortalama<br/>Harf Notu</td></tr>";
    while ($kayit=@mysql_fetch_array($s))
        $scontent
        .="<tr><td>".$kayit[6]."<br/>".$kayit[0]."</td><td>v1: ".$kayit[1]."<br/>v2: ".
        $kayit[2]."<br />f: ".$kayit[3]."<br
        />o: ".$kayit[4]."<br />hn: ".$kayit[5]."</td></tr>";
    $scontent .="</table></p><do type='prev'
label='Geri'><prev/></do>";
    if (@mysql_num_rows($s)==0) $scontent ="<p>Kayıt
    bulunamadı...<br /></p><do type='prev' label='Geri'><prev/></do>";
    }
    }

$W_TITLE=TurkceHarfleriAt($stitle);
$W_ENCODINGTYPE=0;
?>
<card id="main" title="<? echo $W_TITLE ?>">
<?
print TurkceHarfleriAt($scontent);
?>
</card>
</wml>
//Sss//
<?
    vtbaglanvesec();
    $s=@mysql_query("select * from sss order by konu");
    if (!$s) {
        $stitle="Hata!";
        $scontent="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
        exit;
    }

    $stitle="Sık Sorulan Sorular";

    $scontent ="<p><table align='center' columns='1'>";
    while ($kayit=mysql_fetch_array($s))
        $scontent .="<tr><td><a href='sssdetay.php?kod=".$kayit['kodu']. "'> ". $kayit['konu'] .
"</a></td></tr>";
    $scontent .="</table><br /></p><do type='prev' label='Geri'><prev/></do>";
?>
//Sssdetay//
<?
// send wml headers
header("Content-type: text/vnd.wap.wml");
echo "<?xml version='1.0'>\n";
echo '<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3/TR". "n\nt";
echo "'http://www.wapforum.org/DTD/wml13.dtd">';
?>
<wml>
<?
$stitle="";
$scontent="";

```

```

include "library.inc.php";
include "..\kontrolpanel\config.inc.php";

$kod=@$_GET['kod'];
if (!isset($kod))
    {
        $title="Hata!";
        $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
    }
else
    {
        vtbaglanvesec();
        $s=@mysql_query("select * from sss where kodu='$kod'");
        if (!$s) {
            $title="Hata!";
            $content="<p>Hata oluřtu!<br /></p><do type='prev'
label='Geri'><prev/></do>";
        }
        else
            {
                $kayit=@mysql_fetch_array($s);
                $title="SSS Detayı";
                $content = "<p>. $kayit['konu'] . "</p><p>&nbsp;</p><p>".
nl2br($kayit['metin']);
                $content .= "<br /></p><do type='prev' label='Geri'><prev/></do>";
            }
    }

$W_TITLE=TurkceHarfleriAt($title);
$W_ENCODINGTYPE=0;
?>
<card id="main" title="<? echo $W_TITLE ?>">
<?
print TurkceHarfleriAt($content);
?>
</card>
</wml>

//Takvim//
<?
$title="Sınav Takvimi";
$content="<p>Sınav Takvimini Görmek İstedığınız Programı Seçiniz:<br/>";
$content.=programlarlistekutusu_wml();
$content.="</p>";
<do type='accept' label='Göster'>
    <go href='takvimgoster.php?kod=$(kod)'/>
</do>
<do type='prev' label='Geri'><prev/></do>";
?>
//Takvim göster//
<?
// send wml headers
header("Content-type: text/vnd.wap.wml");
echo "<?xml version='1.0'?'>\n";
echo '<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3/TR". "\n\t";
echo "'http://www.wapforum.org/DTD/wml13.dtd">';
?>
<wml>
<?
$title="";
$content="";
include "library.inc.php";
include "..\kontrolpanel\config.inc.php";

```

```

$kod=@$_GET['kod'];
if (!isset($kod))
    {
        $title="Hata!";
        $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
    }
else
    {
        vtbaglanvesec();
        $s=@mysql_query("SELECT d.kodu, d.adi, st.tarih, st.saat FROM dersler d, sinavtakvimi
st
                                WHERE d.kodu = st.derskodu AND
d.programkodu = '$kod'");
        if (!$s) {
            $title="Hata!";
            $content="<p>Hata oluřtu!<br /></p><do type='prev'
label='Geri'><prev/></do>";
        }
        else {
            $title="Sinav Takvimi";
            $content="<p><table align='center' columns='4'>";
            while ($kayit=@mysql_fetch_array($s))
                $content.="<tr><td>". $kayit[0] . "</td><td>". $kayit[1]
. "</td><td>".
                substr($kayit[2],8,2) .
                ".</td><td>". $kayit[2]. "</td><td>".
                substr($kayit[3],0,5) . "</td></tr>";
            $content.="</table><p><do type='prev' label='Geri'><prev/></do>";
            if (@mysql_num_rows($s)==0) $content="<p>Kayıt bulunamadı...<br
/></p><do type='prev' label='Geri'><prev/></do>";
        }
    }
}
$W_TITLE=TurkceHarfleriAt($title);
$W_ENCODINGTYPE=0;
?>
<card id="main" title="<? echo $W_TITLE ?>">
<?
print TurkceHarfleriAt($content);
?>
</card>
</wml>
Tanıtım
<?
$title="Okulumuz ve Tavşanlı";
$content="
<p>&nbsp;</p>
<p>
<table align='center' columns='1'>
<tr><td><a href='?p=okulumuzhakkında'>Okulumuz Hakkında</a></td></tr>
<tr><td><a href='?p=programlar'>Programlar</a></td></tr>
<tr><td><a href='?p=yonetim'>Yönetim</a></td></tr>
<tr><td><a href='?p=personel'>Personel</a></td></tr>
<tr><td><a href='?p=tavşanlıhakkında'>Tavşanlı Hakkında</a></td></tr>
<tr><td><a href='?p=yurtlar'>Yurtlar</a></td></tr>
</table>
</p>
<do type='prev' label='Geri'><prev/></do>
";
?>
//Tavşanlı hakkında//
<?
vtbaglanvesec();

```

```

$s=@mysql_query("select * from statiksayfalar where sayfa='tavsanlihakkinda");
if (!$s) {
    $title="Hata!";
    $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
}
else
{
    $kayit=mysql_fetch_array($s);
    $title=$kayit['sayfabasligi'];
    $content="<p>" . nl2br($kayit['metin']) . "<br /></p><do type='prev'
label='Geri'><prev/></do>";
}
?>
//Yönetim//
<?
    vtbaglanvesec();
    $s=@mysql_query("select * from yonetim");
    if (!$s) {
        $title="Hata!";
        $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
        exit;
    }
    $title="Yönetim";

    $content="<p><table align='center' columns='1'>";
    while ($kayit=mysql_fetch_array($s))
        $content .= "<tr><td><a href='yonetimdetay.php?kod=".$kayit['sicilno']. ">" .
            $kayit['unvan'] . " " . $kayit['adi'] . " " . $kayit['soyadi'] .
"</a></td></tr>";
    $content .= "</table><br /></p><do type='prev' label='Geri'><prev/></do>";
?>
//Yönetim detay//
<?
// send wml headers
header("Content-type: text/vnd.wap.wml");
echo "<?xml version='1.0'>\n";
echo "<!DOCTYPE wml PUBLIC '-//WAPFORUM//DTD WML 1.3/TR' '\n\t";
echo "'http://www.wapforum.org/DTD/wml13.dtd'>";
?>
<wml>
<?
$title="";
$content="";

include "library.inc.php";
include "..\kontrolpanel\config.inc.php";

$kod=@$_GET['kod'];
if (!isset($kod))
{
    $title="Hata!";
    $content="<p>Hata oluřtu!<br /></p><do type='prev' label='Geri'><prev/></do>";
}
else
{
    vtbaglanvesec();
    $s=@mysql_query("select * from yonetim where sicilno='$kod'");
    if (!$s) {
        $title="Hata!";
        $content="<p>Hata oluřtu!<br /></p><do type='prev'
label='Geri'><prev/></do>";
    }
    else {
        $kayit=@mysql_fetch_array($s);

```

```

        $stitle="Yönetici Detay Bilgileri";
        $scontent = "<p>". $kayit['unvan'] . "<br />". $kayit['adi'] . " " . $kayit['soyadi']
. "<br />". $kayit['gorevi'] .
        " </p><p>&nbsp;</p><p>". nl2br($kayit['detay']);
        $scontent .= "<br /></p><do type='prev' label='Geri'><prev/></do>";
    }
}

$W_TITLE=TurkceHarfleriAt($stitle);
$W_ENCODINGTYPE=0;
?>
<card id="main" title="<? echo $W_TITLE ?>">
<?
print TurkceHarfleriAt($scontent);
?>
</card>
</wml>

//Yurtlar//
<?
vtabaglanvesec();
$s=@mysql_query("select * from yurtlar order by adi");
if (!$s) {
    $stitle="Hata!";
    $scontent="<p>Hata oluştu!<br /></p><do type='prev' label='Geri'><prev/></do>";
    exit;
}
$stitle="Yurtlar";
$scontent = "<p><table align='center' columns='1'>";
while ($kayit=mysql_fetch_array($s))
    $scontent .= "<tr><td><a href='yurtlardetay.php?kod='". $kayit['kodu'] . "'>".
$stitle['adi'] . "</a></td></tr>";
    $scontent .= "</table><br /></p><do type='prev' label='Geri'><prev/></do>";
?>
//Yurtlar detay//
<?
// send wml headers
header("Content-type: text/vnd.wap.wml");
echo "<?xml version='1.0'>\n";
echo "<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3/TR". "\n\t";
echo "'http://www.wapforum.org/DTD/wml13.dtd'>";
?>
<wml>
<?
$stitle="";
$scontent="";
include "library.inc.php";
include "..\kontrolpanel\config.inc.php";
$skod=@$_GET['kod'];
if (!isset($skod))
    {
        $stitle="Hata!";
        $scontent="<p>Hata oluştu!<br /></p><do type='prev' label='Geri'><prev/></do>";
    }
else
    {
        vtabaglanvesec();
        $s=@mysql_query("select * from yurtlar where kodu='$skod'");
        if (!$s) {
            $stitle="Hata!";
            $scontent="<p>Hata oluştu!<br /></p><do type='prev'
label='Geri'><prev/></do>";
        }
        else {

```

```

        $kayit=@mysql_fetch_array($s);
        $stitle="Yurt Detay Bilgileri";
        $scontent ="<p>" . $kayit['adi'] . "</p><p>&nbsp;</p><p>".
nl2br($kayit['metin']);
        $scontent .= "<br /></p><do type='prev' label='Geri'><prev/></do>";
    }
}
$W_TITLE=TurkceHarfleriAt($stitle);
$W_ENCODINGTYPE=0;
?>
<card id="main" title="<? echo $W_TITLE ?>">
<?
print TurkceHarfleriAt($scontent);
?>
</card>
</wml>

```