KAHRAMANMARAŞ SÜTÇÜ İMAM ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK-ELEKRONİK MÜHENDİSLİĞİ ANABİLİM DALI

3. NESİL GEZGİN TELEFONLAR ÜZERİNDE ÇALIŞAN AĞ/İNTERNET TABANLI UYGULAMALARIN GÜVENLİĞİNİN ARTIRILMASI

MUSTAFA KARABULUT

YÜKSEK LİSANS TEZİ

KAHRAMANMARAŞ
Şubat - 2007

T.C.
KAHRAMANMARAŞ SÜTÇÜ İMAM ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK-ELEKTRONİK ANABİLİM DALI

3. NESİL GEZGİN TELEFONLAR ÜZERİNDE ÇALIŞAN AĞ/İNTERNET TABANLI UYGULAMALARIN GÜVENLİĞİNİN ARTIRILMASI

MUSTAFA KARABULUT

YÜKSEK LİSANS TEZİ

Kod No:

Bu tez 06/02/2007 Tarihinde Aşağıdaki Jüri Üyeleri Tarafından
Oy Birliği ile Kabul Edilmiştir.

Yrd.Doç.Dr	**Prof. Dr.**	**Yrd. Doç. Dr.**
Abdulhamit SUBAŞI	**M.Kemal KIYMIK**	**Metin ARTIKLAR**
DANIŞMAN	**ÜYE**	**ÜYE**

Yukarıdaki imzaların adı geçen öğretim üyelerine ait olduğunu onaylarım.

Prof. Dr. Özden GÖRÜCÜ
Enstitü Müdürü

Proje No: 2005/1-19

İÇİNDEKİLER

KAHRAMANMARAŞ SÜTÇÜ İMAM ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ

YÜKSEK LİSANS TEZİ

ÖZET

3. NESİL GEZGİN TELEFONLAR ÜZERİNDE ÇALIŞAN AĞ/İNTERNET TABANLI UYGULAMALARIN GÜVENLİĞİNİN ARTIRILMASI

MUSTAFA KARABULUT

Danışman: Yrd.Doç. Dr. Abdülhamit SUBAŞI

Yıl: 2007 Sayfa: 62

Jüri : Yrd.Doç. Dr. Abdülhamit SUBAŞI
: Prof. Dr. M. Kemal KIYMIK
: Yrd.Doç.Dr. Metin ARTIKLAR

İnternet kullanımı yaygınlaştıkça internete katılan cihaz türlerinin sayısıda gün geçtikçe artmaktadır. Başlangıçta sadece PC'lerin oluşturduğu bir istemci topluluğu içinde artık yaygın olarak gezgin telefonlarda bulunmaktadır. İnternette güvenlik ile ilgili uygulamalar ve yöntemler genelde PC tabanlı sistemler için düşünülse de PC tabanlı sistemler işletim sistemi, donanım gibi bazı yönlerden birbirlerinden ayrılabildikleri için yine de istemcinin türünden bir soyutlama söz konusudur. Fakat bir gezgin cihazın üzerinde bir PC'ye göre avantajlı olduğu durumlar bu güvenlik uygulamalarında elbette kullanılmamıştır. Bu cihazların sahip olduğu bazı avantajları kendi iletişim güvenliklerini artırmak ve bazı dezavantajları bertaraf etmek için bazı yöntemler ve yollar araştırılmıştır.

Gezgin cihazların zikredilmesi gereken avantajları donanımsal olarak dahili gelen fotoğraf çekme, ses kaydetme özellikleri yanı sıra Bluetooth gibi kablosuz bağlantı seçenekleri olarak özetlenebilir. Elbette bu özellikler her geçen gün artmaktadır.

Genel olarak çalışmamızda günümüzdeki gezgin cihazlarda gittikçe daha yaygın hale gelen resim çekme gibi özelliklere yoğunlaşıp, bu özellikleri kullanarak güvenliğin artırılabilirliğini araştırdık.

Anahtar kelimeler : Gezgin cihazlar, kimlik doğrulama, güvenlik, resim tabanlı güvenlik

UNIVERSITY OF KAHRAMANMARAŞ SÜTÇÜ İMAM
INSTITUTE OF NATURAL AND APPLIED SCIENCES
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

MSc THESIS

ABSTRACT

IMPROVEMENT OF INTERNET/NETWORK BASED APPLICATIONS' SECURITY ON 3rd GENERATION MOBILE PHONES

MUSTAFA KARABULUT

Supervisor: Assist. Prof. Dr. Abdülhamit SUBAŞI

Year: 2007 Pages: 62

Jury : Assist. Prof. Dr. Abdülhamit SUBAŞI
: Prof. Dr. M. Kemal KIYMIK
: Assist. Prof. Dr. Metin ARTIKLAR

As the Internet evolving , the number of device types connected to the Internet is increasing continuosly. At the very beginning, Internet was meant to be a network for desktop computers as clients, but currently mobile devices are growing to be a popular means to connect to Internet. Though the security infrastructure of the Internet was constructed mainly for desktop computers and seems to be inflexible apparently, a great range of varying operating systems, hardware types forced the infrastructure of Internet have a kind of abstraction for different kinds of clients. But it is for sure the advantages of a mobile device over a desktop machine wasn't properly considered for the security applications. We've researched methods and ways of making a more secure application of mobile devices by using their advantages and considering their disadvantages as well.

If we should mention some advantages of the mobile devices, they may be briefly listed as their built-in hardware support for taking photos and recording voice and also the built-in capability to make wireless connections through some protocols such as Bluetooth. Of course, by the evolving technology, new features are added to the list continuously.

We mainly focused on the built-in photograph capturing capability of the mobile devices. We showed more secure applications could be built using such features.

Keywords : Mobile devices, authentication, security, image-based authentication.

ÖNSÖZ

Gezgin cihazların her geçen gün daha yaygın hale gelmesi ile bu cihazlar için gerekli güvenlik alt yapısı üzerinde daha çok çalışma ihtiyacı hissediliyor. Internet artık sadece masaüstü bilgisayarların bağlandığı bir ağ değil, bu yüzden yeni cihazlar için farklı güvenlik sistemleri, alt yapısı geliştirmek gerekiyor.

Çalışmamızın ana çıkış noktası da bu oldu; artık bu cihazlar için özelleşmiş güvenlik artırım tekniklerini nasıl yapabiliriz, nasıl becerebiliriz sorusunu sorduk öncelikle.

Sonuç olarak da cep telefonları üzerinde kimlik doğrulama işlemini bir masaüstü bilgisayardan nasıl farklı yapabiliriz konusu üzerinde yoğunlaştık. Pek çok teknolojiyi inceleyerek özellikle yüz tanıma yardımıyla güvenliğin artırılabileceğini gösterdik.

Çalışma sırasında sürekli yardımları, akılcı yönlendirmeleri ile tezin oluşuma katkı sağlayan değerli hocam Yrd.Doç.Dr. Abdulhamit SUBAŞI'na teşekkürü bir borç bilirim.

Sürekli olarak yanımda olan ve fikirleri ile bana destek olan değerli arkadaşım Erdal DAYAK'a da teşekkür etmek isterim.

Bunun dışında da isimlerini burada zikredemeyeceğim maddi, manevi her alanda destek olan tüm dostlarıma teşekkür etmek istiyorum.

Ocak 2007

KAHRAMANMARAŞ **MUSTAFA KARABULUT**

ÇİZELGELER DİZİNİ

ŞEKİLLER DİZİNİ

SİMGELER VE KISALTMALAR DİZİNİ

1G : First Generation
2G : Second Generation
3D : Three Dimension
3G : Third Generation
4G : Fourth Generation
ADSL : Asymmetric Digital Subscriber Line
AMPS : Advanced Mobile Phone Service
ANSI : American National Standards Institute
API : Application Programming Interface
ARM : Advanced RISC Machines
ASP : Active Server Pages
AWT : Abstract Windows Toolkit
CDC : Connected Device Configuration
CDMA : Code Division Multiple Access
Cdma2000 : Code Division Multiple Access 2000
CGI : Common Gateway Interface
CLDC : Connected Limited Device Configuration
CODEC : Compressor Decompressor
CORBA : Common Object Request Broker Architecture
CPU : Central Processing Unit
CS : Circuit Switched
CSD : Circuit Switched Data
CSS : Cascading Style Sheets
CVM : Compact Virtual Machine
D-AMPS : Dijital-Advanced Mobile Phone Service
DES : Data Encryption Standard
DECT : Digital Enhanced Cordless Telecommunications
dk. : dakika
DTM : Dual Transfer Mode
E2E : End to end
EDGE : Enhanced Data Rate for GSM Evolution
EMEA : Europe, Middle East and Africa
EMS : Enhanced Message Service
ETSI : European Telecommunications and Standarts Institute
FDD : Frequency Division Duplex
FDMA : Frequency-division multiple access
FOMA : Freedom of Mobile Multimedia Access
FTP : File Transfer Protocol
GIF : Graphics Interchange Format
GPRS : General Packet Radio Service
GPS : Global Positioning System
GSM : Global System for Mobile Communications
GUI : Graphical User Interface
HSCSD : High Speed Circuit Switched Data
HTML : Hypertext Markup Language
HTTP : Hypertext Transfer Protocol
HTTPS : Hypertext Transfer Protocol Secure

IDE : Integrated Development Environment
IMAP : Internet Mail Application Protocol
IMT-2000 : International Mobile Telecommunications-2000
I/O : Input/Output
IP : Internet Protocol
IrDA : Infrared Data Association
ISDN : Integrated Services Digital Network
ISEN : Institut Supérieur d'Électronique et du Numérique
J2EE : Java 2 Enterprise Edition
J2ME : Java 2 Micro Edition
J2SE : Java 2 Standart Edition
JPEG : Joint Photographic Experts Group
JSP : Java Server Pages
JSR : Java Specification Request
JVM : Java Virtual Machine
KB : Kilo Byte
Kbit/s : Kilobit/saniye
Kbps : Kilo bit per second
KVM : Kilobyte Virtual Machine
LAN : Local Area Network
m. : Metre
MB : Mega Byte
Mbit/s : Megabit/saniye
Mbps : Mega bit per second
MHS : Mobile Home Security
Mhz : Mega hertz
MIDP : Mobile Information Device Profile
MLP : Multi Layered Perceptrons
MMAPI : Mobile Media Application Programming Interface
MMS : Multimedia Messaging Service
OMA : Open Mobile Alliance
OTA : Over-the-air provisioning
OS : Operating System
OSI : Open Systems Interconnection
PC : Personal Computer
PCA : Principal Component Analysis
PCMCIA : Personal Computer Memory Card International Association
PDA : Personal Digital Assistant
PHP : Personal Home Page
PIM : Personal Information Management
PKI : Public Key Infrastructure
P2P : Peer To Peer
PNG : Portable Network Graphics
POP3 : Post Office Protocol 3
RMI : Remote Method Invocation
RMS : Record Management System
RSA : Rivest Shamir Adelman
SDK : Software Development Kit
SHA1 : Secure Hash Algorithm 1
SMS : Short Message Service

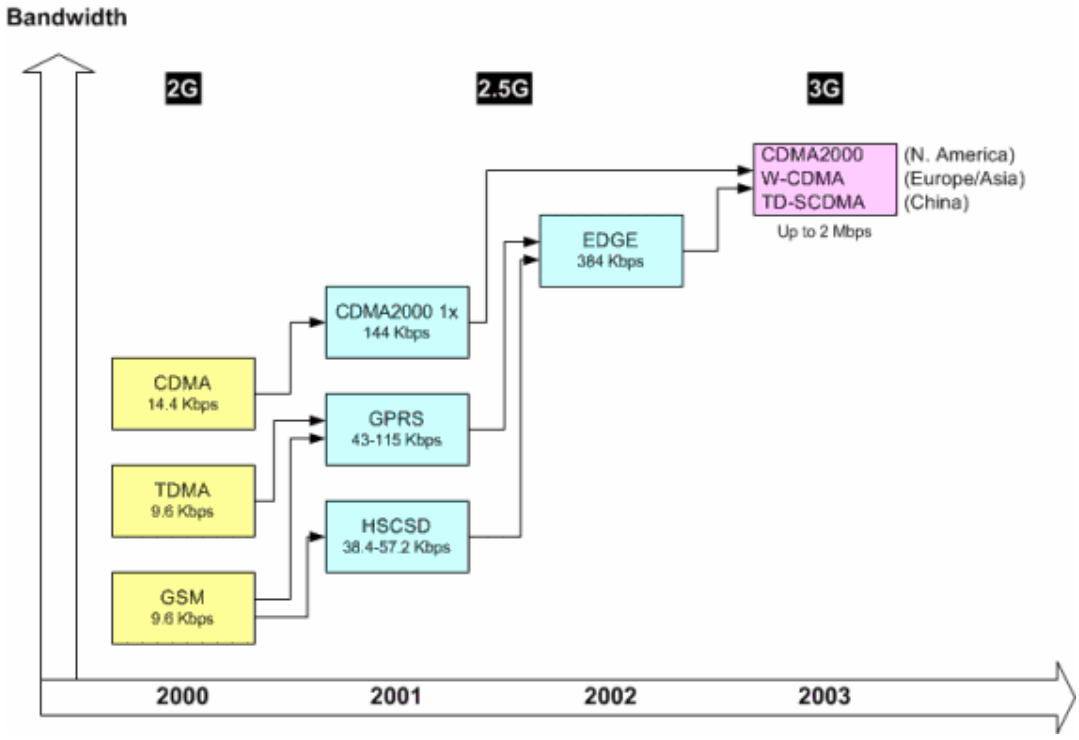
SOAP : Simple Object Access Protocol
SSH : Secure Shell
SSL : Secure Sockets Layer
TDMA : Transmission Control Procotol / Internet Procotol
TD-SCDMA : Time Division-Synchronous Code Division Multiple Access
TD-CDMA : Time Division-Code Division Multiple Access
TCP/IP : Time division multiple access
TLS : Transport Layer Security
UDP/IP : User Datagram Protocol / Internet Procotol
UIQ : User Interface for Symbian OS
UMTS : Universal Mobile Telecommunications System
USIM : UMTS Subscriber Identity Module
UWC : Unified Wireless Service
v.b. : Ve benzeri
WAE : Wireless Application Layer
WAP : Wireless Application Procotol
W-CDMA : Wideband Code Division Multiple Access
WCSS : Wireless Cascading Style Sheets
WDP : Wireless Datagram Protocol
WML : Wireless Markup Lenguage
WSP : Wireless Session Protocol
WTK : Wireless Toolkit
WTP : Wireless Transaction Protocol
WTLS : Wireless Transport Layer Security
XHTML : eXtensible Hypertext Markup Language
XML : eXtensible Markup Language
XSLT : eXtensible Stylesheet Language Transformations

1. GİRİŞ

1.1. 2G- İkinci Nesil ve 3G- Üçüncü Nesil

2G ve 3G ikinci nesil telefon ve üçüncü nesil telefon teknolojileri için kullanılan kısaltmalardır (Second Generation, Third Generation). 2G teknolojisi şu an ülkemizde standart olarak kullanılan teknolojidir. Fakat daha hızlı internet erişimi gibi ihtiyaçların ön plana çıkması ile yerini 3G teknolojisine bırakmaktadır. Henüz ülkemizde 3G teknoloji kullanan bir şebeke bulunmamaktadır (Şekil 1.1).

3G teknolojilerin getirdiği en önemli yenilikler aynı anda hem ses verisinin, hem de diğer veri türlerinin şebeke üzerinde transfer edilebilmesidir. Görüntülü görüşme (hem ses hem görüntü iletimi) bu teknolojiye güzel bir örnek sayılabilir. 3G teknolojiler daha hızlı bağlantıyı daha ucuza mal etmeyi başarmıştır. Şu an itibariyle dünyada yaklaşık 25 ülkede 3G şebeke desteği verildiği tahmin edilmektedir.



Şekil 1.1., 2G'den 3G'ye gelişme (Kaynak Iepsac.org)

Çizelge 1.1. 2G ve 3G'nin Hizmetlerinin karşılaştırmalı tablosu (Selian 2001)

Hizmetler	**2G (İkinci nesil)**	**2G + GPRS**	**3G(Üçüncü nesil)**
Web tarama	Kısa metinler içeren sayfalar	Herhangi 100KB'lık sayfa 30sn'de indirilebilir.	Herhangi 100KB'lık sayfa yaklaşık 2sn'de indirilebilir.
Dosya transferleri	Desteklenmiyor.	500 KB'lık bir dosya yaklaşık 2 dakika sürüyor.	500 KB'lık dosya yaklaşık 10 sn sürüyor.
E-Posta	Sadece SMS var.	Küçük dosya eklentili e-posta gönderimi.	Tam e-posta desteği.
Anında mesajlaşma	SMS	Metin tabanlı	Ses,video dosyaları da dahil.
VoIP (Internet üzerinden ses)	Desteklenmiyor.	Sınırlı.	Tam destek.
Akıcı ses/görüntü	Desteklenmiyor	Sadece kısa filmler.	Tam destek.
Şirket iç ağına erişim(Intranet)	Kısıtlı.	Sadece metin tabanlı	Tam destek.
Şirket uygulamalarına erişim.	Kısıtlı.	Sadece metin tabanlı	Tam destek.

Çizelge 1.2. Yıllara göre kullanılan mobil teknolojiler ve uygulamaları

Süre	**Kullanılan Teknoloji**	**Yeni İç ve Dış Uygulamalar**	**Hız***	**3dk.lık bir MP3 müzik dosyasını indirme süresi***
2000'e kadar	**2G**	• Telefon • E-mesaj • SMS • Sayısal Metin Gönderimi	10Kbps	31–41 dk.
2001–2002	**2.5 G**	• Gezgin Bankacılık • Sesli mesaj, Web • Gezgin Ses Çalıcı • Sayısal Gazete Yayını • Sayısal Ses Dağıtımı • Gezgin Radyo, Karaoke • Lokasyon tabanlı hizmetler İnteraktif şehir haritaları • Gezgin kuponlar, promosyonlar ve bilgi gönderme ile reklam yapma	64–144 Kbps	6–9 dk.

2003 ve ötesi	**3G**	• Gezgin video konferans • Görüntülü Telefon/Mesaj • Uzaktan Tıbbi Müdahale ve Eğitim • Gezgin TV/Video Oynatıcı • Geliştirilmiş Araç Seyrüsefer Cihazı/ Şehir Rehberi • Sayısal Katolog Alışverişi • Sayısal Ses/Görüntü Dağıtımı	144 Kbps- 2 Mbps	11sn.-1.5dk.

1.1.1. İkinci Nesil (2G) telefon sistemlerinin genel özellikleri

İkinci nesil telefonları birinci nesil telefonlardan ayıran en büyük özellikleri şebeke ile haberleşirken analog yerine dijital radyo sinyalleri kullanmalarıdır. İkinci nesil teknolojileri TDMA tabanlı ve CDMA tabanlı olmak üzere ikiye ayrılırlar. En genel 2G teknolojileri şunlardır:

GSM: TDMA tabanlıdır. Avrupa tabanlı bu teknoloji, daha sonraları tüm dünyada kullanılır hale gelmiştir, halen ülkemizde de bu teknoloji ile firmalarımız çalışmaktadır. Bu sistemdeki en temel özellik çoklu erişimi sağlamak için zaman paylaşımı yapılmasıdır.

IDEN: TDMA tabanlıdır. ABD ve Kanada'da kısmen kullanılan GSM'e alternatif bir alt yapıdır.

IS-136 veya D-AMPS: TDMA tabanlıdır, yine sadece Amerika'da kullanılan bir ağ teknolojisidir.

PDC: TDMA tabanlıdır, özellikle Japonya'da kullanılmaktadır.

Telefonlar, şebeke istasyonları ve şebeke arasında dijital olarak haberleşme yapılması iki önemli avantajı beraberinde getirmektedir:

Dijital ses verisi, analog ses verisine göre çok daha iyi sıkıştırılmakta ve çoklanmaktadır. Bu işlem çeşitli CODEC'ler sayesinde yapılabilmektedir. Bu sayede aynı bandwidth ile çok daha fazla telefon görüşmesi yapılabilmektedir.

Dijital sistemler telefonlardan çok daha az radyo gücü çekmektedir. Böylece hücreler daha küçük olabilmekte ve aynı alana daha fazla hücre yerleştirebilmektedir. Bu sayede baz istasyonları ve ilgili donanım da daha ucuza mal edilebilmektedir.

2G teknolojisi özetle; daha az şarj edilen telefonlar, daha uzun ömürlü piller, daha kaliteli ses ve dijital olan diğer hizmetlere (e-posta) erişimi gibi avantajları beraberinde getirmektedir.

1.1.2.Üçüncü Nesil (3G) telefon sistemlerinin genel özellikleri

3G teknolojilerinin 2G teknolojilerine getirdikleri pek çok yenilik vardır. 3G iletişiminde tüm veriler dijital olarak taşınırlar. Bu sayede daha çok veri aynı anda taşınabilmektedir ve görüntülü görüşme, anında mesajlaşma gibi çoklu ortam kullanan

uygulamalar için uygun ortam oluşturulmuştur. 3G teknolojiler, 2G teknolojilere göre daha az pil harcarlar, 3G telefonlar daha uzun pil ömrüne sahiptirler.

En genel 3G teknolojileri şunlandır :

- W-CDMA
- CDMA2000
- CDMA2001
- TD-CDMA / TD-SCDMA
- UWC-136 (Genelde EDGE ile beraber gerçeklenir)
- DECT

1.2. GPRS

GRPS ile doğrudan cep telefonunuzla veya cep telefonunuz üzerinden dizüstü, pda gibi başka bir cihazla internete bağlanabilir ve e-postanıza ulaşabilirsiniz. Ortalama olarak veri transfer hızı donanıma bağlı olarak 26kbps ile 52kbps arasında değişir.

GPRS ihtiyacı olan veya başka bir deyişle GPRS kullanımına uygun uygulamaları şöyle özetleyebiliriz :

E-posta alma veya gönderme uygulamaları. Bu tür uygulamalarda büyük boyutlu dosyaları e-postaya eklemekten kaçınılmalıdır, metin tabanlı e-posta uygulamaları için çok daha uygundur.

GPRS'in paylaşımlı yapısı sebebiyle genellikle GMS şebekeleri bağlantı hizmetini bağlı kalınan süre ile değil de transfer edilen bilgi (gönderilen-alınan) ile ücretlendirir. GPRS teknolojisi, kullanıcıya yüksek erişim hızının yanı sıra, bağlantı süresine göre değil gerçekleştirilen veri alışverişi miktarı özerinden tarifelendirilen ucuz iletişim olanağı sağlamakta ve böylelikle "sürekli bağlantıda, sürekli gerçek zamanda" (always connected/always online) anlayışını sunmaktadır.

Transfer edilen bilgiye göre ücretlendirme olduğu için gezinti sırasında ne kadarlık bilgi çekildiği telefonun sağlayacağı bir yazılımla öğrenilmelidir. Ülkemizin başta gelen GSM şebekelerinin 2006 itibariyle GPRS ücretlenmesi aşağıdaki şekildedir.

Çizelge 1.3. Ülkemizdeki bazı operatörlerin internet bağlantı ücretlendirmesi

	1 KB Ücreti		
Transfer miktarı	**Telsim**	**Turkcell**	**Avea**
0-5 MB	0,001500 YTL	0,001250 YTL	0,001680 YTL
5-10 MB	0,001250 YTL	0,000900 YTL	0,001440 YTL
10-25 MB	0,000900 YTL	0,000600 YTL	0,000960 YTL
25-50 MB	0,000500 YTL	0,000400 YTL	0,000720 YTL
50-100 MB	0,000500 YTL	0,000200 YTL	0,000480 YTL
100 MB – 1GB	0,000400YTL	0,000200 YTL	0,000480 YTL
1GB -10 GB	0,000350 YTL	0,000200 YTL	0,000480 YTL

GSM şebekeleri GPRS hizmetlerinde FDD (Frequency Division Duplex) veya FDMA protokolleri kullanarak çoklu erişimi sağlayabilirler. Bir oturum başlatıldığında, kullanıcıya bir uplink ve downlink frekans kanalı atanır ve kullanıcı paket iletişim modunda bilgi transferi yaptığı için aynı frekans kanalı başka kullanıcılar tarafından da kullanabilir. Tüm paketler GSM ayarlarına uygun olarak sabit uzunluktadır. Downlink içinde "ilk gelen, ilk hizmet alır"

(first-com, first-served) mantığı kullanılmaktadır. Uplink içinde ise ALOHA tahsisat sistemine benzer bir sistem kullanılmaktadır.

1.2.1. GPRS Sınıf Türleri

Bir mobil cihazın sınıfı cihazın GPRS hızını belirler.

Örneğin, GPRS cihazlarının çoğunluğu 24 Kbps (Kilobytes per second – Saniyedeki kilobyte)'e kadar veri indirme (download) hızına sahiptirler. Teorik olarak ise, eğer GMS şebekesi 8 slotun hepsini birden bir kullanıcıya atamışsa, hız en fazla 171.2 kbit/sec olabilir. Ama pratikte bu hız genelde 40-50 Kbps'yi geçmez.

Çizelge 1.4.'de çeşitli protokollerde ki ortalama yakalanabilecek download-upload hızları gösterilmiştir.

Çizelge 1.4. GSM Cihaz Sınıfları

Tür	**Uplink (Gönderim)**	**Downlink (Alım)**
GPRS	14 kbps	28-64 kbps
GSM CSD	9.6-14 kbps	9.6-14 kbps
HSCSD	28 kbps	28 kbps
Dial-UP	56 kbps	56 kbps
ISDN Standard	64 kbps	64 kbps
ADSL	256 kbps	512 kbps
Broadband	2 Mbps	2 Mbps

GSM Cihaz Sınıfları

Cihazın sınıfı cihazın GPRS kullanım yeteneğini, donanımsal desteğini belirler.

A Sınıfı (Class A)
Her iki hizmeti de aynı anda kullanarak GPRS ve GSM hizmetlerine (Ses, SMS v.b.) bağlanabilir. Bu cihazlar şu an piyasa da bulunmaktadır (Nokia N93 gibi).

B Sınıfı (Class B)
GPRS ve GSM hizmetlerinden ikisini de kullanabilirler, fakat bu cihazlar her iki hizmeti aynı anda kullanamazlar. GSM hizmeti başladığında (Sesli görüşme, SMS gibi) GPRS hizmeti durdurulur, GSM hizmeti bitince GPRS hizmetine devam edilir. Piyasada ki çoğu telefon B sınıfına dahil edilebilir.

C Sınıf (Class C)
GPRS veya GSM hizmetlerinden birisini seçerek kullanabilirler. Kullanıcı elle kullanılacak hizmeti seçmelidir, otomatik geçiş veya aynı anda kullanma söz konusu değildir.

Gerçek bir A sınıfı cihaz aynı anda farklı iki frekansta iletişim yapması gerektiğinden iki tane radyoya sahip olmalıdır, bu yüzden bu pahalı gereksinimin üstesinden gelmek için GPRS telefonlar için DTM (Dual Transfer Mode – Çift Transfer Modu) özelliği geliştirilmiştir. Bu tür cihazlar tam olarak olmasa da A sınıfı sayılabilirler. Şu an için tüm şebekeler DTM özelliğini desteklemiyorlar fakat 2007 itibariyle bazı şebekeler destek vermeye başlayacaklarını bildirmişlerdir.

Cihazın GPRS hızının belirleyen ana faktör ise kullanabileceği maksimum slot sayısıdır, bu sayı bazen şebekenin sağladığından az olabilir, bazen ise şebekenin sağladığı kadardır. Cihazın kullanabildiği slot sayısı cihazın dahil olduğu GPRS Multislot Sınıfını belirler.

Çizelge 1.5. Multislot sınıfına göre sağlanan slot sayıları

Multislot Sınıfı	Downlink Slot sayısı	Uplink Slot sayısı	Active Slot sayısı
1	1	1	2
2	2	1	3
3	2	2	3
4	3	1	4
5	2	2	4
6	3	2	4
7	3	3	4
8	4	1	5
9	3	2	5
10	4	2	5
11	4	3	5
12	4	4	5

En genel GPRS multislot sınıfları şunlardır :
Sınıf 2 :
En aşağı seviye GPRS gerçeklemesidir.
Sınıf 4:
Ortalama bir GPRS gerçeklemesidir, Sınıf 2 cihazlara göre bu cihazların download hızı %50 daha fazladır.
Sınıf 6:
Bu da ortalamanın biraz üstünde bir gerçeklemedir, Sınıf 4'den tek farkı upload hızının daha iyi olmasıdır.
Sınıf 8:
Diğerlerine göre daha iyi bir gerçeklemedir, Sınıf 4 ve 6'ya göre %33 daha hızlıdır.
Sınıf 10:
Sınıf 8'e göre daha iyi bir upload hızına sahiptir.
Sınıf 32:
En iyi download performansı (Bir önceki sınıfa göre 25% daha hızlı), şu an piyasada ki tek Sınıf 32 cihazı Nokia N93'dür (Şekil 1.2).

Şekil 1.2. Nokia N93, Sınıf 32 ilk telefon (Anonim 2006a)

1.3. HSCSD : Yüksek Hızlı Şebeke Anahtarlamalı Veri

HSCSD (High-Speed Circuit Switcted Data), Şebeke anahtarlamalı veri iletiminin geliştirilmiş şeklidir. Adından da anlaşılacağı üzere her bir kullanıcıya kanal tahsis etme işlemi anahtarlama ile yapılır. CSD ile HSCSD arasındaki fark kodlama metodları kullanımı ve aynı anda kullanılan zaman dilimi sayısı ile elde edilen yüksek hızdan ileri gelir.

HSCSD'nin getirdiği yeniliklerden birisi veri transferi için değişik hata düzeltme (error correction) metodlarının kullanımına izin vermesidir. GSM şebekelerinde kullanılan asıl hata düzeltme metodlarının en büyük dezavantajı gönderilen veri içinde hata düzeltme kodlarının çok yer tutmasıydı, bu yüzden gönderilen verinin çoğunu hata düzeltme kodları alıyordu. HSCSD değişik hata düzeltme seviyeleri sunar, bu seviyeler radyo bağlantısının kalitesine göre seçilebilir. Bu şu anlama gelmektedir: HSCSD ile 14.4 kbit/s tek bir zaman diliminde taşınabilir, oysa ki CSD ile tek bir zaman diliminde ancak 9.6 kbit/s veri transfer hızı sağlanabilmektedir. HCSCD bu örnekte olduğu gibi %50'ye varan bir iyileştirme sağlayabilmektedir.

HSCSD'nin getirdiği bir başka yenilik ise aynı anda GSM şebekesinin birden fazla zaman dilimini (time slot) kullanabilmektir. Örneğin kullanılan dilim sayısı dört (4) ise , bu 57.6 kbit/s transfer hızına kadar çıkılabilmektedir (14.4 X 4 = 57.6). Ve hata düzeltme kodlarının artırılması gereken kötü radyo iletişim şartlarında dahi, klasik CSD iletişimin dört (4) katına kadar transfer hızı çıkabilmektedir. Eğer 8 GSM şebekesi dilimini aynı anda kullanılabilirse basit bir hesapla hızın 110 kbit/s'nin üzerine çıkabileceğini görmekteyiz.

HSCSD şebekenin her bir zaman dilimini tek bir kullanıcıya tahsis edilmesini gerektirir. Genellikle bu bazı dezavantajlar doğurabilir. Örneğin kullanıcı internete bağlıyken sesli görüşme yapmak istediğinde, bu ayrı bir şebeke zaman diliminin tahsisatı gerektirdiğinden, kullanıcı genelde şebeke ayarlarının izin vermemesi sebebiyle istediği verimi alamaz. Şebeke gerekli ayarlamaları kullanıcı lehinde yapsa bile kullanıcı normalden fazla zaman dilimi kullanacağı için normal ücretlendirmeden daha fazla ödemek zorunda kalabilmektedir. Bu yüzden HSCSD GPRS ile kıyaslandığında daha pahalıya gelebilmektedir ve nispeten daha düşük tercih oranına sahiptir.

HSCSD'nin GPRS'e göre bahsettiğimiz avantajı (tek bir kullanıcıya tüm zaman diliminin ayrılması sebebiyle daha hızlı olması) haricinde, HSCSD GPRS'e göre daha düşük radyo ara yüz gecikmesine sahip olduğu için de yine avantajlıdır. Bunun sebebi HSCSD kullanıcısının şebekeye bir paket gönderirken izin almak için beklememesidir.

HSCSD aynı zamanda paket veri taşıması oranının yüksek olduğu UMTS ve EDGE sistemleri için de bir seçenektir. Bu sistemler için de paket veri iletimine göre bazı avantajlar sunmaktadır.

1.4. EDGE - GSM'in Gelişmesi için Geliştirilmiş Veri Hızları

EDGE (Enhanced Data rates for Global Evolution) GSM'in gelişmesi amacıyla 1997'de Avrupa Telekomunikasyon Standartları Enstitüsü'ne (ETSI) önerilen, radyo tabanlı, yüksek hızlı mobil veri standardıdır. Aynı zamanda bu standart GSM384 olarak da bilinir. EDGE teknolojisi TDMA endüstrisi ile GSM firmalarının üçüncü nesil yüksek hızlı veri transferi standartları çıkarma çalışmalarının bir ürünüdür. EDGE sayesinde mobil operatörler 3G lisansı almadan 3G hizmetleri sunabilmektedirler. EDGE kullanan bir cihaz sadece bir zaman dilimi kullanarak 48 Kbps, sekiz zaman diliminin tümünü kullanarak ise 384 Kbps hıza çıkabilmektedir. EDGE, 800/900/1800/1900 frekans bantlarını desteklemektedir. GSM zaman dilimleme mimarisi ve GSM bant genişliğini kullanmasına rağmen, hiç bir şekilde sadece GSM hücresel sistemlerle kullanılabilir diyebileceğimiz bir kısıtlaması yoktur. Aslında EDGE'in yaptığı iş varolan GSM ve TDMA gibi 2G sistemleri 3G teknolojilerine biraz daha yaklaştırmak ve geliştirmektir.

EDGE teknolojisinin 2G şebekeleri tarafından hayata geçirilmesi oldukça kolaydır, şebeke sadece her hücre için bir EDGE telsiz birimi ekleyerek sistemini EDGE hızına çıkarabilmektedir. EDGE telsizleri aynı zamanda GSM trafiğini kontrol ederek sadece gerekli zamanlarda EDGE moduna geçiş yapabilmektedirler. EDGE şebekesinden faydalanmak için mobil cihazlarında bu teknolojiyi desteklemesi gerekmektedir. Son günlerde çıkan hemen hemen tüm akıllı telefonlar (Smart phone) EDGE ve hatta 3G teknolojilerini desteklemektedirler.

EDGE 2G şebekeleri için IP-tabanlı çoklu ortam hizmetlerini kullanmak için oldukça uygun maliyetde bir seçenek oluşturmaktadır. Bununla birlikte GPRS hizmetleri de tam manasıyla kullanılabilmektedir, yani EDGE yerine geçen değil tamamlayan bir yaklaşımı benimsemiştir.

1.5. UMTS (Evrensel Mobil İletişim Sistemi)

3G, üçüncü nesil kablosuz telefon teknolojisidir. Aynı 1G ve 2G gibi, hücresel bir ağ sistemi kullanır. 3G'ye aynı zamanda Universal Mobile Telecommunications System (yani Evrensel Mobil İletişim Sistemi) anlamına gelen UMTS de denir. Hatta Japonya'da Freedom of Mobile Multimedia Access (Mobil Çoklu Ortam Erişimine Özgürlük) anlamına gelen FOMA ismi de kullanılır.

3G'nin 2G'ye göre getirmiş olduğu en büyük yenilik taban olarak alınan verinin ses değil sayısal veri olmasıdır. Buna ek olarak, 3G sisteminde cihazlar bant genişliğini sadece veri alışverişi sırasında işgal ederler. İlk örnekleri Japonya'da 1998 yılında kullanıma açılan bu teknoloji, 2003'ten itibaren Avrupa'ya da gelmiştir (Anonim 2006b)

UMTS, IMT–2000 olarak bilinen, ITU tarafından tanımlanan ve geliştirilen üçüncü nesil gezgin iletişim sistemlerinin en önemlisi ve önde gelenidir. Diğer bir deyimle UMTS, Avrupa Komisyonu kararıyla ETSI (European Telecommunications and Standars Institute) tarafından ve IMT–2000 çatısı altında geliştirilen bir üçüncü nesil (3G) gezgin iletişim sistemidir. Avrupa'yı üçüncü nesil gezgin iletişim sistemlerinin gelişmesinde öncü olmaya iten en önemli etken ikinci nesil (2G) gezgin iletişim sistemlerinin (GSM ve DECT gibi) başarısıdır. UMTS evrensel gezgin iletişimi sağlamayı amaçlayan bir sistem olup, terminal gezginliğinin yanı sıra kişisel gezginliği ve hizmet gezginliğini desteklemektedir. UMTS 2010 yılında 2 milyar kullanıcıya yüksek kalitede gezgin ve telsiz çoklu ortam hizmet iletişimini sağlayarak, geleceğin pazarını yaratmada anahtar rolü oynayacaktır (Akçay, 2001).

UMTS, geniş band çoklu ortam servislerinin kullanımına olanak sağlayan yüksek veri hızlarını desteklemektedir. Hem simetrik hem de asimetrik veri transferine imkân tanıyarak kaynakların verimli kullanılmasını sağlayan, devre ve paket anahtarlamalı servislerin aynı anda kullanımını mümkün kılan ve IP (Internet Protocol - İnternet Protokolü) protokolünü destekleyen bir hizmettir. Ses kalitesi telli şebekelerle karşılaştırılabilir derecede yüksek olacaktır. Ayrıca, ikinci nesil gezgin haberleşme sistemleri ile kesintisiz geçiş yapabilecek, küresel olarak belirsiz geçişi ve kesintisiz dolaşımı sağlayacaktır (Bayız ve Eken, 2001).

3G'nin getirmiş olduğu birçok yenilik vardır:

- Mesajlaşma, internet erişimi ve yüksek hızda çoklu ortam haberleşme desteği
- Gelişmiş hizmet kalitesi
- Gelişmiş pil ömrü
- Konumlandırma hizmetlerinin sağlanması
- Bütün katma değerli ses hizmetlerinin sağlanabilmesi
- İşletim ve bakım kolaylığı
- Mevcut şebekelerle birlikte çalışabilirlik, 2G'ye dolaşım sağlayabilme
- Mevcut şebekelere geriye doğru uyum sağlayabilme, düşük kurulum maliyeti
- Gelişmiş güvenlik yöntemleri sayesinde mobil ticarete ortam sağlayabilme

(Anonim 2006c)

UMTS Güvenlik Özellikleri

Başlıca UMTS güvenlik özelliklerini şöyle sıralamak mümkündür (Akçay, 2001):

- **Kimlik doğrulama (Authentication):**

a) Bir nesnenin kimliğinin doğrulanması.
b) Veriyi gönderen kaynağın kimliğinin doğrulanması.

- **Gizlilik (Confidentiality):** Bu özellik yetkili olmayan kişilerin gerçek bilgiye erişmelerini engellemektir.

- **Gerçek ismini saklama (Anonymity):** Bir kişinin veya nesnenin yetkili olmayan kişilerce teşhis edilmesini veya tanımlanmasını engellemektir.

- **Erişim kontrolü (Access Control):**

a) Cihaza erişim kontrolü.
b) Hizmete erişim kontrolü.
c) Veriye erişim kontrolü.

- **Bütünlük (Integrity):** Bu özellik verinin yetkisiz kişilerce değiştirilmesini

engellemektir.

- **İnkâr edememe (Non-repudiation):**

a) Veriyi gönderdiğini inkâr edememe.
b) Gönderilen verinin alındığını inkâr edememe.
c) Veriye erişimi inkâr edememe.
d) Hizmete erişimi inkâr edememe.
e) Prosedürü içermeyi inkâr edememe.

- **Bütünleyici (Supplementary):** Bu özellik şebeke operatörünün (network operator) ve hizmet sağlayıcının (service provider) belirli sabit veya gezgin kullanıcılara uçtan-uca güvenlik hizmetlerini sunabildiğini doğrulamaktadır.

2. ÖNCEKİ ÇALIŞMALAR

Aksu (2004), Java/J2ME kullanarak uygulama geliştirmeyi esas aldığı tezinde, 2G ve 3G teknolojilerini, internete bağlanmak için kullanılan protokolleri yan bilgi olarak vermiştir. Örnek olarak yaptığı uygulama mobil cihaz üzerinde çalışan bir sözlük programıdır. Bu uygulamayı yaparken CLDC 1.0 profilinden ve J2ME I/O API'den bahsetmiş ve kullanmıştır. Çalışmasının sonunda yaptığı tartışmada da bu alanda güvenlik ile ilgili yeterince çalışma olmadığını ve bu alanın geliştirilmeye ihtiyaç duyduğunu belirtmiştir.

Özçelik (2006), çalışmasında kablosuz bağlantı teknolojilerinden Bluetooth üzerinde yoğunlaşmış ve bu alanda güvenlik geliştirmeleri önermiştir. Bluetooth ile yakın civardaki cihazların birbirine bağlanabildiğini, veri iletimini güvenli hale getirebilmek için frekans atlama, şifreleme gibi metotlar kullanıldığını fakat yeterli olmadığını belirtmiştir. Kendisi güvenli veri iletimi için anahtar tabanlı şifreleme algoritmalarından DES ile yapmış ve güvenliği artırdığını belirtmiştir (Şekil 2.1).

Şekil 2.1. Bluetooth üzerinde güvenliğin artırılması çalışmasının görünümü (Özçelik, 2006)

Danaher ve Nguyen (2002), cep telefonlarını da içine alan bir ev güvenlik sistemi (MHS, Mobile Home Security) uygulaması geliştirmişlerdir. Bu sistem evde bulunan bir bilgisayarın, web-cam üzerinden çeşitli yazılımlarla elde ettiği görüntüleri GPRS kullanarak mobil telefona gönderilmesi üzerine kuruluydu. Bu çalışmayı daha iyi hale getirebilecek şeylerin başında GPRS'den daha hızlı bir bağlantı hizmeti ve görüntü yakalama işini yapan bilgisayarın bu işi daha kaliteli yapabilmesi olduğunu da belirtmişlerdir.

Ginesu, Giusto ve Onali (2006), mobil cihazların gittikçe daha çok interneti kullanmasıyla ortaya çıkan daha iyi ve güvenilir kimlik doğrulama metodları ihtiyacından bahsetmişler ve resim tabanlı bir kimlik doğrulama (image-based authentication) sistemi önermişlerdir. Çalışmalarında resim dağıtma (scrambling), JPIP ve JPEG2000 gibi teknikler ve protokollerden bahsetmişler ve uygulamışlardır. Sonuç olarak PDA ortamında test ettikleri

uygulamalarının insan hafızasına daha yatkın, daha uygun olduğu için kimlik doğrulama işindeki bazı problemleri çözdüğünü belirtmişlerdir.

Jansen (2004), internette kimlik doğrulama metotlarının insan doğasına zor geldiğini ve bulunması zor parolaların aynı zamanda da hatırlanması zor olanlar olduğunu belirterek, parola sistemlerinin değiştirilmesi gerektiğini söylemiştir. Sunduğu öneride, insan hafızasının anlamsız metinlerden ziyade resimleri ve anlamlı metinleri daha iyi tuttuğunu söylemiştir ve bir resim seçme tabanlı kimlik doğrulama sistemi önermiştir. Bilgi tabanlı kimlik doğrulama sistemlerindeki çeşitli problemlerden bahsetmiş ve önerisinin nasıl bu problemleri çözeceğini detaylarıyla anlatmıştır.

Jermyn (1999), grafik tabanlı parolalar hakkında çalışmasında Draw-a-secret dediği bir metodu üretmiş ve kullanmıştır. Bu metotta kullanıcı resim seçmek yerine hücrelere bölünmüş bir alanda kendisine özel bir şeyler çizerek kimlik doğrulama yapmaktadır. Bu uygulamanın hedef platformu o zamanların gelişmiş mobil cihazları olan PDA'lardır. Sonuç olarak 5x5 bir ızgara üstünde çizilen bir şeylerin metin tabanlı parolalardan daha verimli sonuçlar verdiğini de belirtmiştir.

Perrig ve Song (2000), karışık metni (hashing) görselleştirme adı verdikleri çalışmalarının gerçek dünya güvenlik ihtiyaçlarını karşılayacak bir metot olduğunu söylemişlerdir. Çalışmaları her ne kadar mobil cihazlar üzerine olmasa da mobil cihazlar için varolan bir problemi yine mobil cihazlar üzerinde uygulanabilecek bir metotla çözmeyi önermişlerdir. Çalışmalarının özünde bir metni resme dönüştürerek, bu şekilde insan doğasına daha uygun bir kimlik doğrulama sistemi önermişlerdir. Bu dönüşüm sırasında kullanılacak matematiksel çözümleri, fonksiyonları da çalışmalarında detaylı olarak anlatmışlardır. Sonuç kısmında ise çalışmalarının bundan sonraki yönünün hatırlanması daha kolay olan doğa manzaraları, şehir görüntüsü gibi resimler üretmek olduğunu belirtmişlerdir.

3. MATERYAL VE METOD

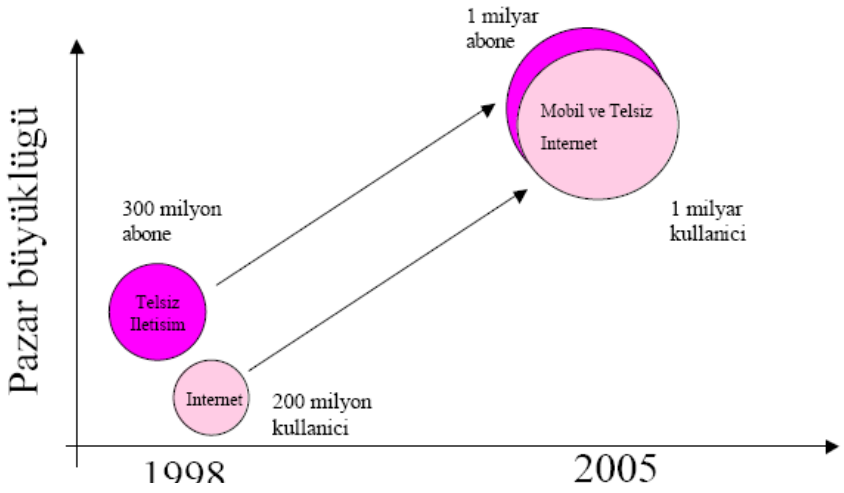
3.1.MATERYAL

3.1.1.Donanım

3.1.1.1. 3G Cihazlar

3G terminalleri, kullanıcılarına günlük hayatlarında, birinci nesil (1G) ve ikinci nesil (2G) gezgin telefonların sağlayabildiğinden daha fazla konfor ve denetim imkânı sağlamışlardır. Onlar e-yardımcı, e-sekreter, e-danışman ve hatta e-idareci olacaklardır. Bunun anlamı, yaşanılan ortamda otomasyonu sağlayan bir kontrol mekanizmasının oluşacak olmasıdır. Şu anda piyasada çoklu ortam özelliklerine sahip birçok 3G gezgin telefon mevcuttur. Bu gezgin telefonlardan en çok tercih edilenlerinden bazıları şunlardır: Nokia 60 Serisi telefonlar, örneğin Nokia 6630, Nokia N90, Nokia N70 v.s. veya Sony Ericcson akıllı telefonlar, örneğin P910 gibi.

Üçüncü nesil telefonların büyük bir çoğunluğu WAP, Bluetooth, IrDA (Infrared Data Association) gibi iletişim standartlarını da desteklemektedir. Bu standartlardan biri kullanılarak dizüstü bilgisayar (dizüstü), PC (Personal Computer) , avuç içi bilgisayar (PDA) gibi cihazlarla gezgin telefonlar arasında bağlantı kurmak, bir ağa bağlanmak mümkündür. Bu bağlantıdan sonra her türlü veri alışverişi mümkün hale gelmektedir (Aksu, Subaşı, Karabulut, Dayak, 2005).



Şekil 3.1. Gezgin cihaz kullanıcıları (Cisco Systems 2005)

Üçüncü Nesil (3G) Multimedya gezgin telefonların öne çıkan önemli özellikleri şöyle sıralanabilir (Büyükçakıroğlu 2003) :

- İşletim sistemleri
- Kullanılan ekranlar
- Donanım ve Performans
- Müzik ve video
- Kamera
- İletişim

Gezgin cihazlara yüklü işletim sistemlerinden yaygın olarak kullanılanları şunlardır: Symbian OS, Windows CE/ Pocket PC, Palm OS, Pocket Linux. Şu an için pazardaki en yaygın işletim sistemi Symbian OS olarak görülmektedir. Pek çok akıllı telefon (Smart Phone) üzerinde Symbian OS bulunmaktadır. Örneğin Nokia 60 Serisi cihazlar. Aşağıda 2006 yılı itibariyle gezgin cihazların pazar payları ve üretilen cihaz sayıları görülmektedir.

Çizelge 3.1. Mobil işletim sistemlerinin pazar payı dağılımı (Anonim 2006d)

İşletim Sistemi	2005, 3.Çeyrek	%	2006, 3. Çeyrek	%	Büyüme
Symbian	8,164,790	59.7%	13,217,980	72.8%	62%*
Linux	3,005,440	22.0%	3,030,220	16.7%	1%
PalmSource	621,700	4.5%	333,340	1.8%	-46%
Microsoft	302,280	2.2%	1,025,540	5.6%	239%
RIM	210,100	1.5%	506,230	2.8%	141%
Diğer	85,580	0.6%	51,308	0.3%	-40%
Toplam	12,389,890	90.5%	18,164,618	100.0%	47%

3.1.1.2. Symbian OS

Symbian OS diğer işletim sistemlerinin aksine sadece mobil cihazlar için tasarlanmış bir işletim sistemidir, başka bir ortam – örneğin masaüstü- için tasarlanmış bir işletim sisteminin uzantısı değildir. Gezgin cihaza özel donanımı yönetmek için tasarlanmış gerçekleştirmeler, kütüphaneler ve API sağlamaktadır, bu sayede cihaz üzerinde yazılım geliştirme yapılabilmektedir. Symbian işletim sistemi ARM işlemciler ile çalışmak için tasarlanmıştır.

Symbian üzerinde programlama yapmak, klasik işletim sistemi API'sini kullanarak proğramlama yapmaktan biraz farklıdır. Çünkü üzerinde çalışılan cihazlar kısıtlı olanaklara sahiptir ve hafıza kullanımı üst düzey öneme sahiptir. Bunun yanında disk kullanımı, işlemci kullanımı da kısıtlı kaynaklar sebebiyle oldukça önemlidir. Bu yüzden her ne kadar Symbian üzerinde olay tabanlı (event-based) programlama yapılsa da ve bu masaüstü işletim sistemlerine çok benzese de, Symbian sistemine özgü pek çok programlama kavramı öğrenilmeden bu platformda verimli programcılık yapılamamaktadır. Bu kavramlardan bazıları : Descriptors, Cleanup-Stack, Active Objects v.b. Bu yüzden bir programcının C++ kullanarak Symbian üzerinde programlama yapması için gereken öğrenim süreci klasik programlama platformlarına göre daha uzun ve zordur.

Fakat Symbian üzerinde tek programlama yapma olanağı C++ değildir, OPL, Python, Visual Basic ve Java (Micro Edition) gibi platform ve diller de desteklenmektedir. Bunun dışında cihaz üreticileri de programlamayı kolaylaştıracak kendi arayüzlerini ve kütüphanelerini tasarlamışlardır. Örneğin Nokia 60 serisi cihazlar üzerinde işletim sisteminin olanakları dışında bu seriye özgü java dili için geliştirilmiş API ve kütüphaneler bulunmaktadır. Sony Ericcson cihazlar üzerinde ise UIQ denilen özel bir arayüz ve API bulunmaktadır. Her ne kadar bu bir zenginlik getirse de, her cihaz üzerinde çalışacak program geliştirme kavramını da zorlaştırmaktadır. Bu durumda genel yazılımlar geliştirmek isteyen programcılar cihaza özgü API ve kütüphanelerden kaçınacak ve genel Symbian API sistemine sarılacaklardır.

Symbian işletim sisteminin getirdiği avantajlardan birisi de kaynak kodunun her ne kadar "Açık kod" kriterlerine göre açık olması tartışılsa da kaynak kodun oldukça büyük kısmının üreticiler tarafından pek çok şekilde dağıtılmasıdır. Bu sayede üzerinde geliştirilen

pek çok açık kaynak kodlu uygulamalarda kolayca bulunmaktadır. Symbian üzerinde ki bazı açık kaynak kodlu uygulamalar şunlardır :

- Putty : SSH erişim yazılımı
- Internet Radio
- Ruby Programlama Dili
- SymTorrent : P2P paylaşım yazılımı
- Apache Web Sunucusu

Symbian OS avantajları yanında bütün işletim sistemlerinin yaşadığı dezavantajları da yaşamaktadır. Örneğin virüsler bu işletim sistemine de bulaşabilmektedir. Masaüstü bilgisayarlar da ağ yoluyla bulaşan virüsler gibi, gezgin cihazlarda da virüsler Bluetooth, GPRS ve diğer bağlantılarla bulaşabilmektedir. Bu sebeple bu cihazlara sahip olanlar kullanmadıkları durumlarda Bluetooth, IrDA gibi kısa mesafe erişimleri dahi kapatmaktadırlar. Symbian üzerinde bazı bilinen yaygın virüsler aşağıdadır:

- Drever.A : Bir SIS dosyası trojanıdır.
- Locknut.B Bu da bir SIS dosyası trojanıdır, özellikle 60 serisini tehdit etmektedir.
- Mabir.A : Bluetooth yoluyla yayılan bir virüstür, MMS erişimine sahiptir.
- Frontal.A: Kendini yayamayan bir SIS dosyası trojanıdır.

Symbian işletim sistemi birden fazla firmanın ortak üretimidir, başlıca sahipleri şöyle sıralanabilir : Ericsson (15.6%), Nokia (47.9%), Panasonic (10.5%), Samsung (4.5%), Siemens AG (8.4%), ve Sony Ericsson (13.1%). Bu durumda bu cihazlar üzerinde Windows veya Linux gibi bir işletim sistemi göremeyeceğimizi şimdiden söyleyebiliriz. (Anonim 2006e)

Symbian işletim sistemini kullanan cihazların bazıları şunlardır:

- Ericsson R380 (2000)
- Nokia 9210 Communicator smartphone
- UIQ arayüzüne sahip cihazlar : Sony Ericsson P800 (2002), P900 (2003), P910 (2004), P990, W950, M600, Motorola A920, A925, A1000, DoCoMo M1000, BenQ P30, P31 ve Nokia 6708.
- Nokia S60 Serisi cihazlar (2002) : Bu teknolojiyi kullanan pek çok cihaz vardır, bunların ilki Nokia 7650 dir. Diğer cihazlar şöyle gelmiştir : Nokia 3650, Nokia 3620/3660, Nokia 6600, Nokia 7610, Nokia 6670 and Nokia 3230. Bu teknoloji Nokia dışındaki firmalar tarafından da kullanılmıştır, örnek cihazlar : Siemens SX1, Sendo X, Panasonic X700, Panasonic X800, Samsung SGH-D730, SGH-D720 ve Samsung SGH-Z600. Nokia 60 serisi cihazları daha da geliştirerek tekrar tekrar kullanmıştır, örneğin 6620, 6630 ve 6680 cihazları bu gruba dahil edebiliriz. Sonları N serisi olarak adlandırılan cihazlar da aslında 60 serisi cihazlardır. Örneğin Nokia N70, N90 gibi.
- Nokia S90 cihazlar (2004) : Nokia 7710 (2004) bu arayüzü kullanmaktadır.

3.1.1.3. SmartPhone – Akıllı Telefonlar

Akıllı telefon kavramı cihazlar üzerinde işletim sistemi ve bu sayede yazılım geliştirme, üçüncü parti yazılımların kurulması gibi işlemlerin yapılabilmesinden sonra ortaya çıkmış bir kavramdır. Akıllı telefon için kesin bir tanımlama yapılamasa da kullanıcının bilgilerini organize eden, bilgisayarlar gibi yazılım geliştirme-kurma işlemlerini destekleyen

cihazlara genelde akıllı denmiştir. Akıllı cihazların en genel özelliği üzerinde Symbian gibi bir işletim sistemi bulunmasıdır.

Bu cihazların tercih edilmesinin sebebi cihazın genişletilebilmesidir. Yani üzerine kurulan yeni yazılımlarla cihazın davranışı, çalışması değiştirilebilmektedir. Örneğin akıllı grubuna dahil edilen bir cihaz aldığınızda telefon rehberi, oyunlar, arama listesi gibi temel özellikleri barındırmasa bile internet üzerinden bu tür yazılımları bulup kurarak cihazınıza bu özellikleri kazandırabilirsiniz. İkinci bir örnek olarak ise bilgisayarda sıkça kullandığınız özellikleri gezgin cihazınıza kazandıracak yazılımların kurulmasını verebiliriz. Mesela Powerpoint sunularını açma ve okuma, PDF dökümanlarını görüntüleme, oyunlar indirme ve oynama gibi. Bu durumda akıllı cihazları sadece telefon olmaktan kopup gittikçe bilgisayara yaklaşan cihazlar olarak tanımlayabiliriz.

Akıllı cihazların genel olarak desteklediği ve içerdiği ortak uygulamalar ve özellikleri şöyle sıralayabiliriz :

- Internet ve e-posta erişimi : İlk telefonlar sadece WAP desteği verirken bu cihazlar HTML yorumlama, geniş ekran ve daha hızlı internet bağlantısı ile bir bilgisayar ile internet erişimine yakın bir internet erişimi sağlamaktadırlar.
- Takvim, Randevu, Rehber Yönetim Yazılımları
- Dahili kamera
- Office, PDF gibi dökümanları açmak için gerekli yazılımlar

Şekil 3.2. Opera web tarayıcısı kurulmuş bir Sony Ericsson akıllı telefon: P910i

Opera masaüstünde görmeye alıştığımız, popüler bir web tarayıcıdır. Sony Ericsson P910i geniş ekranı ile internette gezinti için ideal bir cihazdır.

Şekil 3.3. Çeşitli uygulamaların listelendiği, üzerinde Windows işletim sistemi çalışan bir akıllı telefon : Sagem My S-7

Resimden de görüldüğü gibi cihaz üzerinde Internet Explorer, MSN Messenger gibi masaüstünde görmeye alıştığımız uygulamalar çalışmaktadır.

3.1.1.4. S60 (Series 60) platformu

60 Serisi Nokia tarafından geliştirilen Symbian işletim sistemi ve üzerinde yapılan eklentilerden oluşan bir platformdur. Nokia telefonlar dışında başka üreticilerde zaman zaman kullanmışlardır. Bu firmalardan bazıları LG, Panasonic ve Samsung'dur. S60 serisi cihazlar standart standart Symbian cihazlarının genellikle çoklu ortam çalıcıları, telephony gibi uygulamalar veya kütüphaneler ile güçlendirilmiş halidir. Bu cihazların en genel özelliklerini şöyle listeleyebiliriz :

- Ekran genişliği 176x208 pikseldir. Fakat zaman içinde birden fazla çözünürlük türü desteklenmeye başlanmıştır. Örneğin S60 2. Sürümden itibaren 240x320 ve 352x416 ekran genişlikleri de desteklenmeye başlanmıştır. Örneği Nokia N90 352x416 ekran genişliğini destekleyen ilk S60 cihazdır.
- Java Micro Edition ve Symbian C++ uygulamaları desteklenmektedir. Bu dillerde yazılmış uygulamalar JAR veya SIS uzantılı olarak sisteme kurulabilmektedir.
- Kolay ve hızlı kullanım için tasarlanmıştır.
- Uygulama geliştiriciler için kolaylıklar sağlayan kütüphane ve geliştirme çatısı içermektedir (Framework).

3.1.1.5. 3G Cihazlar ve İnternet Uygulamaları

Üçüncü nesil cihazlar internete bağlanmak için giriş bölümünde bahsettiğimiz bağlantı türlerinden birisini kullanabilirler. Bağlantı üzerinden web sunucular ile iletişim kurarken kullanılacak protokolleri iki ana gruba ayırabiliriz :

1. Klasik WEB
2. WAP

Üçüncü nesil cihazlar gittikçe gelişen donanımsal ve yazılımsal özelliklerle masaüstü cihazlar için tasarlanmış, genel olarak XHTML ve HTML dilinde kodlanmış sayfaları görüntüleyebilirler, yani klasik web'e erişebilirler. Fakat sayfaların düzgün görüntülenebilmesi için cihazın ekran, renk derinliği, işlemci hızı gibi özelliklerinin gerçekten üst seviyede olması gerekmektedir. Bu yüzden her üçüncü nesil cihaz HTML olarak kodlanmış sayfaları düzgün görüntülemeyebilir.

Gezgin cihazların ilk yaygınlaşmaya başladığı zamanlardan beri kullanılan diğer protokol ise WAP'dır. WAP siteleri HTML yerine WML ile kodlanmış sayfalardan oluşur, WML HTML'ye göre daha basit ve daha az elemandan oluşan bir kodlama türüdür, en önemli özelliği ise gezgin cihazların ekran, işlemci ve renk seçenekleri göz önünde bulundurularak tasarlanmış bir kodlama türüdür. Dolayısıyla ikinci nesil cihazlardan itibaren çoğu cihaz WML ile kodlanmış sayfaları oldukça düzgün görüntüleyebilmektedir. WAP bu avantajlarına rağmen HTML'nin ve Web'in zengin arayüz seçenekleri sebebiyle geliştirilmeye ihtiyaç duyulmuş ve ortaya WAP 2.0 fikri atılmıştır. WAP 2.0 klasik Web'in zengin seçenekleri ile donatılmış ama gezgin cihazların da eksikleri göz önünde bulundurulmuştur. Aşağıdaki tablo da bu üç protokol kısaca karşılaştırılmıştır.

Çizelge 3.2. WAP protokollerinin karşılaştırılması (Anonim 2004a)

	WAP 1.0	**WAP 2.0**	**Klasik Web**
Görüntü	Siyah-beyaz, küçük ekranlara göre tasarlanmıştır.	Sınırlı renkli ve kısıtlı görüntü imkanlarına sahip ekranlar	Gerçek renk içeren HTML kodlarını görüntüleyebilen ekranlar.
İçerik	CARD-DECK kodlama mantığına sahip WML kodlaması.	Card-Deck mantığınının yanı sıra klasik Web mantığını da içeren bir XHTML uygulaması	HTML, XHTML tabanlı web sayfaları.
Sunucu	WAP sunucu	WAP Proxy arayüzlü Web veya WAP sunucu	Web sunucu
Protokoller	WAP Protokolleri : • WSP (Wireless Session Protocol), • WTP (Wireless Transaction Protocol), • WTLS (Wireless Transport Layer Security), ve WDP (Wireless Datagram Protocol)	WAP Protokolleri WP ve HTTP (Wireless Profiled HTTP), TLS (Transport Layer Security), ve WP-TCP (Wireless Profiled TCP)	TCP/IP üzerinden HTTP
Mimari	Sunucu ve cihazlar arasında WAP geçidi	Sunucu ve cihazlar arasında WAP geçidi	Tarayıcılar doğrudan web sunuculara bağlı.
İstemci taraflı programlama	WMLScript	WMLScript	JavaScript/ECMAScript

WAP ve WEB arasında pek çok yönden karşılaştırma yapmak mümkün, ikisinin de çeşitli avantaj ve dezavantajları var, bunları kullanıcılar açısından özetlersek

- Web sayfalarının sayısının WAP sayfalarına göre oldukça fazla olması, sadece WAP kullanan kullanıcıları rahatsız edici bir durum oluşturmaktadır. Kullanıcılar her iki içeriğe de erişebilmek ve hatta mümkünse her sayfadaki bilgiyi aynı şekilde edinmek istemektedirler. Bu WAP'ın dezavantajlarından birisidir.
- WAP içeriğinin Web içeriğine göre oldukça fakir olması da yine WAP'ın dezavantajlarından birisidir.

- Web içeriğinin masaüstü bilgisayarlar ile ancak düzgün görüntülenmesi WAP'ın Web üzerindeki avantajlarından birisidir.
- WAP üzerinden daha az bilgi akışı ve daha az ücret ödeyerek bağlı kalmak yine WAP'ın avantajlarından birisidir. Çünkü WAP HTTP veya HTTPS yerine bunların daha optimize edilmiş hali diyebileceğimiz bir protokol kullanmaktadır ve bu durumda veri transferi daha az olmaktadır.

WAP protokolü TCP/IP ve HTTP protokollerinden esinlenerek hazırlanmış bir protokoldür. TCP/IP'nin katmanlarına benzer katmanlara sahip daha çok UDP'yi andıran bir protokol olan WAP ile TCP/IP OSI Referans modelinin karşılaştırmasını şöyle yapabiliriz :

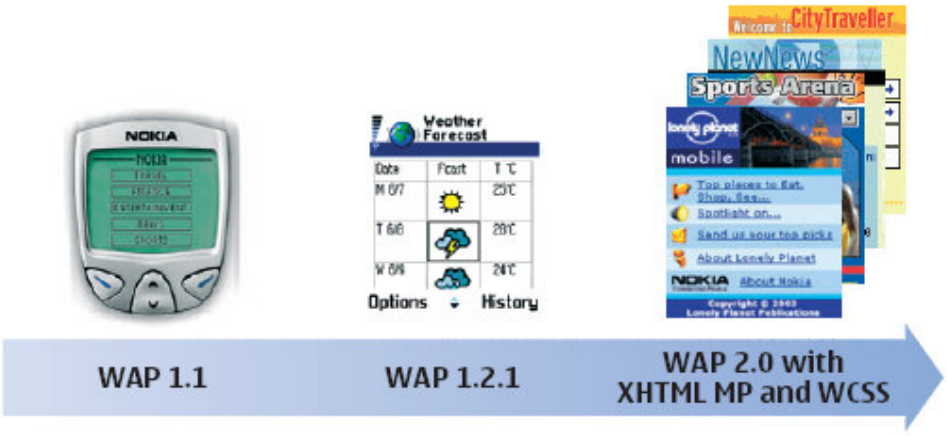
Çizelge 3.3. WAP Protokolü Katmanları

WAP Protokolü Katmanları	
Datagram protokolü	WDP
İletişim ve Güvenlik katmanı	WTLS
İşlem katmanı	WTP
Oturum katmanı	WSP
Uygulama katmanı	WAE

Çizelge 3.4. OSI Modeli Katmanları

OSI Modeli Katmanları	
Fiziksel katman	Physical
Veri katmanı	Data Link
Ağ katmanı	Network
İletişim katmanı	Transport
Oturum katmanı	Session
Sunum katmanı	Presentation
Uygulama katmanı	Application

Önümüzdeki yıllarda WAP atak yapacak fakat ayakta kalmayı başarabilecek mi sorusuna kimse kesin bir yanıt veremiyor. Fakat cep telefonu üreticileri 2006 itibariyle WAP 2.0 için gerekli desteği vermeye başladılar, bu sebeple WAP 2.0 bir süre gündemde kalacaktır.



Şekil 3.4. WAP'ın gelişimi (Anonim , 2006k)

WAP 2.0 ile birlikte yeni teknolojiler kullanılmaktadır :

a) XHTML MP : Sayfaların kodlaması için,

b) WCSS : Sayfaların stil desteğine sahip olması için.

Yeni teknolojilerin varolan WAP teknolojileriyle karşılaştırmasını çizelge 3.5.'de görebiliriz.

Çizelge 3.5. WAP Teknolojilerinin karşılaştırılması

	WML 1.X	**XHTML MP ve WCSS**
Standartların geliştirilmesi	WAP forum tarafından	W3C tarafından geliştirilmiş teknolojiler, OMA tarafından uyarlanmıştır.
Cihazlarda içeriğin görüntülenmesi	İçerik ve yerleşim aynı dökümanda belirtilir ve bu dökümanlardan her cihaz için ayrı ayrı oluşturulmalıdır.	İçerik ve yerleşim farklı dökümanlarda belirtilir, böylece her cihaz için sadece farklı stil dosyası tanımlayarak aynı içeriği kullanması sağlanabilir.
İçerik kodlama	İkili olarak kodlanır.	Gerek yok
Belge yerleşim kontrolü	Basit	WCSS ile gelişmiş kontrol
Renk desteği	Sadece renkli gif'ler. Tam font ve arkaplan desteği yok.	WCSS ile tam font, renk, arkaplan v.b. desteği.

WAP 2.0'ın protokoller bazındaki en büyük getirilerinden bir tanesi, TCP/IP üzerinden iletişime izin vermesi olacaktır. Gezgin cihazların TCP/IP'den türetilmiş bir protokol olan wTCP/IP (Wireless Profiled TCP/IP) protokolünü kullanması öngörülmüştür. Bu yeni protokol TCP/IP'in sağladıklarının tümünü sağladığı gibi bant genişliği için bazı optimizasyonlarda içermektedir, dolayısıyla transfer edilen veri miktarında azalma sağlanmaktadır. wTCP/IP atasına göre daha iyi hata kontrolüne sahip ve daha hızlı bağlantılar sağlamaktadır.

3.1.1.6. 3G, Internet ve Güvenlik

İnternette bir sistemin güvenli olabilmesi için aşağıdaki koşulları sağlayabilmesi gerekmektedir.

- **Mahremiyet :** Bir bilgi gönderilirken, gönderen ve alıcı arasına kimse girememelidir. Girse bile gönderilen veriyi, bilgiyi elde edememelidir.

- **Bütünlüğün korunması:** Gönderilen mesaj alıcıya olduğu gibi gitmelidir. Araya girmek isteyenler olduğunda yaptıkları değişiklikler alıcı tarafından kolayca ve açıkca anlaşılır olmalıdır.
- **Doğrulama :** Gönderilen bilgiyi alan hedef, kaynağın gerçek ve doğru kişi olduğunu doğrulayabilmelidir.
- **İnkar edememe :** Bilgiyi gönderen daha sonra bu bilgiyi gönderdiğini rededememelidir.

Günümüz internetinde bu özellikler

- Şifreleme (Encryption)
- Şifreleme ile anahtar çıkarma (Hashing)
- Dijital imzalar ve dijital sertifikalar (Digital signatures ve Digital Certificates)
- Parola ile koruma (Password protection)

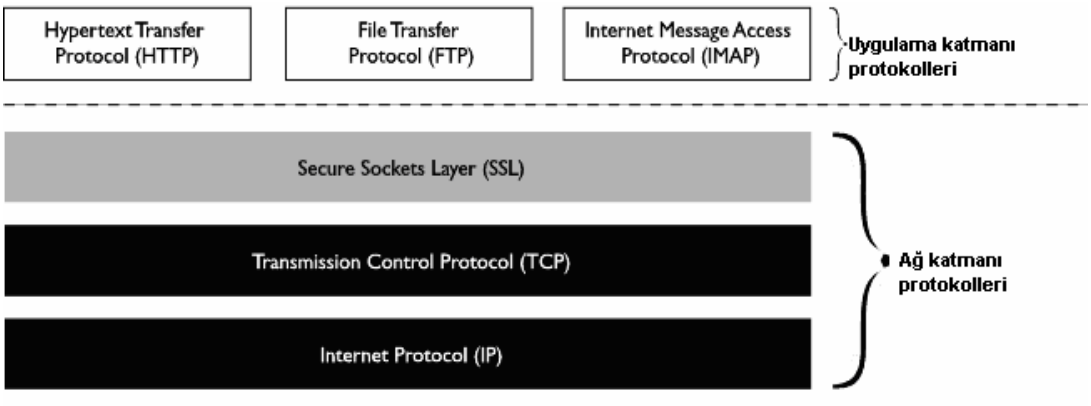
işlemleri ile sağlanmaktadır. Bu özelliklerin hepsinin de aynı anda sağlanması önemlidir, çünkü internet çeşitli sebeplerden güvenli bir ortam değildir. İnternette bilgi bir bilgisayardan diğer bilgisayara doğrudan gitmemektedir, arada pek çok bilgisayardan geçmektedir. Bunlara "Router" denir. Bu bilgisayarlarda yetkili birisi gönderilen tüm verileri inceleyebilir. Bunun yanında internet kullanıcılarının çoğu internete bir LAN vasıtasıyla bağlanmaktadır, dolayısıyla internete veri gönderilmeden önce de kullanıcının bağlı bulunduğu veri ağda çeşitli bilgisayarlara uğramaktadır, yine bu bilgisayarlarda yetkili birileri bu veriye kolayca erişebilir. Bu yüzden internette gönderilen veri, ilk bilgisayardan son bilgisayara sadece alıcının bilebileceği şekilde paketlenmiş olarak gönderilmektedir.

3.1.1.7. SSL (Secure Sockets Layer) ve TLS (Transport Layer Security)

İnternet güvenliğine yönelik çözümlerden ilki Netscape firması tarafından 1994 yılında geliştirilmiştir. Firma tarayıcısına, TCP/IP'nin üstüne bina edilmiş yeni bir protokol katmanı yerleştirmiştir ve bu katmana SSL adı verilmiştir. SSL temel olarak bazı matematiksel işlemler yaparak verinin alıcı-gönderici arasında şifrelenmesi sağlayarak güvenliğin temel esaslarının tümünü sağlamaktadır. Bu şifreleme mekanizması PKI (Public Key Infrastructure) tarzında olmaktadır, yani anahtar paylaşımı esasında şifrelenme bir anahtar ile yapılmaktadır.

SSL ilk zamanlarda Netscape tarafından kullanılırken, sonraları tüm internet altyapısında kullanılır hale gelmiştir. Günümüzdeki populer tarayıcıların hepsi SSL'yi desteklemektedir. Örneğin Internet Explorer, Firefox, Opera gibi. Çünkü 1990'larda gelişmeye başlayan ve 2000'lerde patlayan E-Ticaret (E-Commerce) SSL tekniğine sıkı sıkıya bağlanmıştır, bunun sebebi ise kullanıcının gizli kalması gereken kredi kartı gibi bilgilerininin ticaret amacıyla kullanılmasıdır. SSL daha sonraları TLS (Transport Layer Security) olarak geliştirilmeye devam edilmiştir. TLS'nin 2006 yılı itibariyle en geçerli versiyonu 1.1 dir.

SSL'nin herhangi bir internet tabanlı protokol ile beraber kullanılabilmesi onun yaygınlaşmasında önemli sebeplerden olmuştur. Örneğin HTTP, FTP, IMAP veya POP protokollerinden herhangi birisi de SSL'yi kullanabilmektedir. Şekilde bu entegrasyon gösterilmiştir. Şekil 3.5'de görüldüğü üzere SSL OSI modelinin ağ katmanlarının en üstünde bulunmaktadır.



Şekil 3.5. SSL protokolünün OSI modelindeki yeri

SSL temel güvenlik aşamalarını sağlamak için aşağıdaki metodları kullanmadır :

- **Şifreleme (Cryptography).** Bu mesajları başkalarının araya girerek görmemesi için kullanılan bir bilimdir.
- **Mesaj karıştırma (Message hashing).** Mesaj matematiksel bir işlem dizisine sokularak sonuç olarak mesajdan bir anahtar üretilir. Bu anahtara "Fingerprint" denilir. Eğer mesajın sadece bir harfi veya karakteri değişirse bu mesajdan yeni bir "fingerprint" oluşturulur ve ilk parmak izine hiç bir şekilde benzemez. Bu yolla mesajın değişip değişmediği alıcı tarafından kolayca bilinir.
- **Dijital Sertifika (Digital certificates).** Dijital sertifika bir şahsın kimliğini doğrulayan kısa elektronik dökümandır. Dijital sertifikanın geçerli olabilmesi için bir Sertifika Otoritesinden (Certificate Authority- CA) tarafından onaylanması gerekmektedir. Örneğin VeriSign (www.verisign.com) veya GeoTrust (www.geotrust.com) firmaları bu hizmeti para karşılığı yapan firmalardır. Böylece internet kullanıcısı bilgiyi gönderdiği adresin sahibini tanıyabilmektedir.
- **Dijital İmzalar (Digital signatures).** Dijital imza – günlük hayatta olduğu gibi – mesajın kime ait olduğunu gösteren elektronik bir tekniktir. Teknik olarak ise, bir dijital imza ile mesaj tekrar biçimlendirilir ve bunu sadece ama sadece alıcı kişi tekrar eski haline getirebilir ve okuyabilir. Dijital imzalar ile "inkar edememe" sağlanmaktadır.

SSL şifreleme işlemlerini yoğun bir şifreleme algoritmalarını kullanır, bunlardan bazıları :

- **PKI için :** RSA, Diffie-Hellman, DSA veya Fortezza
- **Simetrik Şifre (Symmetric Cipher) :** RC2, RC4, DES, Triple DES, AES ve Camellia
- **Tek yönlü Şifreleme (One-Way Hashing) :** MD2, MD4, MD5 ve SHA

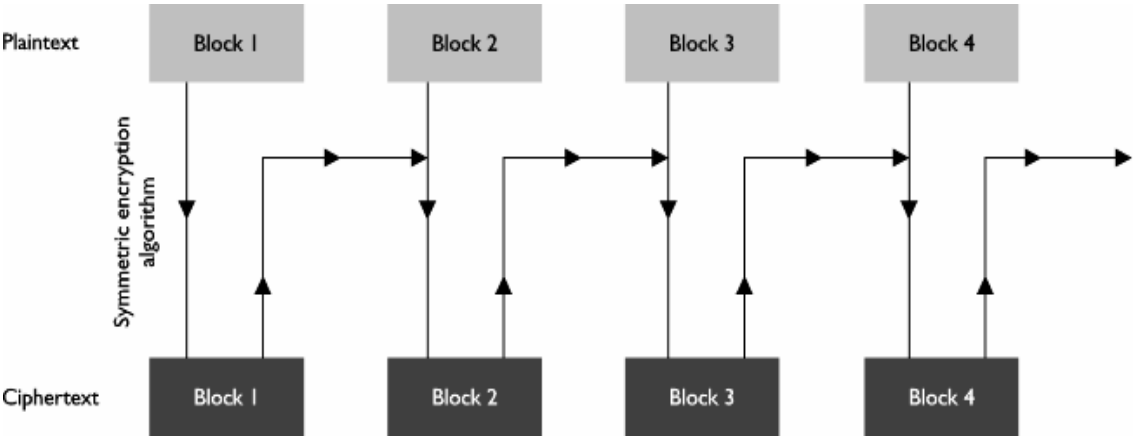
Simetrik Anahtar Şifreleme (Symmetric Key Cryptography)

SSL verinin mahremiyetini/özelliğini korumak için şifreleme kullanır. Verinin alıcıya gönderilirken şifrelenmesine encrpytion, alıcı da geri çevrilmesine decryption denmektedir. Şifreleme yapılırken (encryption) veri belli bir sayısal değeri kullanan matematiksel işlemlerden geçirilir, daha sonra bu veriyi ancak o sayısal değeri bilen bir alıcı tekrar geriye çevirebilir.

Örnek bir basit şifreleme metodu şu olabilir. Anahtar değer 9 olsun, metin (PlainText) "Merhaba olsun". Algoritmamız ise mesajın içindeki her karakterin sayısal koduna (ASCII) 9

değerini ekleyip yeni bir metin çıkarmak olsun. Bu durumda ortaya çıkacak yeni metin (CipherText) "Vn{qjkj" olmaktadır. Bu metni geri çevirmek isteyen kişinin yapması gereken kullanınan algoritmanın tam tersini uygulamaktır. Bu örnekte karakterlerin kod değerlerinin 9 ile bölünmesidir. Buna simetrik şifreleme denmektedir, çünkü her iki tarafta da aynı algoritma kullanılmaktadır.

İnternette kullanılan şifreleme algoritmaları elbetteki bu örnektekinden çok daha karmaşıktır. İlk farklılık şifreleme algoritmasının karakter karakter okuma-değiştirme yerine karakterlerin bit bloklarını alıp işlemeleridir. Buna "Block Cipher Encryption" denmektedir. Bloklar halinde veri çeşitli dönüşüm işlemlerine tabi tutulmaktadır ve genelde anahtar oldukça uzundur, örneğin 112 bit veya 128 bit gibi. Ve bununla birlikte blok zincirlemesi (cipher block chaining) denen bir teknik de uygulanmaktadır, bu teknikte bir dönüşümden çıkan sonuç diğer dönüşüm için girdi olarak kullanılmaktadır. Bu sayede metnin içinde tekrarlanan veri sonuç metninde kaybolmaktadır ve böylece üretilen sonuçtan geri dönmeye çalışan kötü niyetli kişilerin işi oldukça zorlaşmaktadır (Şekil 3.6).



Şekil 3.6. Blok zincirleme işlemleri (Anonim 2002a)

3.1.1.8 .PKI – Açık Anahtar Şifreleme (Public Key Cryptography)

Simetrik şifreleme yöntemleri gerçekten verimli sonuçlar üretmesine rağmen İnternet tabanlı tüm uygulamalar için uygun değildir. Çünkü simetrik şifreleme yapılabilmesi için önce anahtarın güvenli bir şekilde gönderenden alıcıya teslim edilmesi gerekmektedir. Dolayısıyla bu işi sadece simetrik şifreleme ile yapamayacağımız oldukça açıktır.

Bu dezavantajı aşmak için başka bir şifreleme tekniği kullanılmaktadır : Açık anahtar ile şifreleme (Public key cryptography). Aynı zamanda bu tekniğe asimetrik şifreleme de denmektedir. Asimetrik denmesinin sebebi gönderen ve alıcının aynı algoritmayı kullanmadan şifreleme-çevirme (encryption-decryption) yapmalarıdır. Bu algoritma türü matematiksel olan tek-yol fonksiyonlarını (one-way function) yoğun olarak kullanmışlardır.

Bu algoritma türünün temelinde alıcının şifreleme ve çözme işlemini iki ayrı anahtar ile yapmasıdır. İşlemler şöyle gerçekleşir :

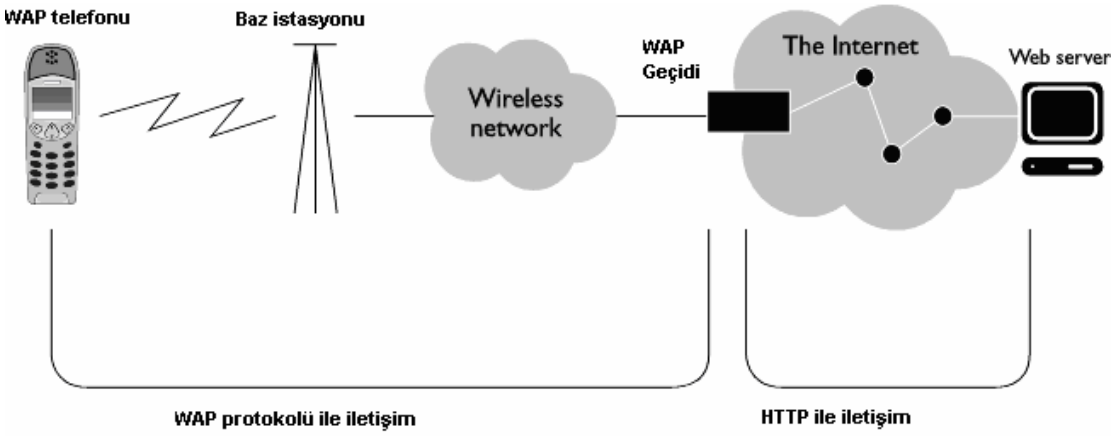
1. Alıcının açık anahtarını gönderene iletmesi
2. Gönderenin açık anahtar ile metni şifrelemesi
3. Alıcının açık anahtar ile şifrelenen metni alması
4. Alıcının gizli anahtar ile şifrelenen metni çözmesi

Görüldüğü gibi bu sistem ile simetrik algoritmalardaki anahtar iletimi problemiş aşılmış olup, oldukça güvenli bir altyapı oluşturulmaktadır.

3.1.1.9. Mobil İnternet ve Güvenlik

Mobil cihazlar klasik internet sitelerini kullanırken yine SSL/TSL kullanırlar. Ve bu sayede mahremiyet, doğrulama, koruma gibi temel güvenlik esaslarını sağlarlar. Fakat SSL/TSL'nin bazı dezavantajları gezgin cihazlarda iyice ortaya çıkmaktadır. Örneğin SSL yoğun şifreleme işlemleri yaptığı için, işlemci yükü oldukça fazladır ve gezgin cihazlar kısıtlı olanaklarıyla SSL için çok uygun değillerdir.

Gezgin cihazlar WAP tabanlı sitelerde ise farklı bir güvenlik protokolü kullanmaktadırlar : WTLS (Wireless Transport Layer Security). Bu protokol WAP protokolü içinde seçimlik bir seçenektir ve WDP ile WTP protokolleri arasına yerleştirilmektedir. Gezgin cihaz olarak WAP'a bağlanmak istediğinde önce sunucu yerine WAP Gateway denilen geçide bağlanır ve bu geçit isteği standart internet isteğine dönüştürür. Eğer güvenlik söz konusu ise önce WTLS ile geçide bağlanılır, daha sonra bu istek SSL/TSL kullanılarak HTTP protokolüne dönüştürülür (Şekil 3.7). Dolayısıyla bir dönüşüm olduğu için bu sistemde çeşitli güvenlik boşlukları oluşabilmektedir. Eğer hedef sunucu, geçit (gateway) ve sunucuyu kendi üstünde bulundurursa güvenlik açıkları kapatılabilmektedir.

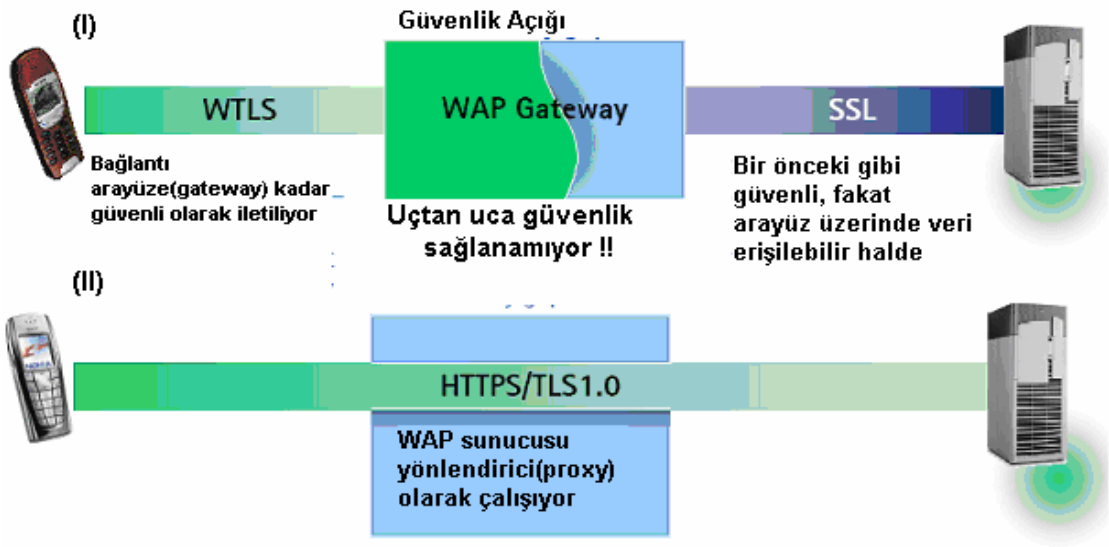


Şekil 3.7. WAP ve HTTP protokolleri ile internet erişimi (Anonim 2002a)

Bu mimarinin çeşitli açıkları internet üzerinde yayınlanmış ve tartışılmaktadır. Birinci olarak WTLS'nin zayıf şifreleme algoritmaları kullanmasıdır. Hatta bazı WAP istemci uygulamaları bu şifreleme seçeneğini de isterlerse kapatabilmektedirler, bu da WAP ağı üzerinde güvensiz bir ortam oluşturmaktadır. İkinci olarak WAP isteğini HTTP protokolüne dönüştüren ve yönlendiren geçidin (gateway) erişilebilir olmasında güvenlik açığı ortaya çıkmaktadır. Nokia ve diğer üreticiler geliştirilen yeni WAP protokolünde bu tür açıkların farkında olduklarını ve kapatmaya çalıştıklarını belirtmişlerdir.

WAP özelliklerinin ileri tarihleri olarak düşünülen sürümleri WAP Forum tarafından açıklanmaktadır. Ve bu sürümlere baktığımızda, WAP İletişim Katmanı E2E Güvenlik özelliklerinde (WAP Transport Layer E2E Security Specification) WTLS'deki bazı açıkların kapatılması yönünde geliştirmeler olduğunu görmekteyiz. Örneğin bu özellik WAP

geçidinden önce bir şifreleme-çözme işlemini çıkarmış ve bunun yerine XML formatında bir dosyanın oluşturulmasına ve içerisinde güvenli olarak yönlenecek sunucu bilgilerinin bulunmasına, bu bilgilerin daha sonra istemciye gönderilmesine karar verilmiştir. Yani istemci WAP geçidine hiç bir veri göndermeyecek, geçite bağlanıp güvenli olarak bağlanabileceği içerik sunucusu ile ilgili bilgileri alacak ve doğrudan bu sunucuya bağlanacaktır. Bu sayede geçit-internet arasında ki dönüşümlerden dolayı açığa çıkan güvenlik problemleri aşılmış olacaktır.



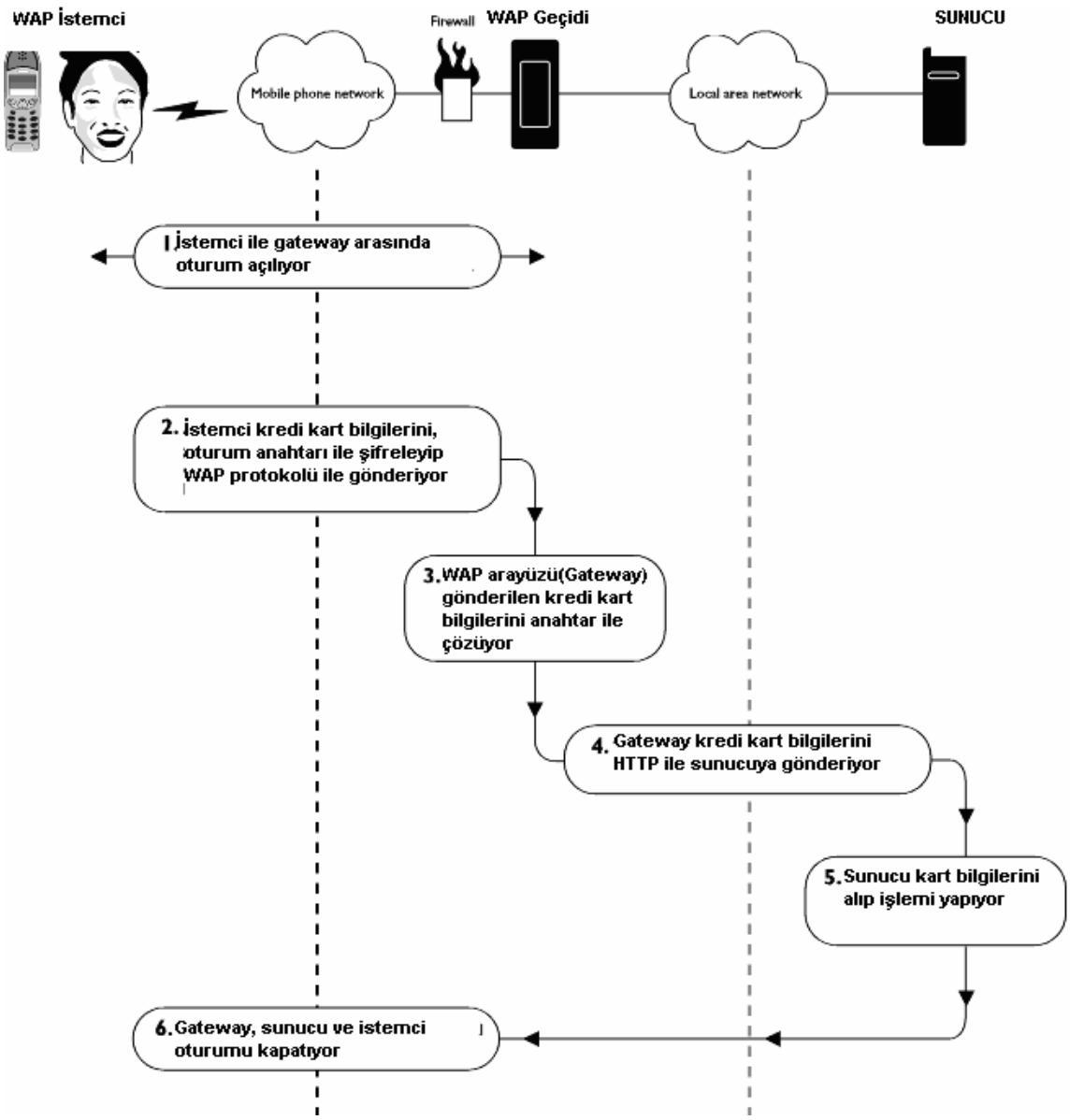
Şekil 3.8. Yeni WAP protokolü ile eskisinin karşılaştırılması (Anonim, 2006k).

Uçtan Uca(E2E) Gezgin İnternet Güvenliğine Genel Bakış

Bir önceki bölümde bahsettiğimiz gibi WTSL gezgin cihazla WAP geçidi (gateway) arasındaki güvenliği sağlamaktadır. Çoğu internet tabanlı işlem ise aynı zamanda WAP geçidi ile web sunucu arasında da güvenli iletişimin yapılmasını gerektirmektedir. Bu iki yolla yapılabilir

- WAP geçidi ile HTTP/Web sunucu arasında SSL/TLS kullanılabilir.Bu metot bir güvenlik açığı içermektedir, çünkü veri WAP geçidine geldiğinde çevrilip tekrar SSL ile gönderime hazırlandığı sırada yabancılar tarafından elde edilebilir olmaktadır. (Buna WAP boşluğu – WAP Gap – denmektedir) Bu yüzden geçide erişim kısıtlanmalı ve veriler hiç bir zaman geçit üzerinde saklanmamalıdır. Eğer aksi durumlar olursa güvenlik zedelenmiş ve banka işlemleri gibi kritik işlemler için hazır sayılamaz.
- WAP geçidi içerik hizmeti veren sunucu ile aynı yerde tutulabilir (hosting). Böylece geçit sunucunun ateş duvarı (firewall) altında güvende olacaktır. Bu teknik ile hep WAP boşluğu engellenmiş hem de sunucu ile geçit arasında SSL gereksinimi ortadan kalkmış olmaktadır. SSL'nin ortadan kalkması maliyet açısından avantajlı bir durumdur. İçerik hizmeti veren kurum böylece tüm internet işlemi bitene kadar bir ISP gibi gezgin cihazın tüm gezinti hizmetini karşılamış olur.

Şekil 3.9.'da bir WAP istemcisi ile sunucu arasındaki iletişim adımlarını göstermektedir. Şekilden de görüldüğü gibi ateş duvarı sunucu ve geçidi korumaktadır ve güvenlik tam olarak sağlanmaktadır.



Şekil 3.9. WAP Geçidinin İçerik Sağlayıcı tarafından host edildiği WTLS işlemi (Anonim 2002a)

Çizelge 3.6. WAP Güvenlik açığını kapatmak için öneriler (İnceoğlu, Kılınç 2003)

Çözüm	Açıklama	Değerlendirme
Ağ geçidini web sunucuya kaydırma	Ağ geçidini, web sunucunun olduğu tarafa kaydırmaktır. Güvenlik zinciri ağ geçidinde yine kırılacaktır ancak bu sadece elektronik ticareti yapan satıcının alanı güvenli değilse zararlı olacaktır ve güvenliği azaltmayacaktır.	MSP'ler için en büyük problem yeni iş imkanlarının (ek hizmet servislerinin) yok olmasıdır. Kablolu ağlarda iletişim trafiği artarsa bu durum istemciye bilgi ulaşmasını geciktirir. İki farklı siteden alış veriş yapmak isteyen bir kullanıcının iki sitenin de ağ geçidini kullanması gerekir. Kullanıcı servis kalitesi gibi problemlerle karşı karşıya kalabilir.
Bağımsız uçtan uca güvenlik	İstemci ve sunucunun hem WAP hem de İnternet protokollerinden bağımsız olarak belirli güvenlik ölçümleri üzerinde anlaşmasıdır. Bu işlem kablolu ve kablosuz ağlar arasındaki ağ geçidi tarafından yapıldığı için, ağ geçidi kullanmamanın dezavantajları ile karşı karşıya kalınacaktır.	WML-Script şifre kütüphanesindeki fonksiyon, istemci tarafta istenen fonksiyonalite için yeterince güçlü değildir. Güvenliğin satıcı tarafından sağlanan programa dayanıyor olması dezavantajdır. Standart ve WAP Forum organizsayonunun desteği eksiktir.
WAP ağ geçidini kullanım dışı bırakma	Bu yaklaşım ile uçtan uca şifrelemeyi kullanan İnternet'teki elektronik ticaret hareketlerinde sağlanan güvenlik seviyesi mobil ortamda da sağlanmış olur.	Ağ geçidinin sağladığı eniyilemeyi de kaybetmek anlamına gelir. Gelecekte daha güçlü mobil telefonlar geliştirilse, kablosuz ağlardaki bant genişliği arttırılsa bile, yüksek miktarda gecikme olması engellenemeyecektir. WAP ağ geçidinin kullanım dışı bırakılması, WAP standardının karmaşıklığını arttıracaktır.

Gezgin cihazlar için şifreleme : Dezavantajlar

PKI ve benzeri şifreleme teknikleri güvenliği sağlama konusunda oldukça başarılı olmalarına rağmen, asıl hedef platformları masaüstü bilgisayarlar olduğu için gezgin cihazlarda çeşitli dezavantajlar yaşamaktadırlar. Bu dezavantajların sebebi her zaman olduğu gibi gezgin cihazların kısıtlı güce ve işlemciye sahip olmasıdır.

Örneğin anahtar boyutunu 1024 bitten 56 bite düşürerek performansını artıran Eliptik Eğri Şifrelemesi (Elliptic Curve) bir masaüstü bilgisayarda bir saniyede yapılırken, ortalam bir cep telefonunda on beş saniye sürmektedir (Anonim 2001a). Bu kullanıcı açısından gerçekten uzun bir zaman olduğundan muhtemelen kullanıcılar bunu beklemeden işlemini yapmayı deneyecektir. Ayrıca bu tür işlemler cihazın pilini de çabuk tüketeceğinden çok fazla bu tür işlem yapıldığında belki de cihaz gücü bittiği için kapanacaktır.

Bu yüzden WAP 2.0 ile beraber kullanılacak yeni bir şifreleme tekniği olan WPKI önerilmiştir ve yakın zamanda kullanıma başlanacaktır.

3.1.2. Yazılım

3.1.2.1. Geliştirme için kullanılacak diller

3.1.2.1.1.Java-J2ME

1990'lı yılların ikinci yarısında ortaya atılan "Java" projesi ancak 2000'lere doğru popülarite kazanmaya başlamıştı. Sloganı "Java heryerde" olan proje, tüm cihazlar, masaüstü, sunucu, telefon ve hatta buzdolapları için tek bir platform tek bir dil amacıyla yola çıkmıştı. Ve bir süre sonra da hedeflerine ulaşmaya başladı. Java platformu oldukça genişlemişti. SUN firması 2001 yılında Java platformunu üç ana kategoriye ayırmaya karar verdi :

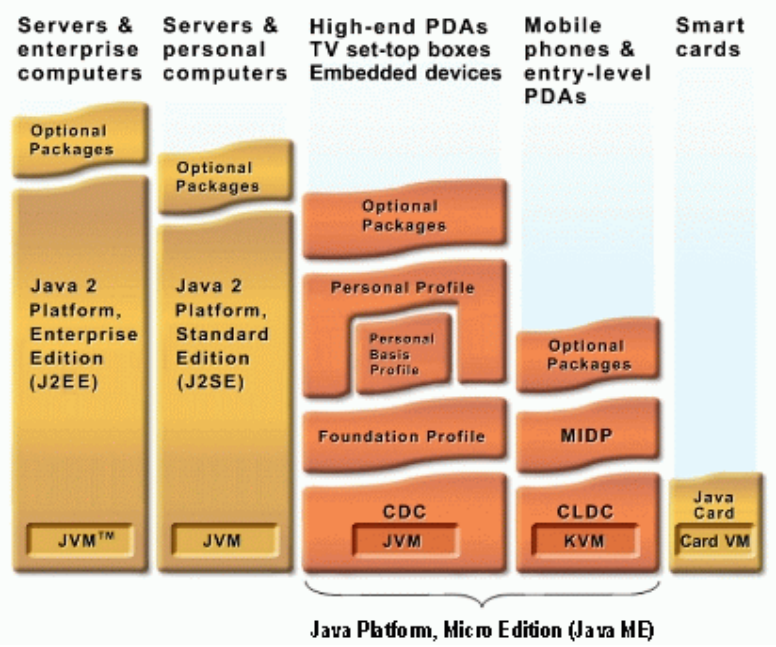
- **Java SE** : Masaüstü pazarına yönelik olan bu bölüm ile Windows, Linux veya MacOS üzerinde yazılacak herhangi bir program tüm platformlarda hiç değiştirilmeden çalışacaktır. Bu herhangi bir işletim sistemi API'sine bağlı kalmadan hepsinde programlama yapmak isteyenler için mükemmel bir fırsattı.
- **Java EE** : Sunucu pazarına yönelik bir ürün olan "Enterprise Edition" ile SUN firması dağıtık uygulamalar, ağ tabanlı uygulamalar, veritabanı uygulamaların üretmek isteyenlere bu platformu sunmuştu. Oldukça başarılı olan bu platformda sonraları özellikle finans ve bankacılık uygulamalarında güvenlik ve sağlamlık sebebiyle en çok tercih edilen platform olacaktı.
- **Java ME** : Kısıtlı görüntü ve işlemci imkanlarına sahip cihazlara yönelik olan platformun bir başka parçasıdır. Özellikle PDA, Palm ve Cep telefonlarında çok yaygınlaşmıştır. Günümüzde piyasadaki cep telefonlarının hemen hepsi Java (Java ME) platformunu desteklemektedir.

2006 yılında SUN firması tüm Java platformunun kaynak kodlarını açmıştır. Java ME kaynak kodları ile yeni bir proje başlatılmış ve projeye phoneME denilmiştir. Java ME ile ilgili önemli bir nokta da, platformun bu parçasının SUN firması tarafından son kullanıcıya şimdiye kadar hiç verilmemiş olmasıdır, SUN bunun sebebini "Cep telefonlarına Java ME desteği vermek üreticilerin işidir" şeklinde yanıtlamıştır. Fakat cep telefonu üreticilerinin standartları takip etmekte disiplinli olmamaları sonucunda geliştiriciler her cep telefonunda farklı kodlama yapmak zorunda kalmışlardır. Bu da Java platformunun "Bir kere yaz, her yerde çalıştır" sloganına ters düşen bir durumdur.

3.1.2.1.1.1.Configuration ve Profile

J2ME konfigürasyonu bir aygıt ailesi için minimum Java platformunu tanımlar. Bu aileye ait aygıtlar benzer bellek ve işlemci yeteneklerine sahiptir. Bir konfigürasyon, sistem düzeyinde programcının o aygıtın desteklediğini bildiği bazı dil ve platform özelliklerinin tanımlamasıdır. Belli bir konfigürasyona giren aygıtlarda hangi Java kütüphanelerinin olduğunu bilir ve ona göre program yazabilir. Özet olarak, bir konfigürasyon şu maddeleri içerir:

- Belli Java programlama dili özellikleri
- Belli Java sanal makinesi özellikleri
- Belli Java kütüphaneleri



Şekil 3.10, Java Platformu bileşenleri (SUN, 2006)

J2ME profili (profile) uygulama düzeyinde verilen servislerden oluşur. Verilen servis herhangi bir konuda olabilir. Ancak belli konulardaki profiller standarttır, o profili destekleyen bütün aygıtlarda aynı servis aynı şekilde bulunur. Profiller bir konfigürasyon üzerinde çalışırlar. Ayrıca bir profil bir başka profil üzerinde çalışabilir. Bir aygıta birden fazla profili destekleyebilir ancak sadece tek bir konfigürasyona sahip olur. Örneğin SMS Mesajlaşması bir profil konusudur. Bu, gezgin telefon konfigürasyonu için çok yaygın bir profile'dır (Aksu 2004).

3.1.2.1.1.2. J2ME'de Tanımlanmış Konfigürasyonlar

Konfigürasyonlar Java 2 Standart Edition (J2SE)'daki bazı sınıfları (class) içerebilirler. Ancak hepsini içermeleri, bu cihazların bilgisayara göre daha az teknik özellik içermesi nedeniyle mümkün değildir. J2SE'deki bir sınıf belli bir konfigürasyonda tanımlıysa mutlaka aynı şekilde tanımlı olmak zorundadır. Farklı bir metot ekleyemez. Buna karşın eski metotları tanımlamak zorunda değildirler.
J2ME iki konfigürasyon tanımlar:

- **CLDC (Connected, Limited Device Configuration) :** Kişisel, kesikli bağlantılı gezgin aygıtlar için.
- **CDC (Connected Device Configuration) :** Sürekli bağlantılı ağ aygıtları için.

CDC daha gelişmiş (örneğin 2 MB'tan fazla belleğe sahip aygıtlar) için tanımlanmışken CLDC daha az gelişmiş (örneğin 160 KB kadar düşük bellekli) aygıtlar için tanımlanmıştır. CDC tanımlaması CLDC'yi tümüyle kapsar. Yani CLDC'de tanımlanmış bütün servis ve kütüphaneler CDC'de de bulunur (Aksu 2004).

3.1.2.1.1.3. CDC (Connected Device Configuration)

CDC, J2SE'de bulunan birçok özelliği barındırmaktadır. CDC için CVM (Compact Virtual Machine) sanal makinesi geliştirilmiştir. JVM (Java Virtual Machine)'de olan hemen hemen bütün özellikleri içermektedir. Ancak sınırlı kaynağa sahip aygıtlar için tasarlanmıştır.

CDC'de tanımlanmış paketler:

- java.lang, java.lang.ref, java.lang.reflect
- java.util, java.util.zip, java.util.jar
- java.net
- java.io
- java.text
- java.security, java.security.cert
- java.math
- javax.microedition.io

CDC konfigürasyonu çeşitli profillerle tamamlanmıştır. İlk olarak tanımlanan profil Foundation Profile'dır. Bu profil java.lang, java.util, java.net, java.io ve java.security'deki eksiklikleri (J2SE'de bulunup, CDC'de bulunmayan nesneleri) eklemektedir. Bu profille birlikte CDC bir programlama ortamı sağlamış olmaktadırlar. Ancak bu ortamda AWR veya Swing gibi bir ara yüz yoktur. Bir programda ara yüz kullanmak için Personal Profile gibi bir profili desteklemelidir. Personal Profile'i bütün AWT (Absract Windows Toolkit) paketlerini desteklemektedir. Ayrıca java.beans ve jabcax.microedition.xlet paketleri bulunmaktadır. Personal Profile'la birlikte Personal Java adlı Java sisteminde J2ME'a bir geçiş yolu sağlamış olur. Bir başka profil de RMI (Remote Method Invocation - Uzaktan Yöntem Çağırma) Profile'dır. Bir nesnenin ağda başka bir aygıtta çalışan bir nesnenin herhangi bir metodunu doğrudan çağırmasını sağlayan RMI nesnelerinin hemem hemen tamamını desteklemektedir (Aksu 2004).

3.1.2.1.1.4. CLDC (Connected, Limited Device Configuration)

Java 2 Micro Edition'da tanımlanmış ikinci konfigürasyon CLDC (Connected, Limited Device Configuration) düşük bellekli, şarj gibi zayıf güç kaynakları olan, kablosuz bağlantılı aygıtlar için tanımlanmıştır.
CLDC'de Java dili desteği olarak bazı özellikler eksik bırakılmıştır: Float (kesirli) tipi Object Serialization ve Error class'ı gibi. Float'ı desteklememesinin nedeni söz konusu düşük yetenekli aygıtların kesirli sayıları destekleyen donanım birimlerinin olmamasıdır.
CLDC'yi destekleyen KVM (Kilobyte Virtual Machine) adlı bir VM (Virtual Machine) bulunmaktadır (Bir kaç KB'lik hafızaya sığdığı için bu ismi almıştır). Bu konfigürasyon şu paketlerden sınıflar içerir:

- java.lang
- java.util
- java.io
- javax.microedition.io

(Aksu 2004)

3.1.2.1.1.5. J2ME'de Tanımlanmış Profiller (Profile)

MIDP Profili

MIDP (Mobile Information Device Profile) mevcut profiller arasında ilk ve en yaygın olanıdır ve uygulamanın yaşam döngüsü, kullanıcı grafik arabirimleri, iletişim ağı ve kalıcı veri depolama ile ilgili kütüphanelerini içerir. MIDP, CLDC konfigürasyonunun üzerine kurulmuştur ve günümüzde Motorola, Nokia, Ericsson ve RIM (Blackburry) gibi sektörün devleri tarafından desteklenmektedir.

MIDP 1.0'ın desteklediği paketler:

- java.io
- java.lang,java.util
- javax.microedition.io
- javax.microedition.lcdui
- javax.microedition.midlet
- javax.microedition.rms

MIDP 2.0'ın desteklediği paketler:

- java.io
- java.lang
- java.util
- javax.microedition.io
- javax.microedition.lcdui
- javax.microedition.lcdui.game
- javax.microedition.media
- javax.microedition.media.control
- javax.microedition.midlet, javax.microedition.pki
- javax.microedition.rms

(Aksu, 2004)

3.1.2.1.1.6 MIDlet

Bir MIDlet Java ME sanal makinesi üzerinde çalışan, telefon, PDA gibi kısıtlı imkanlara sahip cihazlar üzerinde çalışmak için tasarlanan bir Java programıdır. Yani java uzantılı bir dosyanın derlenerek cihaza aktarılmasından ibarettir. Fakat klasik java programlarına göre bir MIDlet programının sağlaması gereken bazı ekstra özellikler vardır:

- Ana sınıf javax.microedition.midlet.MIDlet sınıfından türeyen bir sınıf olmalıdır.
- Derlenen MIDlet sınıfı jar olarak paketlenmelidir. Bunu yapabilmek için Java SDK içinde dahil edilmiş "jar" uygulaması kullanılabilir.
- Jar dosyası "preverify" denen işleme tabi tutulmalıdır.
- Bazı durumlarda, özellikle uygulamanın cihazın depolama, bağlantı gibi özelliklerini kullanması gereken durumlar gibi, uygulama uygulamayı üreten tarafından imzalanmalıdır. Bu güvenlik açısından konulan bir tedbirdir.

Aşağıda hiçbir şey yapmayan ama MIDlet olarak isimlendirebileceğimiz bir java kod parçasını görebilirsiniz.

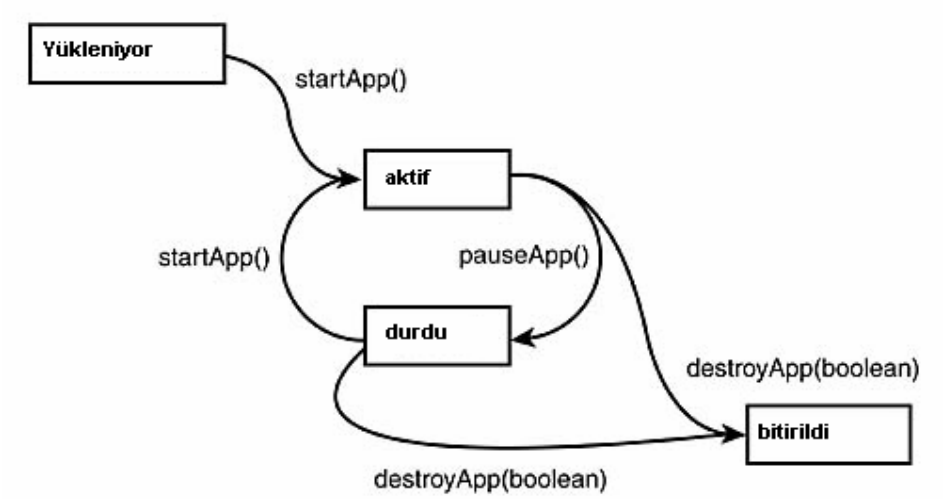
```
import javax.microedition.lcdui.*;
```

```
import javax.microedition.midlet.*;

public class MIDletDeneme  extends MIDlet
{
  public HelloMIDlet() { }

  public void startApp() { }

  public void pauseApp() {}

  public void destroyApp(boolean unconditional) {}

}
```

Bir MIDlet sınıfı için üç temel metod vardır. Bu metodlar gerçekleştirilmezse MIDlet düzgün çalışmaz.

- **startApp() :** Uygulama cihaz tarafından başlatıldığında bu metot cihaz tarafından ilk olarak çalıştırılır. Eğer uygulama içinde başlatılmasını istediğimiz herhangi bir ilk işlem, ilk değişken ataması varsa bu metot içinde yapmak gerekmektedir. Aslında çalışan ilk metot yapıcı metottur fakat yapıcı metodu Java dilinin genel özelliği olarak düşünürsek, bir MIDlet açısından ilk metot startApp metodudur.
- **pauseApp():** Bu metot yine cihaz tarafından otomatik olarak uygulamanın durdurulması gerektiği durumlarda çağrılır. Örneğin uygulama çalışırken gelen bir aramada cihaz uygulamayı durdurabilir, bu tamamen durdurmak veya bitirmek değildir, arama bitince tekrar uygulama kaldığı yerden devam edecektir. Programcının durdurma ve devam etme işlemleri için bir şey yapmasına gerek duyulmaz, bunların hepsinden cihaz sorumludur. Fakat eğer programcı uygulama durdurulduğunda bir şeyler yapmak istiyorsa bu metodun içine işlemleri yapan kodları yazmalıdır.
- **destroyApp() :** Uygulama cihaz tarafından sonlandırılırken bu metot içindeki kodlar çalıştırılır. Programcı burada bazı bellek boşaltma, sonlandırma işlemlerini kendisi yapmak isteyebilir. Aslında Java otomatik çöp toplayıcısına sahip olduğu için (Garbage Collector – GC) çoğu temizlik bu toplayıcı tarafından yapılmaktadır. Fakat gezgin cihazlar bellek kullanımı açısından kritik seviyede olduğu için bazen anından temizliği yapılması en uygun hareket olabilmektedir. Bunu düşünen Java üreticileri bu özelliği destroyApp ile programcının yapabilmesini sağlamışlardır.

Bir MIDlet içinde bunlardan başka metotlar da bulunabilir, bunda sakınca yoktur. Fakat bu metotlar özeldir ve bulunmak zorundadır, çünkü cihaz sadece bu metotları çalıştırmakla yükümlüdür, geriye kalan ise programcının işidir. Şekil 3.11'de bir MIDlet'in yaşam döngüsü gösterilmiştir.



Şekil 3.11. MIDlet yaşam döngüsü (Anonim 2006f)

3.1.2.1.1.6 MIDlet Kullanıcı Arayüzü Bileşenleri

MIDlet kullanıcı arayüzü sınıfları iki gruba ayrılmıştır. (Burada sınıf ve API kavramlarını değişmeli olarak kullandık, çünkü API dediğimiz paketler sınıflardan oluşmaktadır, başka kaynaklarda sınıf ve API iki farklı şey olarak yazılıyor olabilir)

Yüksek seviyeli sınıflar(High Level API) : Bu API ile kullanıcı ekranında gösterileceğimiz TextBox, RadioButton, CheckBox, List gibi kullanıcının masaüstü programlarda görmeye alıştığı nesneleri kullanabiliriz. Bu sınıflardan bazıları ekranın tamamını kaplayan, daha doğrusu bir ekran sağlayan, Displayable sınıfından türetilmiş Form ve Alert gibi ekranda diğer bileşenleri kapsayıcı sınıflardır. Bazıları ise kapsayıcı (konteynır) ekran bileşenlerinin içinde yer alan alt elemanlardır. Örneğin TextField, StringItem, ImageItem, Gauge bileşenleri gibi. Masaüstü yazılım arayüzlerinin aksine yüksek seviyeli API bileşenlerinin konumları, boyutları ile oynama yapma şansımız yoktur. Bileşenlerin sırasını belirleyebiliriz ve bu sırada bileşenleri yerleştirmek istediğimizi söyleriz, cihaz bileşenlerin ekranda nasıl yerleştirileceğine karar vermektedir.

Dolayısıyla yüksek seviyeli API ile oluşturulan ekranlar cihazlar arası farklılıklar gösterebilir. Dahası bu programcı için hem avantaj hem de dezavantajdır. Programcı cihazın ekran boyutunu bilemediği için cihazın otomatik yerleştirmesi bir avantajdır. Fakat ekran tasarımının cihazlar arası ufak farklılıklar göstermesi uygulama tasarımı açısından da istenilen bir şey değildir ve bu da bir dezavantajdır.

Alt seviyeli sınıflar (Low Level API) : Yüksek seviyeli API'nin aksine, alt seviye API programcının tüm ekranı pikseller seviyesinde kontrol etmesine izin verir. Elbette piksel seviyesinde kontrol etmenin de bir maliyeti vardır : Zorluk.

Bu amaç için "lcdui" paketine Canvas denilen özel bir ekran sınıfı yerleştirilmiştir. Canvas sınıfının kendisi hiçbir çizdirme (paint()) metoduna sahip değildir, fakat AWT sınıflarında olduğu gibi paint metoduna bir erişim yolu sağlamaktadır. Cihaz ekranın çizdirilmesine karar verdiğinde, Canvas sınıfı içindeki paint metodunu çağırarak bu işlemin yapılmasını ister, programcı bu "call-back" tarzındaki paint metodu içinde istediği çizim komutlarını yerleştirerek çizimi yaptırmış olur.

Cihaz paint metodunu çağırırken, metoda Graphics nesnesini parametre olarak verir. Bu nesne cihazın ekranına çizim yapma yeteneğine sahip bir nesnedir ve böylece bu nesnesin drawLine(), fillRect() gibi çizim metotlarını kullanarak istenilen şekiller veya pikseller ekrana çizdirilir. Örneğin drawLine ile ekranda belirlenen uzunlukta ve koordinatta bir çizgi çizilir, fillRect ile ise istenilen dört nokta arası boyanarak doldurulur, drawString ise ekrana bir string değeri yazdırır.

Alt seviye API, masaüstündeki Swing ve AWT bileşenlerinin alt yapısına bazı konularda benzemektedir. Fakat onların aksine yüksek seviyeli API ve düşük seviyeli API'nin aynı anda kullanılmasına izin vermemektedir.

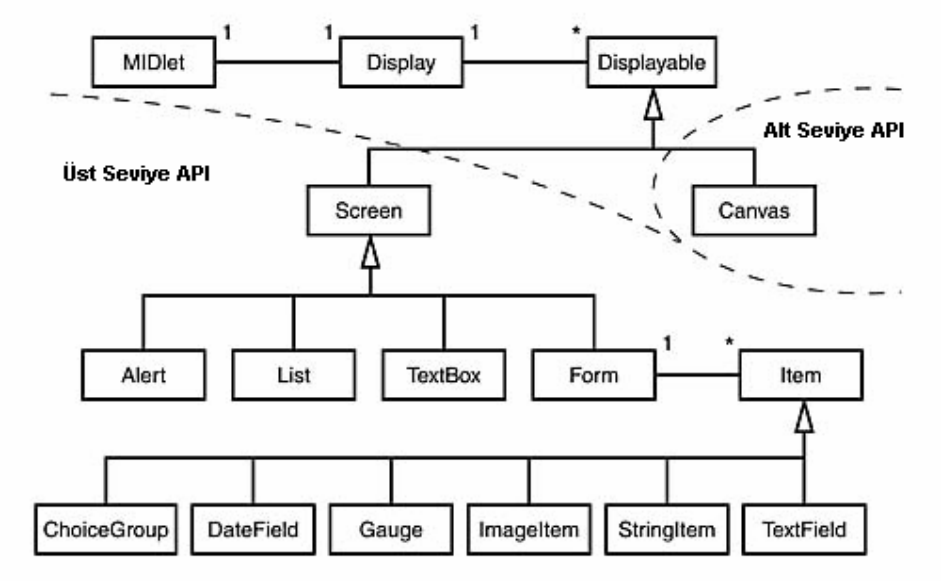
Bu iki bileşen grubunun her iki ile de beraber kullanılabilen, kullanıcı ile etkileşimi sağlayan bir diğer bileşen ise Command (Komut) bileşenleridir. Bunları klasik masaüstü programlarındaki düğmelere benzetebiliriz. Fakat cep telefonunda uygulamanın kullanacağı düğmelerin yeri belli olduğu, ekran üzerinde çoğunlukla hassasiyet olmadığı için düğmeler ekran dışındadır. Komutları eklerken programcı komutlar ile ilgili iki şeyi belirleyebilir :

1. **Komutun türü (Command Type) :** Komutun nereye yerleştirileceği programcı tarafından belirlenmez. Programcı sadece komutun türünü söyler, cihaz türe göre komutu nereye yerleştireceğine karar verir. Bunun sebebi örneğin "Tamam" düğmesinin Nokia telefonlarda farklı bir yerde, Sony telefonlarda farklı bir yerde bulunmasıdır, geleneğin bozulup, kullanıcının düğmeleri alıştığından farklı bir yerde görmemesi için düğmelerin yerini cihaz belirlemektedir. Komut türleri şunlardır :
    a. Command.Back
    b. Command.CANCEL
    c. Command.EXIT
    d. Command.HELP
    e. Command.ITEM
    f. Command.OK
    g. Command.SCREEN
    h. Command.STOP

2. **Komut verildiğinde yapılacak iş (Command Action Code) :** Komuta basıldığında yapılacak işleri tıpkı masaüstü yazılımlarda olduğu gibi bir event-handler metot devralarak, içindeki kodları icra eder.

3.1.2.1.1.7 MIDlet ile MultiMedya Seçeneklerini Yönetme (MMAPI)

MMAPI (Mobile Media API) 1.1. MIDP 2.0 profili ile gelen yeni bir API'dir ve ses, görüntü kaynağı olan pek çok mobil cihaz donanımını kullanabilme özelliğini içermektedir. Elbette tüm gezgin cihazlar kamera, ses kaydı gibi özelliklere sahip olmadıklarından MMAPI tasarlanırken sadece desteklenen özellikleri kullanabilme yeteneğinde üretilmiştir.



Şekil 3.12. Alt ve Üst Seviye API'ye Genel Bakış (Kroll, 2002)

MMAPI geliştirilmeye JSR-135 önerisi olarak başlamıştır ve 2006 itibariyle 1.1 versiyonu mevcuttur. MMAPI ile ilgili en önemli özellik tüm çoklu ortam seçeneklerini en genel şekilde kullanmaya özen göstermesi, cihaza, ortama özel bir sınıf, komut veya özellik içermekten kaçınmasıdır. Bu da tastamam Java'nın "Bir yerde yaz, her yerde çalıştır" felsefesine uygun düşmektedir. API'nin en temel üç sınıfı kastedilen soyutlamayı yapmaktadır, bu sınıflar "Player" ve "Control" arayüzleri (interface: nesneye yönelik programlama da bir sınıf türü) ile "Manager" sınıfıdır. Özetle, Manager sınıfı programcı tarafından Player sınıfından türemiş bir çoklu ortam açıcısı/çalıcısı açmak için kullanılır. Ve bu çalıcılar ise Control sınıfı ile ayarlanabilmektedir. Manager sınıfı pek çok çalıcıyı (Player) oluşturabilir ve programcıya teslim eder. Manager.createPlayer metot çağrısı ile bu işlem yapılmaktadır. CreatePlayer metoduna verilecek çalıcı türü ile Manager sınıfı ne tür bir Player oluşturacağını anlamaktadır. Aşağıdaki tabloda Player sınıfından türemiş çalıcı/oynatıcı/açıcı tüm çoklu ortam türleri listelenmiştir.

Çizelge 3.7. MMAPI'nin desteklediği bazı ortam çeşitleri ve örnek kullanımlar

Ortam türü	**Örnek kullanım**
Ses kaydetme	• "capture://audio" metnini kullanarak varsayılan ses kaydetme donanımını başlatabiliriz. • "capture://devmic0?encoding=pcm" metnini kullanarak ses kaydetme donanımlarından ilkini alarak, PCM formatında kayıt yapılmasını sağlayabiliriz.
Video/Görüntü kaydetme	• "capture://video" metni ile varsayılan kamera kayıt için başlatılır. • "capture://devcam0?encoding=rgb888&width=100&height=50" ile de ikinci bir kameradan belirlenen formatta, genişlik ve yükseklikte kamera kayıt için başlatılıyor.
Radyo	"capture://radio?f=105.1&st=stereo" metni ile 105.1 FM frekansında, stereo modda dinlemeye başlamak için kullanılabilir.
Harici bir kaynaktan ortam indirme	"rtp://host:port/type" dosyanın bulunduğu yere işaret eden bir adres yazarak, dosyayı doğrudan çalmaya başlayabiliriz.
MIDI ve ton çalma	"device://tone" veya "device://midi"

Örneğin aşağıdaki kod parçası fotoğraf çeken bir uygulama için cep telefonu ekranına kamerayı kayıt modunda yerleştiriyor, tabi kamera çalışmaya devam ediyor.

```
Form form = new Form("Kamera gösteriliyor");
Item item = (Item)mVideoControl.initDisplayMode(
   GUIControl.USE_GUI_PRIMITIVE, null);
form.append(item);
```

(Anonim 2003a)

Kamera ekran üzerinde gösterildikten sonra herhangi bir Command (komut) ekleyerek kameradan resim çekmek istediğimizde, Command için yazmamız gereken kod şuna benzer bir şey olması gerekmektedir.

```
byte[] raw = mVideoControl.getSnapshot(null);
Image image = Image.createImage(raw, 0, raw.length);
```

(Anonim 2003a)

Bu iki satır kod ile önce getSnapshot() ile resmin çekilmesini ve çekilen resmin byte dizisi olarak aktarılmasını ve daha sonrada bir resim (Image) nesnesine dönüştürülmesini sağlıyoruz. İsteğe göre bu resim (byte dizisi) istenirse bir sunucuya upload edilebilir, istenirse yerel cihaza RMS veya File System API ile kaydedilebilir. Eğer gerekli parametreler verilirse resim PNG veya JPEG formatlarından herhangi birisinde olabilirdi. MMAPI bu konuda çok çeşitli seçenekler sunmuştur.

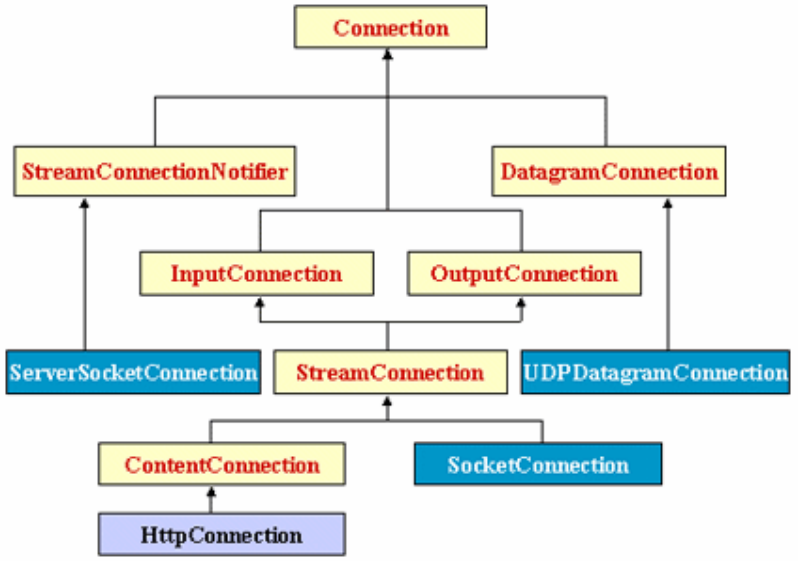
3.1.2.1.1.7. MIDlet ile İnternet Erişimi

Java ME ağ/internet tabanlı API pek çok farklı cihazın ihtiyaçlarına yanıt verebilmek ve cihazdan bağımsız çalışabilmek gibi ihtiyaçlara yanıt verebilmek için tasarlanmıştı. Fakat ağ ve internet bağlantısı gibi ihtiyaçlar genelde birbirinden farklı ve bağımsız, yer yer cihaza özgü öğeler içermekteydi. Bu yüzden Java ME içinde ağ tabanlı API "Genel Bağlantı Çatısı (Generic Connection Framework- GCF)" olarak ortaya çıkarılmıştır.

GCF'nin ana fikri, ağ veya dosya sistemi gibi kaynaklardan gelen veriyi Java arayüzleri ile soyutlamak ve kaynak ne olursa olsun programcının her şekilde aynı mantığı kullanabilmesini sağlamaktır. Bu mimaride API'nin gerçekleştirilmesi cihaz üreticilerine bırakılmış, bir Java programcısı bu alt düzey cihaz bağımlı koddan uzak tutulmuştur. Yani programcı her zaman aynı Java sınıfları ile uğraşacak, sınıfların içleri cihaz üreticisi tarafından doldurulacaktır.

GCF ile temel olarak aşağıdaki bağlantı türleri gerçekleştirilmiştir.

- Sıralı girdi (javax.microedition.io.InputConnection sınıfı ile tanımlanmıştır.)
- Sıralı çıktı (javax.microedition.io.OutputConnection sınıfı ile tanımlanmıştır.)
- Datagram iletişimi (javax.microedition.io.DatagramConnection)
- Soket tabanlı istemci-sunucu uygulamaları için bildirim mekanizması (javax.microedition.io.StreamConnectionNotifier)
- Bir web sunucu ile HTTP iletişimi (javax.microedition.io.HttpConnection)



Şekil 3.13. Bağlantı sınıfları hiyerarşisi (Anonim 2003b)

Tüm bağlantı türleri için Connector sınıfı kullanılmaktadır. Bu sınıf bağlantı türlerini oluşturan bir fabrika gibi çalışmaktadır. Connector sınıfının farklı bağlantı türlerini oluşturması aşağıdaki örneklerde gösterilmiştir.

- HTTP Bağlantısı : Connector.open("http://java.sun.com/wireless")
- Soket Bağlantısı : Connector.open("socket://java.sun.com:**port**")
- Datagram Bağlantısı : Connector.open("datagram://java.sun.com:**port**")
- Seri port Bağlantısı : Connection cc = Connector.open("comm:0;baudrate=9000");
- Dosya okuma/yazma: Connection fc = Connector.open("file://foo.dat");

(Anonim 2003c)

Görüldüğü üzere J2ME ağ paketi ile üst düzey (http gibi) veya alt düzey (socket gibi) bağlantı türleri oluşturabilmekteyiz. Önemli bir nokta uygulama yaparken ağ işlemlerinin her zaman ana işlemden (Main Thread) farklı bir işlemde (Thread) yapmaktır. Aksi takdirde derleyici sizi uyaracak ve belki de uygulamanız kilitlenecektir.

3.1.2.1.2. Servlet-JSP Teknolojisi

Servlet ve JSP teknolojisi Java platformunun web tabanlı uygulamalar için bir parçasıdır. CGI programlama yapmak için Java tabanlı bir çözüm sunar. Servlet ve JSP uygulamaları teknik olarak aynı şeylerdir ve bu uygulamalar web sunucu üzerinde çalışırlar. Asıl hedefleri web sunucuya gelen dinamik istekleri sunucudan almak, veritabanı, dosya gibi kaynaklardan bilgileri çekmek veya kullanıcının bilgilerini işledikten sonra kullanıcıya sonuç olarak yanıtı göndermektir. Genel olarak işleyiş adımlarını şöyle sıralayabiliriz :

1. **Kullanıcı tarafından gönderilen tüm verileri okumak :** Kullanıcı tarafından iki şekilde bilgi gönderilebilir : POST veya GET. Her ikisi de aslında bir formun gönderilmesinden ibarettir. Yani bir kullanıcı sadece bir form aracılığıyla sunucuya bilgi gönderebilir. Servlet tüm bu bilgileri toplayarak programcıya http istemcisinden bağımsız bir şekilde sunar.

2. **HTTP isteğinde bulunan kullanıcının verileri dışındaki verileri toplamak :** Bir HTTP isteğinde kullanıcının bizzat gönderdiği veriler dışında da veriler bulunmaktadır. Örneğin kullanıcının tarayıcısı, işletim sistemi, tarayıcısının özellikleri, cookie politikası gibi.
3. **Bilgileri işle ve sonucu üret :** Burası Servlet içinde asıl iş yapılan yerdir. Verilen girdilere göre sonucu üretme işlemi veritabanı ile iletişim, arka planda başka bir uygulama çalıştırma, uzaktaki başka bir sunucuya CORBA, RMI gibi teknolojilerle bağlanma gibi pek çok işlemi içerebilir. Ama sonuç olarak kullanıcıya bir yanıt (response) üretilmelidir.
4. **Sonucu biçimlendirmek :** Sonuç yani çıktı tarayıcıda görüntülenebilecek bir biçimde olmalıdır. Genellikle çıktı HTML olarak biçimlendirilebilir. Bunun dışında XML-XSLT olarak da çıktı verilebilmektedir.
5. **Gerekli HTTP protokolü biçimlendirmesinin yapılması :** Tarayıcıya gönderilecek yanıt HTTP protokolünde olmalıdır. Bunun içerisinde HTML olarak içerik ve içerik, sunucu hakkında bazı bilgiler içeren üst bilgi vardır. Tüm bu biçimlendirme işlemleri sonucunda tarayıcının istediği ve görüntüleyebileceği bir sayfa ortaya çıkar.
6. **Oluşturulan dökümanı istemciye gönder :** Tüm bu işlemler sonucu oluşturulan döküman, belki HTML, belki GIF veya başka bir tür de olabilir, istemciye gönderilir (Hall, 2000).

Servlet kullanmanın klasik CGI programlamaya göre pek çok avantajından bahsedebiliriz. Bu avantajları şöyle listeleyebiliriz :

- **Verimlilik :** Klasik CGI programlama da her yapılan istek için yeni bir işlem başlatılır ve sunucunun hafıza, işlemci yeterliliğine göre işlenebilecek en fazla işlem sayısı söz konusudur. Servlet'ler ise daha farklı çalışır. Bir Servlet sunucuda ilk kez çalıştırıldığında hafızaya yüklenir ve daha sonraki isteklerde yeni bir Servlet yüklemesi olmaz, fakat yeni bir Thread yüklenir. Dolayısıyla tüm program yeniden yüklenmesi yerine sadece yeni bir Thread yüklenir. Bu da hafıza ve işlemci gücünden tasarruf etmek demektir. Bunun yanında Servlet'ler veritabanı bağlantısı gibi kaynaklara erişimi de havuz sistemi ile tekrar tekrar kullanabilmektedirler. Böylece sunucuyu en çok yoran kaynaklara erişim meselesinden de tasarruf edilmiş olunur.
- **Kullanışlılık:** Servlet sistemi programcı için tüm HTTP bilgilerini, kullanıcı girdilerini yakalayarak programlama ortamına hazır olarak sunar. Ayrıca web programlama için yeni bir dil öğrenilmesini gerektirmez, klasik Java dili ile web programlama yapılmasını sağlar.
- **Güç :** Servlet teknolojisi Session, Application gibi bilgi takibi için çok kullanışlı alt sistemleriyle, web sunucu ile doğrudan iletişime geçebilmesi ile klasik CGI programlama ile yapılması nerdeyse imkansız işlemleri çok rahat yapabilmektedir. Ayrıca Java platformunun gücünü de arkasına almıştır.
- **Taşınabilirlik :** Bir Servlet klasik bir Java programı gibi taşınabilirdir. Yani Linux altında çalışan bir Servlet sisteminiz hiç bir şey yapmadan Windows ortamına ve hatta MacOS ortamına taşınabilmektedir. Çünkü Servlet sunucuya özel bir işlem yapmaz, tüm platforma özel işlemleri Java sanal makinesi halletmektedir.
- **Güvenlik :** Klasik CGI programlamada kullanılan diller genelde betik (script) dilleri olduğu için syntax kontrolü, çalışma zamanında oluşan hataların denetimi gibi işlemlerden yoksundurlar. Java platformu tabanlı Servlet'ler bunların tamamını karşılamaktadır.
- **Ucuzluk :** Servlet sistemlerini açık kaynak kodlu Apache ve Tomcat sunucuları ile çalıştırabilmekteyiz. Ayrıca Java çalışma zamanı platformu ve Java geliştirme platformları da bedava olarak internetten indirilebilmektedir.

(Hall, 2000)

JSP teknolojisi ile Servlet teknolojisinin hemen hemen aynı şey olduğundan bölümün girişinde bahsetmiştik. Aslında her JSP sayfası arka tarafta bir Servlet'e otomatik olarak dönüştürülmektedir. JSP sayfalarının neden kullandığını bir kaç maddede sıralayabiliriz :

- HTML kodlarını tek tek write komutu ile yazmaktan kurtardığı için
- HTML ve Java kodlarının karışımını yapıp hızlıca bir uygulama yapılabildiği için
- Betik dillerinden gelen yeni kullanıcılara alışkın oldukları bir ortam sunduğu için

JSP çeşitli avantajlarına rağmen HTML ve Java kodunun karışık yazılması sebebiyle bazı dezavantajları da beraberinde getirmektedir.

- JSP ile hata ayıklamanın zorluğu
- JSP sayfalarındaki Java kodlarının güncellenmesinin zorluğu
- Bir tasarımcının JSP sayfasını güncellemesi için Java bilgisine ihtiyaç duyması
- Klasik Java programlamaya göre bazı yazım kurallarının da öğrenilmesi gerektirdiği

Aşağıda kullanıcıya "Merhaba Kullanıcı" metnini ve IP adresini gönderen bir Servlet ve bir JSP çözümünün örnek kodları var:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD><TITLE>Hello WWW</TITLE></HEAD>
<BODY>
<H1>Merhaba Kullanıcı</H1>
<FONT Face='Times New Roman'>IP Adresiniz :
<%=request.getRemoteAddr() %>
</FONT>
</BODY>
</HTML>
```

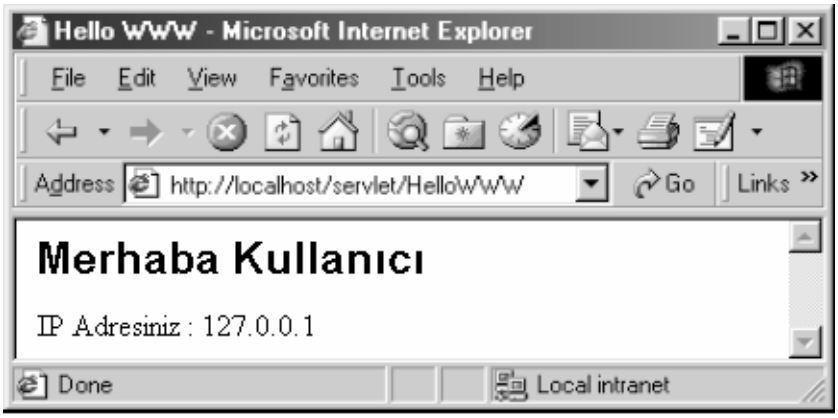
Örnek JSP dosyası

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class HelloWorld extends HttpServlet
{
        public void doGet(HttpServletRequest request,
                HttpServletResponse response)
                throws ServletException, IOException
        {
                PrintWriter out = response.getWriter();
                out.println("<HTML>");
                out.println("<head><title>Hello WWW</title></head>");
                out.println("<body>");

                out.println("<h1>Merhaba Kullanıcı</h1>");
                out.println("<FONT face='Times New Roman' >IP Adresiniz : ");
                out.println( request.getRemoteAddr() );
                out.println("</FONT>");
```

```
            out.println("</body>");
            out.println("</HTML>");
        }
}
```

Örnek Servlet dosyası

Görüldüğü gibi Servlet içinde çok fazla HTML kodu yazmak yorucu, aynı zamanda JSP içinde de çok fazla Java kodu yazmak yorucu ve kötü oluyor. Bu yüzden JSP sayfalarını çok HTML içeren çıktılar için, Servlet'leri de Java ile çok işlem yapılan sayfalar hazırlamak için kullanmak gerekmektedir. Şekil 3.14. de her iki program için de örnek ekran çıktısı tarayıcıda gösterilmiştir.



Şekil 3.14. Örnek JSP ve Servlet programlarının çıktısı

Son zamanlarda HTML içerik hazırlama, Java kodu yazma işini beraber yapmanın zorluğundan dolayı Java EE (Enterprise Edition) için pek çok çatı (framework) üretilmiştir. Bunların başında SUN firması tarafından üretilen JSF (Java Server Faces), Apache tarafından açık kaynak kodlu olarak üretilen Struts söylenebilir.

JSP ve Servlet teknolojisine rakip pek çok dil, teknoloji ve platform bulunmaktadır. Uygulamamızda Servlet teknolojisini seçmemizin sebebi diğer teknolojilere üstünlük kurmasıdır. Tek tek diğerleri ile kısaca karşılaştıralım

PHP ve Servlets

PHP son zamanlarda popüler olan güçlü bir betik dilidir. Windows, Linux gibi ortamların hemen hepsinde düzgün çalışmaktadır. Fakat bazı dezavantajları vardır :

- Derlenmemesi : Doğrudan yorumlanan bir dildir
- Dilin zayıf olması : Açık kaynak kodlu olarak geliştirilen PHP dilinde tüm oluşturulan fonksiyon ve kütüphanelerin arasında bir yazım standartı bulunmamaktadır. Aynı zamanda bazı çalışma zamanı hataları PHP tarafından otomatik tür dönüşümü gibi işlemlerle kapatılmaktadır.

- Güçlü bir IDE bulunmaması : PHP için güçlü bir IDE bulunmamaktadır. Son zamanlarda Zend ve Nusphere firmaları iyi ürünler çıkartmaktadır fakat Java teknolojisinin geldiği seviyeye gelinmesi için biraz zaman gerekmektedir.

Ayrıca gezgin cihaz tarafında Java-J2ME kullandığımız için sunucu tarafında da PHP yerine Java kullanmak işimizi kolaylaştırmıştır.

ASP-ASP.NET ve Servlets

ASP.NET/ASP Microsoft firmasının ürettiği son zamanların Java platformuna en çok benzeyen ve Java platformunun en büyük rakibi olan web tabanlı uygulama geliştirme platformudur. ASP.NET'i Java yerine kullanmamamızın önemli nedenlerinden birisi PHP bölümünde de yazdığımız gibi gezgin cihaz üzerinde ki yazılımımızın Java tabanlı olmasıydı. Onun dışında ASP.NET'in çeşitli dezavantajlarından da bahsedilebilir :

- Sadece Windows ortamında çalışması : Son zamanlarda Mono adlı bir proje ile Linux'a taşınması söz konusu olmasına rağmen, henüz Windows ortamında bile sürümler arası uyuşmazlıkları bulunan bir platformun başka bir platformda sorunsuz çalışması söz konusu olamaz. Zaten Mono projesi Microsoft tarafından doğrudan desteklenen bir proje olmadığından, güncellemeler Microsoft'un sürümlerine göre oldukça geriden gelmektedir. Dolayısıyla kesin olarak söyleyebileceğimiz son söz .NET platformunun Java platformu kadar "platform bağımsızlığı" henüz sağlayamadığıdır.
- C# veya VB.NET gibi yeni bir dilin öğrenilmesini gerektirmesi

Bunların dışında Servlet teknolojisine diğer alternatif teknolojiler : Macromedia firması tarafından üretilen ColdFusion, açık kaynak kodlu Python, C++ ile CGI programlama, Perl ile CGI programlama sayılabilir. Servlet teknolojisi özellikle bizim uygulamamız için diğer bütün alternatif teknolojilere göre de tercih edilebilir olmuştur.

3.1.2.1.2.1 Tomcat Sunucusu

Apache Tomcat açık kaynak kodlu olarak üretilen temelde Java Server Pages ve Servlet teknolojilerini çalıştırmak için kullanılan bir web sunucudur. Bağımsız bir HTTP sunucusu olarak veya başka bir HTTP sunucu ile entegre olarak çalışabilmektedir (Örneğin IIS veya Apache Web Server ile). XML tabanlı bir ayar dosyası ile sunucu üzerinde çalışacak Servlet ve uygulamalar ayarlanır, sunucu ayarları da yine XML tabanlı ayar dosyasında saklanır.

Apache Tomcat diğer Web konteynırların (Web Container) aksine EJB gibi ileri seviye Java EE teknolojilerini desteklememektedir. Tek amacı Servlet ve JSP çalıştırmaktır. Aşağıdaki tabloda Tomcat versiyonları ve hangi Servlet sürümünü desteklediği gösterilmektedir.

Çizelge 3.8. Tomcat sunucusunun versiyonları (Kaynak, Apache Software Foundation)

Servlet/JSP Sürümleri	**Apache Tomcat sürümü**
2.5/2.1	6.0.x
2.4/2.0	5.5.x
2.3/1.2	4.1.x
2.2/1.1	3.3.x

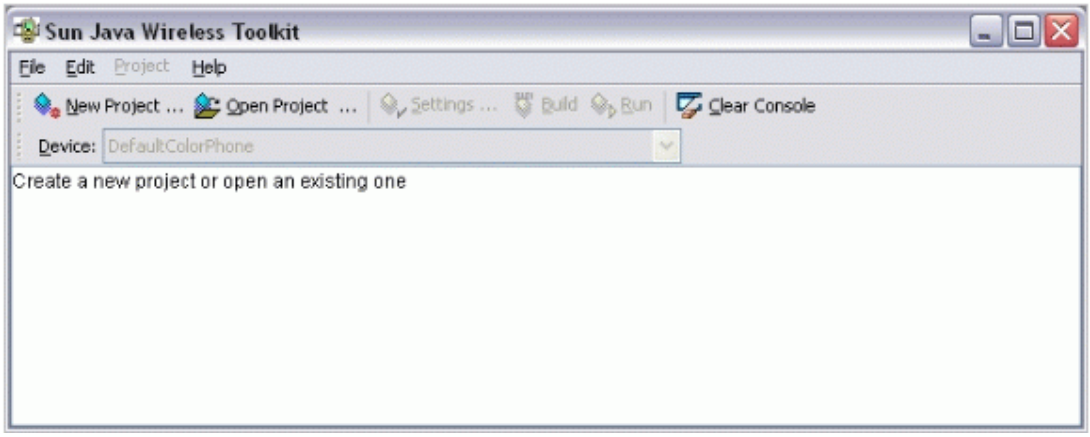
3.1.2.2. Geliştirme Araçları – IDE

3.1.2.2.1. Wireless Toolkit

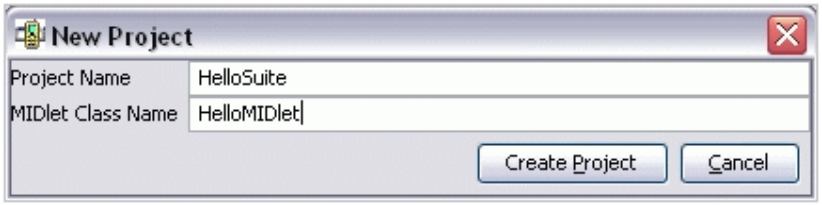
Wireless Toolkit (WTK) SUN firması tarafından geliştirilen mobil cihazlar için yazılım geliştirme aracıdır. Bu araç tek başına kullanılabildiği gibi başka Tümleşik Geliştirme Ortamları (IDE Integrated Development Environment) ile de birleşik çalıştırılabilir. WTK'nın oldukça basit bir yapısı vardır, yazım editörü, görsel tasarım editörü gibi bütünleşik araçlara sahip değildir. En temel özelliği yazılan kodları derlemek, paketlemek ve bir emülatör ile nasıl çalıştığını göstermektir. Emülatörler cihazların birebir kopyaları değildir fakat cihazların destekledikleri standartlara göre üretildikleri için cihaz üzerinde yazılımın nasıl görüneceği konusundan net fikirler vermektedir.

WTK ile bir proje yapmanın adımları şunlardır :

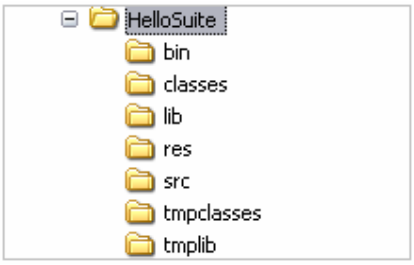
1. **Projeyi oluşturma :** WTK'nin Ktoolbar isimli aracını açtıktan sonra gelen ekranda (Şekil 3.15.) "New Project" düğmesine basılarak, proje için gerekli bilgiler girilir. (Şekil 3.16.). Yeni bir proje oluşturulurken WTK sizin için gerekli klasörleri ve tanımlama, ayar dosyalarını oluşturur. Her bir proje birden fazla alt klasörden oluşur, bu klasörler içine gerekli dosyaların yerleştirilmesi programcıya bırakılmıştır. WTK sadece iskelet yapıyı kurmaktadır (Şekil 3.17.).
2. **Proje dosyalarını yazma ve yerleştirme :** Projedeki kaynak kod dosyalarını (*.java) proje klasörü içindeki "src" klasörüne yerleştirmek gerekmektedir. Böylece WTK src klasöründeki dosyaları derleyip yazım yanlışları varsa programcıya gösterecek, yoksa dosyaları *.class dosyalarına çevirecek ve uygulamayı paketlemek için hazır hale getirecektir.
3. **Dosyaları paketleme ve cihaza programı kurma :** *.class, yani java uzantılı dosyaların derlenmiş hali ve programın kullanacağı harici kaynakların, gif, txt gibi farklı dosya türlerin, telefona yüklenmesi için jar uzantılı olarak paketlenmesi gerekmektedir. Aynı zamanda uygulama hakkında bazı bilgiler veren jad uzantılı özel bir biçime sahip dosyanın da oluşturulmuş olması gerekmektedir. WTK bu işlemlerin hepsini yapmaktadır. Programcının tek yapacağı dosyaları doğru klasörlere yerleştirmektir. Sonuç olarak WTK'nin üreteceği iki dosya olan jar ve jad dosyalarını cihaza yüklemenin temel iki yolu vardır. Birincisi bu dosyaları internette bir sunucuya yerleştirmek ve cihazın internet üzerinden java programı yükleme özelliğini kullanmaktır. Genelde cihazlar jad dosyasını önce okuyup, jad dosyası içeriğine göre jar dosyasını otomatik indirmektedirler. İkinci yol ise çoğu cihaz ile birlikte verilen bilgisayar yazılımlarını kullanarak cihaza USB veya başka bir yol ile doğrudan aktarmaktır. Bu ikinci yol genelde cihaza bağımlıdır ve değişebilmektedir. Ama sonuç olarak her durumda uygulamayı yüklemek oldukça kolaydır.



Şekil 3.15. WTK KtoolBar yazılımının açılış ekranı (Anonim 2006g)



Şekil 3.16. WTK "New Project" Ekranı (Anonim 2006g)



Şekil 3.17. WTK ile oluşturulan örnek proje için klasör yapısı (Anonim, 2006g)

3.1.2.2.2. NetBeans IDE

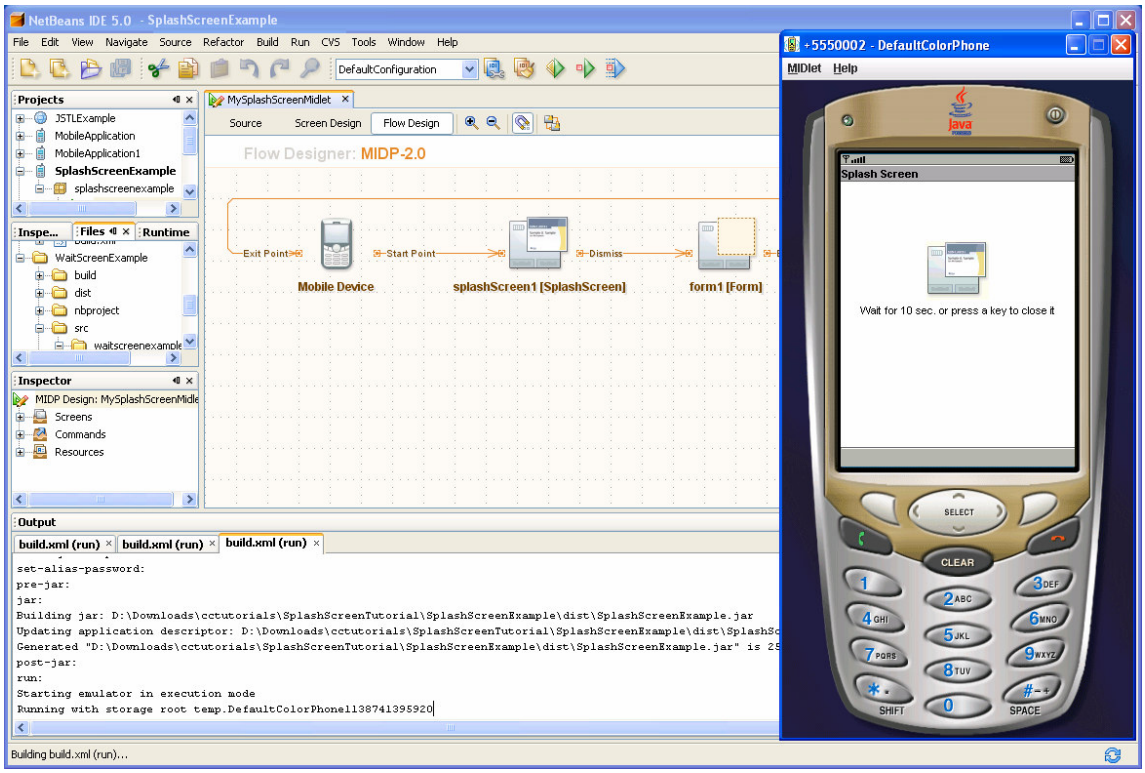
NetBeans IDE'si programcıların yazma, derleme, hata bulma ve yüklemelerini sağlayan bir araç olan geliştirme ortamıdır. Java ile yazılmış olmasına rağmen herhangi bir programlama dilini destekleyebilir. NetBeans IDEsini genişletmek için çok sayıda modül bulunmaktadır. NetBeans IDE nasıl kullanıldığına ilişkin hiçbir sınırlama bulunmayan ücretsiz bir üründür (Anonim 2006h).

NetBeans'ın en güncel sürümü 5.5'dir ve bu sürümle masaüstü, sunucu ve gezgin cihazlara yönelik programlamayı desteklemektedir. Fakat her bir yeni ortam için NetBeans üzerine yine kendi sitesinden bedava indirilebilen eklentilerin (plug-in) kurulması gerekmektedir. Örneğin cep telefonları için yazılım geliştirilmek isteniyorsa sisteme önce "Netbeans" sonra "Netbeans Mobility Pack" kurulmalıdır. Tüm bu yazılımlar tamamen ücretsizdir. Netbeans aşağıdaki özellikleri desteklemektedir :

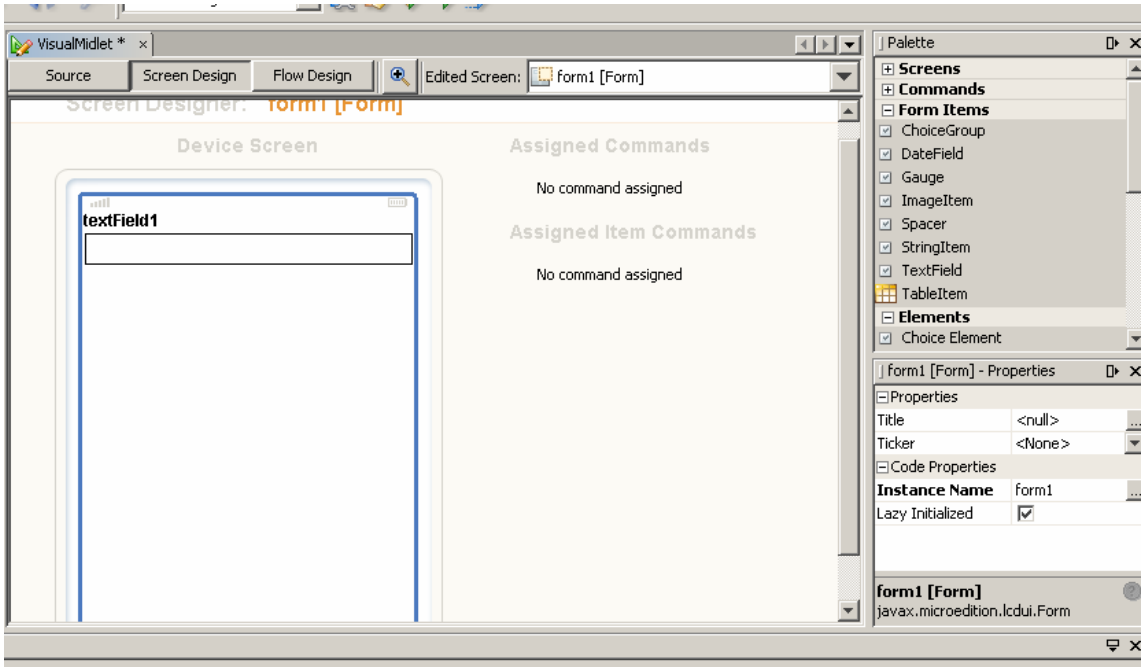
- Birden fazla emülator desteği vermektedir. Böylece uygulamanın tüm cihazlar üzerinde çalıştığı garanti edilmektedir.
- Görsel olarak arayüz tasarlama (Şekil 3.18).

- Görsel olarak programın ekranları arası akışını tasarlayabilme (Şekil 3.19.)
- "Web services" gibi ileri düzey konularda sihirbazları ile otomatik uygulama kodunu oluşturabilme
- Ant uyumlu derleme, çalıştırma işlemleri sayesinde farklı cihazlar ile çalışmak için yeterli esneklik sağlanması.

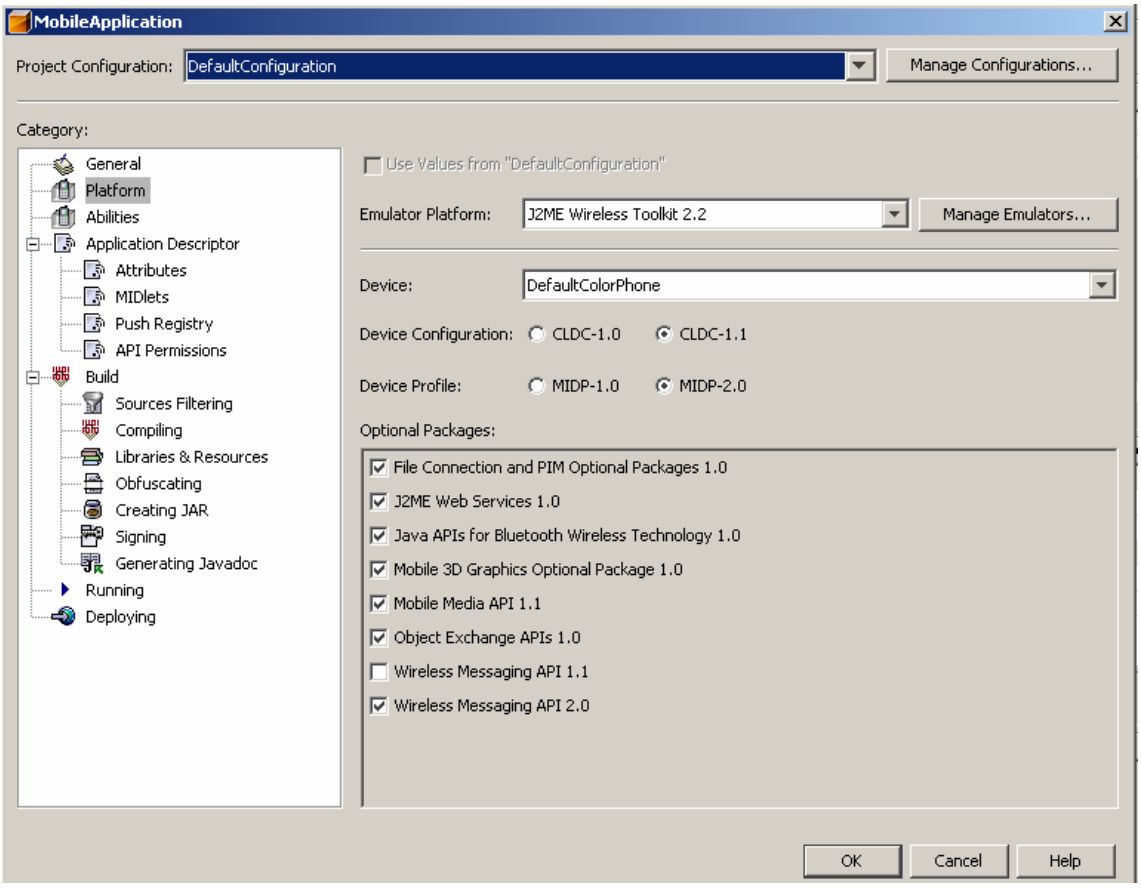
NetBeans IDE bizim projemiz için de seçtiğimiz geliştirme ortamlarından birisi olmuştur. Çünkü gerek kod yazarken otomatik tamamlama yapması, gerek proje yönetimi, gerekse farklı cihazlar için üretilen emülatörlerin üzerine kurulması gibi özellikleriyle yazılım geliştirmeye oldukça yardımcı olmaktadır.



Şekil 3.18. Netbeans Mobil Geliştirme Ortamında Ekranların Akış Tasarımı yapılmakta, ön tarafta da emülatörde uygulama izlenmektedir (Anonim 2006h).



Şekil 3.19. Netbeans ile görsel olarak ekran tasarımı



Şekil 3.20. Netbeans ile uygulamanın hangi standartlara uygun olacağını ayarlamak.

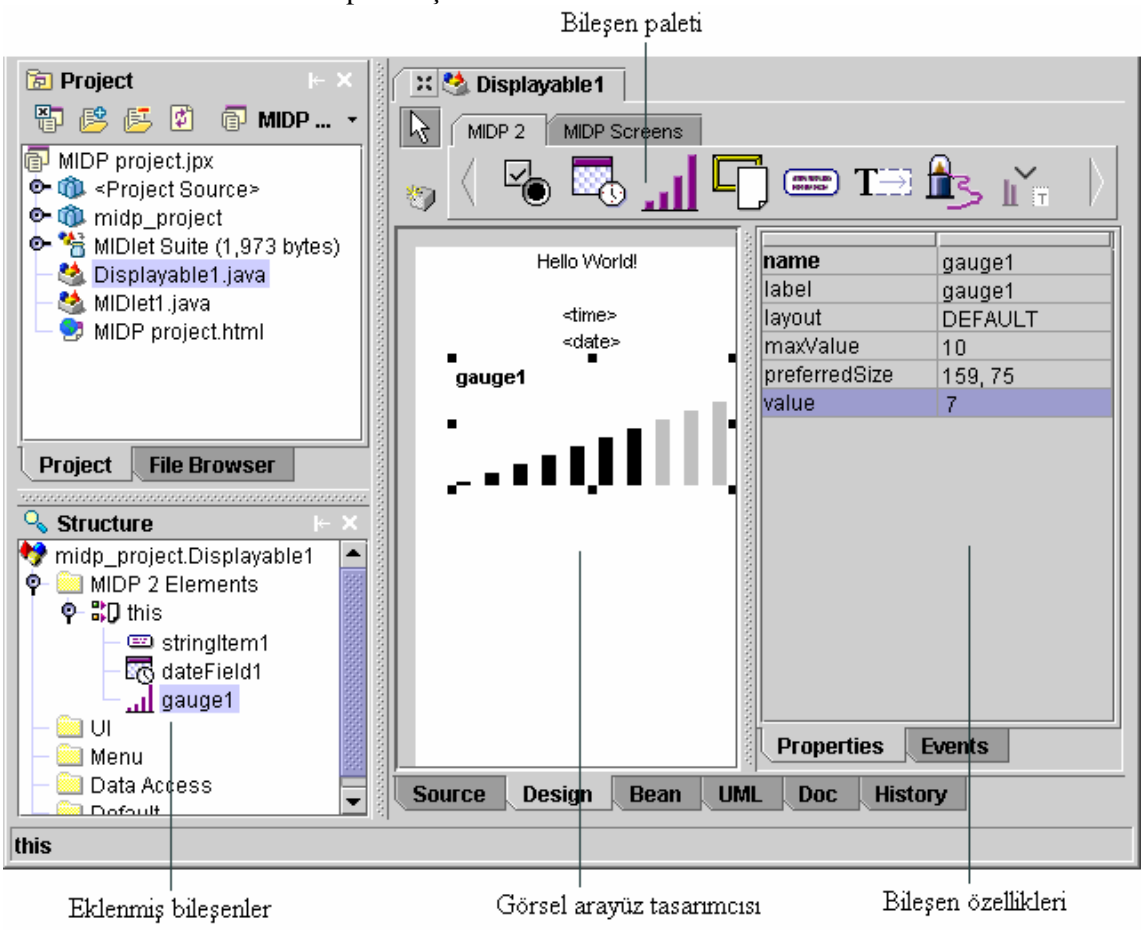
3.1.2.2.3. JBuilder IDE

Borland firması tarafından 1995 yılında üretilen Jbuilder, Java ilk çıktığı yıllarda boy gösteren ilk geliştirme ortamlarından birisidir. 2005 yılına kadar Borland tarafından geliştirilmeye devam edilmiştir, 2005 yılında Borland, IDE pazarından çekildiğini açıklayarak geliştirmesini durdurmuştur. Jbuilder Java ile yazılmış bir geliştirme ortamıdır. Netbeans gibi hem masaüstü, hem kurumsal, hem de gezgin cihazlar için geliştirme seçenekleri sunmuştur. Fakat ücretli bir yazılım olduğu için zamanlar Netbeans elde tuttuğu popülaritesini elinden almıştır.

JBuilder tıpkı NetBeans gibi mobil eklentisi ile pek çok işlemi otomatik olarak yapmaktadır. Desteklediği bazı özellikler şunlardır :

- Görsel olarak ekranları tasarlayabilmek. Bu özellik Netbeans'ın görsel tasarımı kadar gelişmiş değildir
- Görsel olarak ekranlar arası akışı takip edebilmek
- Uygulama için gerekli paketleme işlemlerin (provizyon, OTA gibi) otomatik yapılması
- Birden fazla emülatör ve Java ME sürümü kullanabilmek.

Jbuilder NetBeans için alternatif iyi bir geliştirme ortamıdır. Fakat lisans ücreti geliştiricilerin daha az tercih etmesine sebep olmuştur.



Şekil 3.21. JBuilder ile görsel olarak MIDlet tasarlama (Anonim 2006ı)

3.2. METOD

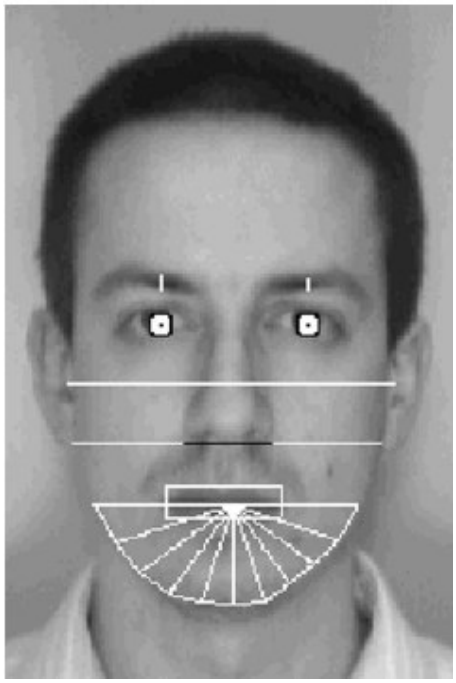
3.2.1. Resimden Yüz Tanıma (Image Based Face Recognition)

Yüz tanıma, günümüzde kişi kimliklendirme işlemlerinde çok sık başvurulan yöntemlerden biridir. Yüz tanıma problemi için bir bilgisayar modeli kurmak oldukça zordur. Bu zorluk, yüzlerin çok boyutlu, karmaşık ve anlamlı görsel öğeler olmalarından kaynaklanmaktadır. Oysa insan beyni, bir an gördüğü bir yüzü, yıllar sonra bile, yaşlanma, saç uzaması, sakal bırakma gibi değişimlere rağmen birkaç saniye içerisinde hatırlayabilmektedir. İnsan beynindeki bu üst düzey yetenek, yüz tanıma esnasında bilginin en iyi ve en kısa şekilde kodlandığı gerçeğini ortaya çıkarmaktadır (Atalay 1996)

Son yıllarda yüz tanıma ile ilgili metotlar iki teknik üzerine yoğunlaşmıştır: 1) Yüz özellikleri tabanlı 2) Şablon tabanlı. Birinci teknik yüzün önden görünümü üzerinde ayırtedici özelliklerin çıkarılması, yüzün üzerindeki öğelerin birbirlerine göre olan göreceli konumu gibi parametreler üzerine kuruludur. Bu özellikler genellikle aşağıdakiler olabilir :

- Kaş kalınlığı ve gözün merkez pozisyonuna göre dikey konumu
- Kaşın kavisleri, eğrisi
- Burnun dikey konumu ve büyüklüğü
- Ağzın dikey konumu, genişliği, yüksekliği. Alt ve üst dudaklar
- Çenenin şeklini tanımlayabilecek yarıçap
- Burundan itibaren yüzün genişliği

Elde edilen ölçümler yüzün pozisyonu, resim içindeki oranı, dönme açısı gibi sonucu bozabilecek parametrelerden soyutlamak için normalize edilirler ve her bir ölçüm kümesi bir vektörün özellikleri olarak saklanır. Her yeni eşleşme işleminde girdi olarak verilen resimdeki vektör değeri veritabanındaki vektörler ile karşılaştırılıp en iyi sonuç bulunmaya çalışılır (Grudin 1999).



Şekil 3.22. Özellik tabanlı yüz tanıma için geometrik ölçümlerin kullanımı (Grudin 1999)

Şablon eşleştirme tabanlı yüz tanıma algoritmalarında ise her bir kişi alanları iki boyutlu bir dizi olan bir veritabanı girdisi olarak saklanır. Bu dizi piksel dizisidir. Fakat özellik tabanlı çözümde olduğu gibi bu dizi içindeki piksellerde normalize edilmiş resmin pikselleridir. Tanıma işlemi herhangi bir resmin bütün veritabanı resimleri ile ilişkinin (correlation) matematiksel fonksiyon olarak kullanılarak karşılaştırılması ile yapılır.

Bu iki yüz tanıma tekniği Brunelli ve Poggio tarafından karşılaştırılmış ve sonuç olarak özellik tabanlı tanıma tekniğinin daha hızlı ve daha az hafıza harcayarak çalıştığı gözlemlenmiştir. Fakat şablon tabanlı tanıma tekniği tanıma oranında bariz bir üstünlük sağlamıştır.

Bundan sonra başlıca resim tabanlı yüz tanıma metotları anlatılacaktır.

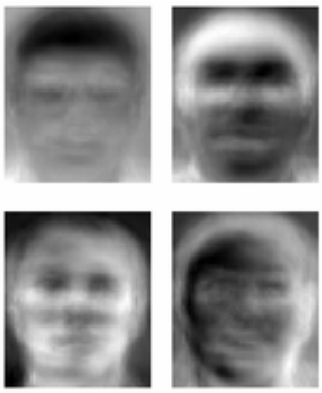
3.2.1.1. Özyüzler - Eigenfaces

Yüz tanıma için bir eigenvektör kümesi kullanan metottur. Bir eigenvektörleri olası insan yüzlerinin ihtimal dağılımının kovaryans matrislerinden türetilmişlerdir.

Layman'e göre, özyüzler "Yüz içeriğinin standartlaştırılmış" kümesinden ibarettir. Ve yüzün birden fazla resminin istatistiksel analizinden türetilmişlerdir. Herhangi bir insan yüzü bu standart yüzlerden birisi olabilir varsayımı yapılmıştır. Yaklaşım şu şekildedir : Bir insanın yüzü %10 birinci yüzden, %24 ikinci yüzden, %X üçüncü yüzden v.s. şeklinde yüzün tamamı için bir küme oluşturulur. Bu şekilde yüz tanıma için saklanan yüzler gerçekte fotoğraf makinesi ile çekilen yüzden daha az yer kaplayarak saklanabilmektedir.

Özyüzler tekniğinde çalışabilmek için önce büyük bir özyüz veritabanının oluşturulması gerekmektedir. Bunun için önce pek çok yüzün fotoğrafları çekilir, sonra bu yüz fotoğraflarının ağız ve göz seviyeleri standart hale getirilir, sonra yüzün resim içindeki hizalanması hepsi için aynı yapılır ve son olarak da hepsi aynı piksel boyutuna getirilir. Sonra bu resimlerin piksel değerlerini içeren eigenvektörleri üretilir. Bu eigenvektörleri ile en sonunda kovaryans matrisi çıkarılmaktadır.

Ortaya çıkan özyüzlerin tam olarak ortalaması alınabilirse ortalama bir insanın, yani herkesin benzeyebileceği bir insanın, gri tonlamada resmi çıkarılabilir.



Şekil 3.23. AT & T Laboratuvarlarından bazı özyüzler (Anonim 2006i)

Özyüzler metodu, Temel Bileşen Analizi (Principal Component Analysis – PCA) dönüşümleri yaparak yüzleri modellediği bir uzaya taşır. Bu dönüşümlere aynı zamanda Karhunen-Loeve dönüşümleri de denilmektedir. PCA insan yüzleri arasında ki değişkenliği belirlemektedir. Örneğin birbirinden farklı iki yüze bakıldığında aslında nelerin değişik olduğunu matematiksel olarak belirlemek için PCA kullanılır. PCA belirleme işlemini yüzleri veya yüz parçalarını kategorize ederek yapmamaktadır. Yani burnun büyüklüğü şu kadar, yüzün genişliği bu kadar şeklinde bir gruplama yaparak özellikleri belirlemez. Bunun yerine pek çok yüz incelenerek insanların yüzleri arasındaki farklılığı oluşturan değişkenler belirlenmektedir. İşte bu değişkenlere eigenfaces – özyüzler denilir. Herhangi bir insanın yüzünün diğer yüzlere benzeme oranı esastır ve bir yüz başka yüzler cinsinden ifade edilir. Şekil 3.24. bunu örneklemektedir.

Şekil 3.24. Bir yüze ait vektörün yüz uzayında şekilsel olarak gösterimi (Balasuriya ve Kodikara, 2001)

3.2.1.2. Yüz tanıma için yapay sinir ağları (Neural Networks)

Yüz tanıma için bazı araştırmacılar ve ticari amaçlı çalışmalar yapay sinir ağlarını da kullanmışlardır. Kullanılan metotların hepsi yapay sinir ağları olarak adlandırılsa da aralarında oldukça büyük farklılıklar vardır, yani hepsini tek bir metot gibi anlatmak oldukça güçtür. Bu bölümde biz daha çok pek çok araştırmacı tarafından kullanılan katmanlı algılayıcı (Multi Layered Perceptrons – MLP) metodu üzerinde duracağız.

Orjinal olarak Werbos tarafından formüle edilen MLP, pek çok birbirlerine tamamen bağlı lineer olmayan nöronlardan oluşan katmanlar içermektedir. Nöronlar arasındaki bağlantılar, çalışma kalıplarından oluşan çalışma uzayının değerlerini belirleyen ağırlıklar içermektedirler. Bu ağırlık değerleri geri-üreme tekniği ile ayarlanarak, ilişkilerin hata oranı en aza indirgenmektedir.

MLP işlemlerinin yüz tanımada ki görevi, özellik çıkartma (feature extraction) da diyebileceğimiz, yüzleri özünü işlemler sırasında temsil edecek yoğun ve olabildiğince kolay işlenebilecek bir model geliştirmektir. Bu yüzden gizli nöronların sayısı hem girdi hem çıktı katmanlarının sayısından daha azdır. Böylece ağ girdileri en önemli bilgileri saklayabilecek daha düşük boyutlu bir alt uzaya kodlayacaktır. Yapılan bir çalışma da lineer gizli nöronların oluşturduğu uzayın, PCA metodu tabanlı en yüksek değerlere sahip eigen-degerlerle oluşturulan uzay ile aynı olduğunu göstermiştir. Aralarındaki en önemli farkın ise varyansın birinci metotta gizli birimlerde dahi dağıtılmış olmasıdır (Grudin 2000).

Özetle denebilir ki MLP tekniği eigenfaces-özyüzler tekniği ile benzer yanlara sahiptir. MLP'nin en önemli farkı, Lanitis'e göre, her uygulamaya özel yapılan eğitim sırasında MLP'nin iç kullanım için modeli oluşturulmaktadır. Her uygulama için bu model değişebilmektedir.

3.2.1.2. Fisher yüzleri - Fisherfaces

Fisherfaces, eigenfaces ile benzeşen bir metottur. Çünkü tıpkı eigenfaces gibi yüzler bir alt uzaya modellenmektedirler. Bu metot ile yüzün ifadesindeki büyük farklılıklara ve ışıklandırma değişikliklerine rağmen yüzler alt uzaya doğru bir şekilde modellenebilmektedirler. Harvard ve Yale üniversitelerinin yüz veritabanları üzerinde yapılan çalışmalara göre "Fisherfaces" "EigenFaces" metoduna göre daha az hata oranına sahip sonuçlar üretmektedir (Belhumeur, Hespanha, Kriegman, 1997).

Aslında Fisher'in Lineer Ayırım (Lineer Discrimination) tekniği pek çok desen tanıma uygulaması için kullanılmaktadır. 1936 yılında ilk kez Robert Fisher tarafından üretilmiş ve taksonomik sınıflandırma için kullanılmıştır. Son zamanlarda yüz tanıma için de araştırmacılar bu metodu kullanmışlardır. Bu tekniğin en önemli getirisi yüzün ışıklandırılmasının farklı yönlerden yapılması sonucu elde edilen resimlerin çok farklı olması (Şekil 3.25.), yüzün ağlama, gülme gibi ifadelerinin yüz resmini çok değiştirmesi gibi değişkenlerin tatmin edici bir düzeyde izole edilmesidir.

Şekil 3.25. Harvard üniversitesi yüz veritabanından görüntüler : Farklı ışıklandırma ortamlarında aynı yüz ve ifade oldukça değişebilmektedir.

4. BULGULAR VE TARTIŞMA

Çalışmamızda daha çok gezgin cihaz üzerinden herhangi bir sunucuda yapılacak kimlik doğrulama (authentication) işlemini farklı bir teknik kullanarak geliştirmeye çalıştık. Günümüzdeki kimlik doğrulama sistemlerinin çoğu metin tabanlı olup, kullanıcıya uzun, karmaşık ve bulunması zor bir parolanın ezberletilmesine dayanmaktadır. Bu tür bir parola sistemi bir insan için zor olup, hatırlanmasında çeşitli problemler yaşanabilmektedir. Dahası bir şekilde bu parolayı tahmin edebilecek birisi sisteme kullanıcı adına giriş yapabilmektedir.

Böyle bir kimlik doğrulama sistemini gezgin cihazın çoklu ortam yetenekleriyle birleştirmek kullanıcı açısından metin tabanlı parolaların getirdiği dezavantajları kaldırmak için iyi bir yaklaşım olacaktır. Yaklaşımımıza göre kimlik doğrulama işlemi için aşağıdakilerden herhangi birisi kullanılabilir :

- İris tanıma : Bilindiği üzere insan gözünün bileşenlerinden olan iris, bir insanı diğerlerinden ayırmak için kullanılabilecek biyolojik bir veridir. Bir insanı iris ile tanıma işlemi uzun süredir araştırmacıları meşgul eden ve hatta günümüzde artık ticari olarak da gerçekleştirilmiş bir çalışmadır.
- Parmak izi tanıma : Parmak izi de iris gibi insanı diğerlerinden ayıran bir veridir. İrise göre parmak izini tanıma günlük hayatta daha sık uygulanan bir tekniktir.
- Yüz tanıma : Tanıma işlemleri içinde belki de en zoru yüz tanıma işlemidir. Çünkü yüz tanıma işlemi a) yüzü bulma, b) özellikleri çıkartma, c) matematiksel olarak modelleme gibi pek çok alt işlem içermektedir. Metot bölümünde yüz tanıma ile ilgili bazı sık kullanılan teknikler hakkında bilgi verilmiştir.

Araştırmacılar ve ticari firmalar iris, parmak izi ve yüz tanıma ile ilgili pek çok çalışma ortaya koymuştur. Fakat genel olarak bunu cep telefonunun çoklu ortam yetenekleri ile birleştirip, internet üzerinden gönderen ve doğrulamayı internet tabanlı uygulamalara odaklayan çalışmalar pek azdır. Çünkü şimdiye kadar gerek cihazların çoklu ortam seçeneklerin azlığı, gerek sunucu ile iletişim hızlarının yavaşlığı gibi sebeplerle gerekli ihtiyaçlara yeterli düzeyde yanıt verecek teknolojinin bulunmaması söz konusuydu, fakat günümüzde bir kaç megapixel'lik fotoğraf çekme yeteneğine sahip cep telefonları, internet üzerinden hızlı veri gönderimi yapılmasını sağlayacak EDGE gibi üçüncü nesil teknolojiler, cep telefonları üzerinde programlamayı kolaylaştıracak akıllı telefon teknolojileri sayesinde bu işlemler hayal olmaktan çıkmıştır.

Çizelge 4.1. Biyolojik verilerin karşılaştırılması

	Biyolojik karakteristikler			**Davranış karakteristikleri**	
	Parmak izi	İris	Yüz	İmza	Ses
Maliyet	Düşük	Yüksek	Orta	Düşük	Düşük
Güvenirlik	Yüksek	Orta	Orta	Düşük	Orta
Uzun vadede stabilite	Yüksek	Orta	Orta	Düşük	Orta
Hata oluşturacak durumlar	Nadir, sahtecilik az işlenmiş	Az, lensler ve gözlükler	Işık, yaş ve yüz ifadesindeki farklılıklar	Zaman zaman stildeki değişiklikler	Kayıt kalitesi, fiziksel şartlar

Çizelge 4.2. Biyolojik verilerin uygulama açısından karşılaştırılması

	Eşleştirme	**Veri boyutu**	**Uygulamalar**
Parmak izi	Ayrıntılar	1000	Çok
Yüz	Burun, göz ve yüz detaylarının geometrisi	1000	Orta çoklukta
İris	İris deseni	500	Az
İmza	Stil ve baskı	1000	Az
Ses	Ses frekansı	1000	Az

4.1. Parmak izi tanıma uygulamaları

Parmak izinin insanlarda tekrar etmediğine dair bilgi 1600'lü yılların sonlarına doğru keşfedilmiştir. Daha sonra 1800'lü yıllarda bu bilgi uygulamalarda kullanılmıştır. Örneğin suçluların tanımlanmasında yoğun olarak kullanılmaya başlanmış, hatta Japonya 1911 yılında bununla ilgili kanunlar çıkartmıştır. 1960'larda parmak izi tanıma artık karakollarda kullanır hale gelmiştir. 1993 yılında ANSI parmak izi resimleri bir standart yayınlamıştır ve böylece parmak izi tanımanın bilgisayar uygulamalarındaki gelişimi için artık bir standart konulmuştur.

Parmak izi tanımanın diğer tanıma uygulamalarına göre önemli olan farkı bir cep telefonunda gerçekleştirilmesinin daha zor olmasıdır. Bu uygulamanın gerçeklenmesi için ya dokunmatik bir ekran ile parmak izinin alınması ya da fotoğraf makinesi ile parmak izinin alınması gerekmektedir. İkinci durumda parmak objektife tam olarak yaklaştırılamayacağı için bu metot ile gerçekleştirme söz konusu değildir. Birinci durumda gerçeklenme biraz daha mümkün görünsede, pazarda varolan cihazlar hemen hemen hiçbirisi ile – dokunmatik ekranlı olanlar dahil – bu iş için tam olarak uygun değillerdir. Ancak harici bir donanım ile bu işlem gerçekleştirilebilmektedir.

Masaüstü ve taşınabilir bilgisayarlar için değişik firmalar parmak izini bilgisayara aktaracak harici donanımlar üretmişlerdir (Şekil 4.1). Bu cihazlar USB gibi bir porttan bilgisayara bağlanıp, parmak izini çeşitli tolerans parametreleri ile – kuruluk, fiziksel hasar gibi – alıp resim formatında veya başka bir formatta bilgisayara aktarabilmektedirler.

Şekil 4.1. Technoimagia firması tarafından üretilen kablolu ve kablosuz iki ayrı model halinde parmak izin tanıma sensörü.

4.2. İris tanıma uygulamaları

İris tanıma uygulamaları pek çok ticari firma ve araştırmacı tarafından gerçeklenmiştir. Örneğin OKI firması 2006 yılının sonlarına doğru dahili kamerası olan cep telefonları ve PDA cihazları için iris tanıma sistemini tanıtmıştır. Firma bu iş için kendi geliştirdikleri iris tanıma algoritmasını kullandıklarını ve optik kamera sistemi bulunan cihazların tümünde uygulamanın çalışabileceğini açıkladı. Ayrıca OKI bu uygulamayı 2007 içinde ticari olarak pazara süreceğini söyledi.

OKI firması bir süreden beri firmalar için iris tabanlı güvenlik sistemleri üretiyordu. Bu sistemler kızıl ötesi kamera kullanıyor ve genelde finans kurumları, devlete bağlı ajanslar gibi güvenliğin önemli olduğu yerlerde kullanılıyordu.

OKI firması uygulamayı klasik PIN kilidinin yerine düşünmektedir. Yani iris tanıma uygulaması tamamen cep telefonu içinde çalışacaktır. İlk akla gelen dezavantajlar böyle bir matematiksel işlemler bütününün nasıl bir cep telefonunda ne kadar süre içinde gerçekleşeceğidir. Şu an için pazardaki telefonların çoğu kısıtlı işlemci gücüne sahip ve bu yüzden RSA gibi firmalar şifreleme algoritmalarının daha düşük işlemci hızında çalışacağı sürümlerini geliştirmeye çalışıyorlar.

Firmanın yaklaşımını da gözardı etmemek gerekiyor. Pek çok kullanıcı cep telefonunda kendisi için önemli ve gizli bilgiler tutmakta ve bu cihazlar çalındığında veya başkasının eline başka bir şekilde geçtiğinde cihazın içindeki bilgilerin kullanılması sonucu güvenlik açıkları çıkabilmektedir. Aslında buna güvenlik açığından çok güvenliğin delinmesi de diyebiliriz. Böylece bir cep telefonu bir sunucu gibi üzerinde gizli bilgilerin bulunduğu, güvenli olarak kimlik doğrulama ve girişin yapılması gereken bir cihaz durumuna gelmiştir.

Şekil 4.2. OKI firmasının uygulaması için sağladığı görüntüler (Anonim 2006j)

4.3. Yüz tanıma uygulamaları

Krithika Venkataramani, Saim Qidwai, and B. V. K. Vijayakumar, 2005 yılında yaptıkları çalışmalarında cep telefonu ile çekilen fotoğrafları yüz tanıma işlemleri için test

edip, yüz tanıma algoritmalarının gerçek yaşamda gerçekten başarılı olup olmadıklarını araştırdırlar. Aşağıdaki maddeleri sonuç olarak çıkarabiliriz :

- Yüz tanıma cep telefonu ile çekilen fotoğraflarla yapılabilir
- Korelasyon filtresi (Correlation Filter) yüz tanıma için kullanılan diğer algoritmalara göre daha başarılı bulunmuştur. Karşılaştırılan diğer algoritmalar : PCA, FisherFaces
- Korelasyon filtresi başarılı olmasına rağmen sürekli güncellenmelidir ve PDA, cep telefonu gibi cihazlar üzerinde çalışmaya çok da müsait değildir (Venkataramani, Qidwai ve Vijayakumar, 2005).

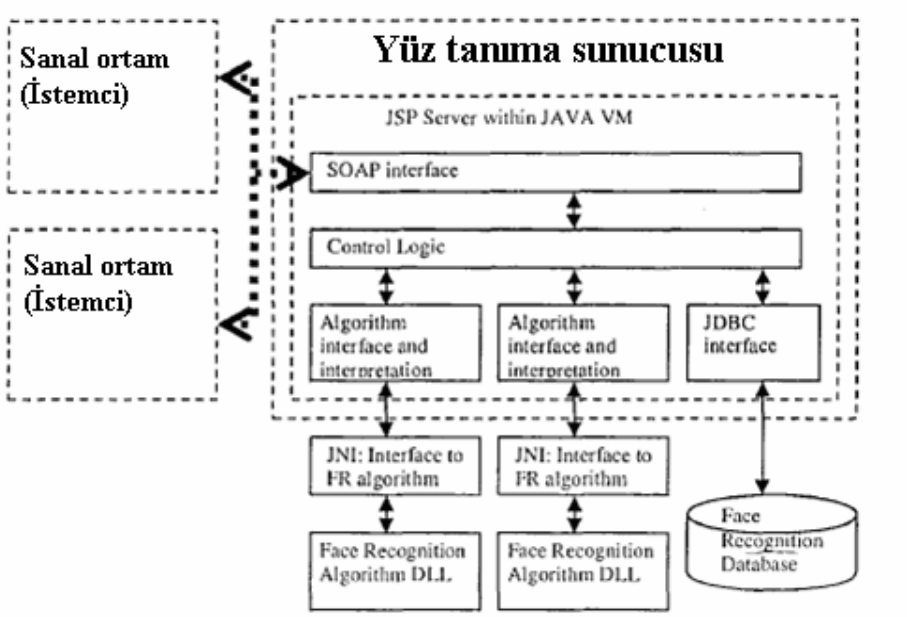
Öte yandan, Visionics firmasının desteğiyle Motorola sadece özel amaçlar için kullanılmak üzere, cep telefonlarında yüz tanıma uygulaması gerçekleştireceklerini açıkladı. Firma Java/J2ME destekli telefonlarda çalışacak şekilde planlanan uygulama daha çok suçluların tespiti gibi işler için düşünülmektedir. Fakat eleştiriler uygulamanın güvenlik ile ilgili geliştirme yapılırken pek çok kişinin mahremiyetini de zedeleyeceğini söylemektedir. Ayrıca uygulamanın cihazda çalışma hızı, çok detaylı olarak yüz tanıma yapabilmenin getireceği işlem maliyeti de gezgin cihazlar için şu sıralar pek uygun görülmemektedir.

Adler, 2001 yılında yaptığı çalışmasında internette anonim olarak dolaşan kişilerin kimlik tespiti ve bazı sitelerde kişilerin gerçekten bilinerek gezinti yapmalarını sağlamak gibi faydalar açısından "Sanal ortamlar için yüz tanıma" başlıklı çalışmasında böyle bir altyapının gereksinimlerinden bahsetmiştir. Önerdiği uygulama Şekil 4.4'de özetlenmiştir (Adler, 2001).

```
<?xml version="1.0"?>
<FacerecEngine>
 <executeCommands>
  <commands>
   <command>
    <action>UPLOAD | QUERY |
            DELETE | CREATE_TEMPLATE|
            SEARCH </action>
    <ref_number> id </ref_number>
    [
     additional parameters
    ]
   <requires> unique id </requires>
   </command>
   <command> ... </command>
   .
  </commands>
 </executeCommands>
</FacerecEngine>
```

```
<?xml version="1.0"?>
<FacerecEngine>
 <executeCommandsResponse>
  <success> 0 | 1 </success>
  <results>
   <result>
    <action>UPLOAD | QUERY |
            DELETE | CREATE_TEMPLATE
            SEARCH </action>
    <success> 0 | 1 </success>
    <error_code> value </error_code>
  [
     action specific data
  ]
   </result>
   <result> ... </result>
   .
  </results>
 </executeCommandsResponse>
</FacerecEngine>
```

Şekil 4.3. Adler'in önerdiği SOAP tabanlı protokol için istek-cevap şablonları



Şekil 4.4. Adler'in önerdiği sanal yüz tanıma hizmetinin akışı (Adler, 2001).

Bu verilere göre yüz tanıma işlemini cep telefonu ile yapmanın en verimli yolu, yüz tanıma işlemlerini cep telefonu dışında bir yerde yapmaktır. Örneğin çekilen resmin internet yoluyla bir sunucuya iletilmesi ve sunucunun üzerinde bulunan yazılım ve veritabanı ile yüzün tanınmasının yapılmasıdır. Böyle bir sistemin getireceği başka bir avantaj da yüz tanımayı bir hizmet gibi tüm istemcilere sunabilmektir.

Böyle bir sistemde tüm kullanıcıların verileri bir sunucu üzerinde tutulacak, böylece her cihazlar üzerinde kalıcı depolama ihtiyacı giderilecektir. Ayrıca böyle bir sistemi gezgin cihazlar, masaüstü bilgisayarlar veya başka tür cihazlar da kullanabileceklerdir. Bu hizmeti genelleştirmek için hizmeti Web Hizmetleri (Web Services) altyapısı ile sunmak ve istekleri SOAP protokolü ile gönderip, almak iyi bir çözüm olacaktır.

4.4. Önerilen uygulama

Tezimizde önereceğimiz uygulama

- Maliyet
- Güvenirlik
- Gezgin cihazların kısıtlı işlem ve hafıza olanakları

göz önünde bulundurulduğunda "İnternet üzerinden kimlik doğrulama işlemlerinin yüz tanıma ile yapılması" olacaktır. Böyle bir uygulama da cep telefonu/gezgin cihaz üzerinde sadece fotoğraf çekme ve bir şekilde programlanabilirlik yetenekleri olması yeterlidir. İşin yorucu kısmı olan depolama, işleme ve doğrulama gibi işlemler bir sunucu üzerinde yapılacaktır. Önerdiğimiz sistem Adler'in önerdiği alt yapıyı kullanabilir. Fakat genel bir hizmet olmayacaksa daha basit bir altyapı çözülmesi daha mantıklı olacaktır.

Uygulamamızdaki adımlar, eğer kullanıcı sunucu üzerinde kayıtlı değilse :

1. Cep telefonu ile internet üzerinden sunucuya bağlanma

2. Sunucuya kayıt amaçlı fotoğrafların ve diğer kişiyi tanımlayıcı bilgilerin metin tabanlı gönderilmesi
3. Sunucunun kişi bilgilerini veritabanına kaydetmesi
4. Sunucunun kişinin fotoğrafını eğitim için işlemesi (training) ve bu fotoğrafdan gerekli eigenfaces/özyüzleri çıkartması. Bu sonraki doğrulamalar için gerekli olacaktır ve bu adımda işlemin yapılması ilerideki doğrulama işlemlerini hızlandıracaktır.
5. Sunucunun gezgin cihaza olumlu sonuç döndürerek, kaydın tamamlandığını bildirmesi

Uygulamamamızdaki adımlar, eğer kullanıcı kayıtlı ve doğrulama yaptırmak istiyorsa

1. Cep telefonu ile internet üzerinden sunucuya bağlanma
2. Fotoğraf ve kimlik bilgisini sunucuya gönderme
3. Sunucuda verilen bilgilerin veritabanı ve yüz tanıma algoritmasının ikisi ile birlikte doğrulanması ve "Doğru/Yanlış" formatında bir sonuç üretilmesi
4. Sonucun istemciye gönderilmesi

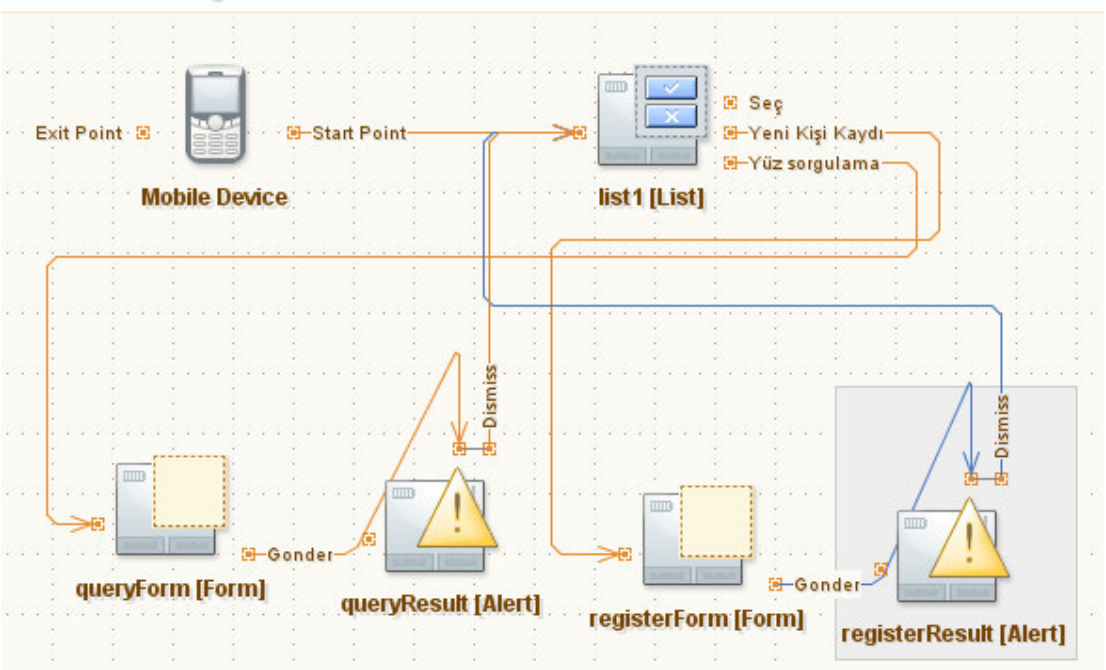
Materyal ve metot bölümünde değindiğimiz gibi, bu uygulamanın tüm adımları Java platformu ile gerçekleştirilmiştir. İstemci tarafında çalışacak uygulama

- Java ME sürümü
- Java MMAPI programlama arayüzü

programlama araçlarını kullanmıştır.

Sunucu tarafında ise

- Tomcat veya başka bir uyumlu JSP kabı, sunucusu
- JSP ve Servlet teknolojisi
- JDBC uyumlu bir veritabanı
- Yüz tanıma için EigenFaces metodunu kullanan Java tabanlı bir uygulama kullanılmıştır.



Şekil 4.5. Gezgin cihaz üzerindeki uygulamamızın basit olarak akışı

5. SONUÇ VE ÖNERİLER

Üçüncü nesil cihazlar ile internet kullanım hızı ve oranı artacağından, yeni uygulamalar interneti daha çok kullanacak ve uygulama geliştiriciler güvenlik açısından eskiden kullanmadıkları, teknik yetersizlikten dolayı kullanamadıkları teknikleri gelişen teknoloji ve internet hızıyla kullanmayı düşünebilecekler.

Örneğin yüz tanıma, iris tanıma, parmak izi tanıma uygulamaları önceden gerek cihazın üzerindeki donanımsal eksiklikler, gerekse cihazın çok kısıtlı işlemci ve hafıza olanakları yüzünden düşünülmeyen uygulamalardı. Günümüzde ise artık neredeyse yüz tanıma cihaz içindeki işlemci üzerinden yapılacak derecede cihazlar geliştirildi.

Bu cihazlar üzerinde programlama için olanaklarda oldukça gelişmiştir. Örneğin, Java teknolojisi geniş yelpazesi ve yüksek seviyeli bir programlama ortamı sunması ile her ortam için tercih edilen bir programlama platformudur. Gerek sunucu taraflı uygulamalar, gerekse gezgin cihazlar üzerindeki uygulamalar için gerekli herşeyi sunmaktadır ve oldukça yüksek yüzdede tüm dünyadaki geliştiriciler ve üreticiler tarafından tercih edilmektedir. Günümüzdeki çoğu ikinci nesil ve üçüncü nesil cep telefonları, PDA'lar gibi gezgin cihazlar Java'yı desteklemektedirler. Dolayısıyla pek çok farklı cihazın iletişim kurmasını gerektiren sanal ortamlar için Java uygun bir platformdur.

Bizim önerdiğimiz uygulama da güvenliğin artırılması için yüz tanıma tabanlı bir kimlik doğrulama sistemini gerçekleştirmeye çalıştık. Bu uygulama protokol bazından varolan güvenlik teknolojilerini kullanacak ama uygulama bazında bazı yenilikler getirecektir. Uygulama aslen parola tabanlı kimlik doğrulama uygulamaları için bir alternatif sunmaktadır.

Bu uygulamanın klasik parola tabanlı uygulamalara göre avantajları :

- Metin tabanlı, karışık, uzun ve tahmin edilemeyecek parolalardan kurtulma
- Taklit edilemez bir bilgi ile kimlik doğrulama
- Anonimlikten kurtulma ve kurtarma. Özellikle bazı kritik uygulamalar, bankacılık gibi, kullanıcının anonim olmasından ziyade, bilinir kişi olmasını tercih etmektedir
- Kullanıcı açısından kolaylık

gibi maddeler olarak listelenebilir.

Bu uygulamanın çeşitli dezavantajları ve uygulama sırasında, pratikte getirdiği zorluklar da vardır. Bunları şöyle özetleyebiliriz

- Yüz tanıma sırasında kullanılacak resimlerin hassas algoritmalar tarafından işlenecekleri için resim kalitesi önemlidir. Bu yüzden cep telefonları üzerindeki fotoğraf çekme yeteneklerinin daha da geliştirilmesi gerekmektedir.
- Durgun resim (Still Image) üzerinden resim tanıma işlemi için çekilen resimlerin boyut, hizalama, yüzün konumu, ışık durumu gibi açılardan benzer veya aynı olması istenmektedir. Bu tür fotoğrafları çekerken kullanıcılar gerek fiziksel şartlar, gerek dikkat eksikliği gibi faktörlerden dolayı gereken düzeyde düzenli fotoğraflar çekmekte pratikte yetersiz kalmaktadırlar.
- Tam olarak %100 oranında güvenirlik sağlayan bir yüz tanıma algoritmasının henüz geliştirilmemiş olması
- Yüz tanıma uygulamalarının ışık, yaş ve yüz ifadesi gibi etkenlerden çok kolay etkilenmesi ve değişik sonuçlar üretebilmesi

Çeşitli dezavantajlarına rağmen yüz tanıma ile kimlik doğrulama oldukça çekici bir konudur. İleride internet üzerinden yüz veritabanı hizmeti veren devlet kurumları olabileceği gibi, herkesin birbirini tanıyabileceği, sahteciliğin yüz tanıma gibi uygulamalarla en aza indirgeneceği günlerde gelecektir. Eğer tüm internet kullanıcıları devlet veritabanı üzerinden yüz tanıma ile kimlik doğrulama yaparak internete bağlanmak zorunda bırakılırsa, alt yapı bunu desteklerse, gerekli protokoller geliştirilirse, elbette bu konuşulanlar hayal olmaktan çıkacaktır.

Çalışmamızda biz bu tür bir uygulamanın akışını, çalışmasını ve eksik yönlerini inceledik. Yüz tanıma işlemini durgun resimler yerine, hareketli filmler veya üç boyutlu resimler ile yaparak doğrulama işleminin güvenirliğini artırabiliriz. Kullanıcı giriş yapacağı sırada sunucu tarafından gelecek talimatlara göre poz verip bu pozları sunucuya gönderecek bir uygulama da bu tür bir güvenlik artırımı sağlamaktadır. Fakat bu tür bir uygulama talimatları güvenli bir şekilde alacak ve işleyecek, kullanıcı tarafından kırılamayacak bir protokol alt yapısı ile ancak çalışabilecektir.

KAYNAKLAR

ADLER A., 2001. Automatic Face Recognition System Architecture For Collaborative Virtual Environments, IEEE Int. Workshop Haptic Virt. Environ., 1:1-6.

AKÇAY, A. R. 2001. 3.Nesil Gezgin İletişim Sistemleri-2. http://turk.internet.com/ haber/yazigoster.php3?yaziid=2048, s1.

AKSU M. 2004. 3. Nesil Cep Telefonları İçin Symbian İşletim Sistemi Üzerinde Çalışabilen C++ Tabanlı Uygulama Geliştirilmesi, K.S.Ü. Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, s48-53

ANONİM, 2001a. Wireless Security And M-Commerce, http://www.thefeaturearchives.com/topic/Archive/Wireless_Security_and_M-Commerce.html.

ANONİM, 2002a. Guide to Mobile Internet Security, http://www.kannel.org/download/kannel-wtls-snapshot/wtls.html.

ANONİM, 2003a. Taking Pictures with MMAPI, http://developers.sun.com/techtopics/mobility/midp/articles/picture/

ANONİM, 2003b. J2ME Networking, http://www.j2medev.com/Article/Class1/Class14/200506/272.html

ANONİM, 2003c. Network Programming With J2ME, http://www.wirelessdevnet.com/channels/java/features/j2me_http.phtml

ANONİM, 2004a. Differences between WAP and Web, http://csc.colstate.edu/summers/Research/Wireless/WAPvsWeb.html.

ANONİM, 2006a. New Nokia 3G N93 Mobile Phone, ww.3g.co.uk/PR/April2006/2976.htm.

ANONİM, 2006b. Wikipedia 3G Article, http://en.wikipedia.org/wiki/3G

ANONİM, 2006c. Wikipedia Türkçe, 3G Makalesi http://tr.wikipedia.org/wiki/3G

ANONİM, 2006d. Symbian Fast Facts, http://www.symbian.com/about/fastfacts/fastfacts.html

ANONİM, 2006e. Wikipedia Symbian OS Article, http://en.wikipedia.org/wiki/Symbian_OS

ANONİM, 2006f. MIDP Programming with J2ME, http://www.developer.com/java/j2me/article.php/10934_1561591_1.

ANONİM, 2006g. Wireless Development Tutorial Part I, http://developers.sun.com/techtopics/mobility/midp/articles/wtoolkit/

ANONİM, 2006h. Netbeans IDE, http://www.netbeans.org/index_tr.html

ANONİM, 2006ı. Developing Mobile Applications : JBuilder 2005, http://info.borland.com/techpubs/jbuilder/jbuilder2006/mobile.pdf

ANONİM, 2006i. Wikipedia EigenFace Article, http://en.wikipedia.org/wiki/Eigenfaces

ANONİM, 2006j. OKI İris Tanıma Teknolojisi Kameralı Telefonlarda, http://www.letsgodigital.org/tr/11528/iris_tanima_teknolojisi_telefonlarda/

ANONİM, 2006k. Nokia White Paper, Next Generation Mobile Browsing : Rich and colorful browsing for mobile users

ATALAY I., 1996. Face Recognition Using Eigenfaces, ITU Fen Bilimleri Enstitüsü Yüksek Lisans Tezi, s.1-5

BALASURIYA L. S., KODIKARA N.D., 2001. Frontal View Human Face Detection and Recognition, Proceedings of the International Information Technology Conference, Colombo

BAYIZ, Y., EKEN, S. K. 2001. GSM'in geleceği: UMTS. İletişim Teknolojileri Bildiriler Kitabı, s31-32.

BELHUMEUR P. N., HESPANHA P., KRIEGMAN D., 1997. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection, IEEE Trans. PAMI, Special Issue on Face Recognition, 19(7), s 711-720

BÜYÜKÇAKIROĞLU, B. 2003. Multimedia (Smartphone) cep telefonları. Avuçici Web Sayfası. http://www.avucici.com/habergoster.php?ID=28

DANAHER M., NGUYEN D., 2002. Mobile Home security with GPRS, Proceedings of the International Conference on. Information Science (ISI 2002), s.376-381

GINESU G., GIUSTO D.D., ONALI T., 2006. Mutual Image-BasedAuthentication Framework with JPEG2000 in Wireless Environment, EURASIP Journal on Wireless Communications and Networking, vol. 2006, Article ID 73685, s.1-14

GRUDIN A. M., 2000. On internal representations in face recognition systems, Pattern Recognition, vol. 33, no. 7, s. 1161-1177.

HALL M., 2000. Core Servlets And Java Server Pages, Chapter 1, s5-10

İNCEOĞLU M. M., KILINÇ E., 2003. WAP Ağ Geçidinde Ortaya Çıkan Güvenlik Boşluklarının Kapatılmasına İlişkin Öneriler, IX. "Türkiye'de Internet" Konferansı, s. 1-3.

JANSEN W., 2004. Authenticating Mobile Device Users Through Image Selection, In The Internet Society: Advances in Learning, Commerce and Security, K. Morgan & M. J. Spector (Editors), v30, s10

JERMYN I., 1999. The Design and Analysis of Graphical Passwords, Proceedings of the 8th USENIX Security Symposium

KROLL M., 2002. Java 2 Micro Edition (J2ME) Application Development, Chapter 3

PERRIG A., SONG D., 1999. Hash Visualization: a New Technique to improve Real-World Security, Proc. 1999 International Workshop on Cryptographic Techniques and E-Commerce (CryTEC '99).

SELIAN, A. 2001. 3G Mobile Licensing Policy: From GSM to IMT-2000-A Comparative Analysis. ITU, http://www.itu.int/osg/spu/ni/3g/casestudies, s20-25.

SUN Microsystems, 2006. Java ME Technology. http://java.sun.com/javame/technology/index.jsp

ÖZÇELİK M. A., 2006. Bluetooth Üzerinden Güvenli Veri İletimi, K.S.Ü. Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, s1-10

VENKATARAMANI K., QIDWAI S., VIJAYAKUMAR B. V. K., 2005. Face Authentication From Cell Phone Camera Images With Illumination and Temporal Variations, Ieee Transactıons On Systems, Man, And Cybernetıcs—Part C: Applications And Reviews, Vol. 35, Issue 3, s.411-418

ÖZGEÇMİŞ

Mustafa KARABULUT 03 Mart 1979 tarihinde Gaziantep'de doğdu. İlkokul eğitimini Gaziantep Özdemirbey İlkokulunda, orta ve lise eğitimini de yine Gaziantep'de Fitnat Nuri Tekerekoğlu Anadolu Lisesinde tamamladı. 1997 yılında Çanakkale 18 Mart Üniversitesi Bilgisayar Mühendisliği bölümünü kazandı ve 2001 yılında buradan mezun oldu.