



T.C.  
MALTEPE ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

**MOBİL KARGO VE TAŞIMACILIK SİSTEMİ**

**H.M.Yıldırım KOÇDAĞ**

Yüksek Lisans Tezi

**Tez Danışmanı**

**Prof. Dr. İlhami YAVUZ**

**İSTANBUL – 2006**



**T.C.  
MALTEPE ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**MOBİL KARGO VE TAŞIMACILIK SİSTEMİ**

**YÜKSEK LİSANS TEZİ**

**H.M.Yıldırım KOÇDAĞ**

**Tez Danışmanı  
Prof. Dr. İlhami YAVUZ**

**İSTANBUL – 2006**

Bu tez çalışması, Maltepe Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 26 / 06 / 2006 tarih ve 2006 / 18 sayılı kararıyla oluşturulan jüri tarafından ***Bilgisayar Mühendisliği Yüksek Lisans Tezi*** olarak kabul edilmiştir.

JÜRİ

Prof. Dr. İlhami YAVUZ

Danışman

Prof. Dr. A.Mesut RAZBONYALI

Üye

Prof. Dr. Kemal Köymen

Üye

## ÖZET

Mobil ticaret gün geçtikçe önemi artan kritik bir satış kanalı olarak bir çok firmanın dikkatlice takip ettiği bir husus olmuştur. Fakat mobil ticaretin getirdiği zorluklar bir çok farklı sektörde çalışan firmanın bu alanda çekingen kalmasına ve sadece izlemeye beklemesine sebep olmuştur. Bu firmalardan bir bölümünde özellikle teslimat ve kargo taşıyan firmalar olmuştur.

Kargo firmalarının yeni teknolojileri kağıt – kalem çözümlere kıyasla tercih etme oranları, mobil cihaz teknolojisi ortaya çıktıktan sonra keskin olarak yükselişe geçmiştir. Bunun yanında bu konuda projeler geliştiren firmalar ve mobil cihaz üreten firmalar, bu trend den çok büyük kazançlar elde etmeye başlamıştır. Şirketler içinde hatta şirketlerin bilgi işlem departmanları içinde sadece mobil cihazlar ile ilgilenen çalışanlar, mühendisler yetiştirilmeye başlanmış, mobil cihaz takımları kurulmaya başlanmıştır.

Projenin amacında bu gelişmeyi dikkate alarak bir mobil cihaz uygulaması geliştirmektir. Projeyi tasarlarak tüm yeni nesil teknoloji altyapısını kullanmak, güvenlik ve uluslararası standartları dikkate almak önem verilecek noktalardır.

**Anahtar Kelimeler :** Mobil ticaret, mticaret, mobil taşımacılık, kargo firmaları için mobil uygulamalar, mobil imza, mobil cihaz üzerinde imza şifreleme, PDA tabanlı taşımacılık sistemi ve taşımacılık sektöründe cep bilgisayarlarının önemi.

## ABSTRACT

Mobile commerce is becoming a critical sales channel for a lot of companies everyday. However, the firms were pesimistic about the new system, because of its unspecified drawbacks. In the last a few years the companies barrels got down, and they started to make a lot of profits and take the advantages of new technology. One of these sectors is the working subject of the companies which are doing public shipment

The percentage of popularity about the shipment companies which are using new technologies such as notebooks, and PDAs is madly growing. By all the mean, A slew of outsources and software makers started to convience companies with stational information of the proliferate turnovers of the others. In this point, the number of mobile applications and the software companies which are taking some strategical decisions about developing mobile system are growing. The aim of this project is considering the mobile commerce as a powerful system and prepare a shipment system which can be able to store the delivery information and also take digital signature of clients.

One of the secure delivery system is capturing signature of clients via PDAs which is also inspected and developed in the project with cryptographic algorithms and digital image processing methods. The other pivotal point which examined is Mobile Database. The mobile technology is supported with online transmission of the delivery information. However, the system should be also designed for difficult scenarios such as being offline or having connection problems. In this point, the importance of the database of the mobile devices can be more. The last point is security concerns center on how to protect transactions as they pass from wireless network or cable network across the m-commerce infrastructure and into corporate back-end systems. On the other hand, how the developed system is deal and surpassed about these concerns.

**Keywords:** Mobile commerce, mcommerce mobile shipment, mobile applications for transportation companies, mobile signature, signature encryption on mobile device, PDA based shipment system and influence of mobile devices on transportation companies.

## **TEŐEKKÖR**

**Danıőman Hocam Prof. Dr. İlhami YAVUZ'a projemi hazırlama süreci boyunca verdiği destekten dolayı teşekkür ederim.**

## İçindekiler Listesi

ÖZET .....	IV
ABSTRACT .....	V
TEŞEKKÜR.....	VI
İçindekiler Listesi.....	VII
Kısaltmalar .....	VIII
Şekiller Listesi.....	IX
Tablolar Listesi .....	X
1. GİRİŞ .....	8
2. YENİ YAPI “mShipment” .....	9
2.1. Cep Bilgisayarlarının Evrimi .....	10
2.1.2. Yeni Nesil Cep Bilgisayarları .....	11
2.1.2. Pazar Liderleri.....	14
2.2. Sistem Mimarisi .....	16
2.3. Geliştirilmiş Varlık İlişki Diyagramı .....	18
2.4. Kullanıcı Senaryoları .....	19
2.4. Yeni Sistemin Gant Diyagramı .....	22
3. SİSTEM FİZİBİLİTESİNİN HAZIRLANMASI .....	23
3.1. Teknik Fizibilite .....	24
3.1.1. Masaüstü Bilgisayar ve Cep Bilgisayarı Üzerinde Kullanılan Yazılımlar... ..	24
3.1.2. İnternet Bağlantısı .....	25
3.1.3. Masaüstü ve Cep Bilgisayarının Minimum Donanım İsterleri .....	26
3.1.4. Çalışan Kaynağı .....	27
3.2. Ekonomik Fizibilite.....	27
3.2.1. Yazılım Tutarı .....	27
3.2.2. İnternet Bağlantısı Tutarı .....	28
3.2.3. Donanım Tutarı .....	29
3.2.4. Proje Çalışanlarının Maaliyetleri .....	29
3.3. Sistem Güvenliği .....	30
3.3.1. XML Standartı .....	30
3.3.2. İleri Şifreleme Sistemi AES (Advance Encryption System).....	34
3.3.2.1. AES’in Adımları .....	34
3.3.2.2. AES’in ‘mShipment’ İçerisinde Uygulanması .....	36
3.4. İmza Yakalama.....	38
3.5. Sistemin Kullanılabilirliği .....	39
5. REFERANSLAR .....	40
EKLER.....	40
A) Sistemin Ekran Görüntüleri .....	40
B) Sistem Kodları.....	47
C) Öz Geçmiş.....	103



## Kısaltmalar

<b>Kısaltma</b>	<b>İngilizcesi</b>	<b>Türkçesi</b>
AES	Advance Encryption System	İleri Şifreleme Sistemi
CPU	Central Processing Unit	Merkezi İşlem Birimi
GPS	Global Position System	Global Yer Belirleme Sistemi
IR	Infra Red	Kızıl Ötesi
PIM	Personal Information Manager	Kişisel Bilgi Yöneticisi
XML	eXtensible Markup Language	Genişletilebilir İşaretleme Dili

## Şekiller Listesi

Şekil 1 – Geliştirilmiş Varlık İlişki Diyagramı .....	19
Şekil 2 – Senaryo 1 .....	20
Şekil 3 – Senaryo 2 .....	21
Şekil 4 – ‘mshipment’ Projesinin Gant Diyagramı .....	23
Şekil 5 – Örnek XML Dökümanı .....	33
Şekil 6 – Dönen Anahtarın Eklenmesi (AddRoundKey) .....	35
Şekil 7 – Alt Veriler (SubBytes) .....	35
Şekil 8 – Satırların Kaydırılması (Shift Rows) .....	36
Şekil 9 – Kolonların Karıştırılması (MixColumns) .....	36
Şekil 10 – XML Dosyası .....	37
Şekil 11 – Şifrelenmiş XML Dosyası .....	38
Şekil 12 – Cep Bilgisayarının Kurulumu .....	41
Şekil 13 – İşletim Sisteminin Başlangıcı .....	41
Şekil 14 – Program Girişi .....	42
Şekil 15 – Ana Menü .....	42
Şekil 16 – Teslimat Menüsü .....	42
Şekil 17 – İmza Ekranı .....	42
Şekil 18 – Veri Aktarımı .....	43
Şekil 19 – Cep Bilgisayarının Veritabanı .....	43
Şekil 20 – Çalışanlar Ekranı .....	43
Şekil 21 – Müşteriler Ekranı .....	44
Şekil 22 – Teslimatlar Ekranı .....	44
Şekil 23 – Müşteri İmza Ekranı .....	45
Şekil 24 – Şehirler Ekranı .....	46
Şekil 25 – İlçeler Ekranı .....	46
Şekil 26 – GPS Harita .....	47

## Tablolar Listesi

Tablo 1 – Geliştirme Anında Masaüstü ve Cep Bilgisayarında Kullanılan Programlar	25
Tablo 2 – Çalışma Anında Masaüstü ve Cep Bilgisayarında Kullanılan Programlar	25
Tablo 3 – Çalışma Anındaki Minimum Donanım İsterler	26
Tablo 4 – Geliştirilme Anındaki Minimum İsterler	26
Tablo 5 – Çalışanlar ve Geliştirme Süreleri	27
Tablo 6 – Programların Tutarı	28
Tablo 7 – İnternet Bağlantısı Tutarı	29
Tablo 8 – Donanım Tutarı	29
Tablo 9 – Proje Çalışanlarının Maaliyetleri	30

## 1. GİRİŞ

Mobil cihazlar gibi yeni çağ teknolojileri sayesinde iş hayatı bir birinden farklı yeni yaklaşımlar ve yeni yetenekler kazanmıştır. Yaşam standartları ve iş şartları, bu yeni teknolojiler ile gittikçe iyileşmektedir. Bu teknolojilerin gelişmesi, bu konudaki eleştirileride her geçen gün arttırmaktadır. Örnek olarak, kağıt ve kalem ile çalışmanın sayısal ortamda çalışmaya göre oldukça basit ve hızlı olduğu sürekli dile getirilmektedir. ‘**mShipment**’ projesi, yeni teknolojileri ve mobil cihazları kullanarak, bu tarz eleştirilere verilebilecek güzel bir cevap niteliği taşıyan sistem olmuştur.

Kağıt kalem odaklı çözümler, mobil işlemci içeren yeni nesil bilgisayarları kullanan çözümler ile popolaritesini yitirmeye başlamıştır. Buna ek olarak, yeni sistemlerin çok daha güvenilir ve kullanıcı dostu olması yeni sistemlere geçişi hızlandırmıştır. Mobil sistemler servis kalitesini ve teslimat hızlarını yükseltmiştir. Ayrıca yeni sistemlerin devreye alınmasından sonra, data kaybı hızlı bir şekilde düşmeye başlamıştır. Bu sistemin en büyük dezavantajı eski teknolojileri kullanan şirketlerin yeni teknolojilere geçip, performansları ve yapabilecekleri analiz ettikten sonra, gün geçtikçe artan yeni isterlerinin oluşmasıdır. Buna örnek olarak ise, sayısal imza ve sıcak satış işlemleri verilebilir.

Fiziksel olarak hafif oluşu, çok daha fazla kullanıcı dostu oluşu ve çok daha az enerji kullanımı ele alındığında, cep bilgisayar sistemleri laptop gibi hantal mobil cihazlar karşısında bir adım önde gözükmektedir. Bir başka açıdan bakarsak cep bilgisayarlarının da devreye girmesinden sonra mobil cihazlı sistem çözümleri hızlı bir artış göstermiştir. Yer ve zaman bağımsız çalışabilen bu tarz çözümler elektronik ticaret konusunda atılım yapmak isteyen firmalar için ilham kaynağı ve ayrıca elektronik ticaret üzerinde projeler geliştiren firmalar için ise potansiyel iş kaynağı oluşturmuştur.

‘**mShipment**’ (**Mobile Shipment System**) projesinin amacı, teslimat yapan firmalar için basit ve esnek bir çözümü yeni teknolojiler, dağıtık veri tabanı yapısı, şifreleme ve imza teknolojileri kullanarak oluşturmaktır.

‘**mshipment**’, teslimat va taşımacılık problemlerine farklı bakış açıları kullanarak, en güçlü ve güvenilir çözümü üretmeyi hedeflemiştir. Paketlerin teslimatı sırasında müşterilerin imzasını almak, güvenliği en yüksek çözümlerden biri olarak ele alınmıştır. Bundan dolayı sayısal imzanın görüntü işleme algoritmaları ile alınıp ileri düzey şifreleme metodları kullanılarak şifrelenmesi proje içerisindeki önemli modüllerden birini oluşturmaktadır.

Üzerinde durulan bir başka önemli nokta ise dağıtılmış veri tabanı modelidir. Mobil cihazlar ile teslimat verisinin anlık iletimini sağlayabiliriz. Teslimat bilgileri alındığı anda direk merkeze yollanabilir. Fakat, sistem offline çalışma veya bağlantı problemleri yaşanan zor şartlarda bile çalışabilecek şekilde tasarlanmalıdır. Bundan dolayı mobil cihazın teslimat verisini kendi üzerinde online olana kadar tutabilecek bir mekanizmaya ihtiyacı vardır. Bu da zor senaryolar karşısında sistemin çalışabilmesi için bir veri tabanına ihtiyacı olduğunu ortaya çıkarmaktadır. Bu database sayesinde bir yandan zor senaryoları aşarken, diğer yandan gerekli durumlarda bağlantısız çalışarak maliyetleri düşürebiliriz. Özetle bu sorundan çıkarabileceğimiz sonuç, dağıtılmış veritabanı yapısının sistem üzerinde beklendiğinden çok daha büyük önem taşıdığı olacaktır.

Sistemin geliştirilmesinde dikkate alınacak son önemli nokta ise veri iletimi güvenliği. Teslimat verisinin veri tabanları üzerinde geçişi sırasında gizlenmesi gereken bilgilerin (örnek olarak sayısal imza bilgisi) şifreli olarak yollanması gerekmektedir. Bu noktada projeye güvenli veri transferini sağlayabilecek, özel ileri teknoloji şifreleme algoritmaları kullanılması gerekmektedir.

## **2. YENİ YAPI “mShipment”**

‘**mShipment**’, kargo firmalarının isterlerinin maksimum’unu cep bilgisayarları kullanarak karşılayacak bir sistem ortaya çıkarmayı hedeflemektedir. Bu bölümde

'mshipment' sistemini detayları ile inceleyeceğiz. Fakat öncelikle, cep bilgisayarlarının (Pocket Digital Assistans, PDA) evrimini gözlemleyelim.

## **2.1. Cep Bilgisayarlarının Evrimi**

Cep bilgisayarları tam olarak nereden gelir? Cep bilgisayarlarının evrimi uzun ve ilginç gelişimlerden biridir. 1970 li yılların sonları, 1980 lı yılların başlarına gidersek, eğer herhangi biri bir arkadaşının telefonunu almak isterse, herhangi bir defter veya cep bilgisayarı teknolojisinin büyük büyük dedeleri olan Rolodex<sup>TM</sup> cihazlarını kullanmaları gerekmektedir. Rolodex<sup>TM</sup> cihazlarının ilerleyen yıllarda isime göre ve girilen herhangi bir harfe göre arama yapılabilen versiyonları çıktı, ve bu dönem bazı bilim adamları tarafında devrim niteliği taşımaktaydı. Bu ürünün en kötü yanı ise çok fazla kullanıcı dostu olmaması ve pahalı olmasıydı. Zamanla yeni cihazlar çıktı ve Rolodex<sup>TM</sup>'lerin yerini aldı. Bu bölümde eski ve yeni cihazları detayları ile inceleyeceğiz.

### **2.1.1. Eski Nesil Cep Bilgisayarları**

1970'li yılların sonları ve 1980'li yılların başlarına gidersek, genel olarak Filofax2 olarak tanımlanan cihazlar mobil asistanlar olarak piyasada ilgi görmeye başladılar. Daha sonra bu cihazlara gençler tarafından yaygın olarak tercih edilmesinden dolayı yuppies[2] ismi takıldı. Küçük bir hafızası olması ve A5 büyüklüğünde olması bile bu cihazların yaygınlaşmasını engelleyemedi. Fiyatları oldukça yüksekti ve koruma kılıfları gerçek deriden oluşmaktaydı. Gençlere yönelik olan, piyasada çok tutulan versiyonları birbirinden farklı renklere sahip dış kaplama içermektedir. Temel fonksiyonları, günlük, not defteri, adres defteri ve telefon rehberiydi. Bunun yanısıra ofislerde kullanılan versiyonlarında ek olarak hesap modülleri bulunmaktaydı. Bu cihazların bir diğer adı da ringbound(yüzüklü) dir. Bu ismi verilmesinin nedeni ise, birbirine bağlı sayfalardan oluşan modülleri olması. Örnek olarak günlük bilgilerini bir sayfada tutuyorsunuz, ama sayfanın sonu sistemin hafızası ile sınırlı. Arama modülleri

veya sayfalama özellikleri henüz bu sistemlerde yoktu. Bu eksiklikleri en büyük dezavantajları olmuştur.

Eksikliklerinin yanında bu sistemlerin en önemli özelliği yeni bir fonksiyon almak için veya yeni hafıza almak için ekstra para ödemenizdir. Bazı satın alınabilen fonksiyonlarına örnek olarak, toplantı notları alma, seyahat bilgileri ve haritalar. Bazı modülleri sayesinde print out(bilgisayar çıktısı) bile alınabildiği söylenmektedir.

Filofax olarak adlandırabileceğimiz ilk cihaz, 6 Haziran 1921'de, Norman & Hill kolektif şirketi tarafından hazırlanmış ve piyasaya sunulmamıştır. Birinci dünya savaşı sırasında bu cihazlar Amerikan Hükümeti tarafından kullanılmıştır.

Elektronik ajandalar (Electronic Organisers), Filofax cihazları piyasadaki pazar payına darbe vurmak için satışa sunulmasına karşın, yüksek fiyatları ve sınırlı fonksiyonlarıyla başarılı olamamışlardır. Fakat kullanıcı dostu ve hafif olmaları, büyük bir yenilgi almadan toparlanarak piyasada belirli bir yer almasına sebep olmuştur. Elektronik ajandaların, Filofax cihazları yenmeleri tam olarak beklenenden çok daha uzun sürmüştür. Ve pazarlamadaki ilk giren ilk kazandır metodolojisini bir kez daha haklı çıkarmıştır.

Teknolojinin ilerlemesi ile bu dönemin yapılabileceklerinin en üst düzeye çıkarıldığını, arama modüllerinin ve yeni fonksiyonların eklendiğini ve cihazların yaygınlaştığını gözlemleyebiliriz, hatta hatta bazı saat üreticisi firmalar mamüllerinin içine bu cihazların belirli özelliklerini eklemek için büyük araştırma – geliştirme çalışmaları yapmışlardır. Bu dönemdeki en kritik nokta ve yeni cihazları eski cihazlardan ayıran evrim noktası, bilgilerin senkronizasyonu, bir cihazdan diğerine bilgi aktarma ve desktop bilgisayarlardan bilgi aktarımı olmuştur.

### **2.1.2. Yeni Nesil Cep Bilgisayarları**

Kişisel Bilgi Yöneticisi (Personal Information Manager, PIM), bu cihazlar bu dönemin ilk ürünleridir ve modern cep bilgisayarlarının en yakın akrabalarıdır. Cep bilgisayarlarının şu anda üzerinde taşıdıkları özelliklerin büyük bir bölümü bu cihazlardan kalan mirastır. Bu özellikler aşağıda sıralanmıştır;

- Masaüstü bilgisayar bağlantısı,
- Standart fonksiyonlar. Örnek olarak, takvim, adres kayıt defteri v.b. ,
- Kullanıcı dostluğu,
- Stylus kalem,
- Bir masaüstü bilgisayar bağlantısı ile üzerindeki programların güncellenmesi.

PIM cihazlarının bu özelliklerini tümü bugüne taşınmıştır. En fazla önemli olanı ise masaüstü bilgisayar ile bağlantı kurup, yedeklemeler alınabiliyor olması, emaillerin aktarılabilir olması ve takvim üzerine kayıt edilmiş alarmların cihaza senkronize edilebilmesi. Bunun yanında bir son dönem PIM cihazları birbirleri arasında da bağlantı kurabilmektedirler. Bu bağlantı, kablo veya kızıl ötesi(IR) iletişim ile sağlanmaktadır. Bu özellikler eski tip cihaz üreticilerinin piyasada büyük bir yara almasına neden olmuştur.

PC bağlantısı kullanıcılara, cep bilgisayarlarına yeni programlar yükleyebilme eski programları güncelleyebilme yetenekleri kazandırmıştır. Kullanıcılar, PIM cihazları sayesinde, cep bilgisayarları üzerinde oyun oynama ve kitap okuma alışkanlıkları kazanmışlardır. Tabi bu özellik, oyun üreticisi ve kitapevlerinin iştahını kabartmıştır.

Önde gelen PIM üretici firmaları, Sharp ve Casio gibi firmaları kendilerine örnek almış bir kısmı bu konuda başarılı olmuş gelişmiş, bir kısmı gene bilgisayar sektörü içinde başka alanlara dallanmış, bir kısmı ise büyük elektronik üreticisi firmalar çatısı altında toplanmıştır. Psion, önemli PIM üreticisi firmalardan biridir. Psion'un ilk ürettiği digital asistan, bir mutant bilimsel hesap makinasıdır ve hemen hemen bir tuğla büyüklüğüne sahiptir. ABC klavyesine sahip ve iki satırlık görüntü alanı vardır. Modern cihazlar ile



karşılaştığımızda sınırlı fonksiyonları ve hantal yapısına rağmen, günümüz teknolojisine kadar taşıdığı özellikleri bulunmaktadır.

PIM cihazların liderliği 2002 yılına kadar sürmüştür. 2002 yılında PDA'lerin (Dijital Cep Asistanları, Pocket Digital Asistant) piyasada yaygınlaşması ile PIM cihazlar bir anda piyasada büyük yenilgi almıştır. PDA'lerin Palm Pilot modelini benzerliği, stylus kalem kullanmaları, touch sensitive ekrana sahip olmaları ve Apple Newton6 modelinden alıntı olduğu dedikodusu, piyasada bir anda popülerleşmesinin en büyük etkileri olduğu söylenmektedir. El yazısı tanıma özelliği stylus kalem kullanımına, kullanıcı dostu özellik kazandırmıştır. El yazısı tanımadaki oran %85'lere varınca, kullanım popülaritesi bir üst adıma taşınmıştır.

En yaygın olarak bilinen PDA özellikleri:

- Renkli ekran,
- Kullanıcı dostu olması ve öğrenilmesi basit olması,
- Stylus kalem içermesi,
- Standard programları: takvim, adres rehberi v.b.,
- Ses,
- Diğer cihazlara göre yüksek hafızası,
- Kablosuz bağlantı özellikleri (IR, Bluetooth, WiFi),
- Bilgisayar bağlantısı,
- Bilgisayar bağlantısı ile yüklenebilecek program çeşitliliği,
- Upgrade (Güncel versiyon yükleme) port'u,
- İnternet bağlantısı özelliği.

Bazı PDA'lerin cep telefonu özelliği de bulunmaktadır, bu modeller SmartPhone olarak adlandırılmaktadır. Bu cihazların ilk sürümleri Nokia 9110 ile piyasaya çıkmıştır. Geniş ekranı ve stylus kalem ile piyasaya başarılı bir giriş yapmış fakat. Bir cep telefonuna göre fiziksel olarak hantal oluşu popülaritesini beklenen seviyelere getirememiştir.

Şu anda en fazla kullanılan ve bilinen PDA modelleri veya benzerleri,

- XDA,
- Blackberry,
- Sony Ericsson P800 ve P900,
- Nokia 6600,
- Motorola MPx200,
- Siemens SX1,
- Motorola A835,
- NEC e808.

### **2.1.2. Pazar Liderleri**

9 Ocak 2004 ten itibaren, Smartphone' lar piyasada PDA'lerin önünde lider durumdalar.

PDA'lerin yaygın olarak kullandığı işletim sistemleri;

- Palm OS
- Symbian
- Windows CE

Palm OS, Palm cihazlar için üretilmiş, kullanıcı dostu işletim sistemlerinden biridir. Word, Excel ve Powerpoint dosyalarını okuyabilme ve açabilme özelliklerine sahiptir.

Symbian işletim sisteminin orijinali Psion tarafından yazılmış daha sonraları açık kod olarak kullanıcı gelişimine açık olarak piyasaya sürülmüştür. Microsoft Windows'a benzer özellikleri vardır. Örnek olarak, kullanıcı klasörler yaratabilir, dosyaları taşıyabilir, kopyalayabilir veya silebilir. Tam olarak olmasada Microsoft Windows ile uyumlu çalıştığını söyleyebiliriz. Hemen hemen tüm office dökümanlarını görüntüleyebilir. Sony Ericsson başta olmak üzere bir çok firma tarafından hala kullanılmaktadır.

Windows CE ise bu piyasaya henüz yeni girmiş bir işletim sistemidir. Toshiba ve Hewlett-Packard başta olmak üzere bir çok firma tarafından özellikle sanayi tipi el bilgisayarlarında yaygın olarak tercih edilmektedir.

Bu noktada PIM'lerin de kendilerine ait bir işletim sistemine sahip olduğunu belirtmemiz gerekir. Sharp firması tarafından geliştirilen Synergy işletim sistemi ve Casio tarafından geliştirilen PVOS işletim sistemi, PIM'lerin en çok tercih edilen işletim sistemleridir.

Bu yeni nesil cihazların birbirleri ile iletişim ağı vardır. Büyük bir bölümü benzer teknolojileri kullanmaktadırlar. Bu teknolojiler;

- Mavi Diş (Bluetooth),
- Radyo Sinyalleri,
- Kızıl Ötesi (IR).

Örnek olarak, herhangi bir veri bir PDA cihazından markası veya türü ne olursa olsun bir başka cihaza kolayca yollanabilir. Bu ortak standart'a **mavi ceket (blue-jacking)** adı verilmektedir. Bu teknoloji sayesinde maaliyetsiz olarak mavi diş teknolojisinin desteklediği mesafedeki her hangi bir kullanıcıya ücretsiz mesaj veya data göndermek mümkün olmuştur.

Bahsettiğimiz cihazların yanı sıra kategorize edilmesi güç olan bir çok mobil cihazda mevcuttur. Örnek olarak, Psion 3C cihazı PIM ve PDA kategorisi arasında kalmış. PIM cihazlara göre yüksek performanslı ve PDA'lere göre sınırlı fonksiyona sahip bir üründür.

Bir başka örnek Psion 5MX'dir. Psion 5MX'in PDA in geliştirilmiş şekli mi? Yoksa bir laptop bilgisayar akrabası mı? karar vermek oldukça zordur. Bu cihaz bir laptop bilgisayarın veya bir desktop bilgisayarın yaptığı bir çok işi yapabilmektedir. Ayrıca boyut olarakta bir cep bilgisayarı kadar küçüktür.

Kategorize etmenin ne kadar zor olacağı konusunda bir başka örnek olarak Sharp firmasının çıkardığı yeni nesil cihazlardan bahsedebiliriz. Sharp ZR-5700E'in kesinlikle bir PIM olduğunu söyleyebilmemize karşın, bu cihazın fonksiyonalitesi ve esnekliği herhangi bir PIM cihaza oranla çok yüksektir. Bir başka örnek ise Sharp ZQ-5650M; bu ürün ise bir elektronik orginaser'a göre oldukça performanslı ve geniş bellek'e sahiptir.

PDA cihazların her geçen gün biraz daha yaygınlaşması ile birlikte bazı problemler de ön plana çıkmaya başladı. Bu problemler içinde en yaygın olanlardan biri, PDA cihazlarına text bilginin girişinin oldukça yavaş olması. PDA'ler ile karşılaştırıldığında, bir çok insan desktop bilgisayarın klavyesi ile çok hızlı yazabilmektedir.

İleriye dönük PDA teknolojileri içinde speech-to-text (konuşmanın text'e dönüşümü) de bulunmaktadır ve bu konuda araştırmalar sürmektedir. Bu teknoloji desktop bilgisayarlarda olmasına karşın, cep bilgisayarı işlemcisi ve hafızası ile PDA'ler üzerinde uygulanmasını oldukça sınırlı kılmaktadır. Bunun yanı sıra, bu teknolojiyi kullanmak için PDA'lere bir de kullanıcının ağzına belirli bir uzaklıkta konuşlandırılacak bir mikrofon ekleyeceğimizi düşünürsek ve ayrıca bu mikrofonun ofis veya sokak gibi kalabalık ve gürültülü ortamlarda kayıt yapamaya çalışacağını düşünürsek; bu teknolojinin yakın süreçte hayata geçmesinin pek mümkün olmadığını görebiliriz. Bu konuda bir başka öneri ise, bir kamera yardımıyla kullanıcının dudaklarını okuyarak görüntüyü text'e çevirme teknolojisi. Fakat bu da cihazın işlemcisi ve görüntü işleme algoritmalarının kompleks yapısı gözönüne alındığında yakın gelecekte gözükmemektedir.

Dezavatajlarına rağmen, yakın gelecekte çok daha fazla PDA görebileceğimiz, bilim adamları tarafından ön görülmektedir. PDA'ler hayatımızdaki bir çok cihazın yerini almaya başlayacak ve yaşantımıza daha çok entegre olacaklardır. Örnek olarak, cep telefonlarının veya televizyon kumandalarının yerini kolaylıkla almaları gibi.

## **2.2. Sistem Mimarisi**

'mshipment' da, PDA cihazları müşteri teslimat süreci içerisinde laptop veya kalem-kağıt çözümleri yerine tercih edilmiştir. Veri bütünlüğünün sağlanması için, tüm alınan veri internet veya local network senkronizasyonları ile merkez veritabanına aktarılır. PDA üzerindeki veritabanı ve merkezi veritabanı arasındaki gerekli senkronizasyon web servisleri veya sistem servisleri tarafından sağlanır. PDA kullanıcılarının tümü sisteme giriş yaptıktan sonra, topladıkları veriyi sisteme aktarabilirler, teslimat bilgilerini kontrol edebilir, raporlar alabilir veya yeni teslimat listeleri oluşturabilirler. Oluşturdukları listedeki belirledikleri route a göre, müşterilerine tek tek ulaşarak gerekli teslimatları yaparlar.

Teslimat sırasında, teslimat bilgileri dağıtılmış veri tabanı mimarisi ile merkezi veri ambarına aktarılır. Bu olay veri bütünlüğünü sağlar. Sistemin PDA ve merkezdeki temel yapısı ve menüleri aşağıda gösterilmiştir.

#### **PDA Tarafı**

- Kullanıcı Girişi / Çıkışı
- Bilgi Aktarım / Güncelleme
- Teslimat Listesi
- Müşteri Islak İmzası Alma

#### **Merkezi Sistem Tarafı**

- Teslimat Bilgileri Ekleme / Değiştirme
- Müşteri Bilgileri Ekleme / Değiştirme
- Teslimat Bilgileri Ekleme / Değiştirme / Onaylama
- Islak İmza
- Kullanıcı Giriş / Çıkışı
- Bilgi Aktarımı
- Kullanıcı Profili Değiştirme
- Teslimat ve Müşteri Raporları

- GPS Tabanlı Harita

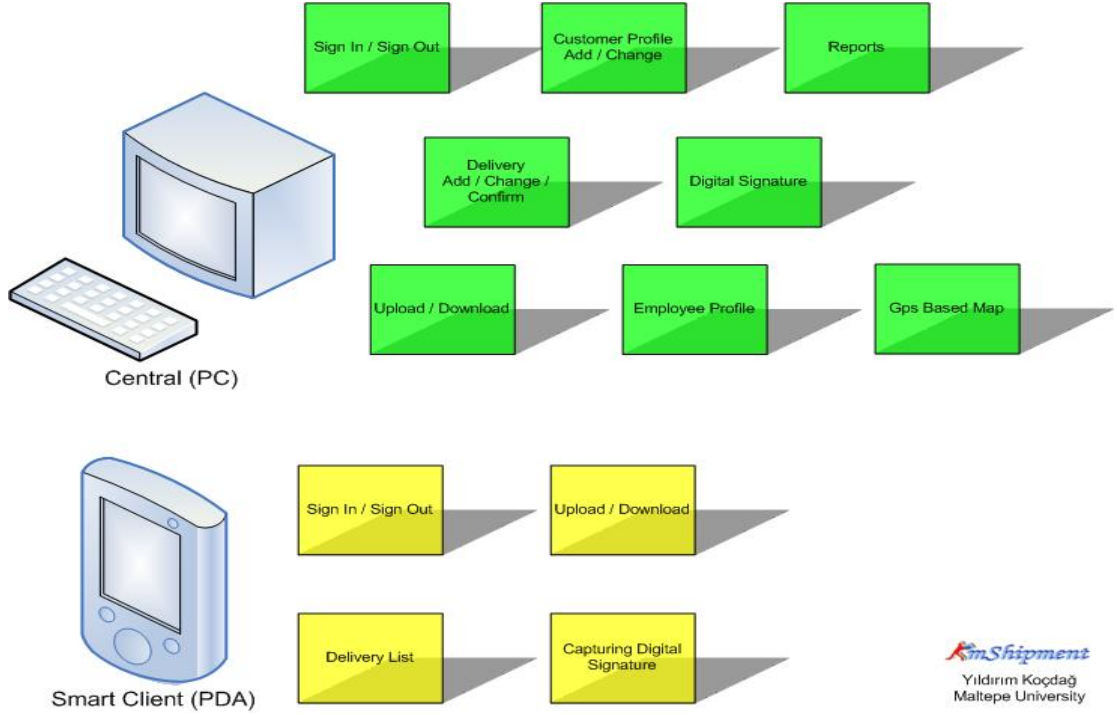
### 2.3. Geliştirilmiş Varlık İlişki Diyagramı

Projenin temel yapısı geliştirilmiş varlık ilişki diyagramı (Şekil 1) ile açıklanmıştır. Veri tabanı PDA'ler ve merkezi sistem arasında dağıtık olarak tutulmuştur. Aşağıda bahsedilen PDA üzerinde gerçekleştirilen işlemler sonucunda oluşturulan bilgi PDA üzerindeki veri tabanında tutulur. Bunun yanısıra merkezi sistemde gerçekleştirilen işlemler sonucu oluşturulan bilgi sistemde merkezi veri tabanında tutulur. Şekil 1'i incelersek, merkezi sistemdeki bazı fonksiyonların, PDA üzerindeki fonksiyonlarının bir geliştirilmiş versiyonu (superset'i) olduğunu görebiliriz. Yeni kullanıcılar, müşteriler eklemek ve değiştirmek ve teslimat detaylarını incelemek PDA'e göre merkezi sistemdeki ekstra fonksiyonlardır.

GPS tabanlı harita sistemi proje içerisinde sadece simülasyon olarak kullanılmıştır. Buradaki temel amaç, teslimat yapacak aracın harita üzerindeki takibidir. Böylelikle sistem tarafından manuel veya otomatik olarak, müşteriye teslim edilecek paketi hakkında detaylı bilgi verilebilir.

Sisteme özel bir raporlama aracı eklenmiştir, bu araç sayesinde teslimat hakkındaki gerekli istatistik bilgileri ve raporlar kolaylıkla alınabilecektir. Dilendiğinde bu raporlar ve istatistik bilgileri müşteriyle paylaşılıp, firma hakkında bilgi verecek ve çalışmaların özeti niteliğini taşıyacaktır.

Projenin bir başka kilit noktası dijital imza noktası, müşteriye teslimat yapıldıktan sonra müşteri tarafından atılacak imzanın sistemde tutulması, ileride karşılaşılabilecek problemlerin önüne geçerek güvenli bir teslimat süreci oluşturacaktır.



Şekil 1 – Geliştirilmiş Varlık İlişki Diyagramı

## 2.4. Kullanıcı Senaryoları

Projedeki iş adımları tek tek incelendiğinde, bir iş adımının tamamlanmadan veya onaylanmadan diğerine geçmenin mümkün olmadığı açıkça görülmektedir. Örnek olarak , sipariş teslim etmeden müşteri imzasını alamayız. Projedeki tüm adımlar çalışanlar tarafından tam olarak anlaşılmalı ve uygulanmalıdır. Bundan dolayı bu bölümün çok iyi anlaşılması gerekmektedir.

‘**mShipment**’ projesinin fonksiyonlarının tek tek irdelenmesi için, kullanıcı senaryolarını incelemeye başlayalım. İki farklı senaryo; senaryo 1 ve senaryo 2 aşağıda incelenmiştir.

### Senaryo 1 :

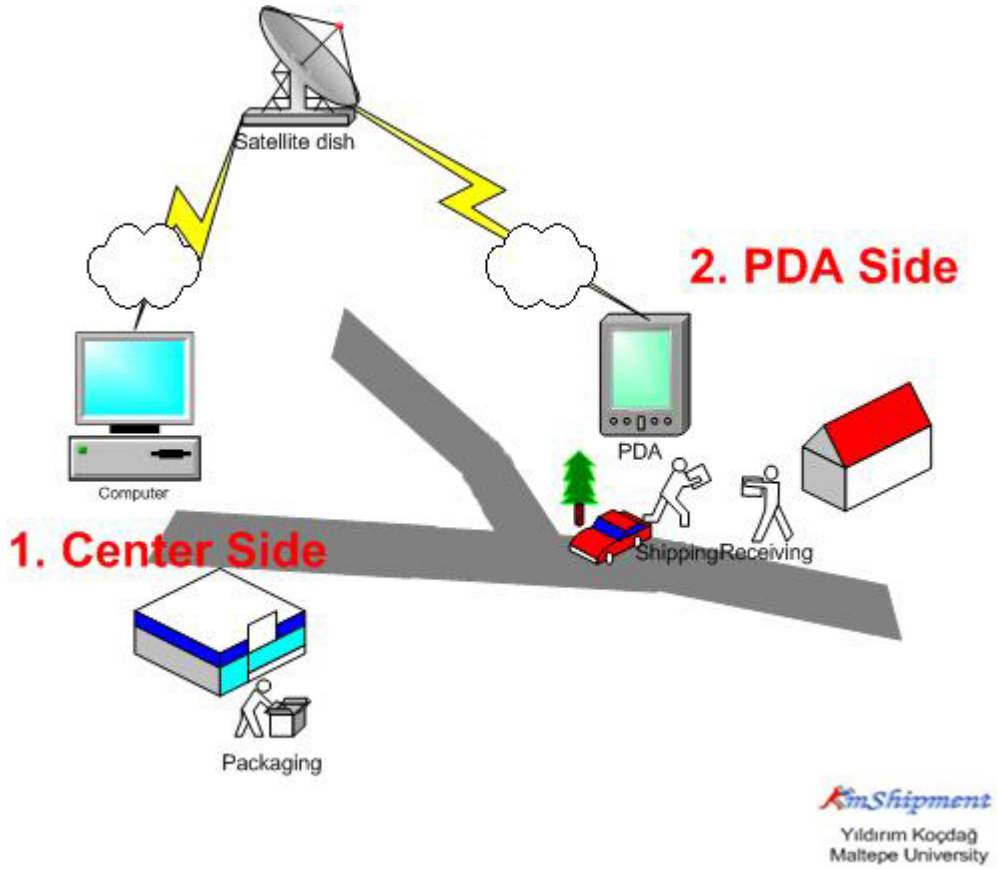
Sisteme ilk bakışımızda iki farklı katman dikkatimizi çekmektedir bunlar;

- PDA Tarafı,

- Merkezi Sistem Tarafı.

İşlemlerin başlangıç noktası merkezi sistemdir. Bu noktada iş adımları sırasıyla, teslimat bilgisini incelenmesi, route'un çalışan personel'e atanması ve hazırlanan verinin PDA'a internet veya herhangi bir local bağlantı ile aktarılması. Bu aktarım süreci Şekil 2'de Uydu, Bilgisayar ve Cep Bilgisayarı ile sembolize edilmiştir. Aktarım süreci esnasında, fiziksel teslimat paketleri ve materyaller teslimat aracına yüklenir.

PDA cihazının üzerindeki müşteri isimleri, adresleri ve teslimat route'u gibi bilgilerin güncellenmesinden sonra teslimat personeli paket teslimatı için hazır hale gelir. Paket teslimatı sırasında, teslim yapan personel müşteriden teslim yapıldığına dair PDA üzerine bir imza alır. İmza alındıktan sonra, teslimat verisi merkezi sisteme direk yollanabilir veya bir sonraki alternatif senkronizasyona kadar PDA üzerinde saklanabilir.

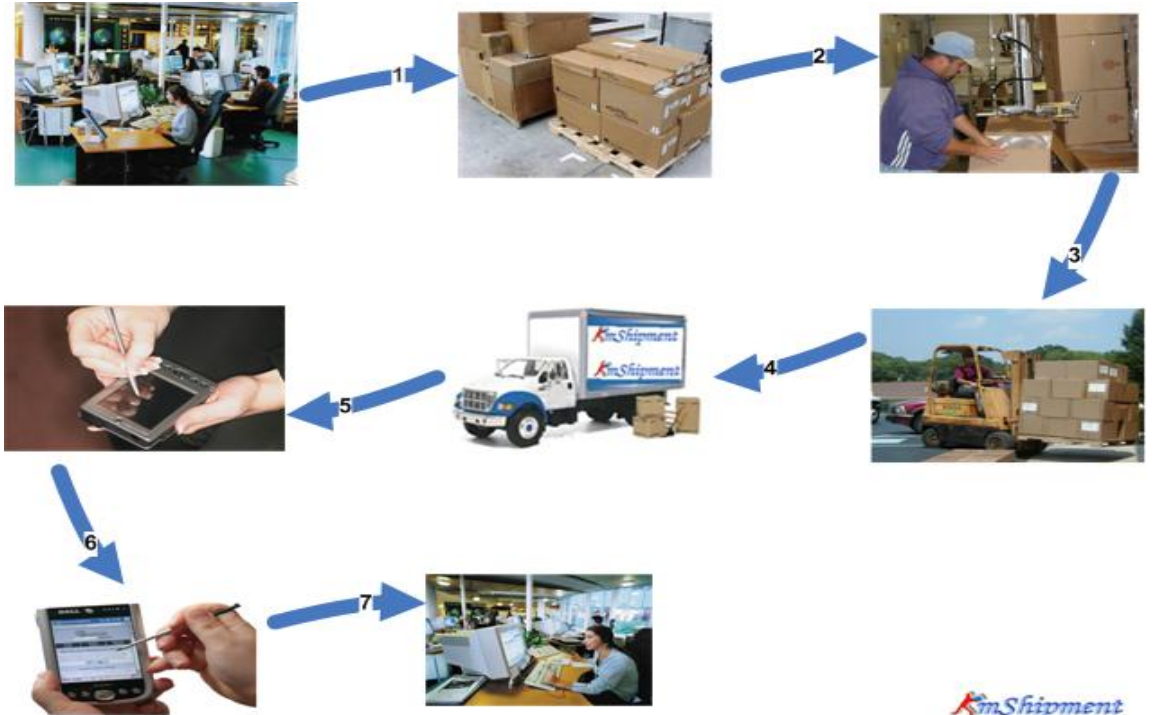


Şekil 2 – Senaryo 1



## Senaryo 2 :

Yönetim ofisimizin Şekil 3 üzerinde sol taraftaki ilk resim şeklinde sembolize edildiğini düşünelim ve oklar ile tek tek yapılan iş adımlarını inceleyelim. Operasyon isteği yönetim ofisinden, depoya yollanır(1 numaralı ok ile gösterilmiştir). Depo çalışanları teslimat paketlerini hazırlarlar ve bu paketleri 2, 3 ve 4 numaralı oklarda gösterildiği şekilde kargo aracına yüklerler. Kargo müşteriye ulaştırıldığı anda, müşteriden teslimat bilgisi olarak imza alırsız (5 numaralı ok ile gösterilmiştir). 6 numaralı ok ise müşteriden alınan imzanın ve teslimat bilgilerinin kargo elemanı tarafından kontrol edildiği iş adımlarını göstermektedir. Son adımda kargo elemanı tarafında kontrol edilen bilgiler yönetim ofisine aktarılır ve tamamlanmış teslimat şeklinde yönetim ofisi tarafından durum bilgisi güncellenir.



**KmShipment**  
Yıldırım Koçdağ  
Maltepe University

Şekil 3 – Senaryo 2

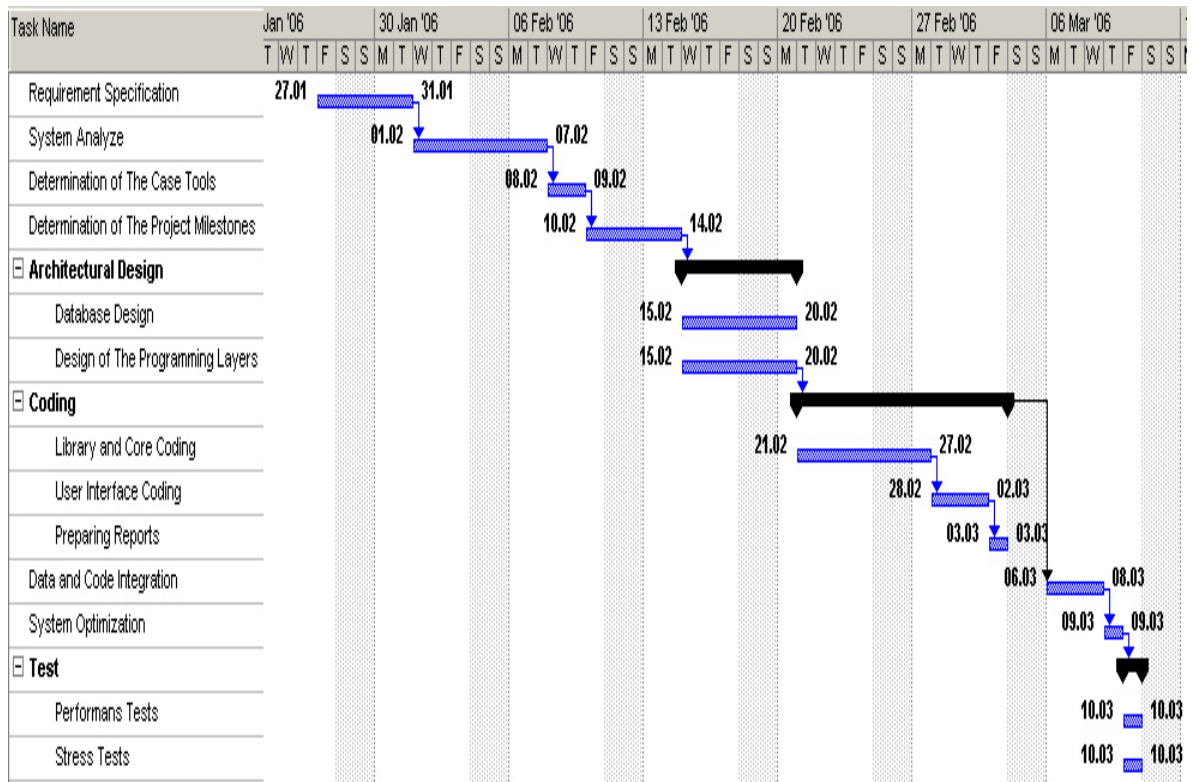
## 2.4. Yeni Sistemin Gant Diyagramı

Projenin geliştirilmesine 27 Ocak 2006 tarihinde başlanmıştır. İsterlerin belirlenmesi ve tanımlanması 10 Mart 2006 tarihinde bitmiştir. Tüm gelişim süreci yaklaşık 2 ay da tamamlanmıştır. Bu süreç boyunca başlıca 9 farklı alanda çalışmalar yapılmıştır. Bu alanlar;

- İsterlerin belirlenmesi,
- Sistem analizi,
- Kullanılacak araçların belirlenmesi,
- Proje kilometre taşlarının belirlenmesi,
- Mimarı tasarım,
- Kodlama,
- Veri ve kod bütünlüğünün oluşturulması,
- Sistem optimizasyonu,
- Test.

Bu adımları sırasıyla kısa kısa inceleyecek olursak; İlk bölümde yönetim, kıdemli çalışanlar ve müşteriler dinlenerek bir sistem isterleri dökümantasyonu hazırlandı. İkinci bölümde bu isterler değerlendirilerek, yeni sistemin nasıl olması hakkında bir analiz yapıldı, problemler ve çözümler saptandı. Problemler ve çözümlerini elde ederken hangi araçlar kullanılacağı belirlendi. Bir sonraki adımda projenin kritik noktaları saptandı ve proje kilometre taşları çıkarıldı. Mimarı yapının tasarlanması aşamasında mobil cihaz ve merkezi sistemler bir arada değerlendirildi, bundan dolayı merkezi veri tabanı ve mobil cihaz veri tabanı yapıları aynı anda tasarlandı. Mimarı tasarımdan sonra kodlama aşamasına geçildi, kodlama sürecinde önce projenin çekirdek kodu yazılarak, kod kütüphaneleri oluşturuldu daha sonra bu kod kütüphaneleri kullanılarak kullanıcı arayüzleri hazırlandı. Merkezi sistem ve mobil cihazlar üzerinde kullanılan veri tabanları ve kullanıcı arayüzleri arasında veri ve kod bütünlüğü oluşturuldu. Son adımda öncelikle sistem optimizasyonu ardından sistem testleri yapıldı. Sistem testleri sırasında öncelikle kara kutu ve beyaz kutu testleri ardından stress tesleri yapıldı.

Projeje ilk bakışta merkezi sistemin ve mobil cihazın üzerinde geliştirilecek sistemlerin birbirinden tamamen farklı olduğunu düşünebiliriz. Fakat geliştirme süresi boyunca bu iki sistemin tamamen birbirine bağlı ve paralel ilerlemesi gerektiği açıkça görüldü. Bir sistemdeki gelişim diğerini etkilediği için paralel bir geliştirme süreci izlendi. Bundan dolayı ‘mShipment’ projesindeki çalışma kesinlikle sistematik olmak zorundaydı. Ayrıca tüm adımlar olabildiğince açık ve basitçe tanımlanmalıydı. Bu geliştirme sürecinin iş adımları Şekil 4’teki Gant diyagramında gün ve gün şeklide görülmektedir.



Şekil 4 – ‘mShipment’ Projesinin Gant Diyagramı

### 3. SİSTEM FİZİBİLİTESİNİN HAZIRLANMASI

Sistem fizibilitesi 4 farklı alanda incelenebilir;

- Teknik Fizibilite,

- Ekonomik Fizibilite,
- Sistem Güvenliđi,
- Sistem Uygulanabilirliđi.

### **3.1. Teknik Fizibilite**

‘**mShipment**’ sistemi yukarıda da bahsedildiđi üzere mobil cihaz, internet teknolojisi ve desktop cihaz üzerine kurulu bir sistemdir. Bundan dolayı sistemin minimum isterleri bir PDA cihazı ve internet bađlantısına sahip güçlü bir desktop bilgisayar olacaktır. Teknik fizibilite 4 farklı ana başlık altında incelenebilir. Bunlar ;

- Masaüstü bilgisayar ve cep bilgisayarı üzerinde kullanılan yazılımlar,
- İnternet bađlantısı,
- Masaüstü bilgisayar ve cep bilgisayarı minumum donanım isterleri,
- Çalışan kaynađı.

#### **3.1.1. Masaüstü Bilgisayar ve Cep Bilgisayarı Üzerinde Kullanılan Yazılımlar**

Projeyi geliştirirken kullanabileceđimiz iki farklı temel araç var, bunlar J2EE and .NET. Her ikiside platform bađımsız çalışabilmekte. J2EE ile PALM ları üzerine program geliştirebilirken, .NET ile PDA ler üzerine program geliştirebiliriz. PALM’ların kısıtlı hafızası ve yavaş işlemcileri dikkate alındığında proje geliştirme aracı olarak en uygununun .NET olduđu açıktır.

Masaüstü bilgisayar ve cep bilgisayarında kullanılan programlar iki farklı alanda incelenebilir, geliştirme anı ve çalışma anı. Bu iki farklı durum Tablo 1 ve Tablo 2 de gösterilmiştir.

<b>Masaüstü Bilgisayar</b>	<b>Cep Bilgisayarı</b>
Ms Visual Studio .NET Final Version	Pocket Pc 2002
Ms Smart Device Extensions	.NET Compact Framework 2.1.
Ms SQL Server 2000	Ms SQL Server CE
Ms Windows XP Professional	
Ms Frontpage 2002	
Ms Office 2003	
Ms Visio 2003	
Ms Project 2003	

**Tablo 1 – Geliştirme Anında Masaüstü ve Cep Bilgisayarında Kullanılan Programlar**

<b>Masaüstü Bilgisayar</b>	<b>Cep Bilgisayarı</b>
Ms SQL Server 2000	Ms SQL Server CE
Ms Windows XP Professional	Pocket PC 2002
Framework 1.1	.NET Compact Framework 2.1.
Virtual Network Driver	

**Tablo 2 – Çalışma Anında Masaüstü ve Cep Bilgisayarında Kullanılan Programlar**

### 3.1.2. İnternet Bağlantısı

İnternet bağlantısı son yıllarda ülkemizin her köşesinde oldukça gelişmiştir. Bağlantı problemleri özel firmalarında bu markete girmesi ile birlikte düşmüştür ve kalite artmıştır. Bağlantı hızları basit bir sistem için 128kbps ve 2Mbps arasında

değişmektedir. Hazırlayacağımız uygulama için gerekli duyulan bağlantı hızı, kullanıcı sayısı arttıkça doğru orantılı olarak aratacaktır. Projenin başında yüksek maliyetli bir yapıya ihtiyaç yoktur. Kullanıcı sayısı arttıkça, internet bağlantı hızı da arttırılabilir.

### 3.1.3. Masaüstü ve Cep Bilgisayarının Minimum Donanım İsterleri

Minumum donanım isterleri Tablo 3 ve Tablo 4 de gösterilmiştir.

<b>Masaüstü Bilgisayar</b>	<b>Cep Bilgisayarı</b>
Pentium 4 – 1000 Mhz CPU	420 Mx Processor
512 MB RAM (.NET Framework İsteri)	32 MB RAM
Graphich Card 32 MB	32 MB ROM
ISDN(512Kbps) or ADSL(512Kbps) Modem	Barcode okuyucu cihaz(opsiyonel)
40 GB Hard Disk	PDA için desktop senkronizasyon kiti
UPS (400 VA)	

**Tablo 3 – Çalışma Anındaki Minimum Donanım İsterler**

<b>Masaüstü Bilgisayar</b>	<b>Cep Bilgisayarı</b>
Pentium 4 – 1700 Mhz CPU	420 Mx Processor
512 MB RAM (.NET Framework and Smart Device Extension İsterleri)	32 MB RAM
Grafik Kartı 32 MB	32 MB ROM
40 GB Hard Disk	PDA için desktop senkronizasyon kiti / Bluetooth / Wifi
Network bağlantı kabloları / Bluetooth / Wifi	

**Tablo 4 – Geliştirilme Anındaki Minimum İsterler**

### 3.1.4. Çalışan Kaynağı

Geliştirme süreci boyunca çalışanlar dört farklı alanda çalışmışlardır. Çalışanlar ve geliştirme süreç zamanları Tablo 4 de gösterilmiştir.

<b>Çalışanlar</b>	<b>Geliştirme Süreleri (Gün/Çalışan)</b>
System Analisti * 2	7
Veritabanı sorumlusu * 2	10
PDA Programcısı * 2	20
Windows Programcısı * 3	20
Test Yapan * 2	10
<b>Toplam Geliştirme Süreci( Gün/Çalışan)</b>	<b>67</b>

**Tablo 5 – Çalışanlar ve Geliştirme Süreleri**

### 3.2. Ekonomik Fizibilite

Ekonomik fizibilite 4 farklı alanda incelenebilir,

- Yazılım Tutarı,
- İnternet Bağlantısı Tutarı,
- Donanım Tutarı,
- Çalışan Maaşları Tutarı.

#### 3.2.1. Yazılım Tutarı

‘mShipment’ projesinin bir mobil cihaz uygulaması olduğundan dolayı, mobil cihaz üzerinde proje geliştirebileceğimiz bir framework’e ihtiyacımız vardır.

Bundan dolayı, PDA üzerinde proje geliştirebilmek için “**smart device extensions**” adı verilen bir .NET eklentisi kullanılmalıdır. “**Compact framework for run time libraries**” sayesinde projenin PDA üzerinde çalışabileceği tüm kütüphaneleri “Pocket PC” işletim sistemine sahip mobil cihaza aktarır.

Tablo 6’da merkezi sistem ve PDA üzerinde kurulması gerekli programlar ve fiyatları belirtilmiştir.

<b>Uygulama</b>	<b>Fiyat</b>
Ms Windows XP Pro	149\$
Ms SQL Server 2000	1100\$
Pocket PC 2002	Bundled with PDA
.Net Framework	Free
Ms Visual Basic .NET Ent.	1800\$
.Net Compact Framework	Free
Ms Office 2003	150\$
Ms Visio 2003	350\$
Ms Project 2003	350\$
Ms Frontpage 2003	400\$
<b>Toplam Tutar</b>	<b>4299\$ + KDV</b>

**Tablo 6 – Programların Tutarı**

### **3.2.2. İnternet Bağlantısı Tutarı**

İnternet bağlantısının yaklaşık tutarı aylık ve kurulum ücretleri olarak Tablo 7’de gösterilmiştir. Modem ücreti kurulum ücretlerine dahildir.



<b>Bağlantı Tipi</b>	<b>Hizmet</b>	<b>Tutar</b>
ADSL(2048/512)	Kurulum	100\$
	Aylık	20\$
ISDN(Primary Access, 512 - )	Kurulum	150\$
	Aylık	30\$

**Tablo 7 – Internet Bağlantısı Tutarı**

### 3.2.3. Donanım Tutarı

Merkezi sistem ve mobil cihaz için gerekli minimum tutarı Tablo 8’de gösterilmiştir.

<b>Donanım</b>	<b>Tutar</b>
PDA	400\$ - 1500\$
Desktop Synch Kit	120\$
Desktop PC	600\$ – 1300\$
UPS ( 400 VA)	230\$
Barcode Okuyucu Cihaz	60\$
<b>Toplam Tutar</b>	<b>1410\$ - 3210\$</b>

**Tablo 8 – Donanım Tutarı**

### 3.2.4. Proje Çalışanlarının Maaliyetleri

Projede çalışan sistem analisti, windows programcısı, mobil cihaz programcısı ve test yapan kişilerin maaliyetleri ve toplam tutarı Tablo 9’da gösterilmiştir.

<b>Çalışanlar</b>	<b>Maaliyet</b>
Sistem Analisti	1020\$
Windows Programcısı	2400\$
Mobil Cihaz Programcısı	3500\$
Test Yapan Çalışan	350\$
<b>Toplam Tutar</b>	<b>7270\$</b>

**Tablo 9 – Proje Çalışanlarının Maaliyetleri**

**Toplam proje tutarı yaklaşık 12,000\$ ila 15,000\$ arası gelişmelere göre değişebilir.**

### **3.3. Sistem Güvenliği**

Masaüstü bilgisayar ile cep bilgisayarı arasında güvenli bir veri iletişimi için, XML (Extensible Markup Language - Genişletilebilir İşaretleme Dili) ve AES (Advance Encryption System) iyi bir kombinasyon oluşturabilir. Cep bilgisayarından merkezi sisteme teslimat bilgisi aktarımı sırasında veya merkezi sistemden cep bilgisayarına veri transferi sırasında, başarılı ve güvenli bir operasyon için XML standartını ve AES ileri şifreleme sistemini kullanabiliriz.

Bu bölümde, XML standardı ve AES ileri şifreleme teknolojisi dışında güvenli teslimatın son noktası olan müşteriden sayısal imza alma yönetimi üzerinde de durulacaktır.

#### **3.3.1. XML Standartı**

XML (Extensible Markup Language - Genişletilebilir İşaretleme Dili) yapılandırılmış belge ve verilerin tanımlanması için hazırlanmış uluslararası bir formatıdır. Herhangi bir verinin biçimlenmesi, tanımlanması ve veri aktarımları için kullanılır. XML dili bir çok cihaz tarafından okunabilen ve anlaşılabilir bir dil olduğu için cihazlar arasında iletişimi

sağlamak amacıyla da çok sık kullanılır. Yollanılan verinin doğruluğuda gene kendi içinde yer alan doğrulama yapısı yardımıyla sağlandığında veri aktarımları sırasında eksik bilgi veya bozuk bilgiye rastlanması çok enderdir[3].

### **3.3.1.1. XML Standartının Güçlü ve Gücsüz Noktaları**

XML standartının başlıca avantajları ;

- İnsan ve makina tarafından kolaylıkla okunabilmesi,
- Unicode desteği olduğundan dolayı, bir çok iletişim dilini içinde barındırabiliyor olması(Çince,Arapça v.s.),
- Yetenekleri sayesinde bilgisayar bilminde kullanılan veri yapılarına, kayıtlara, listelere ve ağaçlara destek verebiliyor olması,
- Dökümanite yapısı yardımı ile, verileri ve alan isimlerini kolaylıkla anlaşılır kılabilmesi,
- Zorunlu alanları ve söz dizim sırası ile üzerine kurulu algoritmaların basit, hızlı ve tutarlı çalışabilmesini sağlamaktadır,
- Bu format sayesinde yüksek verilerin veri tabanı olmaksızın offline ve online tutulabilmesi imkanı sağlanmıştır,
- Uluslararası standartlara dayanarak veri bütünlüğü sağlanır,
- Hiyerarşik yapısı sayesinde bir çok dökümana kolaylıkla destek verebilir(Word, Excel),
- Lisans sıkıntısı yoktur, herkes tarafından kolaylık kullanılabilir,
- Platform bağımsız çalışmasından dolayı, her ortamda her cihazda kolaylıkla kullanılabilir ve sistemler tarafından okunabilir,
- 1986 yılından beri kullanılan SGML teknolojisini baz aldığı için SGML teknolojisinin edindiği tecrübenin tümüne sahiptir.

XML'in genel güçsüz noktaları ;

- Söz diziminden kaynaklanan ekstra alan bilgisi yüzünden çok büyük alan kaplamaktadır, bundan dolayı büyük videolar veya resimlerin XML standardı kullanılarak aktarılması oldukça maaliyetlidir, yüksek bant genişliği ister,
- SGML yapısını kullanmasına karşın, aslında SGML'in minimize edilmiş şeklidir. Bundan dolayı SGML programcılarının SGML kullanarak yaptığı çalışmaların XML'e dönüştürülmesi mümkün değildir.
- Noktalı sayılarda ve bunların doğrulanması sırasında problemler yaşadığı söylenmektedir, bu problemlerin çözümleri olabildiğince komplekstir..
- Eğer hiyerarşik bir yapısı olmayan bir dökümanın aktarımını yapacaksak, text formatına göre oldukça maaliyetlidir ve gereksiz bir ekstra efor harcamış oluruz.

### **3.3.1.2. XML Söz Dizimi**

Örnek XML dökümanı Şekil 5'te gösterilmektedir.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
  <note>
    <to>Tove</to>
    <from>Jani</from>
    <heading>Reminder</heading>
    <body>Don't forget me this weekend!</body>
  </note>
```

Şekil 5 – Örnek XML Dökümanı

XML dökümanları kendi içerisindeki bilgiyi tanımlayan özel bir basit söz dizim yapısı kullanmaktadırlar.

XML dosyasının ilk satırı, XML tanımlaması şeklinde adlandırılır ve XML versiyonunu ve karakter kodlamasını gösterir. Şekil 5'te gösterilen örnek dökümanda XML versiyonu 1.0 ve karakter kodlaması ISO-8859-1 (Latin-1/West European) olarak verilmiştir.

Daha sonra gelen satırlardaki, ilk kısım dökümanın kök elemanıdır(<note>). Kök elemanın ardından gelen ilk 4 satır, kök elemana bağlı kök elemanın çocukları gösterir (<to>, <from>, <heading>, and <body>). Son satır ise dökümanın bittiğini gösteren kök elemanın son satırıdır (</note>).

Bir XML dökümanı oluşturulurken, dikkat edilmesi gereken noktalar vardır. Bunlar;

- Tüm XML döküman elemanları tag lar ile kapatılmalıdır(</note>),
- XML tagları büyük küçük harflere duyarlıdır, bundan dolayı tagları açıp kapatırken dikkat edilmesi gerekir,
- Tüm XML elemanları uygun şekilde dallanmış olmalıdır,
- Tüm XML dökümanlarının bir kök elemanı olmalıdır.

### **3.3.2. İleri Şifreleme Sistemi AES (Advance Encryption System)**

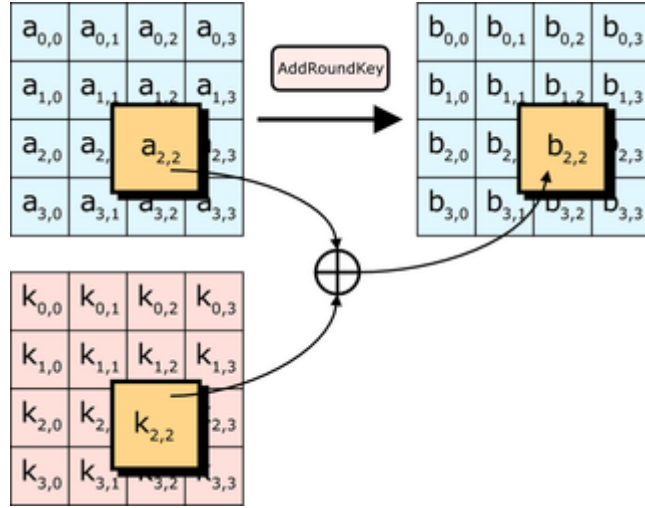
Kriptografide, Rijndael olarak bilinen İleri Şifreleme Sistemi (AES, Advance Encryption System), Amerikan Hükümeti tarafından geliştirilmiş bir blok şifreleme standartıdır. AES Ulusal Standartlar ve Teknoloji Enstitüsü (National Institute of Standards and Technology, NIST) tarafından Kasım 2001 yılında US FIPS PUB 197 adıyla ve yaklaşık 5 yıl süren bir standartlaşma sürecinden sonra yayınlanmıştır[5].

İlk ve basit şekli iki belçikalı kriptografi uzmanı Joan Daemen ve Vincent Rijmen tarafından bulunmuştur[6]. Oldukça güçlü ve günümüzde de kullanılan bir şifreleme sistemidir. Temel olarak kompleks algortima yapısına dayanamamasına karşın, şifreleme yapısı matrisler üzerine kurulu olduğundan basit şifreleme yöntemlerine oranla pahalı bir sistemdir. Dönen bir şifreleme anahtarına sahiptir, ayrıca look up tablolarına ve kaydırma fonksiyonlarına ihtiyaç duyar.

#### **3.3.2.1. AES'in Adımları**

Önemli ve güncel bir şifreleme sistemi olan AES dört temel adımdan oluşmaktadır. Bu bölümde bu adımları ve adımlar süresinde yapılan işlemleri inceleyeceğiz.

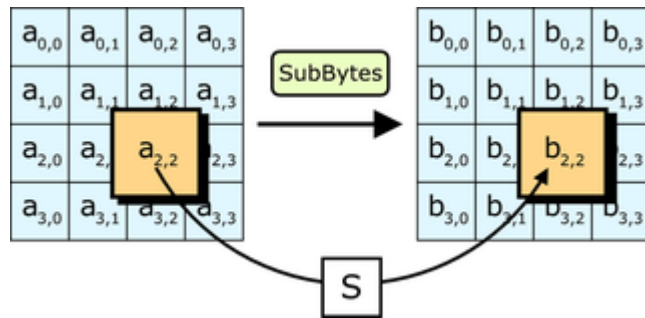
**a)** Dönen şifreleme anahtarının eklenmesi (AddRoundKey), her bir veri dönen şifreleme anahtarı ile birleştirilir. Dönen şifreleme anahtarı şifreleme sistemi tarafından oluşturulmuş bir plana göre her seferinde farklı bir anahtar oluşturur. Şekil 6'da bu şifreleme yapısı gösterilmiştir.



Şekil 6 – Dönen Anahtarın Eklenmesi (AddRoundKey)

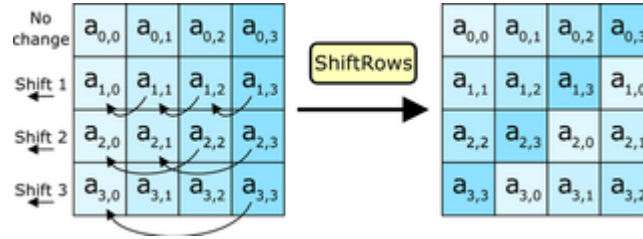
Şekil 6’da da gösterildiği üzere K matrisindeki dönen anahtar, A matrisindeki temel veri ile XOR’lanarak B matrisi oluşturulur.

b) Her bir alt veri(SubBytes), düzlemsel olmayan ve lookup tablosu kullanılarak bir yer değiştirme adımından geçer. Şekil 7’de alt veri yer değiştirme adımı gösterilmiştir.



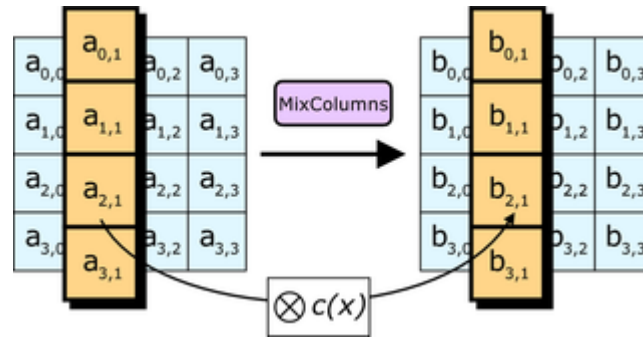
Şekil 7 – Alt Veri (SubBytes)

c) Satır Kaydırma(ShiftRows), bu adımda her bir satır daha önceden belirlenmiş bir sayı kadar kaydırma işleminden geçer. İlk satır alt verisi bu işlemde geçmez. İkinci satır bir sol kaydırma ile, üçüncü ve dördüncü satırlarda sırasıyla iki ve üç sol kaydırma ile bu adımdan geçer.



Şekil 8 – Satır Kaydırma (Shift Rows)

d) Kolonların karıştırılması (MixColumns), bu adımda kolonlar düzlemsel bir şekilde tek tek karıştırılır. Şekil 9’da gösterildiği üzere her bir kolonun dört byte’ı işleme girer ve yine dört byte’lık bir çıktı oluşturulur.



Şekil 9 – Kolon Karıştırma (MixColumns)

### 3.3.2.2. AES’in ‘mShipment’ İçerisinde Uygulanması

Bu bölümde, ‘mShipment’ içerisinde merkezi sistem ve mobil cihaz arasındaki güvenli veri transferinin nasıl gerçekleştiğini inceleyeceğiz. Her bir teslimat bilgisi birleştirilerek bir xml dökümanı oluşturulur. Şekil 10’da yollanamaya hazır bir teslimat verisi gösterilmiştir.



```
<Table>
  <Delivery>
    <ID>1</ID>
    <CusName>YILDIRIM</CusName>
    <CusSurname>KOCDAG</CusSurname>
    <CusAddress>KRIZANTEM SOK.GOZCAN APT.4/5</CusAddress>
    <CusDistrictID>1</CusDistrictID>
    <CusCityID>1</CusCityID>
    <CusPhone>+902163052460</CusPhone>
  </Delivery>
  <Delivery>
    <ID>2</ID>
    <CusName>HAKAN</CusName>
    <CusSurname>KOCDAG</CusSurname>
    <CusAddress>Vardar Apt.No:42</CusAddress>
    <CusDistrictID>5</CusDistrictID>
    <CusCityID>1</CusCityID>
    <CusPhone>+902122990000</CusPhone>
  </Delivery>
</Table>
```

**Şekil 10 – XML Dosyası**

Cep bilgisayarı ve merkezi sistem arasında senkronizasyon gerçekleştirilmeden önce, Sade XML dökümanı AES ile Şekil 11 de gözüktüğü şekilde şifrelenir. Bu işlem sırasında 256 Bit'lik "ACDGRYTUEISAXODKDKJSDJKDASQWEUOINCXZ" anahtar kullanılmıştır. Şifrelemeden sonra dosya iki taraf arasında güvenli bir şekilde iletilir.

```
197 67 1 133 123 1 252 1 19 14 49 64 175 167 47 94 9 184 190 246 135 230 127 80 215
175 36 207 232 186 65 33 60 77 118 32 22 80 222 15 100 157 32 19 195 70 115 186 136
163 219 122 92 202 126 225 188 227 147 219 131 49 132 78 116 172 222 226 7 222 233
228 100 250 44 132 14 146 140 0 125 8 183 6 81 234 214 91 156 79 119 90 54 55 0 208
204 177 166 53 41 209 215 124 57 99 88 100 195 102 106 30 159 227 69 15 51 68 31 54
151 94 166 241 168 43 217 253 199 107 104 246 71 152 174 250 196 193 91 193 67 123
244 100 127 216 192 29 102 244 148 112 132 130 6 63 146 254 134 200 43 124 188 19
59 29 25 154 92 22 225 98 221 94 50 48 111 215 186 219 131 9 231 92 97 89 126 185
235 156 104 234 111 215 44 66 164 183 48 88 111 241 52 133 45 230 212 239 152 120
165 71 121 118 82 169 239 214 21 195 239 74 201 252 30 114 35 0 15 85 154 207 72
69 221 193 72 229 150 131 10 165 45 62 236 8 253 225 212 78 46 132 37 129 218 38
136 0 219 0 81 149 163 216 83 179 89 28 27 12 242 8 121 221 97 47 161 189 108 165
225 41 24 240 228 174 122 104 125 207 10 151 124 213 222 157 178 218 236 218 28 49
244 189 207 31 17 143 43 66 214 20 96 9 230 28 209 1 153 226 181 182 203 163 6 215
217 251 102 46 244 192 91 50 60 133 146 141 207 82 247 197 51 46 227 190 98 44 181
231 188 127 45 195 56 121 31 18 146 53 36 198 237 118 120 12 128 171 163 227 167 12
111 247 89 158 37 116 39 239 92 32 145 166 117 253 20 53 132 142 247 2 198 122 166
115 177 218 246 252 4 70 248 210 214 184 158 159 231 26 248 199 206 248 215 223 130
30 20 115 53 3 168 150 43 143 87 171 104 32 114 56 8 189 56 1 58 122 149 54 222 60
208 236 63 188 131 25 230 193 52 43 208 16 201 211 179 140 40 179 76 220 134 19
```

Şekil 11 – Şifrelenmiş XML Dosyası

AES en iyi ve en güvenli şifreleme sistemi olmamakla beraber, mobil cihazın işlemci hızı ve kısıtlı belleğini göz önüne alınırsa, ‘**mShipment**’ sistemi için en uygun çözümlerden biri olduğu söylenebilir. Burada dikkat edilecek noktalardan bir diğeri ise, istenirse tüm teslimat bilgisi değil sadece gerekli görülen bilgiler, örnek olarak imza bilgisi şifrelenebilir. Böylelikle veriyi şifrelemek ve şifrelenmiş veriyi açmak için gerekli işlemci masrafından kurtulmuş olur ve sisteme hız katabiliriz.

### 3.4. İmza Yakalama

Güvenli teslimatı tamamlamak adına, müşterinin ıslak imzasını almak son adımdır. İmza alma sürecinde iki önemli işlem vardır, birincisi imza alma, ikincisi ise alınan imzanın kaydedilmesi. İmza stylus kalem ve ekranın hassas dokunmatik özelliği sayesinde kolayca alınabilir. Fakat imzayı sakalayabilmek, imzayı aldığımız kadar kolay değildir. PDA’lerin hafızasının limitli olmasından dolayı bu verinin transforme edilip, sıkıştırılarak saklanması gereklidir. Ayrıca gene PDA in kısıtlı CPU (Central Processing Unit, Merkezi İşlem Birimi) işlemcisini gözönüne alırsak bu sıkıştırma ve

transforme işlemi olabildiğince basit olmak zorundadır. Bu problemin çözümünün adımları aşağıda gösterilmiştir;

- İmza alındıktan sonra, imza üzerindeki çizili noktalar 1 ile boş kalan noktalar ise 0 ile yer değiştirilir,
- 0 olan noktalar ihmal edilir,
- 1 olan noktaların koordinatları (X, Y) şeklinde belirlenir,
- Bir sonraki adımda bu noktalar içinde aynı doğruyu olan noktalar arasında eleme yapılır ve aynı doğruyu oluşturan noktalardan başlangıç ve bitiş noktaları kayıt edilir.

### **3.5. Sistemin Kullanılabilirliği**

Sistem fizibilitesi, proje tutarı ve güvenlik incelendikten sonra, projenin başarıyla hayata geçirilebileceği ve bir çok teslimat şirketi için uygun bir çözüm olabileceğini rahatlıkla söyleyebiliriz.

## **4. SONUÇ**

**'mShipment'** bir kargo taşımacılığı yapan firma için yeni teknolojileri kullanarak teslimatını en güvenilir, en hızlı ve modern şekilde müşterisine ulaştırmasını amaç edinmiş bir projedir. Teknolojinin ve yeniliklerin artması, yıllık ciroları ülkemizde bile milyar dolarları aşan bu sektör için yeni cihazların kullanılmasını ve projelerin hazırlanmasını kaçınılmaz hale getirmiştir.

Cep bilgisayarları ve sayısal imzalar ise sadece bu sektörde değil bankacılıktan, muhasebeye, bahis şirketlerinden, medya devlerine kadar hemen hemen her alanda

kullanılabilecek büyük bir yelpazeye sahiptirler. Böylelikle küçük bir öngörü ile bu alanda atılacak adımların ileriki günlerde çok daha fazla ses getirebileceğini rahatlıkla söyleyebiliriz.

Sonuç olarak kullandığı kargolama ve taşımacılık sektörü için sayısal ıslak imzanın alınması ve bu imzanın sıkıştırılıp, şifreli iletimi bu alanda ilk niteliğinde olmuştur. Bu sisteme ek olarak GPS tabanlı harita, cep bilgisayarları ve .NET teknolojileri ile de bu projenin benzer taşımacılık ve kargo projeleri arasında iyi bir yer alabileceğini söyleyebiliriz.

## **5. REFERANSLAR**

- [1] “History of the Personal Data Assistant (PDA)”, [www.bbc.co.uk](http://www.bbc.co.uk), March 2004
- [2] “Evolution of PDAs”, Evan Koblentz, May 2005
- [3] “What is XML”, Kendall Clark, [www.xml.com](http://www.xml.com), March 2006
- [4] “XML Syntax”, Jan Egil Refsnes, [www.xmlfiles.com](http://www.xmlfiles.com), September 2004
- [5] “Report on the Development of the Advanced Encryption Standard (AES)”, James Nechvatal, Elaine Barker, Lawrence Bassham, William Burr, Morris Dworkin, James Foti, and Edward Roback, <http://cr.yip.to/bib/2001/nechvatal.pdf>, June 2001
- [6] “Encrypt It: Keep Your Data Secure with the New Advanced Encryption Standard”, James McCaffrey, MSDN Magazine - The Microsoft Journal for Software Developers, Nov 2003

## **EKLER**

### **A) Sistemin Ekran Görüntüleri**



Şekil 12 – Cep Bilgisayarının Kurulumu



Şekil 13 – İşletim Sisteminin Başlangıcı



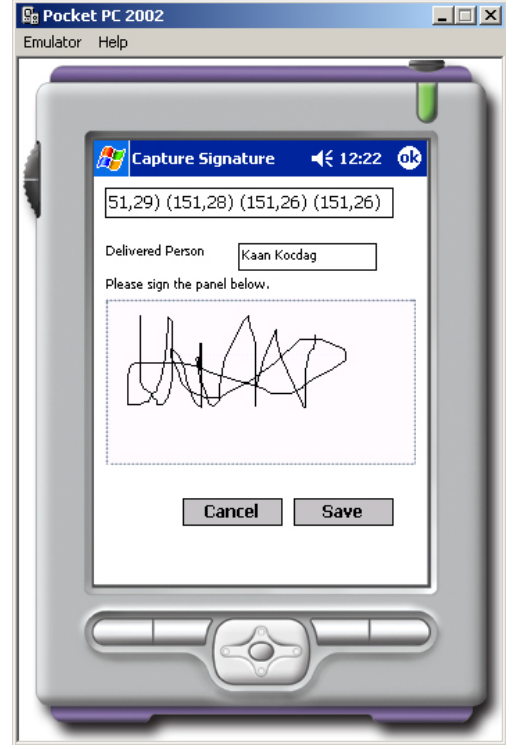
Şekil 14 – Program Girişi



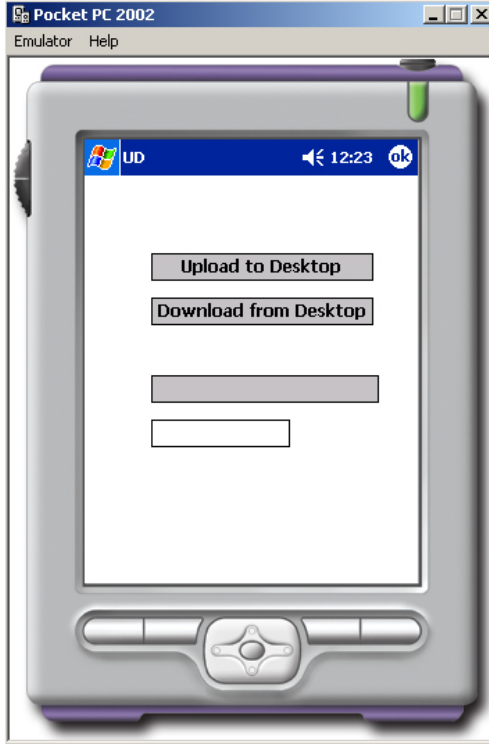
Şekil 15 – Ana Menü



Şekil 16 – Teslimat Menüü



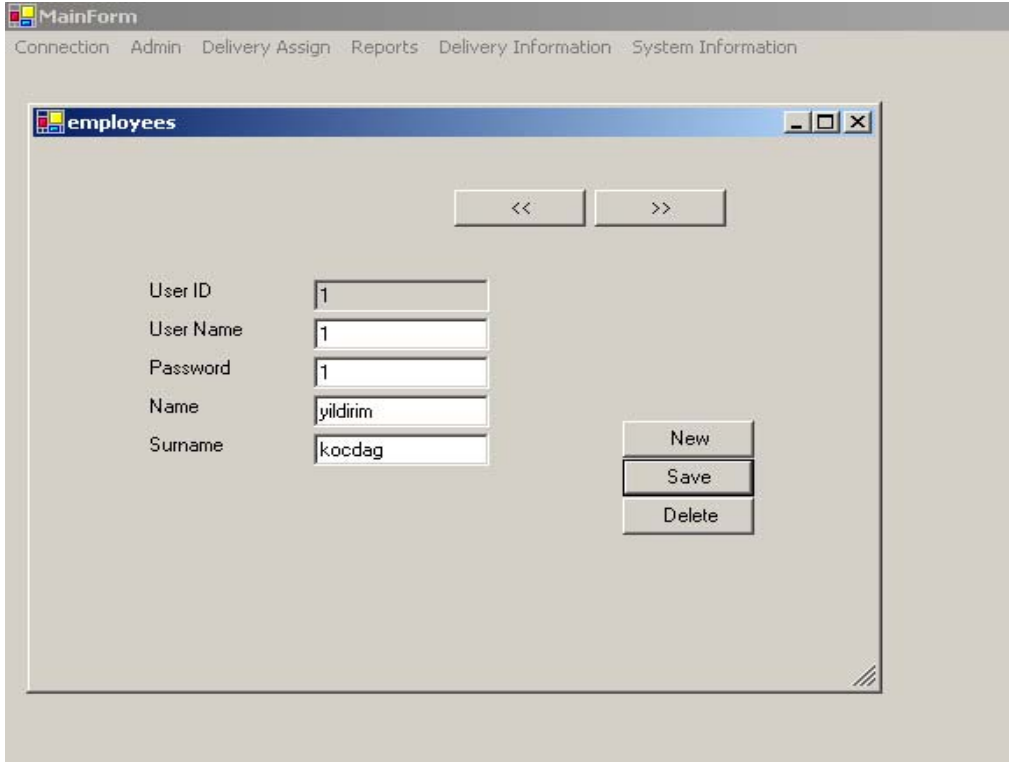
Şekil 17 – İmza Ekranı



Şekil 18 – Veri Aktarımı



Şekil 19 – Cep Bilgisayarının Veritabanı



Şekil 20 – Çalışanlar Ekranı

MainForm  
 Connection Admin Delivery Assign Reports Delivery Information System Information

Customers

<< >>

Customer ID: 1

Name: kocdag

Surname: kocdag

Email: kaankocdag@yahoo.com

Phone: 2163052460

Address: krizantem sokak no 4 dai 5

City: istanbul

Town: Maltepe

New  
 Save  
 Delete

Şekil 21 – Müşteriler Ekranı

MainForm  
 Connection Admin Delivery Assign Reports Delivery Information System Information

deliveries

<< >>

Delivery ID: 1

Customer ID/Name: 1 / Bkocdag Ara

Contact Name: BAHRI

Address: YOK

City: istanbul

Town: Maltepe

Phone: 2163052460

Status: Delivered

New  
 Save  
 Delete  
 Signature

Delivery Information

Assigned Employee: yildirim-kocdag

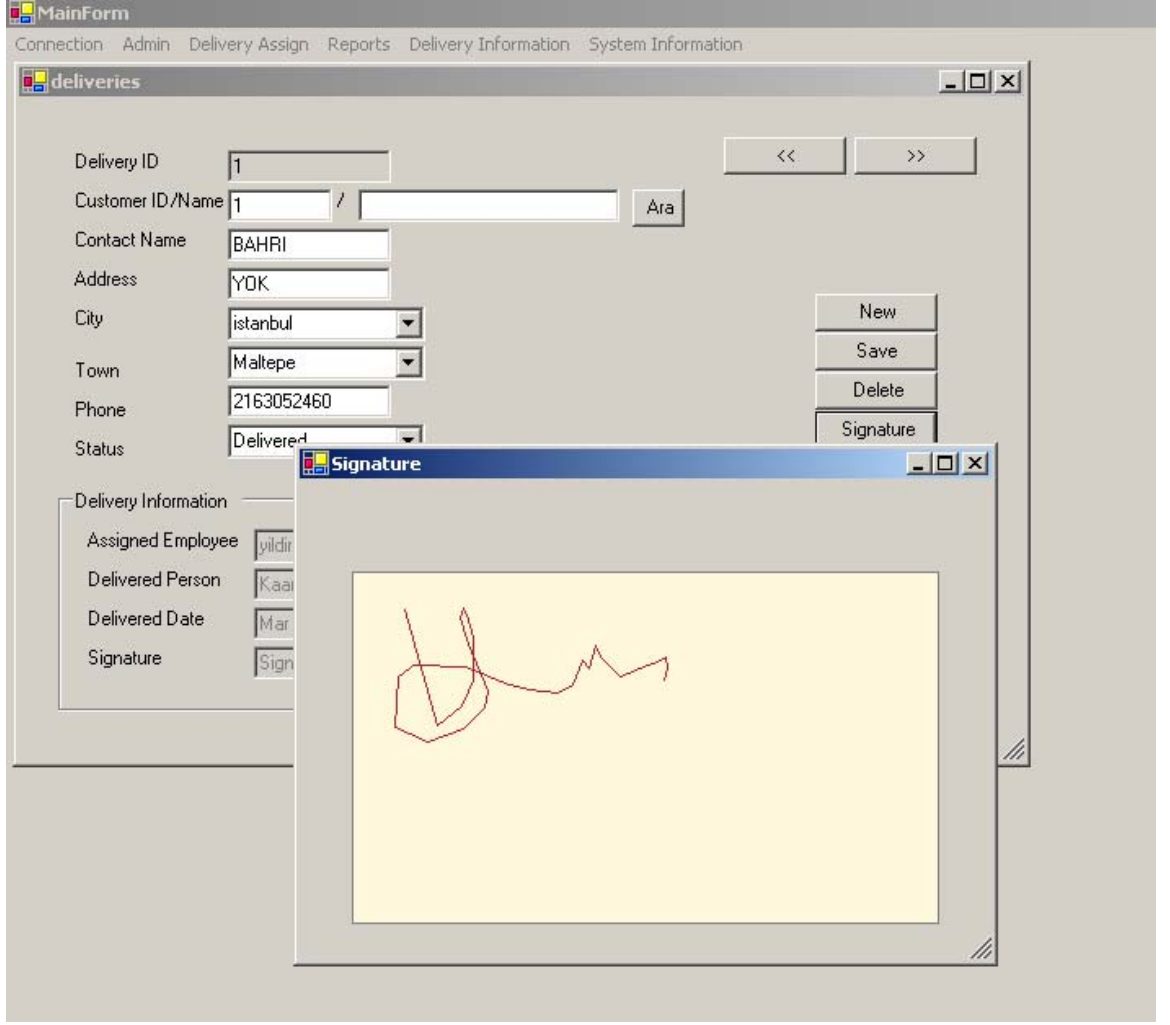
Delivered Person: Kaan Kocdag

Delivered Date: Mar 8 2006 3:28A

Signature: Signature detailsNumber of segments : 1Segment 1 (33 points)[32,23] [52,94]

Şekil 22 – Teslimatlar Ekranı





Şekil 23 – Müşteri İmza Ekranı

MainForm  
Connection Admin Delivery Assign Reports Delivery Information System Information

Cities

<< >>

City ID 1  
City Name istanbul

New  
Save  
Delete

Şekil 24 – Şehirler Ekranı

MainForm  
Connection Admin Delivery Assign Reports Delivery Information System Information

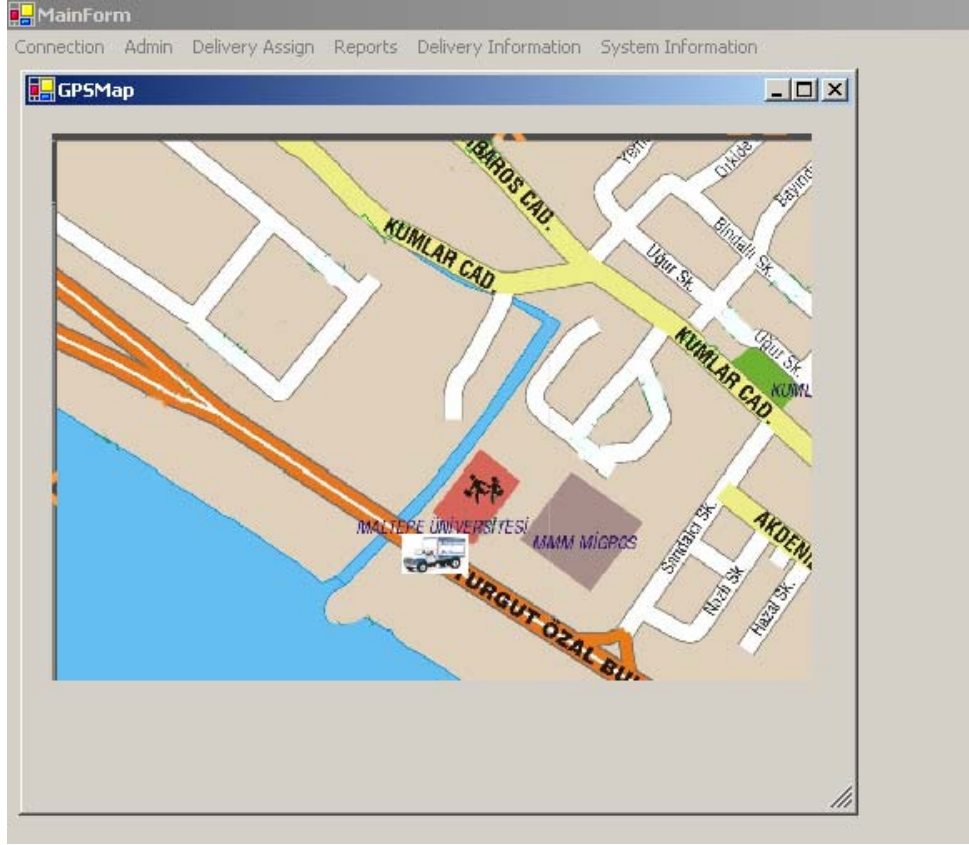
Towns

<< >>

Town ID 1  
Town Name Maltepe  
City Name istanbul

New  
Save  
Delete

Şekil 25 – İlçeler Ekranı



Şekil 26 – GPS Harita

## B) Sistem Kodları

### Masaüstü Bilgisayar Çözümü

## Giriş Ekranı

```
Public Class Login
    Inherits System.Windows.Forms.Form

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents Label1 As System.Windows.Forms.Label
    Friend WithEvents Label2 As System.Windows.Forms.Label
    Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
    Friend WithEvents TextBox2 As System.Windows.Forms.TextBox
    Friend WithEvents Button1 As System.Windows.Forms.Button
    Friend WithEvents Button2 As System.Windows.Forms.Button
    <System.Diagnostics.DebuggerStepThrough> Private Sub InitializeComponent()
        Me.Label1 = New System.Windows.Forms.Label
        Me.Label2 = New System.Windows.Forms.Label
        Me.TextBox1 = New System.Windows.Forms.TextBox
        Me.TextBox2 = New System.Windows.Forms.TextBox
        Me.Button1 = New System.Windows.Forms.Button
        Me.Button2 = New System.Windows.Forms.Button
        Me.SuspendLayout()
        '
        'Label1
        '
        Me.Label1.Location = New System.Drawing.Point(40, 48)
        Me.Label1.Name = "Label1"
        Me.Label1.TabIndex = 0
        Me.Label1.Text = "User Name"
        '
        'Label2
        '
        Me.Label2.Location = New System.Drawing.Point(40, 72)
        Me.Label2.Name = "Label2"
        Me.Label2.TabIndex = 1
        Me.Label2.Text = "Password"
        '
        'TextBox1
        '
        Me.TextBox1.Location = New System.Drawing.Point(144, 48)
        Me.TextBox1.Name = "TextBox1"
        Me.TextBox1.TabIndex = 2
        Me.TextBox1.Text = "1"
        '
        'TextBox2
        '
        Me.TextBox2.Location = New System.Drawing.Point(144, 72)
        Me.TextBox2.Name = "TextBox2"
    End Sub
End Class
```

```

Me.TextBox2.PasswordChar = Microsoft.VisualBasic.ChrW(42)
Me.TextBox2.TabIndex = 3
Me.TextBox2.Text = "1"
'
'Button1
'
Me.Button1.Location = New System.Drawing.Point(192, 184)
Me.Button1.Name = "Button1"
Me.Button1.TabIndex = 4
Me.Button1.Text = "Login"
'
'Button2
'
Me.Button2.Location = New System.Drawing.Point(112, 184)
Me.Button2.Name = "Button2"
Me.Button2.TabIndex = 5
Me.Button2.Text = "Exit"
'
>Login
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(292, 266)
Me.Controls.Add(Me.Button2)
Me.Controls.Add(Me.Button1)
Me.Controls.Add(Me.TextBox2)
Me.Controls.Add(Me.TextBox1)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.Label1)
Me.Name = "Login"
Me.Text = "Login"
Me.ResumeLayout(False)

End Sub

#End Region

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim cn As System.Data.SqlClient.SqlConnection
    Dim cmd As System.Data.SqlClient.SqlCommand
    Dim dtr As System.Data.SqlClient.SqlDataReader
    Dim str As String
    cn = New System.Data.SqlClient.SqlConnection(GlobalModule.conn)

    ' Open the database.
    cn.Open()
    ' (username, password, activity)
    str = "SELECT * FROM Users WHERE username='" + TextBox1.Text + "' AND
password='" + TextBox2.Text + "'"
    cmd = New System.Data.SqlClient.SqlCommand(str, cn)
    dtr = cmd.ExecuteReader()
    ' Load the list into the customer label2 .
    If dtr.Read() Then
        loginid = dtr("userid")
        dtr.Close()
        cn.Close()
        'Dim frmMain As New Form1
        Dim b As New MainForm
        loginUSER = TextBox1.Text
        Me.Hide()
        b.ShowDialog()
    Else
        MsgBox("Incorrect Login Name or Password")
        dtr.Close()
        cn.Close()
    End If

End Sub

Private Sub Login_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load

End Sub

End Class

```

## Ana Menü

```
Public Class MainForm
    Inherits System.Windows.Forms.Form

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents MainMenu1 As System.Windows.Forms.MainMenu
    Friend WithEvents MenuItem1 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem2 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem3 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem5 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem6 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem7 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem8 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem9 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem10 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem11 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem12 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem13 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem14 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem15 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem16 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem17 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem18 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem19 As System.Windows.Forms.MenuItem
    Friend WithEvents MenuItem20 As System.Windows.Forms.MenuItem
    <System.Diagnostics.DebuggerStepThrough() Private Sub InitializeComponent()
        Me.MainMenu1 = New System.Windows.Forms.MainMenu
        Me.MenuItem1 = New System.Windows.Forms.MenuItem
        Me.MenuItem2 = New System.Windows.Forms.MenuItem
        Me.MenuItem3 = New System.Windows.Forms.MenuItem
        Me.MenuItem5 = New System.Windows.Forms.MenuItem
        Me.MenuItem6 = New System.Windows.Forms.MenuItem
        Me.MenuItem7 = New System.Windows.Forms.MenuItem
        Me.MenuItem8 = New System.Windows.Forms.MenuItem
        Me.MenuItem19 = New System.Windows.Forms.MenuItem
        Me.MenuItem20 = New System.Windows.Forms.MenuItem
        Me.MenuItem9 = New System.Windows.Forms.MenuItem
        Me.MenuItem10 = New System.Windows.Forms.MenuItem
        Me.MenuItem14 = New System.Windows.Forms.MenuItem
        Me.MenuItem11 = New System.Windows.Forms.MenuItem
```

```

Me.MenuItem12 = New System.Windows.Forms.MenuItem
Me.MenuItem13 = New System.Windows.Forms.MenuItem
Me.MenuItem15 = New System.Windows.Forms.MenuItem
Me.MenuItem16 = New System.Windows.Forms.MenuItem
Me.MenuItem17 = New System.Windows.Forms.MenuItem
Me.MenuItem18 = New System.Windows.Forms.MenuItem
'
'MainMenu1
'
Me.MainMenu1.MenuItems.AddRange(New System.Windows.Forms.MenuItem()
{Me.MenuItem1, Me.MenuItem5, Me.MenuItem9, Me.MenuItem11, Me.MenuItem15, Me.MenuItem17})
'
'MenuItem1
'
Me.MenuItem1.Index = 0
Me.MenuItem1.MenuItems.AddRange(New System.Windows.Forms.MenuItem()
{Me.MenuItem2, Me.MenuItem3})
Me.MenuItem1.Text = "Connection"
'
'MenuItem2
'
Me.MenuItem2.Index = 0
Me.MenuItem2.Text = "Log IN/OUT"
'
'MenuItem3
'
Me.MenuItem3.Index = 1
Me.MenuItem3.Text = "Upload/Download"
'
'MenuItem5
'
Me.MenuItem5.Index = 1
Me.MenuItem5.MenuItems.AddRange(New System.Windows.Forms.MenuItem()
{Me.MenuItem6, Me.MenuItem7, Me.MenuItem8, Me.MenuItem19, Me.MenuItem20})
Me.MenuItem5.Text = "Admin"
'
'MenuItem6
'
Me.MenuItem6.Index = 0
Me.MenuItem6.Text = "Employees"
'
'MenuItem7
'
Me.MenuItem7.Index = 1
Me.MenuItem7.Text = "Customers"
'
'MenuItem8
'
Me.MenuItem8.Index = 2
Me.MenuItem8.Text = "Deliveries"
'
'MenuItem19
'
Me.MenuItem19.Index = 3
Me.MenuItem19.Text = "Towns"
'
'MenuItem20
'
Me.MenuItem20.Index = 4
Me.MenuItem20.Text = "Cities"
'
'MenuItem9
'
Me.MenuItem9.Index = 2
Me.MenuItem9.MenuItems.AddRange(New System.Windows.Forms.MenuItem()
{Me.MenuItem10, Me.MenuItem14})
Me.MenuItem9.Text = "Delivery Assign"
'
'MenuItem10
'
Me.MenuItem10.Index = 0
Me.MenuItem10.Text = "Delivery Employee Assign"
'
'MenuItem14

```

```

    Me.MenuItem4.Index = 1
    Me.MenuItem4.Text = "Town Employee Assign"
    'MenuItem11
    Me.MenuItem11.Index = 3
    Me.MenuItem11.MenuItems.AddRange(New System.Windows.Forms.MenuItem()
{Me.MenuItem12, Me.MenuItem13})
    Me.MenuItem11.Text = "Reports"
    'MenuItem12
    Me.MenuItem12.Index = 0
    Me.MenuItem12.Text = "Delivery Reports"
    'MenuItem13
    Me.MenuItem13.Index = 1
    Me.MenuItem13.Text = "Employee Reports"
    'MenuItem15
    Me.MenuItem15.Index = 4
    Me.MenuItem15.MenuItems.AddRange(New System.Windows.Forms.MenuItem()
{Me.MenuItem16})
    Me.MenuItem15.Text = "Delivery Information"
    'MenuItem16
    Me.MenuItem16.Index = 0
    Me.MenuItem16.Text = "GPS Based Map"
    'MenuItem17
    Me.MenuItem17.Index = 5
    Me.MenuItem17.MenuItems.AddRange(New System.Windows.Forms.MenuItem()
{Me.MenuItem18})
    Me.MenuItem17.Text = "System Information"
    'MenuItem18
    Me.MenuItem18.Index = 0
    Me.MenuItem18.Text = "About"
    'MainForm
    Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
    Me.ClientSize = New System.Drawing.Size(656, 446)
    Me.Menu = Me.MainMenu1
    Me.Name = "MainForm"
    Me.Text = "MainForm"

End Sub

#End Region

Private Sub MenuItem2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MenuItem2.Click
    loginUSER = ""
    Dim logn As New Login
    logn.ShowDialog()
    Me.Close()

End Sub

Private Sub MenuItem4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
    Dim down As New Download
    down.ShowDialog()
End Sub

```



```

        Private Sub MenuItem19_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MenuItem19.Click
            Dim twn As New Towns
            twn.ShowDialog()
        End Sub

        Private Sub MenuItem20_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MenuItem20.Click
            Dim cts As New Cities
            cts.ShowDialog()
        End Sub

        Private Sub MenuItem7_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MenuItem7.Click
            Dim Cust As New Customers
            Cust.ShowDialog()
        End Sub

        Private Sub MenuItem6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MenuItem6.Click
            Dim emp As New employees
            emp.ShowDialog()
        End Sub

        Private Sub MenuItem8_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MenuItem8.Click
            Dim dlw As New deliveries
            dlw.ShowDialog()
        End Sub

        Private Sub MenuItem3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MenuItem3.Click
            Dim UPL As New UPLForm
            UPL.ShowDialog()
        End Sub

        Private Sub MenuItem16_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MenuItem16.Click
            Dim gpsmp As New GPSMap
            gpsmp.ShowDialog()

        End Sub
End Class

```

## Şehirler

```

Public Class Cities
    Inherits System.Windows.Forms.Form

    #Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

```

```

'Required by the Windows Form Designer
Private components As System.ComponentModel.IContainer

'NOTE: The following procedure is required by the Windows Form Designer
'It can be modified using the Windows Form Designer.
'Do not modify it using the code editor.
Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents Label2 As System.Windows.Forms.Label
Friend WithEvents btnNext As System.Windows.Forms.Button
Friend WithEvents btnPrevious As System.Windows.Forms.Button
Friend WithEvents txtCity_ID As System.Windows.Forms.TextBox
Friend WithEvents txtCity_Name As System.Windows.Forms.TextBox
Friend WithEvents btnSave As System.Windows.Forms.Button
Friend WithEvents btnDelete As System.Windows.Forms.Button
Friend WithEvents btnNew As System.Windows.Forms.Button
<System.Diagnostics.DebuggerStepThrough(> Private Sub InitializeComponent()
    Me.btnSave = New System.Windows.Forms.Button
    Me.btnDelete = New System.Windows.Forms.Button
    Me.Label1 = New System.Windows.Forms.Label
    Me.Label2 = New System.Windows.Forms.Label
    Me.txtCity_ID = New System.Windows.Forms.TextBox
    Me.txtCity_Name = New System.Windows.Forms.TextBox
    Me.btnNext = New System.Windows.Forms.Button
    Me.btnPrevious = New System.Windows.Forms.Button
    Me.btnNew = New System.Windows.Forms.Button
    Me.SuspendLayout()
    '
    'btnSave
    '
    Me.btnSave.Location = New System.Drawing.Point(296, 136)
    Me.btnSave.Name = "btnSave"
    Me.btnSave.TabIndex = 0
    Me.btnSave.Text = "Save"
    '
    'btnDelete
    '
    Me.btnDelete.Location = New System.Drawing.Point(296, 160)
    Me.btnDelete.Name = "btnDelete"
    Me.btnDelete.TabIndex = 1
    Me.btnDelete.Text = "Delete"
    '
    'Label1
    '
    Me.Label1.Location = New System.Drawing.Point(40, 72)
    Me.Label1.Name = "Label1"
    Me.Label1.TabIndex = 2
    Me.Label1.Text = "City ID"
    '
    'Label2
    '
    Me.Label2.Location = New System.Drawing.Point(40, 96)
    Me.Label2.Name = "Label2"
    Me.Label2.TabIndex = 3
    Me.Label2.Text = "City Name"
    '
    'txtCity_ID
    '
    Me.txtCity_ID.Location = New System.Drawing.Point(144, 72)
    Me.txtCity_ID.Name = "txtCity_ID"
    Me.txtCity_ID.ReadOnly = True
    Me.txtCity_ID.TabIndex = 4
    Me.txtCity_ID.Text = ""
    '
    'txtCity_Name
    '
    Me.txtCity_Name.Location = New System.Drawing.Point(144, 96)
    Me.txtCity_Name.Name = "txtCity_Name"
    Me.txtCity_Name.TabIndex = 5
    Me.txtCity_Name.Text = ""
    '
    'btnNext
    '
    Me.btnNext.Location = New System.Drawing.Point(304, 16)

```

```

Me.btnNext.Name = "btnNext"
Me.btnNext.TabIndex = 6
Me.btnNext.Text = ">>"
'
'btnPrevious
'
Me.btnPrevious.Location = New System.Drawing.Point(224, 16)
Me.btnPrevious.Name = "btnPrevious"
Me.btnPrevious.TabIndex = 7
Me.btnPrevious.Text = "<<"
'
'btnNew
'
Me.btnNew.Location = New System.Drawing.Point(296, 112)
Me.btnNew.Name = "btnNew"
Me.btnNew.TabIndex = 8
Me.btnNew.Text = "New"
'
'Cities
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(400, 266)
Me.Controls.Add(Me.btnNew)
Me.Controls.Add(Me.btnPrevious)
Me.Controls.Add(Me.btnNext)
Me.Controls.Add(Me.txtCity_Name)
Me.Controls.Add(Me.txtCity_ID)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.Label1)
Me.Controls.Add(Me.btnDelete)
Me.Controls.Add(Me.btnSave)
Me.Name = "Cities"
Me.Text = "Cities"
Me.ResumeLayout(False)

End Sub

#End Region
Dim dt As New DataTable
Dim dt_max As Integer
Dim i As Integer

Private Sub Cities_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
Dim cnst As String
Dim sql As String
' Dim cmd As New SqlCommand
Dim da As New SqlDataAdapter
cnst = GlobalModule.conn
Dim cn As New SqlConnection(cnst)
cn.Open()
sql = " Select * from Cities order by City_id"
da = New SqlDataAdapter(sql, cn)
da.Fill(dt)
dt_max = dt.Rows.Count - 1
If dt_max < 0 Then
i = -1
cn.Close()
Exit Sub
End If
If dt_max < 1 Then
Me.btnPrevious.Enabled = False
Me.btnNext.Enabled = False
Else
Me.btnPrevious.Enabled = True
Me.btnNext.Enabled = True
End If
i = 0
ds_bind(0)
cn.Close()
End Sub

Sub ds_bind(ByVal record_no As Integer)
Me.txtCity_ID.Text = dt.Rows(record_no)("City_id")

```

```

        Me.txtCity_Name.Text = isSysNull(dt.Rows(record_no)("City_name"))
    End Sub
    Private Sub btnNext_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnNext.Click
        If i < dt_max Then
            i = i + 1
            ds_bind(i)
        Else
            If i > 0 Then
                ds_bind(i)
            End If
        End If
    End Sub
    Private Sub btnPrevious_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnPrevious.Click
        If i > 0 Then
            i = i - 1
            ds_bind(i)
        Else
            If i > 0 Then
                ds_bind(i)
            End If
        End If
    End Sub
    Private Sub btnNew_Record_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnNew.Click
        Me.txtCity_ID.Text = ""
        Me.txtCity_Name.Text = ""
        Me.btnPrevious.Enabled = True
        Me.btnNext.Enabled = True
        MsgBox("New Record is Writing")
    End Sub

    Private Sub btnSave_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSave.Click
        Dim sql As String
        If Me.txtCity_ID.Text <> "" Then
            sql = " Update Cities set City_name='" & Me.txtCity_Name.Text & "' where
City_id=" & Me.txtCity_ID.Text
        Else
            sql = " insert into Cities (City_Name) values " & _
                " ('" & Me.txtCity_Name.Text & "')"
        End If

        Dim cn As New SqlClient.SqlConnection(GlobalModule.conn)
        cn.Open()
        Dim cmd As New SqlClient.SqlCommand(sql, cn)
        cmd.ExecuteNonQuery()
        If Me.txtCity_ID.Text <> "" Then
            Dim da As New SqlClient.SqlDataAdapter
            sql = " Select * from Cities order by City_id"
            da = New SqlClient.SqlDataAdapter(sql, cn)
            dt.Rows.Clear()
            da.Fill(dt)
            ds_bind(i)
            MsgBox("Record Updated Successfully")
        Else
            Dim da As New SqlClient.SqlDataAdapter
            sql = " Select * from Cities order by City_id"
            da = New SqlClient.SqlDataAdapter(sql, cn)
            dt.Rows.Clear()
            da.Fill(dt)
            dt_max = dt.Rows.Count - 1
            i = dt_max
            ds_bind(dt_max)
            MsgBox("Record Saved Successfully")
            If dt_max < 1 Then
                Me.btnPrevious.Enabled = False
                Me.btnNext.Enabled = False
            Else
                Me.btnPrevious.Enabled = True
                Me.btnNext.Enabled = True
            End If
        End If
    End Sub

```

```

        End If
        cn.Close()
    End Sub

    Private Sub btnDelete_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnDelete.Click

        If Me.txtCity_ID.Text <> "" Then
            Dim sql As String
            sql = " Delete from Cities where City_id=" & txtCity_ID.Text
            Dim cn As New SqlClient.SqlConnection(GlobalModule.conn)
            cn.Open()
            Dim cmd As New SqlClient.SqlCommand(sql, cn)
            cmd.ExecuteNonQuery()
            Me.txtCity_ID.Text = ""
            Me.txtCity_Name.Text = ""
            MsgBox("Record Deleted Successfully")
            Dim da As New SqlClient.SqlDataAdapter
            sql = " Select * from Cities order by City_id"
            da = New SqlClient.SqlDataAdapter(sql, cn)
            dt.Rows.Clear()
            da.Fill(dt)
            dt_max = dt.Rows.Count - 1
            If dt_max > -1 Then
                i = 0
                ds_bind(0)
            End If
            If dt_max < 1 Then
                Me.btnPrevious.Enabled = False
                Me.btnNext.Enabled = False
            Else
                Me.btnPrevious.Enabled = True
                Me.btnNext.Enabled = True
            End If
            cn.Close()
        End If
    End Sub

End Class

```

## Müşteriler

```

Public Class Customers
    Inherits System.Windows.Forms.Form

    #Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

```

```

'Required by the Windows Form Designer
Private components As System.ComponentModel.IContainer

'NOTE: The following procedure is required by the Windows Form Designer
'It can be modified using the Windows Form Designer.
'Do not modify it using the code editor.
Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents Label2 As System.Windows.Forms.Label
Friend WithEvents Label3 As System.Windows.Forms.Label
Friend WithEvents Label4 As System.Windows.Forms.Label
Friend WithEvents Label5 As System.Windows.Forms.Label
Friend WithEvents Label6 As System.Windows.Forms.Label
Friend WithEvents Label7 As System.Windows.Forms.Label
Friend WithEvents Label8 As System.Windows.Forms.Label
Friend WithEvents ID As System.Windows.Forms.TextBox
Friend WithEvents Surname As System.Windows.Forms.TextBox
Friend WithEvents address As System.Windows.Forms.TextBox
Friend WithEvents phone As System.Windows.Forms.TextBox
Friend WithEvents email As System.Windows.Forms.TextBox
Friend WithEvents city As System.Windows.Forms.ComboBox
Friend WithEvents town As System.Windows.Forms.ComboBox
Friend WithEvents btnNew As System.Windows.Forms.Button
Friend WithEvents btnPrevious As System.Windows.Forms.Button
Friend WithEvents btnNext As System.Windows.Forms.Button
Friend WithEvents btnDelete As System.Windows.Forms.Button
Friend WithEvents btnSave As System.Windows.Forms.Button
Friend WithEvents txtname2 As System.Windows.Forms.TextBox
<System.Diagnostics.DebuggerStepThrough(> Private Sub InitializeComponent()
    Me.Label1 = New System.Windows.Forms.Label
    Me.Label2 = New System.Windows.Forms.Label
    Me.Label3 = New System.Windows.Forms.Label
    Me.Label4 = New System.Windows.Forms.Label
    Me.Label5 = New System.Windows.Forms.Label
    Me.Label6 = New System.Windows.Forms.Label
    Me.Label7 = New System.Windows.Forms.Label
    Me.Label8 = New System.Windows.Forms.Label
    Me.ID = New System.Windows.Forms.TextBox
    Me.Surname = New System.Windows.Forms.TextBox
    Me.address = New System.Windows.Forms.TextBox
    Me.phone = New System.Windows.Forms.TextBox
    Me.email = New System.Windows.Forms.TextBox
    Me.city = New System.Windows.Forms.ComboBox
    Me.town = New System.Windows.Forms.ComboBox
    Me.btnNew = New System.Windows.Forms.Button
    Me.btnPrevious = New System.Windows.Forms.Button
    Me.btnNext = New System.Windows.Forms.Button
    Me.btnDelete = New System.Windows.Forms.Button
    Me.btnSave = New System.Windows.Forms.Button
    Me.txtname2 = New System.Windows.Forms.TextBox
    Me.SuspendLayout()
    '
    'Label1
    '
    Me.Label1.Location = New System.Drawing.Point(48, 56)
    Me.Label1.Name = "Label1"
    Me.Label1.TabIndex = 0
    Me.Label1.Text = "Customer ID"
    '
    'Label2
    '
    Me.Label2.Location = New System.Drawing.Point(48, 80)
    Me.Label2.Name = "Label2"
    Me.Label2.TabIndex = 1
    Me.Label2.Text = "Name"
    '
    'Label3
    '
    Me.Label3.Location = New System.Drawing.Point(48, 128)
    Me.Label3.Name = "Label3"
    Me.Label3.TabIndex = 3
    Me.Label3.Text = "Email"
    '
    'Label4

```

```

|
Me.Label4.Location = New System.Drawing.Point(48, 104)
Me.Label4.Name = "Label4"
Me.Label4.Size = New System.Drawing.Size(104, 23)
Me.Label4.TabIndex = 2
Me.Label4.Text = "Surname"
|
'Label5
|
Me.Label5.Location = New System.Drawing.Point(48, 152)
Me.Label5.Name = "Label5"
Me.Label5.TabIndex = 5
Me.Label5.Text = "Phone"
|
'Label6
|
Me.Label6.Location = New System.Drawing.Point(48, 176)
Me.Label6.Name = "Label6"
Me.Label6.TabIndex = 4
Me.Label6.Text = "Address"
|
'Label7
|
Me.Label7.Location = New System.Drawing.Point(48, 224)
Me.Label7.Name = "Label7"
Me.Label7.TabIndex = 7
Me.Label7.Text = "Town"
|
'Label8
|
Me.Label8.Location = New System.Drawing.Point(48, 200)
Me.Label8.Name = "Label8"
Me.Label8.TabIndex = 6
Me.Label8.Text = "City"
|
'ID
|
Me.ID.Location = New System.Drawing.Point(152, 56)
Me.ID.Name = "ID"
Me.ID.ReadOnly = True
Me.ID.TabIndex = 8
Me.ID.Text = ""
|
'Surname
|
Me.Surname.Location = New System.Drawing.Point(152, 104)
Me.Surname.Name = "Surname"
Me.Surname.Size = New System.Drawing.Size(136, 20)
Me.Surname.TabIndex = 10
Me.Surname.Text = ""
|
'address
|
Me.address.Location = New System.Drawing.Point(152, 176)
Me.address.Name = "address"
Me.address.Size = New System.Drawing.Size(288, 20)
Me.address.TabIndex = 13
Me.address.Text = ""
|
'phone
|
Me.phone.Location = New System.Drawing.Point(152, 152)
Me.phone.Name = "phone"
Me.phone.Size = New System.Drawing.Size(136, 20)
Me.phone.TabIndex = 12
Me.phone.Text = ""
|
'email
|
Me.email.Location = New System.Drawing.Point(152, 128)
Me.email.Name = "email"
Me.email.Size = New System.Drawing.Size(136, 20)
Me.email.TabIndex = 11
Me.email.Text = ""

```

```

'city
'
Me.city.Location = New System.Drawing.Point(152, 200)
Me.city.Name = "city"
Me.city.Size = New System.Drawing.Size(121, 21)
Me.city.TabIndex = 14
'
'town
'
Me.town.Location = New System.Drawing.Point(152, 224)
Me.town.Name = "town"
Me.town.Size = New System.Drawing.Size(121, 21)
Me.town.TabIndex = 15
'
'btnNew
'
Me.btnNew.Location = New System.Drawing.Point(424, 224)
Me.btnNew.Name = "btnNew"
Me.btnNew.TabIndex = 20
Me.btnNew.Text = "New"
'
'btnPrevious
'
Me.btnPrevious.Location = New System.Drawing.Point(280, 16)
Me.btnPrevious.Name = "btnPrevious"
Me.btnPrevious.TabIndex = 19
Me.btnPrevious.Text = "<<"
'
'btnNext
'
Me.btnNext.Location = New System.Drawing.Point(360, 16)
Me.btnNext.Name = "btnNext"
Me.btnNext.TabIndex = 18
Me.btnNext.Text = ">>"
'
'btnDelete
'
Me.btnDelete.Location = New System.Drawing.Point(424, 272)
Me.btnDelete.Name = "btnDelete"
Me.btnDelete.TabIndex = 17
Me.btnDelete.Text = "Delete"
'
'btnSave
'
Me.btnSave.Location = New System.Drawing.Point(424, 248)
Me.btnSave.Name = "btnSave"
Me.btnSave.TabIndex = 16
Me.btnSave.Text = "Save"
'
'txtname2
'
Me.txtname2.Location = New System.Drawing.Point(160, 80)
Me.txtname2.Name = "txtname2"
Me.txtname2.TabIndex = 21
Me.txtname2.Text = ""
'
'Customers
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(608, 382)
Me.Controls.Add(Me.txtname2)
Me.Controls.Add(Me.btnNew)
Me.Controls.Add(Me.btnPrevious)
Me.Controls.Add(Me.btnNext)
Me.Controls.Add(Me.btnDelete)
Me.Controls.Add(Me.btnSave)
Me.Controls.Add(Me.town)
Me.Controls.Add(Me.city)
Me.Controls.Add(Me.address)
Me.Controls.Add(Me.phone)
Me.Controls.Add(Me.email)
Me.Controls.Add(Me.Surname)
Me.Controls.Add(Me.ID)

```



```

Me.Controls.Add(Me.Label7)
Me.Controls.Add(Me.Label8)
Me.Controls.Add(Me.Label5)
Me.Controls.Add(Me.Label6)
Me.Controls.Add(Me.Label3)
Me.Controls.Add(Me.Label4)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.Label11)
Me.Name = "Customers"
Me.Text = "Customers"
Me.ResumeLayout(False)

End Sub

#End Region
Dim dt As New DataTable
Dim dt_max As Integer
Dim i As Integer
Dim da_city As New SqlClient.SqlDataAdapter
Dim dt_city As New DataTable

Private Sub Customers_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

    Dim cnst As String
    Dim sql As String
    ' Dim cmd As New SqlClient.SqlCommand
    Dim da As New SqlClient.SqlDataAdapter
    cnst = GlobalModule.conn
    Dim cn As New SqlClient.SqlConnection(cnst)
    cn.Open()
    sql = " Select * from customers order by customer_id"
    da = New SqlClient.SqlDataAdapter(sql, cn)
    da.Fill(dt)
    dt_max = dt.Rows.Count - 1
    sql = " Select city_id,city_name from cities"
    da_city = New SqlClient.SqlDataAdapter(sql, cn)
    da_city.Fill(dt_city)
    Me.city.DataSource = dt_city
    Me.city.DisplayMember = "city_name"
    Me.city.ValueMember = "city_id"
    If dt_max < 0 Then
        i = -1
        cn.Close()
        Exit Sub
    End If
    Me.city.SelectedIndex = 0
    filltown(1)
    If dt_max < 1 Then
        Me.btnPrevious.Enabled = False
        Me.btnNext.Enabled = False
    Else
        Me.btnPrevious.Enabled = True
        Me.btnNext.Enabled = True
    End If
    i = 0
    ds_bind(0)
    cn.Close()
End Sub

Sub ds_bind(ByVal record_no As Integer)
    Me.ID.Text = dt.Rows(record_no)("customer_id")
    Me.txtname2.Text = isSysNull(dt.Rows(record_no)("customer_name"))
    Me.Surname.Text = isSysNull(dt.Rows(record_no)("customer_surname"))
    Me.phone.Text = isSysNull(dt.Rows(record_no)("customer_phone"))
    Me.email.Text = isSysNull(dt.Rows(record_no)("customer_email"))
    Me.address.Text = isSysNull(dt.Rows(record_no)("customer_address"))
    Me.city.SelectedValue = dt.Rows(record_no)("customer_city")
    filltown(Me.city.SelectedValue)
    Me.town.SelectedValue = dt.Rows(record_no)("customer_town")
End Sub
Private Sub btnNext_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnNext.Click
    If i < dt_max Then

```

```

        i = i + 1
        ds_bind(i)
    Else
        If i > 0 Then
            ds_bind(i)
        End If
    End If
End Sub
Private Sub btnPrevious_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnPrevious.Click
    If i > 0 Then
        i = i - 1
        ds_bind(i)
    Else
        If i > 0 Then
            ds_bind(i)
        End If
    End If
End Sub

Private Sub btnNew_Record_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnNew.Click

```

```

    Me.ID.Text = ""
    Me.txtname2.Text = ""
    Me.Surname.Text = ""
    Me.phone.Text = ""
    Me.email.Text = ""
    Me.address.Text = ""
    Me.city.SelectedIndex = -1
    filltown(-1)
    Me.btnPrevious.Enabled = True
    Me.btnNext.Enabled = True
    MsgBox("New Record is Writing")
End Sub

```

```

Private Sub btnSave_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSave.Click
    Dim sql As String
    If Me.ID.Text <> "" Then
        sql = " Update customers set " & _
            " customer_name=" & Me.txtname2.Text & "," & _
            " customer_surname=" & Me.Surname.Text & "," & _
            " customer_email=" & Me.email.Text & "," & _
            " customer_phone=" & Me.phone.Text & "," & _
            " customer_address=" & Me.address.Text & "," & _
            " customer_city=" & Me.city.SelectedValue & "," & _
            " customer_town=" & Me.town.SelectedValue & _
            " where customer_id=" & Me.ID.Text
    Else
        sql = " insert into customers " & _
            "(customer_name,customer_surname,customer_email," & _
            "customer_phone,customer_address,customer_city,customer_town) values " & _
            "(" & Me.txtname2.Text & "," & _
            " " & Me.Surname.Text & "," & _
            " " & Me.email.Text & "," & _
            " " & Me.phone.Text & "," & _
            " " & Me.address.Text & "," & _
            Me.city.SelectedValue & "," & _
            Me.town.SelectedValue & ")"
    End If

    Try
        Dim cn As New SqlConnection(GlobalModule.conn)
        cn.Open()
        Dim cmd As New SqlCommand(sql, cn)
        cmd.ExecuteNonQuery()

        If Me.ID.Text <> "" Then
            Dim da As New SqlDataAdapter
            sql = " Select * from customers order by customer_id"

```

```

        da = New SqlClient.SqlDataAdapter(sql, cn)
        dt.Rows.Clear()
        da.Fill(dt)
        ds_bind(i)
        MsgBox("Record Updated Successfully")
    Else
        Dim da As New SqlClient.SqlDataAdapter
        sql = " Select * from customers order by customer_id"
        da = New SqlClient.SqlDataAdapter(sql, cn)
        dt.Rows.Clear()
        da.Fill(dt)
        dt_max = dt.Rows.Count - 1
        i = dt_max
        ds_bind(dt_max)
        MsgBox("Record Saved Successfully")
        If dt_max < 1 Then
            Me.btnPrevious.Enabled = False
            Me.btnNext.Enabled = False
        Else
            Me.btnPrevious.Enabled = True
            Me.btnNext.Enabled = True
        End If
        End If
        cn.Close()
    Catch ex As Exception
        MsgBox(ex.Message)
    End Try

End Sub

Private Sub btnDelete_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnDelete.Click

    If Me.ID.Text <> "" Then
        Dim sql As String
        sql = " Delete from customers where customer_id=" & ID.Text
        Dim cn As New SqlClient.SqlConnection(GlobalModule.conn)
        cn.Open()
        Dim cmd As New SqlClient.SqlCommand(sql, cn)
        cmd.ExecuteNonQuery()

        Me.ID.Text = ""
        Me.txtname2.Text = ""
        Me.Surname.Text = ""
        Me.phone.Text = ""
        Me.email.Text = ""
        Me.address.Text = ""
        Me.city.SelectedIndex = -1
        filltown(-1)
        MsgBox("Record Deleted Successfully")
        Dim da As New SqlClient.SqlDataAdapter
        sql = " Select * from customers order by customer_id"
        da = New SqlClient.SqlDataAdapter(sql, cn)
        dt.Rows.Clear()
        da.Fill(dt)
        dt_max = dt.Rows.Count - 1
        If dt_max > -1 Then
            i = 0
            ds_bind(0)
        End If
        If dt_max < 1 Then
            Me.btnPrevious.Enabled = False
            Me.btnNext.Enabled = False
        Else
            Me.btnPrevious.Enabled = True
            Me.btnNext.Enabled = True
        End If
        cn.Close()
    End If
End Sub
Sub filltown(ByVal cityid As Integer)
    Dim da_town As New SqlClient.SqlDataAdapter
    Dim dt_town As New DataTable
    Dim cnst As String

```

```

Dim sql As String
' Dim cmd As New SqlClient.SqlCommand
Dim da As New SqlClient.SqlDataAdapter
cnst = GlobalModule.conn
Dim cn As New SqlClient.SqlConnection(cnst)
cn.Open()
sql = " Select town_id,town_name from towns where city_id=" & cityid
da_town = New SqlClient.SqlDataAdapter(sql, cn)
da_town.Fill(dt_town)
Me.town.DataSource = dt_town
Me.town.DisplayMember = "town_name"
Me.town.ValueMember = "town_id"
End Sub

Private Sub city_SelectedIndexChanged(ByVal sender As Object, ByVal e As
System.EventArgs) Handles city.SelectedIndexChanged
Try
If city.SelectedIndex <> -1 Then
filltown(city.SelectedValue)
Else
filltown(-1)
End If
Catch
End Try
End Sub
End Class

```

## Teslimatlar

```

Imports System.Text
Imports System.Drawing.Drawing2D

Public Class deliveries
Inherits System.Windows.Forms.Form

#Region " Windows Form Designer generated code "

Public Sub New()
MyBase.New()

' This call is required by the Windows Form Designer.
InitializeComponent()

' Add any initialization after the InitializeComponent() call

End Sub

' Form overrides dispose to clean up the component list.
Protected Overrides Sub Dispose(ByVal disposing As Boolean)
If disposing Then
If Not (components Is Nothing) Then
components.Dispose()
End If
End If
MyBase.Dispose(disposing)
End Sub

' Required by the Windows Form Designer
Private components As System.ComponentModel.IContainer

' NOTE: The following procedure is required by the Windows Form Designer
' It can be modified using the Windows Form Designer.
' Do not modify it using the code editor.
Friend WithEvents Label5 As System.Windows.Forms.Label
Friend WithEvents Label3 As System.Windows.Forms.Label

```

```

Friend WithEvents Label4 As System.Windows.Forms.Label
Friend WithEvents btnNew As System.Windows.Forms.Button
Friend WithEvents btnPrevious As System.Windows.Forms.Button
Friend WithEvents btnNext As System.Windows.Forms.Button
Friend WithEvents deliveryID As System.Windows.Forms.TextBox
Friend WithEvents Label2 As System.Windows.Forms.Label
Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents btnDelete As System.Windows.Forms.Button
Friend WithEvents btnSave As System.Windows.Forms.Button
Friend WithEvents Label6 As System.Windows.Forms.Label
Friend WithEvents Label7 As System.Windows.Forms.Label
Friend WithEvents Label8 As System.Windows.Forms.Label
Friend WithEvents GroupBox1 As System.Windows.Forms.GroupBox
Friend WithEvents Label9 As System.Windows.Forms.Label
Friend WithEvents Label10 As System.Windows.Forms.Label
Friend WithEvents Label11 As System.Windows.Forms.Label
Friend WithEvents Label12 As System.Windows.Forms.Label
Friend WithEvents Button1 As System.Windows.Forms.Button
Friend WithEvents phone As System.Windows.Forms.TextBox
Friend WithEvents address As System.Windows.Forms.TextBox
Friend WithEvents contactname As System.Windows.Forms.TextBox
Friend WithEvents customerid As System.Windows.Forms.TextBox
Friend WithEvents deliveredperson As System.Windows.Forms.TextBox
Friend WithEvents deliveredDate As System.Windows.Forms.TextBox
Friend WithEvents town As System.Windows.Forms.ComboBox
Friend WithEvents city As System.Windows.Forms.ComboBox
Friend WithEvents status As System.Windows.Forms.ComboBox
Friend WithEvents assignedperson As System.Windows.Forms.ComboBox
Friend WithEvents Label13 As System.Windows.Forms.Label
Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
Friend WithEvents signature As System.Windows.Forms.TextBox
Friend WithEvents Button2 As System.Windows.Forms.Button
<System.Diagnostics.DebuggerStepThrough() Private Sub InitializeComponent()
    Me.phone = New System.Windows.Forms.TextBox
    Me.address = New System.Windows.Forms.TextBox
    Me.contactname = New System.Windows.Forms.TextBox
    Me.Label5 = New System.Windows.Forms.Label
    Me.Label3 = New System.Windows.Forms.Label
    Me.Label4 = New System.Windows.Forms.Label
    Me.btnNew = New System.Windows.Forms.Button
    Me.btnPrevious = New System.Windows.Forms.Button
    Me.btnNext = New System.Windows.Forms.Button
    Me.customerid = New System.Windows.Forms.TextBox
    Me.deliveryID = New System.Windows.Forms.TextBox
    Me.Label2 = New System.Windows.Forms.Label
    Me.Label1 = New System.Windows.Forms.Label
    Me.btnDelete = New System.Windows.Forms.Button
    Me.btnSave = New System.Windows.Forms.Button
    Me.Label6 = New System.Windows.Forms.Label
    Me.Label7 = New System.Windows.Forms.Label
    Me.Label8 = New System.Windows.Forms.Label
    Me.GroupBox1 = New System.Windows.Forms.GroupBox
    Me.signature = New System.Windows.Forms.TextBox
    Me.assignedperson = New System.Windows.Forms.ComboBox
    Me.deliveredDate = New System.Windows.Forms.TextBox
    Me.deliveredperson = New System.Windows.Forms.TextBox
    Me.Label9 = New System.Windows.Forms.Label
    Me.Label10 = New System.Windows.Forms.Label
    Me.Label11 = New System.Windows.Forms.Label
    Me.Label12 = New System.Windows.Forms.Label
    Me.town = New System.Windows.Forms.ComboBox
    Me.city = New System.Windows.Forms.ComboBox
    Me.status = New System.Windows.Forms.ComboBox
    Me.Button1 = New System.Windows.Forms.Button
    Me.Label13 = New System.Windows.Forms.Label
    Me.TextBox1 = New System.Windows.Forms.TextBox
    Me.Button2 = New System.Windows.Forms.Button
    Me.GroupBox1.SuspendLayout()
    Me.SuspendLayout()
    '
    'phone
    '
    Me.phone.Location = New System.Drawing.Point(128, 176)
    Me.phone.Name = "phone"

```

```

Me.phone.TabIndex = 38
Me.phone.Text = ""
'
'address
'
Me.address.Location = New System.Drawing.Point(128, 104)
Me.address.Name = "address"
Me.address.TabIndex = 37
Me.address.Text = ""
'
'contactname
'
Me.contactname.Location = New System.Drawing.Point(128, 80)
Me.contactname.Name = "contactname"
Me.contactname.TabIndex = 36
Me.contactname.Text = ""
'
'Label5
'
Me.Label5.Location = New System.Drawing.Point(32, 128)
Me.Label5.Name = "Label5"
Me.Label5.TabIndex = 35
Me.Label5.Text = "City"
'
'Label3
'
Me.Label3.Location = New System.Drawing.Point(32, 104)
Me.Label3.Name = "Label3"
Me.Label3.TabIndex = 34
Me.Label3.Text = "Address"
'
'Label4
'
Me.Label4.Location = New System.Drawing.Point(32, 80)
Me.Label4.Name = "Label4"
Me.Label4.TabIndex = 33
Me.Label4.Text = "Contact Name"
'
'btnNew
'
Me.btnNew.Location = New System.Drawing.Point(488, 120)
Me.btnNew.Name = "btnNew"
Me.btnNew.TabIndex = 32
Me.btnNew.Text = "New"
'
'btnPrevious
'
Me.btnPrevious.Location = New System.Drawing.Point(432, 24)
Me.btnPrevious.Name = "btnPrevious"
Me.btnPrevious.TabIndex = 31
Me.btnPrevious.Text = "<<"
'
'btnNext
'
Me.btnNext.Location = New System.Drawing.Point(512, 24)
Me.btnNext.Name = "btnNext"
Me.btnNext.TabIndex = 30
Me.btnNext.Text = ">>"
'
'customerid
'
Me.customerid.Location = New System.Drawing.Point(128, 56)
Me.customerid.Name = "customerid"
Me.customerid.Size = New System.Drawing.Size(64, 20)
Me.customerid.TabIndex = 29
Me.customerid.Text = ""
'
'deliveryID
'
Me.deliveryID.Location = New System.Drawing.Point(128, 32)
Me.deliveryID.Name = "deliveryID"
Me.deliveryID.ReadOnly = True
Me.deliveryID.TabIndex = 28
Me.deliveryID.Text = ""

```

```

'
'Label2
'
Me.Label2.Location = New System.Drawing.Point(32, 56)
Me.Label2.Name = "Label2"
Me.Label2.Size = New System.Drawing.Size(112, 23)
Me.Label2.TabIndex = 27
Me.Label2.Text = "Customer ID/Name"
'
'Label1
'
Me.Label1.Location = New System.Drawing.Point(32, 32)
Me.Label1.Name = "Label1"
Me.Label1.TabIndex = 26
Me.Label1.Text = "Delivery ID"
'
'btnDelete
'
Me.btnDelete.Location = New System.Drawing.Point(488, 168)
Me.btnDelete.Name = "btnDelete"
Me.btnDelete.TabIndex = 25
Me.btnDelete.Text = "Delete"
'
'btnSave
'
Me.btnSave.Location = New System.Drawing.Point(488, 144)
Me.btnSave.Name = "btnSave"
Me.btnSave.TabIndex = 24
Me.btnSave.Text = "Save"
'
'Label6
'
Me.Label6.Location = New System.Drawing.Point(32, 184)
Me.Label6.Name = "Label6"
Me.Label6.Size = New System.Drawing.Size(96, 23)
Me.Label6.TabIndex = 40
Me.Label6.Text = "Phone"
'
'Label7
'
Me.Label7.Location = New System.Drawing.Point(32, 160)
Me.Label7.Name = "Label7"
Me.Label7.Size = New System.Drawing.Size(96, 23)
Me.Label7.TabIndex = 39
Me.Label7.Text = "Town"
'
'Label8
'
Me.Label8.Location = New System.Drawing.Point(32, 208)
Me.Label8.Name = "Label8"
Me.Label8.TabIndex = 41
Me.Label8.Text = "Status"
'
'GroupBox1
'
Me.GroupBox1.Controls.Add(Me.signature)
Me.GroupBox1.Controls.Add(Me.assignedperson)
Me.GroupBox1.Controls.Add(Me.deliveredDate)
Me.GroupBox1.Controls.Add(Me.deliveredperson)
Me.GroupBox1.Controls.Add(Me.Label9)
Me.GroupBox1.Controls.Add(Me.Label10)
Me.GroupBox1.Controls.Add(Me.Label11)
Me.GroupBox1.Controls.Add(Me.Label12)
Me.GroupBox1.Location = New System.Drawing.Point(24, 240)
Me.GroupBox1.Name = "GroupBox1"
Me.GroupBox1.Size = New System.Drawing.Size(528, 136)
Me.GroupBox1.TabIndex = 42
Me.GroupBox1.TabStop = False
Me.GroupBox1.Text = "Delivery Information"
'
'signature
'
Me.signature.Enabled = False
Me.signature.Location = New System.Drawing.Point(120, 96)

```

```

Me.signature.Name = "signature"
Me.signature.Size = New System.Drawing.Size(368, 20)
Me.signature.TabIndex = 50
Me.signature.Text = ""
'
'assignedperson
'
Me.assignedperson.Enabled = False
Me.assignedperson.Location = New System.Drawing.Point(120, 24)
Me.assignedperson.Name = "assignedperson"
Me.assignedperson.Size = New System.Drawing.Size(121, 21)
Me.assignedperson.TabIndex = 48
'
'deliveredDate
'
Me.deliveredDate.Enabled = False
Me.deliveredDate.Location = New System.Drawing.Point(120, 72)
Me.deliveredDate.Name = "deliveredDate"
Me.deliveredDate.TabIndex = 47
Me.deliveredDate.Text = ""
'
'deliveredperson
'
Me.deliveredperson.Enabled = False
Me.deliveredperson.Location = New System.Drawing.Point(120, 48)
Me.deliveredperson.Name = "deliveredperson"
Me.deliveredperson.TabIndex = 46
Me.deliveredperson.Text = ""
'
'Label9
'
Me.Label9.Location = New System.Drawing.Point(16, 72)
Me.Label9.Name = "Label9"
Me.Label9.TabIndex = 45
Me.Label9.Text = "Delivered Date"
'
'Label10
'
Me.Label10.Location = New System.Drawing.Point(16, 96)
Me.Label10.Name = "Label10"
Me.Label10.TabIndex = 44
Me.Label10.Text = "Signature"
'
'Label11
'
Me.Label11.Location = New System.Drawing.Point(16, 48)
Me.Label11.Name = "Label11"
Me.Label11.TabIndex = 43
Me.Label11.Text = "Delivered Person"
'
'Label12
'
Me.Label12.Location = New System.Drawing.Point(16, 24)
Me.Label12.Name = "Label12"
Me.Label12.Size = New System.Drawing.Size(104, 23)
Me.Label12.TabIndex = 42
Me.Label12.Text = "Assigned Employee"
'
'town
'
Me.town.Location = New System.Drawing.Point(128, 152)
Me.town.Name = "town"
Me.town.Size = New System.Drawing.Size(121, 21)
Me.town.TabIndex = 43
'
'city
'
Me.city.Location = New System.Drawing.Point(128, 128)
Me.city.Name = "city"
Me.city.Size = New System.Drawing.Size(121, 21)
Me.city.TabIndex = 44
'
'status
'

```



```

Me.status.Location = New System.Drawing.Point(128, 200)
Me.status.Name = "status"
Me.status.Size = New System.Drawing.Size(121, 21)
Me.status.TabIndex = 45
'
'Button1
'
Me.Button1.Location = New System.Drawing.Point(376, 56)
Me.Button1.Name = "Button1"
Me.Button1.Size = New System.Drawing.Size(32, 23)
Me.Button1.TabIndex = 46
Me.Button1.Text = "Ara"
'
'Label13
'
Me.Label13.Location = New System.Drawing.Point(192, 56)
Me.Label13.Name = "Label13"
Me.Label13.Size = New System.Drawing.Size(8, 23)
Me.Label13.TabIndex = 47
Me.Label13.Text = "/"
'
'TextBox1
'
Me.TextBox1.Location = New System.Drawing.Point(208, 56)
Me.TextBox1.Name = "TextBox1"
Me.TextBox1.Size = New System.Drawing.Size(160, 20)
Me.TextBox1.TabIndex = 48
Me.TextBox1.Text = ""
'
'Button2
'
Me.Button2.Location = New System.Drawing.Point(488, 192)
Me.Button2.Name = "Button2"
Me.Button2.TabIndex = 49
Me.Button2.Text = "Signature"
'
'deliveries
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(616, 406)
Me.Controls.Add(Me.Button2)
Me.Controls.Add(Me.TextBox1)
Me.Controls.Add(Me.Label13)
Me.Controls.Add(Me.Button1)
Me.Controls.Add(Me.status)
Me.Controls.Add(Me.city)
Me.Controls.Add(Me.town)
Me.Controls.Add(Me.GroupBox1)
Me.Controls.Add(Me.Label8)
Me.Controls.Add(Me.Label6)
Me.Controls.Add(Me.Label7)
Me.Controls.Add(Me.phone)
Me.Controls.Add(Me.address)
Me.Controls.Add(Me.contactname)
Me.Controls.Add(Me.Label5)
Me.Controls.Add(Me.Label3)
Me.Controls.Add(Me.Label4)
Me.Controls.Add(Me.btnNew)
Me.Controls.Add(Me.btnPrevious)
Me.Controls.Add(Me.btnNext)
Me.Controls.Add(Me.customerid)
Me.Controls.Add(Me.deliveryID)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.Label11)
Me.Controls.Add(Me.btnDelete)
Me.Controls.Add(Me.btnSave)
Me.Name = "deliveries"
Me.Text = "deliveries"
Me.GroupBox1.ResumeLayout(False)
Me.ResumeLayout(False)

```

End Sub

#End Region

```

Dim dt As New DataTable
Dim dt_max As Integer
Dim ri As Integer
Dim da_city As New SqlClient.SqlDataAdapter
Dim dt_city As New DataTable
Dim da_status As New SqlClient.SqlDataAdapter
Dim dt_status As New DataTable
Dim da_employee As New SqlClient.SqlDataAdapter
Dim dt_employee As New DataTable
Dim datan() As Point
Dim dds As Point()
Dim k1 As Integer

Private Sub deliveries_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    Dim cnst As String
    Dim sql As String
    ' Dim cmd As New SqlClient.SqlCommand
    Dim da As New SqlClient.SqlDataAdapter
    cnst = GlobalModule.conn
    Dim cn As New SqlClient.SqlConnection(cnst)
    cn.Open()
    sql = " Select * from delivery order by deliveryid"
    da = New SqlClient.SqlDataAdapter(sql, cn)
    da.Fill(dt)
    dt_max = dt.Rows.Count - 1
    sql = " Select city_id,city_name from cities"
    da_city = New SqlClient.SqlDataAdapter(sql, cn)
    da_city.Fill(dt_city)
    Me.city.DataSource = dt_city
    Me.city.DisplayMember = "city_name"
    Me.city.ValueMember = "city_id"

    sql = " Select deliverystatus_id,deliverystatus_name from deliverystatus"
    da_status = New SqlClient.SqlDataAdapter(sql, cn)
    da_status.Fill(dt_status)
    Me.status.DataSource = dt_status
    Me.status.DisplayMember = "deliverystatus_name"
    Me.status.ValueMember = "deliverystatus_id"

    sql = " Select userid,name + '-' + surname c from users"
    da_employee = New SqlClient.SqlDataAdapter(sql, cn)
    da_employee.Fill(dt_employee)
    Me.assignedperson.DataSource = dt_employee
    Me.assignedperson.DisplayMember = "c"
    Me.assignedperson.ValueMember = "userid"

    If dt_max < 0 Then
        ri = -1
        cn.Close()
        Exit Sub
    End If
    If dt_max < 1 Then
        Me.btnPrevious.Enabled = False
        Me.btnNext.Enabled = False
    Else
        Me.btnPrevious.Enabled = True
        Me.btnNext.Enabled = True
    End If
    ri = 0
    ds_bind(0)
    cn.Close()

End Sub

Sub ds_bind(ByVal record_no As Integer)
    Me.deliveryID.Text = dt.Rows(record_no)("deliveryID")
    Me.customerid.Text = isSysNull(dt.Rows(record_no)("deliveryCustomer"))
    Me.contactname.Text = isSysNull(dt.Rows(record_no)("deliveryContactName"))
    Me.address.Text = isSysNull(dt.Rows(record_no)("deliveryAddress"))

```

```

Me.phone.Text = isSysNull(dt.Rows(record_no)("deliveryTel"))
Me.deliveredperson.Text = isSysNull(dt.Rows(record_no)("deliveredPerson"))
Me.signature.Text = isSysNull(dt.Rows(record_no)("deliveredSignature"))
Me.deliveredDate.Text = isSysNull(dt.Rows(record_no)("deliveredDate"))
Me.city.SelectedValue = dt.Rows(record_no)("deliveryCity")
filltown(Me.city.SelectedValue)
Me.town.SelectedValue = dt.Rows(record_no)("deliveryTown")
Me.status.SelectedValue = dt.Rows(record_no)("deliveryCheck")
Me.assignedperson.SelectedValue = dt.Rows(record_no)("assignedpersonid")
fillsignature(signature.Text)
End Sub

Private Sub btnNext_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnNext.Click
    If ri < dt_max Then
        ri = ri + 1
        ds_bind(ri)
    Else
        If ri > 0 Then
            ds_bind(ri)
        End If
    End If
End Sub
Private Sub btnPrevious_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnPrevious.Click
    If ri > 0 Then
        ri = ri - 1
        ds_bind(ri)
    Else
        If ri > 0 Then
            ds_bind(ri)
        End If
    End If
End Sub

Private Sub btnNew_Record_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnNew.Click

Me.deliveryID.Text = ""
Me.customerid.Text = ""
Me.contactname.Text = ""
Me.address.Text = ""
Me.phone.Text = ""
Me.deliveredperson.Text = ""
Me.signature.Text = ""
Me.deliveredDate.Text = ""
Me.city.SelectedIndex = -1
filltown(-1)
' Me.town.Selectedindex = -1
Me.status.SelectedIndex = -1
Me.assignedperson.SelectedIndex = -1
Me.btnPrevious.Enabled = True
Me.btnNext.Enabled = True
MsgBox("New Record is Writing")
End Sub

Private Sub btnSave_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSave.Click
Dim sql As String
If Me.deliveryID.Text <> "" Then
    sql = " Update delivery set "
    sql += " deliveryCustomer=" & Me.customerid.Text & ","
    sql += " deliveryContactName='" & contactname.Text & "',"
    sql += " deliveryAddress='" & address.Text & "',"
    sql += " deliveryTel='" & phone.Text & "',"
    sql += " deliveredPerson='" & deliveredperson.Text & "',"
    sql += " deliveryCity=" & city.SelectedValue & ","
    sql += " deliveryTown=" & town.SelectedValue & ","
    sql += " deliveryCheck=" & status.SelectedValue
    sql += " where deliveryid=" & Me.deliveryID.Text
Else
    sql = " insert into delivery"

```

```

        sql += " (deliveryCustomer,deliveryContactName,deliveryAddress,"
        sql += " deliveryTel,deliveredPerson,deliveryCity,"
        sql += " deliveryTown,deliveryCheck,"
        sql += " deliveredSignature,deliveredDate,assignedpersonid)"
        sql += " values "
        sql += " (" & Me.customerid.Text & ",',"
        sql += contactname.Text & " ',',"
        sql += address.Text & " ',',"
        sql += phone.Text & " ',',"
        sql += deliveredperson.Text & " ',',"
        sql += city.SelectedValue & " ',',"
        sql += town.SelectedValue & " ',',"
        sql += status.SelectedValue & " ','0',Null,0)"
    End If
    Dim cn As New SqlClient.SqlConnection(GlobalModule.conn)
    cn.Open()
    Dim cmd As New SqlClient.SqlCommand(sql, cn)
    cmd.ExecuteNonQuery()
    If Me.deliveryID.Text <> "" Then
        Dim da As New SqlClient.SqlDataAdapter
        sql = " Select * from delivery order by deliveryid"
        da = New SqlClient.SqlDataAdapter(sql, cn)
        dt.Rows.Clear()
        da.Fill(dt)
        ds_bind(ri)
        MsgBox("Record Updated Successfully")
    Else
        Dim da As New SqlClient.SqlDataAdapter
        sql = " Select * from delivery order by deliveryid"
        da = New SqlClient.SqlDataAdapter(sql, cn)
        dt.Rows.Clear()
        da.Fill(dt)
        dt_max = dt.Rows.Count - 1
        ri = dt_max
        ds_bind(dt_max)
        MsgBox("Record Saved Successfully")
        If dt_max < 1 Then
            Me.btnPrevious.Enabled = False
            Me.btnNext.Enabled = False
        Else
            Me.btnPrevious.Enabled = True
            Me.btnNext.Enabled = True
        End If
    End If
    cn.Close()
End Sub

Sub filltown(ByVal cityid As Integer)
    Dim da_town As New SqlClient.SqlDataAdapter
    Dim dt_town As New DataTable
    Dim cnst As String
    Dim sql As String
    ' Dim cmd As New SqlClient.SqlCommand
    Dim da As New SqlClient.SqlDataAdapter
    cnst = GlobalModule.conn
    Dim cn As New SqlClient.SqlConnection(cnst)
    cn.Open()
    sql = " Select town_id,town_name from towns where city_id=" & cityid
    da_town = New SqlClient.SqlDataAdapter(sql, cn)
    da_town.Fill(dt_town)
    Me.town.DataSource = dt_town
    Me.town.DisplayMember = "town_name"
    Me.town.ValueMember = "town_id"
End Sub

Private Sub city_SelectedIndexChanged(ByVal sender As Object, ByVal e As
System.EventArgs) Handles city.SelectedIndexChanged
    Try
        If city.SelectedIndex <> -1 Then
            filltown(city.SelectedValue)
        Else
            filltown(-1)
        End If
    End Try

```

```

    Catch
    End Try
End Sub

Sub fillsignature(ByVal tline As String)
    Dim k, i, j, il, n, t As Integer
    Dim _signature As SignatureData
    ' k = Len(tline) - 1

    Dim s As String()
    Dim r As String()
    Dim p As String
    Dim tum As String
    tum = tline
    Dim datan() As Point
    s = Split(tum, "points")
    k = s(0).IndexOf("(")
    p = Mid(s(0), k + 2)
    il = CInt(p) - 1
    ReDim dds(il)
    Try
        tline = Mid(s(1), 2)
        r = Split(tline, "(")
        j = r.Length - 2
        For i = 0 To j
            p = Replace(r(i + 1), ")", "")
            p = Replace(p, "(", "")
            s = Split(p, ",")
            Dim fd As New Point
            fd.X = CInt(s(0))
            fd.Y = CInt(s(1))

            dds(i) = fd

        Next
        'k = tline.IndexOf("(")
        'Do While k <> -1
        '    j = tline.IndexOf(")")
        '    p = Mid(tline, k + 1)

        '    s = Split(p, ",")
        '    datan(t).X = CInt(s(0))
        '    datan(t).Y = CInt(s(1))
        '    tline = Mid(tline, j)
        '    t = t + 1
        'Loop

        's = Split(tline, ",")
        'k = s.Length - 2
        'Dim data As Byte()
        'ReDim data(k + 1)
        'For j = 0 To k
        '    ' data(j) = Asc(Mid(tline, j + 1, 1))
        '    data(j) = s(j)
        '    ' data(j) = fs.ReadByte
        'Next

        'il = 0
        'j = 0
        'n = k / 2
        'ReDim datan(n)
        'Do While il < k
        '    datan(j).X = s(il)
        '    datan(j).Y = s(il + 1)
        '    j = j + 1
        '    il = il + 2
        'Loop

        '_signature = New SignatureData(data)
        '    pictSignature.Invalidate()

        ' update the segments view
    
```

```

        'Dim sb As New StringBuilder
        'sb.AppendFormat("Number of segments : {0}", _signature.Lines.Count)

        ' loop through each segment and build up string
        'Dim i As Integer
        'For i = 0 To _signature.Lines.Count - 1
        '    Dim points As Point() = CType(_signature.Lines(i), Point())
        '    sb.AppendFormat(ControlChars.Cr + ControlChars.Lf + ControlChars.Cr +
ControlChars.Lf + _
        '    "Segment {0} ({1} points)" + ControlChars.Cr + ControlChars.Lf, i + 1,
points.Length)

        '    Dim pt As Point
        '    For Each pt In points
        '        sb.AppendFormat("({0},{1}) ", pt.X, pt.Y)
        '    Next pt
        'Next i

        'textPoints.Text = sb.ToString()

    Catch ex As Exception
        ' something went wrong, this happens if the
        ' wrong encryption passphrase is used
        Dim msg As String = "The signature data could not be decrypted. Additional
information:" + _
            ControlChars.Cr + ControlChars.Lf + ControlChars.Cr + ControlChars.Lf +
ex.Message

        ' display error message
        ' textPoints.Text = msg
        ' textDecrypt.Text = msg

        ' clear signature
        _signature = Nothing
        ' pictSignature.Invalidate()
    End Try
End Sub

Private Sub pictSignature_Paint(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.PaintEventArgs)
    ' draw the signature segments

    DrawSignature(e.Graphics, datan)
End Sub

Private Sub DrawSignature(ByVal g As Graphics, ByVal pot() As Point)

    ' background
    k1 = k1 + 1
    If k1 > 2 Then
        Exit Sub
    End If
    g.Clear(Color.Cornsilk)

    ' border
    ' g.DrawRectangle(Pens.Gray, 0, 0, pictSignature.Width - 1,
pictSignature.Height - 1)

    Dim il As Integer
    il = dds.Length - 1
    ' ReDim datan(il)
    Try
        ' datan = dds
        g.DrawLines(Pens.Firebrick, dds)
        'Dim b As System.Drawing.Image
        'b = PictureBox1.Image
        'Dim tt As Integer
        'For tt = 0 To 20
        '    ' g.DrawImage(b, dds(tt).X, dds(tt).Y)
        '    g.DrawImage(b, tt, tt)
        'Next

        ' g.DrawLines(Pens.Firebrick, dds)
        ' g.DrawImage(b, dds(0).X, dds(0).Y)
    End Try

```

```

        ' g.DrawIcon(b, dds(0).X, dds(0).Y)
    Catch ex As Exception
        MsgBox(ex.Message)
    End Try

    ' return if don't have a signature
    'If _signature Is Nothing Then
    '    Return
    'End If

    'If _signature.Width = 0 Or _signature.Height = 0 Then
    '    Return
    'End If

    '' setup drawing surface
    'g.SmoothingMode = SmoothingMode.AntiAlias

    '' scale the signature
    'Dim matrix As New Matrix
    'matrix.Scale( _
    '    CSng(pictSignature.Width) / CSng(_signature.Width), _
    '    CSng(pictSignature.Height) / CSng(_signature.Height))
    'g.Transform = matrix

    '' draw each line segment
    'Dim line As Point()
    'For Each line In _signature.Lines
    '    If Not line Is Nothing Then
    '        If line.Length > 0 Then
    '            ' draw lines or curves
    '            ' If checkSmooth.Checked Then
    '                ' g.DrawCurve(Pens.Firebrick, line, 0.5F)
    '            'Else
    '                g.DrawLines(Pens.Firebrick, line)
    '                g.DrawLines()
    '            End If
    '        End If
    '    End If
    'Next line
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    Dim pict As New picta
    pict.a = dds
    pict.ShowDialog()
End Sub
End Class

```

## Çalışanlar

```

Public Class employees
    Inherits System.Windows.Forms.Form

    #Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then

```

```

        components.Dispose()
    End If
End If
MyBase.Dispose(disposing)
End Sub

'Required by the Windows Form Designer
Private components As System.ComponentModel.IContainer

'NOTE: The following procedure is required by the Windows Form Designer
'It can be modified using the Windows Form Designer.
'Do not modify it using the code editor.
Friend WithEvents btnNew As System.Windows.Forms.Button
Friend WithEvents btnPrevious As System.Windows.Forms.Button
Friend WithEvents btnNext As System.Windows.Forms.Button
Friend WithEvents Username As System.Windows.Forms.TextBox
Friend WithEvents ID As System.Windows.Forms.TextBox
Friend WithEvents Label2 As System.Windows.Forms.Label
Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents btnDelete As System.Windows.Forms.Button
Friend WithEvents btnSave As System.Windows.Forms.Button
Friend WithEvents Label3 As System.Windows.Forms.Label
Friend WithEvents Label4 As System.Windows.Forms.Label
Friend WithEvents Label5 As System.Windows.Forms.Label
Friend WithEvents pass As System.Windows.Forms.TextBox
Friend WithEvents txtusername As System.Windows.Forms.TextBox
Friend WithEvents surname As System.Windows.Forms.TextBox
<System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
    Me.btnNew = New System.Windows.Forms.Button
    Me.btnPrevious = New System.Windows.Forms.Button
    Me.btnNext = New System.Windows.Forms.Button
    Me.Username = New System.Windows.Forms.TextBox
    Me.ID = New System.Windows.Forms.TextBox
    Me.Label2 = New System.Windows.Forms.Label
    Me.Label1 = New System.Windows.Forms.Label
    Me.btnDelete = New System.Windows.Forms.Button
    Me.btnSave = New System.Windows.Forms.Button
    Me.Label3 = New System.Windows.Forms.Label
    Me.Label4 = New System.Windows.Forms.Label
    Me.Label5 = New System.Windows.Forms.Label
    Me.pass = New System.Windows.Forms.TextBox
    Me.txtusername = New System.Windows.Forms.TextBox
    Me.surname = New System.Windows.Forms.TextBox
    Me.SuspendLayout()
    '
    'btnNew
    '
    Me.btnNew.Location = New System.Drawing.Point(336, 176)
    Me.btnNew.Name = "btnNew"
    Me.btnNew.TabIndex = 17
    Me.btnNew.Text = "New"
    '
    'btnPrevious
    '
    Me.btnPrevious.Location = New System.Drawing.Point(240, 32)
    Me.btnPrevious.Name = "btnPrevious"
    Me.btnPrevious.TabIndex = 16
    Me.btnPrevious.Text = "<<"
    '
    'btnNext
    '
    Me.btnNext.Location = New System.Drawing.Point(320, 32)
    Me.btnNext.Name = "btnNext"
    Me.btnNext.TabIndex = 15
    Me.btnNext.Text = ">>"
    '
    'Username
    '
    Me.Username.Location = New System.Drawing.Point(160, 112)
    Me.Username.Name = "Username"
    Me.Username.TabIndex = 14
    Me.Username.Text = ""
    '
    'ID

```



```

'
Me.ID.Location = New System.Drawing.Point(160, 88)
Me.ID.Name = "ID"
Me.ID.ReadOnly = True
Me.ID.TabIndex = 13
Me.ID.Text = ""
'
'Label2
'
Me.Label2.Location = New System.Drawing.Point(64, 112)
Me.Label2.Name = "Label2"
Me.Label2.TabIndex = 12
Me.Label2.Text = "User Name"
'
'Label1
'
Me.Label1.Location = New System.Drawing.Point(64, 88)
Me.Label1.Name = "Label1"
Me.Label1.TabIndex = 11
Me.Label1.Text = "User ID"
'
'btnDelete
'
Me.btnDelete.Location = New System.Drawing.Point(336, 224)
Me.btnDelete.Name = "btnDelete"
Me.btnDelete.TabIndex = 10
Me.btnDelete.Text = "Delete"
'
'btnSave
'
Me.btnSave.Location = New System.Drawing.Point(336, 200)
Me.btnSave.Name = "btnSave"
Me.btnSave.TabIndex = 9
Me.btnSave.Text = "Save"
'
'Label3
'
Me.Label3.Location = New System.Drawing.Point(64, 160)
Me.Label3.Name = "Label3"
Me.Label3.TabIndex = 19
Me.Label3.Text = "Name"
'
'Label4
'
Me.Label4.Location = New System.Drawing.Point(64, 136)
Me.Label4.Name = "Label4"
Me.Label4.TabIndex = 18
Me.Label4.Text = "Password"
'
'Label5
'
Me.Label5.Location = New System.Drawing.Point(64, 184)
Me.Label5.Name = "Label5"
Me.Label5.TabIndex = 20
Me.Label5.Text = "Surname"
'
'pass
'
Me.pass.Location = New System.Drawing.Point(160, 136)
Me.pass.Name = "pass"
Me.pass.TabIndex = 21
Me.pass.Text = ""
'
'txtusername
'
Me.txtusername.Location = New System.Drawing.Point(160, 160)
Me.txtusername.Name = "txtusername"
Me.txtusername.TabIndex = 22
Me.txtusername.Text = ""
'
'surname
'
Me.surname.Location = New System.Drawing.Point(160, 184)
Me.surname.Name = "surname"

```

```

Me.surname.TabIndex = 23
Me.surname.Text = ""
'
'employees
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(480, 342)
Me.Controls.Add(Me.surname)
Me.Controls.Add(Me.txtusername)
Me.Controls.Add(Me.pass)
Me.Controls.Add(Me.Label5)
Me.Controls.Add(Me.Label3)
Me.Controls.Add(Me.Label4)
Me.Controls.Add(Me.btnNew)
Me.Controls.Add(Me.btnPrevious)
Me.Controls.Add(Me.btnNext)
Me.Controls.Add(Me.Username)
Me.Controls.Add(Me.ID)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.Label1)
Me.Controls.Add(Me.btnDelete)
Me.Controls.Add(Me.btnSave)
Me.Name = "employees"
Me.Text = "employees"
Me.ResumeLayout(False)

End Sub

#End Region
Dim dt As New DataTable
Dim dt_max As Integer
Dim i As Integer
Private Sub employees_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

    Dim cnst As String
    Dim sql As String
    ' Dim cmd As New SqlCommand
    Dim da As New SqlDataAdapter
    cnst = GlobalModule.conn
    Dim cn As New SqlConnection(cnst)
    cn.Open()
    sql = " Select * from users order by userid"
    da = New SqlDataAdapter(sql, cn)
    da.Fill(dt)
    dt_max = dt.Rows.Count - 1
    If dt_max < 0 Then
        i = -1
        cn.Close()
        Exit Sub
    End If
    If dt_max < 1 Then
        Me.btnPrevious.Enabled = False
        Me.btnNext.Enabled = False
    Else
        Me.btnPrevious.Enabled = True
        Me.btnNext.Enabled = True
    End If
    i = 0
    ds_bind(0)
    cn.Close()
End Sub

Sub ds_bind(ByVal record_no As Integer)
    Me.ID.Text = dt.Rows(record_no)("userid")
    Me.Username.Text = isSysNull(dt.Rows(record_no)("username"))
    Me.pass.Text = isSysNull(dt.Rows(record_no)("password"))
    Me.txtusername.Text = isSysNull(dt.Rows(record_no)("name"))
    Me.surname.Text = isSysNull(dt.Rows(record_no)("surname"))
End Sub
Private Sub btnNext_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnNext.Click
    If i < dt_max Then
        i = i + 1

```

```

        ds_bind(i)
    Else
        If i > 0 Then
            ds_bind(i)
        End If
    End If
End Sub
Private Sub btnPrevious_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnPrevious.Click
    If i > 0 Then
        i = i - 1
        ds_bind(i)
    Else
        If i > 0 Then
            ds_bind(i)
        End If
    End If
End Sub
Private Sub btnNew_Record_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnNew.Click
    Me.ID.Text = ""
    Me.Username.Text = ""
    Me.pass.Text = ""
    Me.txtusername.Text = ""
    Me.surname.Text = ""
    Me.btnPrevious.Enabled = True
    Me.btnNext.Enabled = True
    MsgBox("New Record is Writing")
End Sub

Private Sub btnSave_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSave.Click
    Dim sql As String
    If Me.ID.Text <> "" Then
        sql = " Update users set" & _
            " username='" & Me.Username.Text & "'," & _
            " password='" & Me.pass.Text & "'," & _
            " name='" & Me.txtusername.Text & "'," & _
            " surname='" & Me.surname.Text & "'" & _
            " where userid=" & Me.ID.Text
    Else
        sql = " insert into users (username,password,name,surname) values " & _
            "(" & Me.Username.Text & "','" & _
            Me.pass.Text & "','" & _
            Me.txtusername.Text & "','" & _
            Me.surname.Text & "'" & _
    End If

    Dim cn As New SqlClient.SqlConnection(GlobalModule.conn)
    cn.Open()
    Dim cmd As New SqlClient.SqlCommand(sql, cn)
    cmd.ExecuteNonQuery()
    If Me.ID.Text <> "" Then
        Dim da As New SqlClient.SqlDataAdapter
        sql = " Select * from users order by userid"
        da = New SqlClient.SqlDataAdapter(sql, cn)
        dt.Rows.Clear()
        da.Fill(dt)
        ds_bind(i)
        MsgBox("Record Updated Successfully")
    Else
        Dim da As New SqlClient.SqlDataAdapter
        sql = " Select * from users order by userid"
        da = New SqlClient.SqlDataAdapter(sql, cn)
        dt.Rows.Clear()
        da.Fill(dt)
        dt_max = dt.Rows.Count - 1
        i = dt_max
        ds_bind(dt_max)
        MsgBox("Record Saved Successfully")
        If dt_max < 1 Then
            Me.btnPrevious.Enabled = False
            Me.btnNext.Enabled = False
        End If
    End If
End Sub

```

```

        Else
            Me.btnPrevious.Enabled = True
            Me.btnNext.Enabled = True
        End If
    End If
    End If
    cn.Close()
End Sub

Private Sub btnDelete_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnDelete.Click

    If Me.ID.Text <> "" Then
        Dim sql As String
        sql = " Delete from users where userid=" & ID.Text
        Dim cn As New SqlClient.SqlConnection(GlobalModule.conn)
        cn.Open()
        Dim cmd As New SqlClient.SqlCommand(sql, cn)
        cmd.ExecuteNonQuery()

        Me.ID.Text = ""
        Me.Username.Text = ""
        Me.pass.Text = ""
        Me.txtusername.Text = ""
        Me.surname.Text = ""

        MsgBox("Record Deleted Successfully")
        Dim da As New SqlClient.SqlDataAdapter
        sql = " Select * from users order by userid"
        da = New SqlClient.SqlDataAdapter(sql, cn)
        dt.Rows.Clear()
        da.Fill(dt)
        dt_max = dt.Rows.Count - 1
        If dt_max > -1 Then
            i = 0
            ds_bind(0)
        End If
        If dt_max < 1 Then
            Me.btnPrevious.Enabled = False
            Me.btnNext.Enabled = False
        Else
            Me.btnPrevious.Enabled = True
            Me.btnNext.Enabled = True
        End If
        cn.Close()
    End If
End Sub
End Class

```

## İmza Ekranı

```

Imports System.Text
Imports System.Drawing.Drawing2D

Public Class picta
    Inherits System.Windows.Forms.Form
    Public a As Point()
    #Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)

```

```

        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents PictureBox1 As System.Windows.Forms.PictureBox
    <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
        Me.PictureBox1 = New System.Windows.Forms.PictureBox
        Me.SuspendLayout()
        '
        'PictureBox1
        '
        Me.PictureBox1.BackColor = System.Drawing.SystemColors.Highlight
        Me.PictureBox1.Location = New System.Drawing.Point(32, 56)
        Me.PictureBox1.Name = "PictureBox1"
        Me.PictureBox1.Size = New System.Drawing.Size(360, 216)
        Me.PictureBox1.TabIndex = 52
        Me.PictureBox1.TabStop = False
        '
        'picta
        '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
        Me.ClientSize = New System.Drawing.Size(424, 294)
        Me.Controls.Add(Me.PictureBox1)
        Me.Name = "picta"
        Me.Text = "Signature"
        Me.ResumeLayout(False)

    End Sub

#End Region

    Dim datan() As Point
    Dim dds As Point()
    Dim k1 As Integer

    Private Sub picta_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
        Dim tline As String
        ' Dim _signature As SignatureData
        dds = a
    End Sub
    Private Sub pictSignature_Paint(ByVal sender As System.Object, ByVal e As
System.Windows.Forms.PaintEventArgs) Handles PictureBox1.Paint
        ' draw the signature segments

        DrawSignature(e.Graphics, dds)
    End Sub

    Private Sub DrawSignature(ByVal g As Graphics, ByVal pot() As Point)
        ' background
        k1 = k1 + 1
        If k1 > 2 Then
            Exit Sub
        End If
        g.Clear(Color.Cornsilk)
        ' border
        g.DrawRectangle(Pens.Gray, 0, 0, Me.PictureBox1.Width - 1, PictureBox1.Height -
1)

        Dim il As Integer
        il = dds.Length
        ReDim datan(il)
        datan = dds
        g.DrawLines(Pens.Firebrick, dds)
    End Sub

```

```
End Sub
End Class
```

## Cep Bilgisayarı Çözümü

### Giriş Ekranı

```
Public Class login
    Inherits System.Windows.Forms.Form
    Friend WithEvents Button1 As System.Windows.Forms.Button
    Friend WithEvents Button2 As System.Windows.Forms.Button
    Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
    Friend WithEvents TextBox2 As System.Windows.Forms.TextBox
    Friend WithEvents Label1 As System.Windows.Forms.Label
    Friend WithEvents Label2 As System.Windows.Forms.Label
    Friend WithEvents MainMenu1 As System.Windows.Forms.MainMenu
    Dim cmd As System.Data.SqlClient.SqlCommand
    Dim cn As System.Data.SqlClient.SqlConnection
    Dim SQL As String
    Dim kt As Integer
    Dim SQLEngine As System.Data.SqlClient.SqlCeEngine
    Dim dtr As System.Data.SqlClient.SqlCeDataReader

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As Boolean)
        MyBase.Dispose(disposing)
    End Sub

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Private Sub InitializeComponent()
        Me.MainMenu1 = New System.Windows.Forms.MainMenu
        Me.Button1 = New System.Windows.Forms.Button
        Me.Button2 = New System.Windows.Forms.Button
        Me.TextBox1 = New System.Windows.Forms.TextBox
        Me.TextBox2 = New System.Windows.Forms.TextBox
        Me.Label1 = New System.Windows.Forms.Label
        Me.Label2 = New System.Windows.Forms.Label
        '
        'Button1
        '
        Me.Button1.Location = New System.Drawing.Point(56, 136)
        Me.Button1.Size = New System.Drawing.Size(72, 24)
        Me.Button1.Text = "Login"
        '
        'Button2
        '
        Me.Button2.Location = New System.Drawing.Point(144, 136)
        Me.Button2.Size = New System.Drawing.Size(72, 24)
        Me.Button2.Text = "Exit"
        '
    End Sub
End Class
```

```

        'TextBox1
    ,
    Me.TextBox1.Location = New System.Drawing.Point(112, 40)
    Me.TextBox1.Text = ""
    ,
    'TextBox2
    ,
    Me.TextBox2.Location = New System.Drawing.Point(112, 72)
    Me.TextBox2.PasswordChar = Microsoft.VisualBasic.ChrW(42)
    Me.TextBox2.Text = ""
    ,
    'Label1
    ,
    Me.Label1.Location = New System.Drawing.Point(16, 40)
    Me.Label1.Size = New System.Drawing.Size(80, 20)
    Me.Label1.Text = "User Name :"
    ,
    'Label2
    ,
    Me.Label2.Location = New System.Drawing.Point(16, 72)
    Me.Label2.Size = New System.Drawing.Size(80, 20)
    Me.Label2.Text = "Password :"
    ,
    'login
    ,
    Me.Controls.Add(Me.Label2)
    Me.Controls.Add(Me.Label1)
    Me.Controls.Add(Me.TextBox2)
    Me.Controls.Add(Me.TextBox1)
    Me.Controls.Add(Me.Button2)
    Me.Controls.Add(Me.Button1)
    Me.Menu = Me.MainMenu1
    Me.Text = "login"

End Sub

#End Region

Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
    database()
End Sub
Private Sub database()
    cn = New System.Data.SqlClient.SqlCeConnection(connection_string)
    Try
        If Not System.IO.File.Exists(db_string) Then
            SQLEngine = New System.Data.SqlClient.SqlCeEngine(connection_string)
            SQLEngine.CreateDatabase()
            cn.Open()
            SQL = "CREATE TABLE users (username nvarchar(10) Primary Key, password
nvarchar (10),namesurname nvarchar (60))"
            Dim cmd As New System.Data.SqlClient.SqlCeCommand(SQL, cn)
            cmd.CommandType = CommandType.Text
            cmd.ExecuteNonQuery()
            SQL = "CREATE TABLE delivery (deliveryID int Primary
Key,deliveryContactName nvarchar(50), deliveryAddress nvarchar(50),"
            SQL = SQL + "deliverytown nvarchar(50), deliverycity nvarchar(50),
deliverytel nvarchar(50),deliveredperson nvarchar(50), deliveredsignature
ntext,delivereddate nvarchar(50), deliverycheck int)"
            cmd.CommandText = SQL
            cmd.ExecuteNonQuery()
            'users (username nvarchar(10) Primary Key, password nvarchar (10),
plakano nvarchar(15))
            SQL = "INSERT INTO users (username, password,namesurname"
            SQL = SQL + ") VALUES ("
            SQL = SQL + "'1','1','Yildirim Kocdag')"
            cmd.CommandText = SQL
            cmd.ExecuteNonQuery()

            ' delivery (deliveryID int Primary Key,deliveryContactName nvarchar(50),
deliveryAddress nvarchar(50),"
            ' deliverytown nvarchar(50), deliverycity nvarchar(50), deliverytel
nvarchar(50),deliveredperson nvarchar(50), deliveredsignature
nvarchar(250),delivereddate nvarchar(50), deliverycheck int)"

```

```

        SQL = "INSERT INTO delivery (deliveryID, deliveryContactName,
deliveryAddress, deliverytown, deliverycity, deliverytel,deliveredperson,
deliveredsignature, delivereddate, deliverycheck)"
        SQL = SQL + " VALUES ("
        SQL = SQL + "1,'Kaan Kocdag','Inonu cad.No
4','istanbul','maltepe','02163052460','0','0','0',0)"
        cmd.CommandText = SQL
        cmd.ExecuteNonQuery()
        cn.Close()
    End If
Catch
    MsgBox("Database Error")
End Try
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    ShowWaitCursor(True)
    cn = New System.Data.SqlClient.SqlCeConnection(connection_string)
    cn.Open()
    SQL = "Select * from users where (username='" + Trim(TextBox1.Text) + "') AND ("
    SQL = SQL + "password='" + Trim(TextBox2.Text) + "')"
    cmd = New System.Data.SqlClient.SqlCeCommand(SQL, cn)
    dtr = cmd.ExecuteReader
    If dtr.Read Then
        usernamesurname = dtr("namesurname")
        dtr.Close()
        cn.Close()
        Dim frm As New mainmenu
        frm.ShowDialog()
        Me.Close()
    Else
        MsgBox("Username or password is incorrect!")
        TextBox1.Text = ""
        TextBox2.Text = ""
        dtr.Close()
        cn.Close()
    End If
    ShowWaitCursor(False)
End Sub
End Class

```

## Teslimatlar

```

Public Class frmdelivery
    Inherits System.Windows.Forms.Form
    Dim cmd As System.Data.SqlClient.SqlCeCommand
    Dim cn As System.Data.SqlClient.SqlCeConnection
    Dim SQL As String
    Dim dt_i As Integer
    Dim dt_max As Integer
    Dim da As System.Data.SqlClient.SqlCeDataAdapter
    Dim dt As New System.Data.DataTable
    #Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    Protected Overrides Sub Dispose(ByVal disposing As Boolean)
        MyBase.Dispose(disposing)
    End Sub

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.

```



```

'Do not modify it using the code editor.
Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents Label2 As System.Windows.Forms.Label
Friend WithEvents Label3 As System.Windows.Forms.Label
Friend WithEvents Label4 As System.Windows.Forms.Label
Friend WithEvents Label5 As System.Windows.Forms.Label
Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
Friend WithEvents TextBox2 As System.Windows.Forms.TextBox
Friend WithEvents TextBox3 As System.Windows.Forms.TextBox
Friend WithEvents TextBox4 As System.Windows.Forms.TextBox
Friend WithEvents TextBox5 As System.Windows.Forms.TextBox
Friend WithEvents Label6 As System.Windows.Forms.Label
Friend WithEvents TextBox6 As System.Windows.Forms.TextBox
Friend WithEvents Button1 As System.Windows.Forms.Button
Friend WithEvents Button2 As System.Windows.Forms.Button
Friend WithEvents MainMenu1 As System.Windows.Forms.MainMenu
Friend WithEvents MenuItem1 As System.Windows.Forms.MenuItem
Friend WithEvents MenuItem2 As System.Windows.Forms.MenuItem
Friend WithEvents MenuItem3 As System.Windows.Forms.MenuItem
Friend WithEvents MenuItem7 As System.Windows.Forms.MenuItem
Friend WithEvents MenuItem4 As System.Windows.Forms.MenuItem
Friend WithEvents MenuItem5 As System.Windows.Forms.MenuItem
Friend WithEvents MenuItem6 As System.Windows.Forms.MenuItem
Friend WithEvents Button3 As System.Windows.Forms.Button
<System.Diagnostics.DebuggerStepThrough() Private Sub InitializeComponent()
    Me.Label1 = New System.Windows.Forms.Label
    Me.Label2 = New System.Windows.Forms.Label
    Me.Label3 = New System.Windows.Forms.Label
    Me.Label4 = New System.Windows.Forms.Label
    Me.Label5 = New System.Windows.Forms.Label
    Me.TextBox1 = New System.Windows.Forms.TextBox
    Me.TextBox2 = New System.Windows.Forms.TextBox
    Me.TextBox3 = New System.Windows.Forms.TextBox
    Me.TextBox4 = New System.Windows.Forms.TextBox
    Me.TextBox5 = New System.Windows.Forms.TextBox
    Me.Label6 = New System.Windows.Forms.Label
    Me.TextBox6 = New System.Windows.Forms.TextBox
    Me.Button1 = New System.Windows.Forms.Button
    Me.Button2 = New System.Windows.Forms.Button
    Me.MainMenu1 = New System.Windows.Forms.MainMenu
    Me.MenuItem1 = New System.Windows.Forms.MenuItem
    Me.MenuItem2 = New System.Windows.Forms.MenuItem
    Me.MenuItem3 = New System.Windows.Forms.MenuItem
    Me.MenuItem7 = New System.Windows.Forms.MenuItem
    Me.MenuItem4 = New System.Windows.Forms.MenuItem
    Me.MenuItem5 = New System.Windows.Forms.MenuItem
    Me.MenuItem6 = New System.Windows.Forms.MenuItem
    Me.Button3 = New System.Windows.Forms.Button
    '
    'Label1
    '
    Me.Label1.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
    Me.Label1.Location = New System.Drawing.Point(8, 56)
    Me.Label1.Text = "Name Surname"
    '
    'Label2
    '
    Me.Label2.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
    Me.Label2.Location = New System.Drawing.Point(8, 80)
    Me.Label2.Text = "Address"
    '
    'Label3
    '
    Me.Label3.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
    Me.Label3.Location = New System.Drawing.Point(8, 136)
    Me.Label3.Size = New System.Drawing.Size(100, 16)
    Me.Label3.Text = "Town"
    '
    'Label4
    '

```

```

        Me.Label4.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.Label4.Location = New System.Drawing.Point(8, 160)
        Me.Label4.Text = "City"
    '
    'Label5
    '
        Me.Label5.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.Label5.Location = New System.Drawing.Point(8, 184)
        Me.Label5.Text = "Phone"
    '
    'TextBox1
    '
        Me.TextBox1.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.TextBox1.Location = New System.Drawing.Point(88, 56)
        Me.TextBox1.Size = New System.Drawing.Size(136, 19)
        Me.TextBox1.Text = "TextBox1"
    '
    'TextBox2
    '
        Me.TextBox2.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.TextBox2.Location = New System.Drawing.Point(88, 80)
        Me.TextBox2.Multiline = True
        Me.TextBox2.Size = New System.Drawing.Size(136, 48)
        Me.TextBox2.Text = "TextBox2"
    '
    'TextBox3
    '
        Me.TextBox3.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.TextBox3.Location = New System.Drawing.Point(88, 136)
        Me.TextBox3.Size = New System.Drawing.Size(136, 19)
        Me.TextBox3.Text = "TextBox3"
    '
    'TextBox4
    '
        Me.TextBox4.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.TextBox4.Location = New System.Drawing.Point(88, 160)
        Me.TextBox4.Size = New System.Drawing.Size(136, 19)
        Me.TextBox4.Text = "TextBox4"
    '
    'TextBox5
    '
        Me.TextBox5.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.TextBox5.Location = New System.Drawing.Point(88, 184)
        Me.TextBox5.Size = New System.Drawing.Size(136, 19)
        Me.TextBox5.Text = "TextBox5"
    '
    'Label6
    '
        Me.Label6.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.Label6.Location = New System.Drawing.Point(8, 32)
        Me.Label6.Text = "Delivery ID"
    '
    'TextBox6
    '
        Me.TextBox6.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.TextBox6.Location = New System.Drawing.Point(88, 32)
        Me.TextBox6.Size = New System.Drawing.Size(136, 19)
        Me.TextBox6.Text = "TextBox6"
    '
    'Button1
    '
        Me.Button1.Location = New System.Drawing.Point(152, 8)
        Me.Button1.Text = ">>"
    '
    'Button2

```

```

    Me.Button2.Location = New System.Drawing.Point(80, 8)
    Me.Button2.Text = "<<"
    'MainMenu1
    Me.MainMenu1.MenuItems.Add(Me.MenuItem1)
    Me.MainMenu1.MenuItems.Add(Me.MenuItem4)
    'MenuItem1
    Me.MenuItem1.MenuItems.Add(Me.MenuItem2)
    Me.MenuItem1.MenuItems.Add(Me.MenuItem3)
    Me.MenuItem1.MenuItems.Add(Me.MenuItem7)
    Me.MenuItem1.Text = "System"
    'MenuItem2
    Me.MenuItem2.Text = "Upload / Download"
    'MenuItem3
    Me.MenuItem3.Text = "Sign In / Sign Out"
    'MenuItem7
    Me.MenuItem7.Text = "Exit"
    'MenuItem4
    Me.MenuItem4.MenuItems.Add(Me.MenuItem5)
    Me.MenuItem4.MenuItems.Add(Me.MenuItem6)
    Me.MenuItem4.Text = "Delivery Process"
    'MenuItem5
    Me.MenuItem5.Text = "Delivery"
    'MenuItem6
    Me.MenuItem6.Text = "Report"
    'Button3
    Me.Button3.Location = New System.Drawing.Point(152, 216)
    Me.Button3.Text = "Delivered"
    'frmdelivery
    Me.ControlBox = False
    Me.Controls.Add(Me.Button3)
    Me.Controls.Add(Me.Button2)
    Me.Controls.Add(Me.Button1)
    Me.Controls.Add(Me.TextBox6)
    Me.Controls.Add(Me.Label6)
    Me.Controls.Add(Me.TextBox5)
    Me.Controls.Add(Me.TextBox4)
    Me.Controls.Add(Me.TextBox3)
    Me.Controls.Add(Me.TextBox2)
    Me.Controls.Add(Me.TextBox1)
    Me.Controls.Add(Me.Label5)
    Me.Controls.Add(Me.Label4)
    Me.Controls.Add(Me.Label3)
    Me.Controls.Add(Me.Label2)
    Me.Controls.Add(Me.Label1)
    Me.Menu = Me.MainMenu1
    Me.Text = "Delivery"

End Sub

#End Region

Private Sub frmdelivery_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

```

```

ShowWaitCursor(True)
cn = New System.Data.SqlClient.SqlCeConnection(connection_string)
cn.Open()
SQL = "Select * from delivery "
da = New System.Data.SqlClient.SqlCeDataAdapter(SQL, cn)
da.Fill(dt)
dt_max = dt.Rows.Count - 1
cn.Close()
If dt_max > -1 Then
    dt_i = 0
    dt_bind(dt_i)
End If
ShowWaitCursor(False)
End Sub
Sub dt_bind(ByVal ki As Integer)
    ' deliveryContactName nvarchar(50), deliveryAddress nvarchar(50),"
    ' deliverytel nvarchar(50),deliveredperson nvarchar(50), deliveredsignature
nvarchar(250),delivereddate nvarchar(50), deliverycheck int)"
    TextBox6.Text = dt.Rows(dt_i)("DeliveryID")
    TextBox1.Text = dt.Rows(dt_i)("deliveryContactName")
    TextBox2.Text = dt.Rows(dt_i)("deliveryAddress")
    TextBox3.Text = dt.Rows(dt_i)("deliverytown")
    TextBox4.Text = dt.Rows(dt_i)("deliverycity")
    TextBox5.Text = dt.Rows(dt_i)("deliverytel")
End Sub

Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
    Dim frm4 As New frmdeliverysignature
    frm4.deliveryid = TextBox6.Text
    frm4.deliveryperson = TextBox1.Text
    frm4.ShowDialog()
    Me.Close()

End Sub
End Class

```

## Teslimat İmzası

```

Public Class frmdeliverysignature
    Inherits System.Windows.Forms.Form
    Friend WithEvents Panell1 As System.Windows.Forms.Panel
    Friend WithEvents Label2 As System.Windows.Forms.Label
    Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
    Friend WithEvents Button1 As System.Windows.Forms.Button
    Friend WithEvents Button2 As System.Windows.Forms.Button
    Private WithEvents _signature As SignatureControl = New SignatureControl

    Private WithEvents _client As ClientSocket = New ClientSocket
    Private _processCommand As EventHandler

    ' hold socket notification arguments
    Private _notifyCommand As NotifyCommand
    Private _notifyData As Object
    Public deliveryid As Integer
    Public deliveryperson As String

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call
        _signature.Location = Panell1.Location
        _signature.Size = Panell1.Size
    End Sub

```

```

        Me.Controls.Add(_signature)
    End Sub

    Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
        MyBase.Dispose(disposing)
    End Sub

    'NOTE: The following procedure is required by the Windows Form Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents txtDetails As System.Windows.Forms.TextBox
    Friend WithEvents Label1 As System.Windows.Forms.Label
    <System.Diagnostics.DebuggerStepThrough() Private Sub InitializeComponent()
        Me.Panel1 = New System.Windows.Forms.Panel
        Me.Label2 = New System.Windows.Forms.Label
        Me.TextBox1 = New System.Windows.Forms.TextBox
        Me.Button1 = New System.Windows.Forms.Button
        Me.Button2 = New System.Windows.Forms.Button
        Me.txtDetails = New System.Windows.Forms.TextBox
        Me.Label1 = New System.Windows.Forms.Label
        '
        'Panel1
        '
        Me.Panel1.BackColor = System.Drawing.Color.LightGray
        Me.Panel1.Location = New System.Drawing.Point(8, 88)
        Me.Panel1.Size = New System.Drawing.Size(224, 120)
        Me.Panel1.Visible = False
        '
        'Label2
        '
        Me.Label2.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.Label2.Location = New System.Drawing.Point(8, 48)
        Me.Label2.Text = "Delivered Person"
        '
        'TextBox1
        '
        Me.TextBox1.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.TextBox1.Location = New System.Drawing.Point(104, 48)
        Me.TextBox1.Text = ""
        '
        'Button1
        '
        Me.Button1.Location = New System.Drawing.Point(144, 232)
        Me.Button1.Text = "Save"
        '
        'Button2
        '
        Me.Button2.Location = New System.Drawing.Point(64, 232)
        Me.Button2.Text = "Cancel"
        '
        'txtDetails
        '
        Me.txtDetails.Location = New System.Drawing.Point(8, 8)
        Me.txtDetails.Size = New System.Drawing.Size(208, 22)
        Me.txtDetails.Text = ""
        '
        'Label1
        '
        Me.Label1.Font = New System.Drawing.Font("Tahoma", 7.0!,
System.Drawing.FontStyle.Regular)
        Me.Label1.Location = New System.Drawing.Point(8, 72)
        Me.Label1.Size = New System.Drawing.Size(168, 16)
        Me.Label1.Text = "Please sign the panel below."
        '
        'frmdeliverysignature
        '
        Me.Controls.Add(Me.Label1)
        Me.Controls.Add(Me.txtDetails)
        Me.Controls.Add(Me.Button2)
        Me.Controls.Add(Me.Button1)
        Me.Controls.Add(Me.TextBox1)
        Me.Controls.Add(Me.Label2)
    End Sub

```

```

        Me.Controls.Add(Me.Pane1)
        Me.Text = "Capture Signature"

    End Sub

#End Region

    Private Sub frmdeliverysignature_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        TextBox1.Text = deliveryperson
        _signature.Clear()
        txtDetails.Text = ""
    End Sub

    Private Sub UpdateSignatureDetails()
        ' display info on line segments
        Dim sb As New System.Text.StringBuilder
        sb.AppendFormat(Nothing, "Signature details" + ControlChars.Cr + ControlChars.Lf
+ "Number of segments : {0}", _signature.Lines.Count)

        ' loop through all line segments
        Dim i As Integer
        For i = 0 To _signature.Lines.Count - 1
            ' get points for this line segment
            Dim points As Point() = CType(_signature.Lines(i), Point())
            sb.AppendFormat(Nothing, ControlChars.Cr + ControlChars.Lf + ControlChars.Cr
+ ControlChars.Lf + "Segment {0} ({1} points)" + ControlChars.Cr + ControlChars.Lf, i +
1, points.Length)
            Dim pt As Point
            For Each pt In points
                sb.AppendFormat(Nothing, "{0},{1} ", pt.X, pt.Y)
            Next pt
        Next i
        txtDetails.Text = sb.ToString()
    End Sub

    Private Sub _signature_SignatureUpdate(ByVal sender As Object, ByVal e As
System.EventArgs) Handles _signature.SignatureUpdate
        ' a new segment was added to the signature
        UpdateSignatureDetails()
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Dim sql As String
        Dim cn As SqlConnection
        Dim cmd As New SqlCommand
        Try
            cn = New SqlConnection(connection_string)
            cn.Open()
            sql = " update delivery set deliverycheck=1," & _
" delivereddate=getdate(),deliveredperson='" & TextBox1.Text & "'," & _
" deliveredsignature ='" & txtDetails.Text & "' where deliveryid=" &
deliveryid
            cmd.Connection = cn
            cmd.CommandType = CommandType.Text
            cmd.CommandText = sql
            cmd.ExecuteNonQuery()
            MsgBox("Saved Successfully")
            Dim mn As New mainmenu
            mn.ShowDialog()
            Me.Close()
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Dim mn As New mainmenu
        mn.ShowDialog()
        Me.Close()
    End Sub

```

```

    End Sub
End Class

```

## AES Kütüphanesi

```

using System;
using System.Data;

namespace AES
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    public class clsAES
    {
        public enum KeySize { Bits128, Bits192, Bits256 }; // key size, in bits,
for construtor
        private int Nb; // block size in 32-bit words. Always 4 for AES.
(128 bits).
        private int Nk; // key size in 32-bit words. 4, 6, 8. (128,
192, 256 bits).
        private int Nr; // number of rounds. 10, 12, 14.

        private byte[] key; // the seed key. size will be 4 * keySize from
ctor.
        private byte[,] Sbox; // Substitution box
        private byte[,] iSbox; // inverse Substitution box
        private byte[,] w; // key schedule array.
        private byte[,] Rcon; // Round constants.
        private byte[,] State; // State matrix

        private const int BlockSize = 16; // block size for encrypt and decrypt.
should always be 16

        #region Initialize
        /// <summary>
        /// Initializes the AES engine
        /// </summary>
        /// <param name="keySize">Size of the key</param>
        /// <param name="Password">the password</param>
        public clsAES(KeySize keySize, string Password)
        {
            // convert the password to a byte array, then call the other init
func
            byte[] keyBytes = ConvertStringToByteArray(Password);
            Init(keySize, keyBytes);
        }

        /// <summary>
        /// Initializes the AES engine
        /// </summary>
        /// <param name="keySize">Size of the key</param>
        /// <param name="keyBytes">the password, in a byte array</param>
        public clsAES(KeySize keySize, byte[] keyBytes)
        {
            Init(keySize, keyBytes);
        } // Aes constructor

        private void Init(KeySize keySize, byte[] keyBytes)
        {
            int i = 0;
            int keyLen = 0, kbLen = 0;
            byte nextByte = 1;

            try
            {
                SetNbNkNr(keySize);

                // set the key

```

```

        this.key = new byte[this.Nk * 4]; // 16, 24, 32 bytes

        // if the password is the right size, just copy the array
        if (this.key.Length == keyBytes.Length)
            keyBytes.CopyTo(this.key, 0);
        else // password is different size, so manually copy
        {
            // get the key lengths
            keyLen = this.key.Length;
            kbLen = keyBytes.Length;

            // manually add the password
            for (i=0; i<keyLen; i++)
            {
                // make sure we can use the keyBytes
                if (i < kbLen)
                    this.key[i] = keyBytes[i];
                else // we need to add some extra bytes
                    this.key[i] = nextByte++;
            }

            // build the two matrixes
            BuildSbox();
            BuildInvSbox();
            BuildRcon();
            KeyExpansion(); // expand the seed key into a key schedule
    and store in w
        }

        catch (Exception excec)
        { AES_ShowError(excec, "Init"); }
    }

#endregion // Initialize

#region Public Encrypt and Decrypt

    /// <summary>
    /// Encrypts the input string
    /// </summary>
    /// <param name="input">the string to encrypt -
    /// Note: the base encryption algorithm uses a byte array, but it's not
always
conversion.
    /// possible to convert the byte array to a string, using a 1:1
larger
return
string.
    /// Instead, we will convert each byte to a string, and add a space. So,
    /// when you are encrypting a string, the return result will always be
    /// in length than the original. When you are decrypting a string, the
    /// result will always be smaller in length than the original encrypted
    /// </param>
    public string Encrypt(string input)
    {
        string rText = "";
        int i=0, sLen = 0;

        try
        {
            // convert the string to a byte
            byte[] bInput = ConvertStringToByteArray(input);

            // encrypt
            byte[] bText = Encrypt(bInput);

            // convert the byte array to a string
            sLen = bText.Length;
            for (i=0; i<sLen; i++)
                rText += bText[i].ToString().Trim() + " ";
        }

        catch (Exception excec)

```



```

        { AES_ShowError(excep, "Encrypt"); }

        // return the text
        return rText.Trim();
    }

    /// <summary>
    /// Encrypts the input byte array
    /// </summary>
    /// <param name="input">the byte array to encrypt</param>
    public byte[] Encrypt(byte[] input)
    {
        int i=0, iLen = input.Length;
        byte[] output = new byte[0];
        byte[] newInput;
        byte[] inBuffer = new byte[BlockSize];
        byte[] buffer = new byte[BlockSize];
        int count=0;

        try
        {
            // we need to resize the arrays so they are 16 byte blocks
            count = GetArraySize(input.Length);
            output = new byte[count];
            newInput = new byte[count];

            // copy the data from input to newInput
            System.Array.Copy(input, 0, newInput, 0, input.Length);

            // we need to send the cipher function 16 bytes at a time
            for (i=0; i<iLen; i= i + BlockSize)
            {
                // copy the input into the input buffer array
                System.Array.Copy(newInput, i, inBuffer, 0,
                BlockSize); // copy all 16 bytes

                // encrypt this block
                System.Array.Copy( Cipher(inBuffer), 0, output, i,
                BlockSize);
            }

            catch (Exception excep)
            { AES_ShowError(excep, "Encrypt"); }

            // return the byte array
            return output;
        }

        /// <summary>
        /// Decrypts the input string
        /// </summary>
        /// <param name="input">the string to decrypt -
        /// Note: the base encryption algorithm uses a byte array, but it's not
        always
        conversion.
        /// possible to convert the byte array to a string, using a 1:1
        /// Instead, we converted each byte to a string, and added a space. So,
        /// when you are encrypting a string, the return result will always be
        larger
        /// in length than the original. When you are decrypting a string, the
        return
        /// result will always be smaller in length than the original encrypted
        string.
        /// </param>
        public string Decrypt(string input)
        {
            string rText = "";
            int i=0, sLen = 0;

            try
            {
                // convert the string to a byte array

```

```

        string[] tByte = input.Split(' ');
        sLen = tByte.Length;
        byte[] bInput = new byte[sLen];
        for (i=0; i<sLen; i++)
            bInput[i] = System.Convert.ToByte(tByte[i]);

        // decrypt and convert to the byte array to a string
        rText = ConvertByteArrayToString( Decrypt(bInput));
    }

    catch (Exception excep)
    { AES_ShowError(excep, "Decrypt"); }

    // return
    return rText;
}

/// <summary>
/// Decrypts the input byte array
/// </summary>
/// <param name="input">the byte array to decrypt</param>
public byte[] Decrypt(byte[] input)
{
    int i=0, iLen = input.Length;
    byte[] inBuffer = new byte[BlockSize];
    byte[] buffer = new byte[BlockSize];
    byte[] output = new byte[input.Length];
    int count=0;

    try
    {
        // we need to send the cipher function 16 bytes at a time
        for (i=0; i<iLen; i= i + BlockSize)
        {
            // copy the input into the input buffer array
            System.Array.Copy(input, i, inBuffer, 0,
                BlockSize); // copy all 16 bytes

            // decrypt this block
            System.Array.Copy( InvCipher(inBuffer), 0, output,
                i, BlockSize);
        }

        catch (Exception excep)
        { AES_ShowError(excep, "Decrypt"); }

        // return the byte array
        return output;
    }

#endregion // Public Encrypt and Decrypt

#region Private Encrypt and Decrypt

/// <summary>
/// Encrypt the data in a 16 bit block. returns:
/// 16 byte block of encrypted characters
/// </summary>
/// <param name="input">16 byte block of characters to encrypt</param>
/// <param name="retCount">the number of bytes actually used</param>
/// <returns>16 byte block of encrypted characters</returns>
private byte[] Cipher(byte[] input) // encipher 16-bit input
{
    byte[] output = new byte[16];

    try
    {
        // state = input
        this.State = new byte[4,Nb]; // always [4,4]
        for (int i = 0; i < (4 * Nb); ++i)
        {
            this.State[i % 4, i / 4] = input[i];
        }
    }
}

```

```

    }
    AddRoundKey(0);

    for (int round = 1; round <= (Nr - 1); ++round) // main
round loop
    {
        SubBytes();
        ShiftRows();
        MixColumns();
        AddRoundKey(round);
    } // main round loop

    SubBytes();
    ShiftRows();
    AddRoundKey(Nr);

    // output = state
    for (int i = 0; i < (4 * Nb); ++i)
    {
        output[i] = this.State[i % 4, i / 4];
    }

    catch (Exception excep)
    { AES_ShowError(excep, "Cipher"); }

    return output;
} // Cipher()

/// <summary>
/// Decrypts a 16 byte block of text
/// </summary>
/// <param name="input">16 byte block to decrypt</param>
/// <returns>16 byte block of decrypted bytes</returns>
private byte[] InvCipher(byte[] input) // decipher 16-bit input
{
    byte[] output = new byte[16];

    try
    {
        // state = input
        this.State = new byte[4,Nb]; // always [4,4]
        for (int i = 0; i < (4 * Nb); ++i)
        {
            this.State[i % 4, i / 4] = input[i];
        }

        AddRoundKey(Nr);

        for (int round = Nr-1; round >= 1; --round) // main round
loop
        {
            InvShiftRows();
            InvSubBytes();
            AddRoundKey(round);
            InvMixColumns();
        } // end main round loop for InvCipher

        InvShiftRows();
        InvSubBytes();
        AddRoundKey(0);

        // output = state
        for (int i = 0; i < (4 * Nb); ++i)
        {
            output[i] = this.State[i % 4, i / 4];
        }

    }

    catch (Exception excep)
    { AES_ShowError(excep, "InvCipher"); }

    return output;
}

```

```

    } // InvCipher()

#endregion // Private Encrypt and Decrypt

#region AES Tables and Work Body

private void SetNbNkNr(KeySize keySize)
{
    this.Nb = 4; // block size always = 4 words = 16 bytes = 128
bits for AES

    if (keySize == KeySize.Bits128)
    {
        this.Nk = 4; // key size = 4 words = 16 bytes = 128 bits
        this.Nr = 10; // rounds for algorithm = 10
    }
    else if (keySize == KeySize.Bits192)
    {
        this.Nk = 6; // 6 words = 24 bytes = 192 bits
        this.Nr = 12;
    }
    else if (keySize == KeySize.Bits256)
    {
        this.Nk = 8; // 8 words = 32 bytes = 256 bits
        this.Nr = 14;
    }
} // SetNbNkNr()

private void BuildSbox()
{
    this.Sbox = new byte[16,16] { // populate the Sbox matrix
1      2      3      4      5      6      7      8      9      a      b      c      d      e      f /* 0
/*0*/
{0x63, 0x7c, 0x77, 0x7b, 0xf2, 0x6b, 0x6f, 0xc5, 0x30, 0x01, 0x67, 0x2b, 0xfe, 0xd7,
0xab, 0x76},
/*1*/
{0xca, 0x82, 0xc9, 0x7d, 0xfa, 0x59, 0x47, 0xf0, 0xad, 0xd4, 0xa2, 0xaf, 0x9c, 0xa4,
0x72, 0xc0},
/*2*/
{0xb7, 0xfd, 0x93, 0x26, 0x36, 0x3f, 0xf7, 0xcc, 0x34, 0xa5, 0xe5, 0xf1, 0x71, 0xd8,
0x31, 0x15},
/*3*/
{0x04, 0xc7, 0x23, 0xc3, 0x18, 0x96, 0x05, 0x9a, 0x07, 0x12, 0x80, 0xe2, 0xeb, 0x27,
0xb2, 0x75},
/*4*/
{0x09, 0x83, 0x2c, 0x1a, 0x1b, 0x6e, 0x5a, 0xa0, 0x52, 0x3b, 0xd6, 0xb3, 0x29, 0xe3,
0x2f, 0x84},
/*5*/
{0x53, 0xd1, 0x00, 0xed, 0x20, 0xfc, 0xb1, 0x5b, 0x6a, 0xcb, 0xbe, 0x39, 0x4a, 0x4c,
0x58, 0xcf},
/*6*/
{0xd0, 0xef, 0xaa, 0xfb, 0x43, 0x4d, 0x33, 0x85, 0x45, 0xf9, 0x02, 0x7f, 0x50, 0x3c,
0x9f, 0xa8},
/*7*/
{0x51, 0xa3, 0x40, 0x8f, 0x92, 0x9d, 0x38, 0xf5, 0xbc, 0xb6, 0xda, 0x21, 0x10, 0xff,
0xf3, 0xd2},
/*8*/
{0xcd, 0x0c, 0x13, 0xec, 0x5f, 0x97, 0x44, 0x17, 0xc4, 0xa7, 0x7e, 0x3d, 0x64, 0x5d,
0x19, 0x73},
/*9*/
{0x60, 0x81, 0x4f, 0xdc, 0x22, 0x2a, 0x90, 0x88, 0x46, 0xee, 0xb8, 0x14, 0xde, 0x5e,
0x0b, 0xdb},
/*a*/
{0xe0, 0x32, 0x3a, 0x0a, 0x49, 0x06, 0x24, 0x5c, 0xc2, 0xd3, 0xac, 0x62, 0x91, 0x95,
0xe4, 0x79},
/*b*/
{0xe7, 0xc8, 0x37, 0x6d, 0x8d, 0xd5, 0x4e, 0xa9, 0x6c, 0x56, 0xf4, 0xea, 0x65, 0x7a,
0xae, 0x08},
/*c*/
{0xba, 0x78, 0x25, 0x2e, 0x1c, 0xa6, 0xb4, 0xc6, 0xe8, 0xdd, 0x74, 0x1f, 0x4b, 0xbd,
0x8b, 0x8a},

```

```

                                                                    /*d*/
{0x70, 0x3e, 0xb5, 0x66, 0x48, 0x03, 0xf6, 0x0e, 0x61, 0x35, 0x57, 0xb9, 0x86, 0xc1,
0x1d, 0x9e},
                                                                    /*e*/
{0xe1, 0xf8, 0x98, 0x11, 0x69, 0xd9, 0x8e, 0x94, 0x9b, 0x1e, 0x87, 0xe9, 0xce, 0x55,
0x28, 0xdf},
                                                                    /*f*/
{0x8c, 0xa1, 0x89, 0x0d, 0xbf, 0xe6, 0x42, 0x68, 0x41, 0x99, 0x2d, 0x0f, 0xb0, 0x54,
0xbb, 0x16} };

    } // BuildSbox()

private void BuildInvSbox()
{
    this.iSbox = new byte[16,16] { // populate the iSbox matrix
1      2      3      4      5      6      7      8      9      a      b      c      d      e      f /*
{0x52, 0x09, 0x6a, 0xd5, 0x30, 0x36, 0xa5, 0x38, 0xbf, 0x40, 0xa3, 0x9e, 0x81, 0xf3,
0xd7, 0xfb},
                                                                    /*1*/
{0x7c, 0xe3, 0x39, 0x82, 0x9b, 0x2f, 0xff, 0x87, 0x34, 0x8e, 0x43, 0x44, 0xc4, 0xde,
0xe9, 0xcb},
                                                                    /*2*/
{0x54, 0x7b, 0x94, 0x32, 0xa6, 0xc2, 0x23, 0x3d, 0xee, 0x4c, 0x95, 0x0b, 0x42, 0xfa,
0xc3, 0x4e},
                                                                    /*3*/
{0x08, 0x2e, 0xa1, 0x66, 0x28, 0xd9, 0x24, 0xb2, 0x76, 0x5b, 0xa2, 0x49, 0x6d, 0x8b,
0xd1, 0x25},
                                                                    /*4*/
{0x72, 0xf8, 0xf6, 0x64, 0x86, 0x68, 0x98, 0x16, 0xd4, 0xa4, 0x5c, 0xcc, 0x5d, 0x65,
0xb6, 0x92},
                                                                    /*5*/
{0x6c, 0x70, 0x48, 0x50, 0xfd, 0xed, 0xb9, 0xda, 0x5e, 0x15, 0x46, 0x57, 0xa7, 0x8d,
0x9d, 0x84},
                                                                    /*6*/
{0x90, 0xd8, 0xab, 0x00, 0x8c, 0xbc, 0xd3, 0x0a, 0xf7, 0xe4, 0x58, 0x05, 0xb8, 0xb3,
0x45, 0x06},
                                                                    /*7*/
{0xd0, 0x2c, 0x1e, 0x8f, 0xca, 0x3f, 0x0f, 0x02, 0xc1, 0xaf, 0xbd, 0x03, 0x01, 0x13,
0x8a, 0x6b},
                                                                    /*8*/
{0x3a, 0x91, 0x11, 0x41, 0x4f, 0x67, 0xdc, 0xea, 0x97, 0xf2, 0xcf, 0xce, 0xf0, 0xb4,
0xe6, 0x73},
                                                                    /*9*/
{0x96, 0xac, 0x74, 0x22, 0xe7, 0xad, 0x35, 0x85, 0xe2, 0xf9, 0x37, 0xe8, 0x1c, 0x75,
0xdf, 0x6e},
                                                                    /*a*/
{0x47, 0xf1, 0x1a, 0x71, 0x1d, 0x29, 0xc5, 0x89, 0x6f, 0xb7, 0x62, 0x0e, 0xaa, 0x18,
0xbe, 0x1b},
                                                                    /*b*/
{0xfc, 0x56, 0x3e, 0x4b, 0xc6, 0xd2, 0x79, 0x20, 0x9a, 0xdb, 0xc0, 0xfe, 0x78, 0xcd,
0x5a, 0xf4},
                                                                    /*c*/
{0x1f, 0xdd, 0xa8, 0x33, 0x88, 0x07, 0xc7, 0x31, 0xb1, 0x12, 0x10, 0x59, 0x27, 0x80,
0xec, 0x5f},
                                                                    /*d*/
{0x60, 0x51, 0x7f, 0xa9, 0x19, 0xb5, 0x4a, 0x0d, 0x2d, 0xe5, 0x7a, 0x9f, 0x93, 0xc9,
0x9c, 0xef},
                                                                    /*e*/
{0xa0, 0xe0, 0x3b, 0x4d, 0xae, 0x2a, 0xf5, 0xb0, 0xc8, 0xeb, 0xbb, 0x3c, 0x83, 0x53,
0x99, 0x61},
                                                                    /*f*/
{0x17, 0x2b, 0x04, 0x7e, 0xba, 0x77, 0xd6, 0x26, 0xe1, 0x69, 0x14, 0x63, 0x55, 0x21,
0x0c, 0x7d} };

    } // BuildInvSbox()

private void BuildRcon()
{
    this.Rcon = new byte[11,4] { {0x00, 0x00, 0x00, 0x00},
                                {0x01, 0x00, 0x00, 0x00},
                                {0x02, 0x00, 0x00, 0x00},
                                {0x04, 0x00, 0x00, 0x00},

```

```

                                                                 {0x08, 0x00, 0x00, 0x00},
                                                                 {0x10, 0x00, 0x00, 0x00},
                                                                 {0x20, 0x00, 0x00, 0x00},
                                                                 {0x40, 0x00, 0x00, 0x00},
                                                                 {0x80, 0x00, 0x00, 0x00},
                                                                 {0x1b, 0x00, 0x00, 0x00},
                                                                 {0x36, 0x00, 0x00, 0x00}
};

    } // BuildRcon()

private void AddRoundKey(int round)
{
    for (int r = 0; r < 4; ++r)
    {
        for (int c = 0; c < 4; ++c)
        {
            this.State[r,c] = (byte) ( (int)this.State[r,c] ^
(int)w[(round*4)+c,r] );
        }
    }
} // AddRoundKey()

private void SubBytes()
{
    for (int r = 0; r < 4; ++r)
    {
        for (int c = 0; c < 4; ++c)
        {
            this.State[r,c] = this.Sbox[ (this.State[r,c] >>
4), (this.State[r,c] & 0x0f) ];
        }
    }
} // SubBytes

private void InvSubBytes()
{
    for (int r = 0; r < 4; ++r)
    {
        for (int c = 0; c < 4; ++c)
        {
            this.State[r,c] = this.iSbox[ (this.State[r,c] >>
4), (this.State[r,c] & 0x0f) ];
        }
    }
} // InvSubBytes

private void ShiftRows()
{
    byte[,] temp = new byte[4,4];
    for (int r = 0; r < 4; ++r) // copy State into temp[]
    {
        for (int c = 0; c < 4; ++c)
        {
            temp[r,c] = this.State[r,c];
        }
    }

    for (int r = 1; r < 4; ++r) // shift temp into State
    {
        for (int c = 0; c < 4; ++c)
        {
            this.State[r,c] = temp[ r, (c + r) % Nb ];
        }
    }
} // ShiftRows()

private void InvShiftRows()
{
    byte[,] temp = new byte[4,4];
    for (int r = 0; r < 4; ++r) // copy State into temp[]
    {
        for (int c = 0; c < 4; ++c)
        {

```

```

        temp[r,c] = this.State[r,c];
    }
}
for (int r = 1; r < 4; ++r) // shift temp into State
{
    for (int c = 0; c < 4; ++c)
    {
        this.State[r, (c + r) % Nb ] = temp[r,c];
    }
}
} // InvShiftRows()

private void MixColumns()
{
    byte[,] temp = new byte[4,4];
    for (int r = 0; r < 4; ++r) // copy State into temp[]
    {
        for (int c = 0; c < 4; ++c)
        {
            temp[r,c] = this.State[r,c];
        }
    }

    for (int c = 0; c < 4; ++c)
    {
        this.State[0,c] = (byte) ( (int)gfmultby02(temp[0,c]) ^
(int)gfmultby03(temp[1,c]) ^
(int)gfmultby01(temp[2,c]) ^
(int)gfmultby01(temp[3,c]) );
        this.State[1,c] = (byte) ( (int)gfmultby01(temp[0,c]) ^
(int)gfmultby02(temp[1,c]) ^
(int)gfmultby03(temp[2,c]) ^
(int)gfmultby01(temp[3,c]) );
        this.State[2,c] = (byte) ( (int)gfmultby01(temp[0,c]) ^
(int)gfmultby02(temp[1,c]) ^
(int)gfmultby03(temp[2,c]) ^
(int)gfmultby01(temp[3,c]) );
        this.State[3,c] = (byte) ( (int)gfmultby03(temp[0,c]) ^
(int)gfmultby01(temp[1,c]) ^
(int)gfmultby02(temp[2,c]) ^
(int)gfmultby01(temp[3,c]) );
    } // MixColumns

private void InvMixColumns()
{
    byte[,] temp = new byte[4,4];
    for (int r = 0; r < 4; ++r) // copy State into temp[]
    {
        for (int c = 0; c < 4; ++c)
        {
            temp[r,c] = this.State[r,c];
        }
    }

    for (int c = 0; c < 4; ++c)
    {
        this.State[0,c] = (byte) ( (int)gfmultby0e(temp[0,c]) ^
(int)gfmultby0b(temp[1,c]) ^
(int)gfmultby09(temp[2,c]) ^
(int)gfmultby09(temp[3,c]) );
        this.State[1,c] = (byte) ( (int)gfmultby09(temp[0,c]) ^
(int)gfmultby0e(temp[1,c]) ^
(int)gfmultby0b(temp[2,c]) ^
(int)gfmultby0d(temp[3,c]) );
        this.State[2,c] = (byte) ( (int)gfmultby0d(temp[0,c]) ^
(int)gfmultby09(temp[1,c]) ^
(int)gfmultby0e(temp[2,c]) ^
(int)gfmultby0b(temp[3,c]) );
        this.State[3,c] = (byte) ( (int)gfmultby0b(temp[0,c]) ^
(int)gfmultby0d(temp[1,c]) ^
(int)gfmultby09(temp[2,c]) ^
(int)gfmultby0e(temp[3,c]) );
    }
}
}

```

```

} // InvMixColumns

private static byte gfmultby01(byte b)
{
    return b;
}

private static byte gfmultby02(byte b)
{
    if (b < 0x80)
        return (byte)(int)(b <<1);
    else
        return (byte)( (int)(b << 1) ^ (int)(0x1b) );
}

private static byte gfmultby03(byte b)
{
    return (byte) ( (int)gfmultby02(b) ^ (int)b );
}

private static byte gfmultby09(byte b)
{
    return (byte)( (int)gfmultby02(gfmultby02(gfmultby02(b))) ^
        (int)b );
}

private static byte gfmultby0b(byte b)
{
    return (byte)( (int)gfmultby02(gfmultby02(gfmultby02(b))) ^
        (int)gfmultby02(b) ^
        (int)b );
}

private static byte gfmultby0d(byte b)
{
    return (byte)( (int)gfmultby02(gfmultby02(gfmultby02(b))) ^
        (int)gfmultby02(gfmultby02(b)) ^
        (int)(b) );
}

private static byte gfmultby0e(byte b)
{
    return (byte)( (int)gfmultby02(gfmultby02(gfmultby02(b))) ^
        (int)gfmultby02(gfmultby02(b)) ^
        (int)gfmultby02(b) );
}

private void KeyExpansion()
{
    this.w = new byte[Nb * (Nr+1), 4]; // 4 columns of bytes
corresponds to a word

    for (int row = 0; row < Nk; ++row)
    {
        this.w[row,0] = this.key[4*row];
        this.w[row,1] = this.key[4*row+1];
        this.w[row,2] = this.key[4*row+2];
        this.w[row,3] = this.key[4*row+3];
    }

    byte[] temp = new byte[4];

    for (int row = Nk; row < Nb * (Nr+1); ++row)
    {
        temp[0] = this.w[row-1,0]; temp[1] = this.w[row-1,1];
        temp[2] = this.w[row-1,2]; temp[3] = this.w[row-1,3];

        if (row % Nk == 0)
        {
            temp = SubWord(RotWord(temp));

            temp[0] = (byte)( (int)temp[0] ^
(int)this.Rcon[row/Nk,0] );

```



```

temp[1] = (byte)( (int)temp[1] ^
(int)this.Rcon[row/Nk,1] );
temp[2] = (byte)( (int)temp[2] ^
(int)this.Rcon[row/Nk,2] );
temp[3] = (byte)( (int)temp[3] ^
(int)this.Rcon[row/Nk,3] );
}
else if ( Nk > 6 && (row % Nk == 4) )
{
temp = SubWord(temp);
}

// w[row] = w[row-Nk] xor temp
this.w[row,0] = (byte) ( (int)this.w[row-Nk,0] ^
(int)temp[0] );
this.w[row,1] = (byte) ( (int)this.w[row-Nk,1] ^
(int)temp[1] );
this.w[row,2] = (byte) ( (int)this.w[row-Nk,2] ^
(int)temp[2] );
this.w[row,3] = (byte) ( (int)this.w[row-Nk,3] ^
(int)temp[3] );

} // for loop
} // KeyExpansion()

private byte[] SubWord(byte[] word)
{
byte[] result = new byte[4];
result[0] = this.Sbox[ word[0] >> 4, word[0] & 0x0f ];
result[1] = this.Sbox[ word[1] >> 4, word[1] & 0x0f ];
result[2] = this.Sbox[ word[2] >> 4, word[2] & 0x0f ];
result[3] = this.Sbox[ word[3] >> 4, word[3] & 0x0f ];
return result;
}

private byte[] RotWord(byte[] word)
{
byte[] result = new byte[4];
result[0] = word[1];
result[1] = word[2];
result[2] = word[3];
result[3] = word[0];
return result;
}

public void Dump()
{
Console.WriteLine("Nb = " + Nb + " Nk = " + Nk + " Nr = " + Nr);
Console.WriteLine("\nThe key is \n" + DumpKey() );
Console.WriteLine("\nThe Sbox is \n" + DumpTwoByTwo(Sbox));
Console.WriteLine("\nThe w array is \n" + DumpTwoByTwo(w));
Console.WriteLine("\nThe State array is \n" +
DumpTwoByTwo(State));
}

public string DumpKey()
{
string s = "";
for (int i = 0; i < key.Length; ++i)
s += key[i].ToString("x2") + " ";
return s;
}

public string DumpTwoByTwo(byte[,] a)
{
string s = "";
for (int r = 0; r < a.GetLength(0); ++r)
{
s += "["+r+"]" + " ";
for (int c = 0; c < a.GetLength(1); ++c)
{
s += a[r,c].ToString("x2") + " ";
}
s += "\n";
}
}

```

```

    }
    return s;
}

#endregion // AES Tables and Work Body

#region Byte and String Conversions

/// <summary>
/// Converts a string to a byte array
/// </summary>
/// <param name="StrToConvert">the string to convert</param>
private byte[] ConvertStringToByteArray(string StrToConvert)
{
    return System.Text.ASCIIEncoding.ASCII.GetBytes(StrToConvert);
}

/// <summary>
/// Converts a byte array to a string
/// </summary>
/// <param name="ByteToConvert"></param>
/// <returns></returns>
private string ConvertByteArrayToString(byte[] ByteToConvert)
{
    string tempStr = "";
    for (int i=0; i<ByteToConvert.Length; i++)
    {
        // do not convert 0
        if (ByteToConvert[i] > 0)
            tempStr += System.Convert.ToChar(ByteToConvert[i]);
    }

    return tempStr;
}

#endregion // Byte and String Conversions

#region Misc Functions

/// <summary>
/// Display an error message
/// </summary>
/// <param name="excep">the exception</param>
/// <param name="FunctionName">the name of the function that caused the
error</param>
private void AES_ShowError(Exception excep, string FunctionName)
{
    string errMsg = "";
    string nl = "\r\n";

    try
    {
        // build the error string
        errMsg += "An error has occurred." + nl;
        errMsg += "Function: clsAES." + FunctionName.Trim() + nl;
        errMsg += "Source: " ;
        errMsg += "Error Description: " + nl + excep.Message;

        // show the message
        // MessageBox.Show(errMsg, "AES Encryption",
MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    catch
    {}
}

/// <summary>
/// Returns the size of the array needed to meet BlockSize
/// </summary>
/// <param name="ArrayLen">the length of the current array</param>
private int GetArraySize(int ArrayLen)
{
    // if this is divisible by blocksize, return arraylen
    if ( (BlockSize % ArrayLen) == 0)

```

```

        return ArrayLen;

        // return the new array size
        return ( ((ArrayLen / BlockSize) + 1) * BlockSize);
    }

    #endregion // Misc Functions
}
}

```

## C) Öz Geçmiş

### YILDIRIM KOCDAG

Professional  
experience

Agu 2006 - BSH (Bosch-Siemens)  
MUNICH / LOS ANGELES  
System Analyst & Programmer

ISTANBUL /

BSH is one of the biggest white goods companies of world. It produces refrigerators, washing machines, dish washer and etc.

Some of the main responsibilities,

- Creating, managing, maintaining, optimizing and configuring databases(SQL Server),
- Developing and implementing information systems for .NET project development,
- Developing and implementing policies and procedures for Web based system development,
- Discussing with clients and identifying documents and projects requirements,
- Integrating and implementing information systems projects solutions,
- Writing, modifying, integrating and testing computer codes for .NET and Web projects.

Projects,

- Cosmos Project is a service project which is implemented in 13 different countries.
- Service Power Project is an integration project with a third party company and BSH.

Dec 2005 - Agu 2006 Arcelik  
System Analyst & Team Leader

ISTANBUL

Arcelik is one of the biggest white goods and electronic devices company in Turkey. It produces TVs, computers, refrigerators, washing machines, dish washer and etc.

Some of the main responsibilities,

- Creating, managing, maintaining, optimizing and configuring

- databases(SQL Server and Oracle),
- Developing and implementing information systems for .NET project development,
- Developing and implementing policies and procedures for Web based system development,
- Collecting the information and user's requirements to develop logical and physical specifications,
- Assessing the feasibility of the new technology,
- Evaluating and synthesizing technical information and use them to design, develop and test the new technology system,
- Evaluating the information from the existing system and develop data, process and network models to reduce the non-functional requirements,
- Creating Business Flow of the system.

#### Projects,

- Marduk Project is a web based call center project. Some of the major operations which can be done via Marduk Pages are new customer entry, customer survey, address detail update, repair maintenance order, customer complaints and general statistics. Almost 70 agents of Arcelik are using the project as a customer relation management system. More than 200 customer's calls are entered everyday by agents via this system.
- WebWasos England is a web based call center system for Arcelik England. Modules of the system are storing new customer information, checking the warranty period of the appliances, assigning the technician for repair tasks, technician and service information update and route sheet management system of technicians, create invoices and send them to central database. After the system completion, some of the integration problem was occurred and the client decided to cancel the project.

July 2004 - Feb 2005    BIS  
System Analyst & Programmer

ISTANBUL

BIS is a software company which has a lot of clients in finance and energy sector.

#### Some of the main responsibilities,

- Developing the baseforms in .NET and SQL Server,
- Researching, evaluating and synthesizing technical information to optimize the system,
- Implementing the generic reports,
- Researching for performance tips,
- Designing the database.

#### Projects,

- BIS Framework, the purpose of the project was to create generic baseforms and controls which can be a base for every form in .NET.
- EDBYS is a product for energy sector companies. The integrated Utility Billing Product of BIS has been implemented at

Palgaz which is a natural gas distribution company in Turkey.

July 2003 – July 2004 Philip Morris, SABANCI ISTANBUL  
System Analyst & Programmer

Philip Morris is a leading global tobacco company.

Some of the main responsibilities,

- Writing, integrating and testing software codes in .NET,
- Maintaining on-line computer programs,
- Identifying technical problems, processes and solutions,
- Prepare reports on the status, operation and maintenance of the software,
- Collecting documentation of user's requirements
- Developing logical and physical specifications.

Projects,

- E-Expense is an application used by Philip Morris staff to enter domestic and international expenditures via internet, and to submit them via internet for approval.
- CarFleet.Net is a system under which various details of Philip Morris's vehicles are kept. Due to the improvement of the service, change of the project from Visual Basic 6.0 to Visual Basic.Net is ensured.

Information  
Technology

In-depth knowledge of Microsoft Products like VB.NET, ASP. NET, ADO.NET, C#.Net, Visual Basic.NET, SQL Server 2000-2005.

**Certifications**

**MCS.D.NET**

**MCDBA**

**MCTS SQL Server 2005**

Advance web programming experience in AJAX, Javascript and ATLAS.

Good exposure to entire Software life cycle phases (system analysis, design, coding, testing & maintenance)

Good experience in Reporting Tools such as Reporting Services, Crystal Reports and XML Report.

Good Experience in other database software such as Sybase and Oracle.

Education

2006 - 2007 MALTEPE UNIVERSITY ISTANBUL  
Business Administration

- MBA

2005 ST. MARKS COLLEGE SYDNEY  
General English

---

- Advance Level, IELTS 6.5

2005 SYDNEY COLLEGE SYDNEY

General English and Academic English

- Advance Level
- English Academic Purposes

2004 – 2006 MALTEPE UNIVERSITY ISTANBUL

Computer Science and Engineering

- MSc

1998 – 2003 YILDIZ TECHNICAL UNIVERSITY ISTANBUL

Computer Science and Engineering

- BSc, GPA 3.08.
- Microsoft Awarded Graduation Project.

---

Languages

English – IELTS 6.5

- READING 8.5
- LISTENING 6
- SPEAKING 6
- WRITING 6

German – Basic