Bütünleşik Maaş Hesabı Otomasyon Sistemi

Veli Eski[1], Doç.Dr. İsmail Rakıp Karaş[2]

[1] Pamukkale Üniversitesi, Bilgi İşlem Daire Başkanlığı, Denizli

[2] Karabük Üniversitesi Bilgisayar Mühendisliği Bölümü, Karabük

veski@pau.edu.tr, ismail.kara@karabuk.edu.tr

Özet: Maaş hesabı otomasyonu üniversitenin tüm birimlerinde 2547 Sayılı Yükseköğretim Kanunu ve 657 Sayılı Devlet Memurları Kanununa göre çalışan akademik ve idari personele ait aylık ücretlerin hesaplanması amacıyla tasarlanmıştır. Bu otomasyon ile birimler arasındaki veri iletiminde yaşanan sorunların giderilmesi, gerçek zamanlı verilerle işlem yapılması, kullanıcıdan ve sistemden kaynaklanan hataların ortadan kaldırılması amaçlanmıştır. Bu kapsamda otomasyon, Delphi programlama dili ve MsSql veritabanı kullanılarak hazırlanmıştır.

Anahtar Sözcükler: Maaş hesabı, yazılım, veritabanı.

Salary Account Automation System

Abstract: Abstract: The salary accounting automation system was designed to calculate the montly charges of all academic and administrative staff who work with 2547 higher education law and 657 civil service law in university. The propose of this automation is to eliminate data transmission problems between the units, to make transactions with real-time data and to eliminate errors caused by the users and systems. In this context, the automation is developed by using Delphi programing language and MsSql database.

Keywords: Salary account, software, database,

1. Giriş

Üniversite bünyesinde yer alan fakülte, meslek yüksekokulu, yüksekokul, enstitü, hastane, araştırma merkezleri ve iktisadi işletmelerde çalışan çok sayıda ve farklı sınıflarda personel mevcuttur. Çalışan personel temelde akademik ve idari olarak iki sınıfa ayrılır. İdari personel de kendi içinde 4A, 4B ve 4C olarak üç farklı sınıflandırmaya tabiidir.

Bu şekildeki personel sınıflandırması sonucu temel olarak ödenen aylık maaşlar personelin statüsüne göre hesaplanması farklılık göstermektedir.

Aylık maaşların dışında personel sınıflandırmasına göre ek ders, yaz okulu, sınav kâğıdı okuma ücretleri, fazla mesai gibi farklı ücret hesaplamaları da yapılmaktadır.

Bu kadar farklı birim, sınıf ve ücret tipi olduğu için birimler arasında veri paylaşımı sağlıklı olmamaktadır. Hatta bazı zamanlarda aynı birim içindeki ek ders ile aylık maaş yapan personel arasında bile veri iletiminde sorunlar yaşanabilmektedir.

Bu gibi sorunlar nedeniyle yeni bir yazılımın tasarlanması gerekliliği ortaya çıkmıştır.

2. Materyal ve Metod

Yeni oluşturulacak olan yazılımda tüm verilerin tek bir merkezde toplanması ve tek bir yazılımın kullanılması amaçlanmıştır. Bu sayede pek çok sorunun üstesinden gelinebilecektir.

2.1.Veritabanı

Yapılan analizlerde mevcut sistemdeki sorunların ana kaynağının farklı veritabanları kullanılması olduğu görülmüştür. Yazılımda kullanılacak olan veritabanının aşağıdaki ihtiyaçları karşılar durumda olmasına dikkat edilmiştir.

- Mevcut sistemde kullanılan tüm veritabanlarının özelliklerini karşılamalıdır.
- Veriler hiyerarşik bir düzen içerisinde saklanabilmelidir.
- Kolay yönetilebilmedir.
- Veri bütünlüğünü koruyabilmelidir.
- Veri güvenliğini sağlayabilmelidir.
- Gerektiğinde yazılım dışında sorgulamalar kolay yapılabilmelidir.
- Veri yapısı yazılım içinden ya da veritabanı arayüzü ile rahatça güncellenebilmelidir.
- Veritabanı günün ihtiyaçlarına göre sürekli yazılımını güncellemelidir.
- Ürün desteği olmalıdır.
- Hemen her yazılımcı rahatça kullanabilmelidir.
- Kuruma fazladan maddi yük getirmemelidir.
- Her türlü yetkilendirme yapılabilmelidir.
- Otomatik yedek alabilmeli, gerektiğinde uyarı mesajları gönderebilmelidir.
- Ücret verilenin kırk beş yıl gibi bir saklanma zorunluluğu olduğundan uzun süreli ve büyük miktarda veriyi saklayabilmelidir.
- Barındırdığı büyük verilere karşılık performansı yeteri derecede iyi olmalıdır.

Yukarıda sayılan tüm ihtiyaçlar göz önüne alındığında ilişkisel bir veritabanının kullanılması zorunluluğu ortaya çıkar. İlişkisel veritabanı, günümüzde en çok kullanılan veritabanı yönetim sistemidir. İlişkisel veritabanı verileri tablolar halinde birbirleriyle ilişkili bir biçimde saklar. Normalde disk üzerinde veriler dosyalarda tutulur. İlişkisel veritabanı ise içinde tablolar barındıran büyük dosyalardır. Bu tablolar arasındaki ilişkiyi ve veri bütünlüğünü index yapıları ile sağlar.

Bu nedenlerden dolayı ve sayılan ihtiyaçları en çok karşılayan veritabanı olarak Microsoft SQL Server veritabanının 2012 sürümünün kullanılması tercih edilmiştir.

2.1.1 Veritabanı Tasarımı

Veritabanı tasarlanırken aşağıda belirtilen sorulara cevap verilerek yazılım için en uygun, en pratik ve hızlı şekilde çalışabilecek bir veritabanı ortaya çıkarılmaya çalışılmıştır.

- Sistem ne iş yapacak?
- Veritabanı ne gibi verileri saklayacak?
- Bu verileri için hangi tablolar gerekecek?
- Tablolarda hangi veriler tutulacak?
- Tablolardaki kolonlar neler olacak?
- Tablolar arasındaki ilişkiler neler olacak?
- Hangi veri türleri kullanılacak?
- Verilerin büyüklüğü ne kadar olacak?
- Kısıtlamalar neler olacak? [1].

2.2. Yazılım Dili

Oluşturulacak yazılımın Windows işletim sisteminde çalıştırılması amaçlandığından işletim sistemine uyumlu görsel bir yazılım dili tercih edildi. Yazılım dilinin seçiminde göz önünde bulundurulan kriterler aşağıda listelenmiştir.

- Kullanılacak olan yazılım dili görsel olmalı.
- Kolay kullanımı olmalı.
- Kolay anlaşılabilmeli.
- Dilin terimleri günlük kullanılan dile yakın olmalı.
- Basit olmasına karşın güçlü bir yapısı bulunmalı.
- Gerektiğinde basit dosya işlemlerinden web sayfası yapımına kadar geniş bir yelpazesi olması.
- Bazı durumlarda oluşturulan yazılımın Linux işletim sisteminde de kolayca çalıştırılabilmesini sağlayabilmesi.
- Seçilen veritabanı ile etkileşimli çalışabilmesi.
- Çok fazla bilgisayar kaynağı kullanmaması.
- Üretilen yazılımın hızlı çalışmasını sağlaması.

Delphi, Pascal tabanlı bir dil olup nesneye yönelik programlama yapabilme özelliğine sahiptir. Öğrenimi kolay olması nedeniyle çoğu

programcı tarafından kullanılır. Tamamen görsel bir programlama dilidir [2].

Windows işletim sistemi için hızlı uygulama geliştirme (RAD – Rapid Application Development) aracıdır. Delphi derleyicisi dünyanın en hızlı derleyicisi konumundadır. Aynı zamanda yazılan kodun derleyici tarafından optimize edilmesiyle daha hızlı çalışan kod üretimi sağlamaktadır.

Tüm bu özellikler dikkate alındığında Delphi yazılım dilinin XE2 versiyonu ile otomasyonun geliştirilmesi kararlaştırılmıştır.

2.2.1 Kullanılan Bileşenler

Yazılım geliştirilirken tüm modüller için ortak pek çok fonksiyon ve kod oluşturulması gerektiği görülmüştür. Bu nedenle de ortak olan tüm işlemler DLL (Dynamic Link Library) dosyaları içerisinde tanımlanmıştır. Bu DLL dosyaları ve görevleri kısaca şu şekildedir.

- **Database.dll** Veritabanı ile ilgili tüm işlemler buradan gerçekleştirilmektedir.
- **DateTime.dll** Yazılım içerisinde kullanılan tarih ve saat işlemleri bu DLL içerisinden yapılmaktadır.
- **Registery.dll** Windows kayıt defterinde yapılacak olan işlemler burada kodlanmıştır.
- **Sabitler.dll** Sistemde kullanılan ve pek çok yerde karşımıza çıkan işlemler bu bölümde yapılandırılmıştır.
- **Hesapla.dll** Hesaplama işlemlerinin çoğunluğu bu dosyada kodlanmıştır.

DLL dosyaları dışında bazı hazır bileşenler de kullanılmıştır. Bu bileşenlerden ilki Rave Report bileşenidir. Bu bileşen kullanılarak listeler, ana/detay raporlar, çapraz raporlar (crosstab) gibi çok farklı raporlar hazırlanabilir [3]. Otomasyondaki raporlar Rave Report bileşeni kullanılarak tasarlanmıştır.

Kullanılan bir başka bileşen TMSSoftware tarafından üretilen bileşen paketidir. Bu paket içerisinde ekran tasarımları ve veritabanı bağlantıları için kullanımı kolay ve görsel bileşenler içerir. Yazılımdaki bazı ekranların tasarım işlemlerinde bu bileşenler kullanılmıştır.

Bir diğer bileşen paketi de JCL (Jedi Code Library) – JVCL (Jedi Visual Code Library) bileşen kütüphanesidir. Bu kütüphanede de yüzlerce tasarım ve iletişim bileşenleri yer almaktadır. Yazılım içerisindeki bazı ekranların tasarımında ve çevre cihazlarla iletişim işlemlerinde bu kütüphaneden faydalanılmıştır.

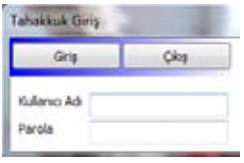
3. Otomasyon

Tüm verilerin tek bir merkezde toplanması, yazılımın tek olması amacıyla bu otomasyon geliştirilmiştir.

Tasarlanan yazılım iki ana modülden oluşmaktadır. Bunlar, yönetici ve kullanıcı modülleridir.

3.1. Yönetici Modülü

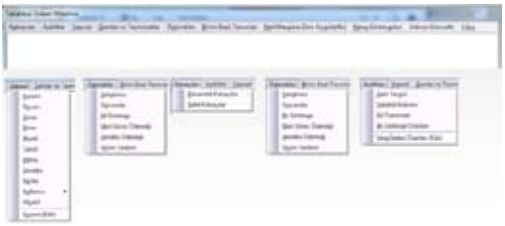
Kurum içerisinde Personel Daire Başkanlığında görevli müdür ve yetkili şefler için tasarlanmıştır. Bu görevliler, tasarlanan bu modül sayesinde tüm hesaplamalarda kullanılan tanım verilerini ekleme, silme ve güncelleme yetkisine sahiptirler.



Şekil 3.1. Yönetici modülü giriş ekranı.

Yönetici modülüne giriş işlemi şekil 3.1'de görülen ekrandan yapılmaktadır. Bu ekranda yetkili personel hem kullanıcı adını hem de parolayı girmek zorundadır. Güvenlik açısından burada kullanıcı isimleri ekrana getirilmemiştir.

Modülün ana menüsü şekil 3.2'de görüldüğü üzere pulldown menü olarak tasarlanmıştır. Menü tasarımı yapılırken birbiriyle ilgili olan işlemler bir menü başlığı altında toplanmıştır.



Şekil 3.2. Yönetici modülü ana ekranı.

Menüdeki herhangi bir seçenek tıklandığında gelen ekran şablonu şekil 3.3'te görülmektedir. Bu şekilden de görüldüğü gibi ekranların genel tasarımında pencere başlık çubuğunda ilgili ekranın küçük bir açıklaması, hemen altında, kullanılabilecek olan işlem butonları yer alır. Butonların altında ise veri girişi yapılabilecek metin alanları, açılır kutular ve seçenekler yerleştirilmiştir. Bu alanın altına da gridler yerleştirilerek veriler görüntülenmiştir. Dikkat edilirse işlem butonlarının arasına silme işlemi için gerekli buton yerleştirilmemiştir. Silme işlemi tüm ekranlarda farenin sağ butonuna atanan kısayol menülerine eklenmiştir. Ayrıca grid üzerindeki veriler grid başlığına tıklatılarak yeniden sıralanması sağlanmıştır. Hem iki modülde de tasarım bu ilkelere göre yapılmıştır.



Şekil 3.3. Unvan tanım ekranı.

3.2. Kullanıcı Modülü

Kullanıcı modülü memurların işlem yapacakları modüldür. Bu modülün girişi şekil 3.4'te görülen ekran aracılığı ile gerçekleştirilmektedir. Bu ekranda kullanıcıların isimleri ekrana getirilmekte ve kullanıcıdan sadece parolasını girmesi ve yapacağı işlem tipini seçmesi beklenmektedir.



Şekil 3.4. Kullanıcı girişi.

3.2.1. Yetkilendirme

Kullanıcı modülünde birden fazla yetkilendirme yöntemi kullanılmıştır. Bu yetkilendirme işlemleri kullanıcının hata yapmasının önüne geçilmesi için tasarlanmıştır.

İlk olarak kullanıcılar yazılım içerisinde yönetici ve kullanıcı olarak ayrılmıştır. Bu yetkilendirme seviyesi ile normal kullanıcıların yönetici modülüne giriş yapmasına izin verilmemiştir. Böylece normal bir kullanıcının tanımlara ulaşması engellenmiştir. Yönetici olarak giriş yapıldığında her iki modülde de tüm kısıtlamalar kaldırılmaktadır.

İkinci olarak kullanıcılara sadece görev yapacakları modüller tanımlanmıştır. Başka bir deyişle bir personel fazla mesai işleminde yetkili ise şekil 3.4'te görülen giriş ekranında modül kutusunda sadece "Fazla Mesai" seçeneği getirilmiştir. Eğer personel birden fazla işlem yapmaya yetkili ise giriş ekranındaki modül kutusuna bu modüller getirilmiştir. Bu sayede kullanıcının kendisini ilgilendirmeyen modülleri görmesi ve üzerinde işlem yapması engellenerek hataların önüne geçilmiştir.

Üçüncü seviye yetkilendirme ise giriş ekranı geçildikten sonra oluşturulan seviyedir. Bu yetkilendirmede kullanıcıya hangi ekranlarda yetki verilmiş ise şekil 3.5'te görülen programın ana ekranında yer alan ağaç menüsünde sadece o ekranları içeren menüler getirilmekte ve başka ekranlara girişi engellenmektedir.

Dördüncü yetkilendirme seviyesinde ise her kullanıcıya yetkili olduğu birim tanımlanmıştır. Böylece kullanıcının başka bir birimdeki personel üzerinde işlem yapması engellenmiştir. Kullanıcı bu yetkilendirme işlemi ile diğer birimlerdeki personeli görebilmekte ancak üzerinde işlem yapamamaktadır. Ancak bazı işlemlerde, fazla mesai hesaplaması, sınav kâğıdı okuma işlemleri gibi durumlarda bu yetki ilkesine bağlı kalınarak kişisel bazda bazı şilemler üzerinde kısıtlı da olsa yetki tanınmıştır.

Son yetkilendirme seviyesinde ise her kullanıcıya, ulaşabildiği ekran üzerinde de yetki tanımlaması yapılmıştır. Bu tanımlama ile kullanıcıya ekranda okuma, kaydetme, silme, yazıcıdan çıktı alma ve export yetkileri tanımlanmıştır. Bu sayede kullanıcıların ekran bazında yapabileceği işlemleri sınırlandırılmıştır.

3.2.2. Kullanıcı Modülü Ana Menüsü



Şekil 3.5. Kullanıcı modülü ana ekranı.



Şekil 3.6. Unvan tanım ekranı.

Şekil 3.5'de yazılımın ana ekranı görülmektedir. Sisteme giren kullanıcının karşısına iki menü çıkmaktadır. Ekranın en üst satırında yer alan menü standart olup sadece sistem içerisindeki tanımlar ve katsayıları göstermektedir. Bu menü yönetici modülündeki menünün aynısıdır. Ancak bu menüden girilen ekranlar bilgi amaçlı olduğundan şekil 3.6'da görüldüğü gibi hiçbir işlem yetkisi yoktur. Ana ekranın solunda yer alan menünün ise ana kısmı standart olup alt menülere geçildiğinde kullanıcının yetki seviyesine göre şekillenmektedir.

3.2.3. Modül Değiştirme

Şekil 3.7'de görülen durum çubuğundaki modül alanına çift tıklatılması durumunda giriş ekranında seçilen modülden başka bir modüle geçiş yapılmasını sağlayan ve şekil 3.8'de görülen ekran getirilir. Bu ekran aracılığı ile kullanıcının programdan çıkmadan istediği modüle geçiş yapması sağlanmıştır.



Şekil 3.7. Durum çubuğu.



Şekil 3.8. Modül değiştirme ekranı.

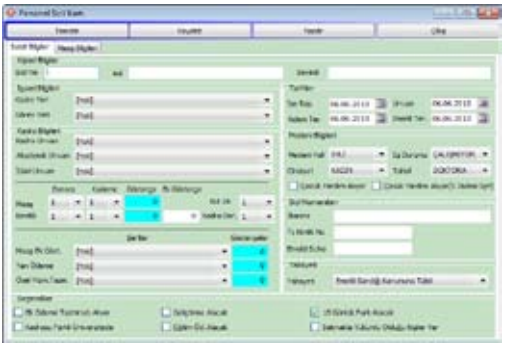
3.2.4. İşlem Ekranlarının Özellikleri

Daha önceki bölümde anlatılan ekran şablonuna ilaveten veri giriş ekranlarında bazı ek ilkeler kullanılmıştır. Şekil 3.9'daki personel sicil kartında görüldüğü gibi veri girişi yapılacak alanlar farklı farklı renklendirilmiştir. Bu tasarımdaki renkler ve anlamları şu şekildedir.

Beyaz renk: Bu alanlar tamamen kullanıcının kontrolündedir. Bu alanlara kullanıcı istediği verileri girebilir. Ancak tarih ve sayısal veri girişi yapılan alanlara metin ifadesi girmesi engellenmiştir.

Gri renk: Bu renkteki alanlar açılır kutular için kullanılır. Bu alanlarda kullanıcı sadece seçim yapabilmektedir. Genel olarak kullanıcı ilgili ekrana girdiğinde bu açılır kutuların içeriği doldurulmaktadır. Ancak bazı ekranlarda ise açılır kutular birbirine bağlı olarak çalışmaktadır. Bu durumda ise ilk kutu seçilmeden kullanıcıya ikinci kutu üzerinde seçim hakkı tanınmamaktadır. Bu ekranlarda kullanıcı ilk kutuyu seçtikten sonra ona bağlı olan ikinci kutunun içeriği yapılan seçime göre doldurulur ve aktif hale getirilir. Her iki halde de kullanıcıya sadece seçim yapma yetkisi verilmiş olup herhangi bir değişiklik ya da yeni bir bilgi girmesi engellenmiştir.

Aqua renk: Bu renge sahip alanlar ise yanlarında bulunan değerlere göre hesaplanarak getirilen alanlardır. Tanımlardaki oranların gösterilmesinde kullanılır. Kullanıcı hiçbir şekilde o alana konumlanamaz ve müdahale edemez.



Şekil 3.9 Sicil kartı ekranı.

3.2.5. Temel İşlem Ekranları



Şekil 3.10 Vergi matrahı izleme ekranı.

Şekil 3.10'da personelin yaptığı tüm ücret işlemlerinden elde edilen GV matrahı ekranı görülmektedir. Bu ekranda kişisel bazda inceleme, kayıt ekleme ve silme işlemleri yapılabilmektedir. Yazılımın haricinde herhangi bir işlemden dolayı GV matrahı oluşmuş ise bu matrah, belirtilen ekrandan GV türü seçilerek girilebilmektedir. Bu ekrandaki silme işleminde ise sadece buradan girilen işlemler için silme yapılabilir. Yani kişinin GV matrahı hesaplama sonucu oluşmuş ise bu ekrandan silme yetkisi yoktur. Ancak hesaplama yapılan ekrandan ilgili hesaplama işlemi silinirse o zaman buradan silinmiş olur. Bu şekilde veri bütünlüğünün kullanıcı tarafından bozulmasının önüne geçilmiştir.



Şekil 3.11. İcra kesinti ekranı.

Şekil 3.11'de personelden kesilecek olan icra tutarlarının girildiği ekran görülmektedir. Personelden yapılacak olan tüm kesinti ekranları bu şekildedir. Kesintinin yapılacağı ilk ay seçildikten sonra kesinti yapılacak personelin bilgisi sicil numarası, as ya da soyadına göre bulunur. Daha sonra kesinti yapılacak toplam tutar, aylık sabit kesinti tutarı ya da oranı girilir. Personel kesintilerinde aynı anda aylık sabit tutar ve oran girilememektedir. Bu iki seçenekten herhangi birine sıfırdan farklı bir sayı girildiğinde diğer alan otomatik olarak sıfır yapılmaktadır. Böylece yazılım tarafından aynı anda hem oran hem de tutar bilgisi girilmesinin önüne geçilmiştir.



Şekil 3.12. Ücret hesaplama ekranı.

Şekil 3.12'de görülen ücret hesaplama ekranı genel bir ekrandır. Kullanıcı hangi modülü seçerse seçsin hesaplama işlemine girdiğinde karşısına bu ekran gelmektedir. Bu ekran aracılığı ile her türlü hesaplama işlemi gerçekleştirilmektedir. Burada kişi ya da birim bazında işlem yapılabilmektedir. İşlem yapılacak ay ve katsayı dönemi seçme izni vardır. Personel her ne kadar işlem yapabileceği ayı seçebilse de geçmişe yönelik işlemleri yönetici onaylamış ise silme ya da yeniden hesaplama işlemlerini yaptıramaz. Onaylanmış bilgiler üzerinde sadece görme yetkisine sahiptir.

4. Raporlama ve Listeler

Otomasyon içindeki ekranların yaklaşık olarak %95 oranında liste ve raporlamalar alınabilmektedir.

4.1.Veri Giriş Ekranlarında Listeleme

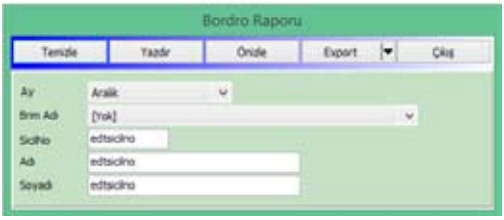
Veri giriş ekranlarının tamamında liste alınabilmektedir. Yazıcıya gönderilen bu listeler aynı zamanda ön izleme yapılabilmektedir. Ayrıca istenildiğinde bu listeler şekil 4.1'de görüldüğü gibi Word, excel, pdf olarak da listeler alınabilmektedir.



Şekil 4.1. Liste ekranları

4.2. Raporlama Ekranlarında Alınan Raporlar

İşlem sonuçlarında ilgili birimlere iletilmek üzere alınan raporlar hazırlanmıştır. Bu raporların yanı sıra zaman zaman gerekli olan ya da yılda bir kere kullanılan özel raporlar da mevcuttur. Listelerde olduğu gibi bu raporlar da yazdırılabilir, ön izlenebilir ve export edilebilir durumdadır. Raporların giriş ekranı genel olarak şekil 4.2 deki gibidir. Şekil 4.3'te ise oluşturulan bir rapor görülmektedir.



Şekil 4.2 Raporlama giriş ekranı



Şekil 4.3. Rapor ön izleme sonucu

5. Sonuç

Bu projede mevcut sistemin yetersiz kaldığı durumlar göz önüne alınarak farklı bir bakış açısı ile yeni bir yazılım geliştirilmiştir.

Bu yazılımdaki tasarımlarla kullanıcılar rahat bir kullanıma ulaştırılmış, kullanıcının yapacağı işlemler neredeyse otomatik hale getirilmiştir. Yazılımın genelinde uygulanan yetki kontrolü ve veri girişlerinde uygulanan ilkeler sonucunda kayıt ve silme gibi işlemlerden önce yapılan son kontrollerle kullanıcının hatalı veri girişinin önüne geçilmiş ve veri güvenilirliği arttırılmıştır.

Veriler tek merkezde toplandığı için veri tekrarı ortadan kaldırılmaktadır. Veri tekrarının ortadan kaldırılması, veri girişindeki hata ve farklılıkların da ortadan kalkmasını sağlamıştır.

Tek merkezden yönetim sayesinde veriler ara-

sındaki tutarsızlıklar da ortadan kaldırılmış, raporların güvenilir ve hızlı bir şekilde alınması sağlanmıştır.

Yazılım güncelleme sayısı fazlalığı ortadan kaldırılmıştır. Bu da güncellemelerin yazılımcılar açısından daha rahat yapılmasına, zamandan kazanılmasına, kullanıcılar için de işlerin zamanında yapılmasını kolaylaştırmıştır.

Ortaya çıkarılan tek tip veritabanı sayesinde personel maaşları daha basit ve daha kolay bir web sayfası yardımıyla ilgili personelin görebileceği şekilde web ortamında yayınlanması imkanı oluşmuştur.

Personelin her türlü kazançlarının bir birimden diğer birime bildirilmesi sorunu ortadan kaldırılmıştır. Bu sayede vergi kayıpları, icra takibi gibi sorunların önünde geçilmiş, kurumun ve kişilerin haksız kazanç ve fazla kesintilerin yapılmasının önüne geçilmiştir.

Tüm bu işlemler sayesinde kurumda çalışan her türlü personel için büyük kolaylıklar oluşturulmuştur. Ayrıca hesaplamalar sonucu ortaya çıkan kayıplar ve hatalı hesaplamaların önüne geçilmiştir, kurum ile personel arasındaki mahkemelik durumların da ortadan kaldırılması sağlanmıştır. Dolayısı ile kurumda çalışan yönetici, tahakkuk şubesi personeli ve diğer tüm personelin memnuniyeti arttırılmıştır.

- Çalışmanın bundan sonraki aşamalarında ise şu işlemlerin yapılması öngörülmektedir.
- Tüm bu ücret hesaplamalarının personelin her yerden görebilmesi için bir sayfa tasarlanmalıdır. Bu sayfada kullanıcının seçtiği ay içerisinde elde ettiği tüm gelir çeşitleri gösterilmeli ve seçtiği ücret türüne göre ilgili veriler ekrana yansıtılmalıdır.
- Farklı üniversitelerde ders veren ya da bir şekilde görevlendirilen personelin orada elde ettiği verilerin girilebilmesi için ve buradaki bazı bilgilerinin oraya aktarılabilmesi için web üzerinden veri giriş çıkışını sağlayan bir servis hazırlanmalıdır.
- Personel Daire Başkanlığının kullandığı yazılım ile irtibatlı hale getirilmeli ve veri girişlerinin daha da azaltılarak otomatik veri alınması sağlanmalıdır.
- Ders veren personel için ileride öğrenci işleri otomasyonları ile bağlantı kurulmalı ve personelin ders verdiği saatler ile sınava giren öğrenci sayıları öğrenci işleri otomasyonundan çekilmelidir.
- İleriki dönemlerde yazılım tarafında yöneticinin ilgili ücret hesaplamalarını onayladığı anda kişilerin elektronik posta adreslerine ücret bordroları gönderilmelidir.
- Web sayfasında yayınlanan ücret bilgileri akıllı telefonlarda kullanılabilir hale getirilmelidir.

6. Kaynaklar

[1] Çiçek, M., "Veritabanı Tasarımı ve SQL Sorgulama Dili", ***Nirvana Yayınları***, Ankara, 5-22 (2010).

[2] Kayık, M., "Delphi programlama dili ile yükseköğretim kurumlarında sınav programlarının düzenlenmesine yönelik bir model önerisi ve Dumlupınar Üniversitesi İktisadi ve İdari Bilimler Fakültesi'nde uygulama örneği", Yüksek Lisans Tezi, ***Dumlupınar Üniversitesi Sosyal Bilimleri Enstitüsü***, Kütahya, 23 (2006).

[3] Kayık, M., "Delphi programlama dili ile yükseköğretim kurumlarında sınav programlarının düzenlenmesine yönelik bir model önerisi ve Dumlupınar Üniversitesi İktisadi ve İdari Bilimler Fakültesi'nde uygulama örneği", Yüksek Lisans Tezi, ***Dumlupınar Üniversitesi Sosyal Bilimleri Enstitüsü***, Kütahya, 38 (2006).