

Web Tabanlı Hibrit Bir Uygulama Modeliyle Personel Bilgi Sistemi Tasarımı

Zafer CÖMERT¹, A. Fatih KOCAMAZ², Musa ÇIBUK¹

¹ Bitlis Eren Üniversitesi, Mühendislik-Mimarlık Fakültesi, Bilgisayar Mühendisliği

² İnönü Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği

zcomert@beu.edu.tr, fatih.kocamaz@inonu.edu.tr, mcibuk@beu.edu.tr

Özet: Personel Bilgi Sistemi, üniversite personellerinin web ortamında kendilerine ait bir alan adı üzerinden modüler bir yapı ile bilgi paylaşmasını sağlayan ve personel ile site ziyaretçileri arasındaki iletişimi ve etkileşimi arttıran web tabanlı bir otomasyondur. PBS, güncel web teknolojileri ve nesne yönelimli programlama teknikleri kullanılarak hazırlanmıştır. Uygulama, hem kendi üzerindeki verileri başka uygulamalar ile paylaşabilecek hem de başka uygulamalar üzerindeki verileri alarak yeni içerikler üretebilecek nitelikte hibrit bir model mantığı ile geliştirilmiştir. Bu kapsamda, SOAP/WSDL web servisleri senkron ve asenkron olarak kullanılmış ve uygulamanın AD-LDAP entegrasyonu yapılmıştır. Uygulamanın blog, kişisel web sayfaları gibi sistemler ve Google Maps ile birlikte çalışması ayrıca bilgi paylaşımı için her bir modülün kullanıcı tercihinin bağlı olarak RSS'ler üretmesi sağlanmıştır. PBS hibrit modelin dışında tema desteği, akıllı URL yapısı ve çoklu dil tercihleriyle ön plana çıkmaktadır.

Anahtar Sözcükler: Web Tabanlı Personel Bilgi Sistemi, Web Servisleri, Nesne-Yönelimli Programlama

Abstract: Personnel Information System, is a web-based automation that provides to share information, to enhance communication and interaction between site visitors and personnel with modular structure on their own domain name. PIS has been prepared by using current web technologies and object-oriented programming techniques. The application has been developed with a quality that both can be shared its own information with other applications and can be produced new content by getting new information from other applications in hybrid model logic. In this context, SOAP/WSDL is used as synchronous and asynchronous and AD-LDAP integration of the application has been made. The application has been provided to work together with such as blog, personal website and Google Maps, furthermore it has been made to produce RSS depending on the preferences of each user module for information sharing. PIS comes to fore with its theme support, smart URL structure and multi-language preference without the hybrid model.

Keywords: Web-based Personnel Information System, Web Services, Object-Oriented Programming

1. Giriş

Yazılımlar, günümüz koşullarında eğitimden alışverişe, bankacılık işlemlerinden kamuya uzanan geniş bir yelpazede kullanılmaktadır. Yazılımlar sıklıkla farklı platformlarda, farklı yazarlık dilleri kullanılarak, farklı işletim sistemleri üzerinde, farklı teknolojiler ile geliştirilmekte ve farklı iletişim protokolleri kullanmaktadır. Bu dağıtık ortamda hemen hemen tüm sistemler 80 numaralı portunu açarak birbirleriyle konuşmaya başlamıştır. Son dönemde kişisel bir bilgisayar ya da yerel bir ağda çalışan yazılımlar, kesintisiz hizmet veren bulut teknolojileri üzerinde küresel bir ağa doğru taşınma eğilimi göstermektedir. Bu sürecin en önemli bileşenlerini ise web teknolojileri oluşturmaktadır. Uygulamalar arasında yetki ya da rol tabanlı veri erişimi ve paylaşımını sağlamak aynı zamanda bu işlemi teknolojik platformlardan soyutlayarak yapmak için tercih edilen en yaygın yöntem web servislerdir [1]. Teknik bir perspektif açısından web ser-

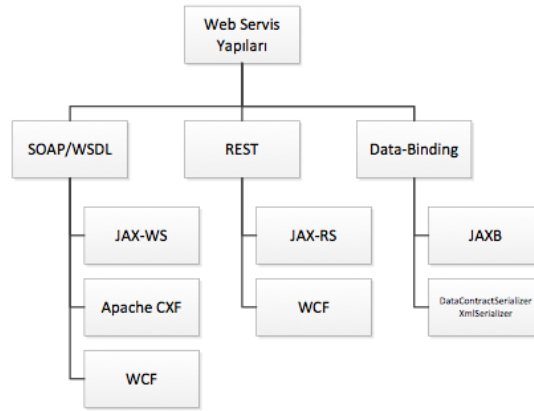
vis kavramı, oturum açma ve yetkilendirme gibi bir görevi yerine getirmek üzere tasarlanan herhangi bir yazılım fonksiyonunun farklı sistemler tarafından ve HTTP üzerinden kullanımını ifade etmektedir. Web servislerinin her an erişilebilir olması, güvenlik açıklarına neden olmaması, oluşan yükü dengelemesi, zamanında cevap verebilmesi, gelen isteklerin kaybolmaması, genişleyebilir ve sürdürülebilir bir nitelik taşıması için önceden belirlenmiş bir tasarım modeline göre hazırlanması gerekir [2]. Bu amaç için en uygun geliştirme yöntemi nesne yönelimli programlamadır (NYP). Yazılımların karmaşıklığı ve boyutlarının sürekli olarak artmasına karşın 1960'lı yılların sonuna doğru NYP ile bilgisayarlı programlama alanında yeni bir yaklaşım geliştirilmiştir. Bu yaklaşım pek çok modern programlama dili tarafından da desteklenmektedir. Dinamik sevk, sarma (kapsülleme), çok şekillilik, kalıtım ve kendisini çalışma zamanında bilme gibi nitelikleriyle NYP ön plana çıkmaktadır. NYP

yapısından kaynaklanan sorunlar ise tasarım modelleri ile çözülmektedir [3]. NYP günümüzde masaüstü, web ve mobil uygulamalarda sıklıkla kullanılmakla birlikte modern yazılım mimarisinin önemli bileşenlerinden biri olmuştur.

2. Web Servisleri

Web servis kavramı, oturum açma ve yetkilendirme gibi bir görevi yerine getirmek üzere tasarlanan herhangi bir yazılım fonksiyonunun farklı sistemler tarafından ve HTTP üzerinden kullanımını ifade etmektedir. Bir başka ifadeyle

web servisleri, yerel nesnelerin dağıtık nesnelere haline gelmesini sağlayan teknolojidir. Web servisleri mobil, masaüstü ve web uygulamaları gibi farklı türdeki istemcilerin ortak paylaşım noktasını oluşturmaktadır. Tüm web servisleri, asgari olarak HTTP ve veri-değişimi için XML veya JSON gibi ortak medya türlerini kullanmak durumundadır. Şekil 1’de teknoloji tercihleri göz önüne alındığında birkaç farklı yapının ön plana çıktığı görülmektedir [4]. Ayrıca verimli bir web servisi oluşturmak için JSON, XML, XSDL, XPath, XSLT ve WSDL teknolojileri de hakim olmalıdır.

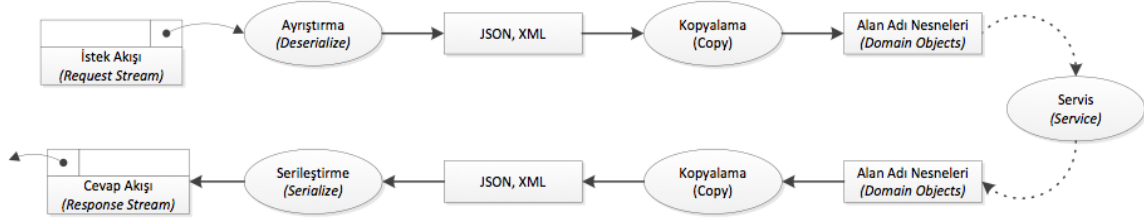


Şekil 1 Web Servis Yapıları

Servis-Yönelimli Mimaride (SOA, Service-Oriented Architecture), uygulama detayları gizlenmelidir ve istemciler asla servisin veri tabanı ile olan ilişkisini bilmemelidir [5]. Bunu sağlamak amacıyla NYP'nin özelliklerinden kapsülleme kullanılmalıdır. Servis ve istemci geliştiricileri tarafından web servislerindeki olası aksamalar dikkate alınmalı, ağ bağlantısı, cevaplama ve yorumlama sürelerindeki gecikmeler, sunucu yoğunluğu gibi kısmi başarısızlık senaryoları oluşturulmalıdır. Verilerin ikili (binary) olarak kodlanması ve iletilmesi bu anlamda önemli bir avantaj sağlayacaktır. Gecikme, ağ trafiği kaynaklı sorunları nispeten engelleme ve servis cevaplarının uygun sürede verilmesini sağlamak üzere önerilen bir diğer yöntem de Proxy metodlarının uygulanmasıdır. Web servisleri tasarlanırken özel bir etkileşim stili düşünülmelidir. Çıkış noktası, servis ile istemcilerin nasıl iletişim kuracağıdır. Servis-istemci etkileşim modelleri:

1. İstek/Cevap (Request/Response)
2. İstek/Bilgi (Request/Acknowledge)
3. Medya Türü Değerlendirmesi (Media Type Negotiation)
4. Bağlantılı Hizmetler (Linked Service)

Bu noktada, en temel yaklaşım servis ile istemci arasında noktadan-noktaya iletişimidir. Servis-istemci etkileşimi içinde istek/cevap modeli yaygın olarak kullanılmaktadır. Bu model istemcilerin servis üzerinde hızlıca ve bekleme olmaksızın genel bir fonksiyonun çalıştırılması ya da bir görevin hemen yaptırılması için kullanılır. İstek/Bilgi modelinde, istemci servis ile iletişim kurduğunda isteğin doğrudan çalıştırılması gerekli değildir. İstek/Bilgi modeli, sistem kaynaklarının (işlemci, disk, bant genişliği, ağ trafiği, veri tabanı bağlantı sayısı gibi...) verimli kullanılmasını sağlamak, aşırı kaynak tüketimini engellemek ve anormal durumların oluşmasını önlemek için idealdir. Bu modelde, servis bir istek aldığı anda isteği arka planda çalıştıracak şekilde iletir, daha sonra isteği tanımlayan tekil bir bilgiyi istemciye döndürür, gelen isteği kuyruğa alıp bir süre sonra cevabı oluşturur. Medya türü değerlendirme modeli, genellikle kaynak API'ler ile birlikte kullanılır. Amaç farklı medya türlerini benzersiz URI formatları ile döndürmektir. Bağlantılı servislerde gelen istek alındıktan sonra bir



Şekil 2 Veri Transfer Nesnesi

adres ile geri dönüş yapılır ve istemci gerekli ayrıştırma işlemini yaptıktan sonra bir başka servise yönlendirilir [6]. Neticede istemci tarafından yapılacak işlemlerle ilgili olarak servis yerleri ve URL modelleri keşfedilebilir.

Web servislerinde bir diğer önemli konu istek ve cevapların yönetilmesidir. İstek ve cevapların yönetilmesi için Servis Denetçisi (Service Controller) bir ön kontrol mekanizması oluşturur. Servis denetçisi gelen isteklerin JAX-WS, JAX-RS, Axis2 ve WCF gibi yapısını (framework) kontrol eder ve yapıya uygun işleyiciyi (handler) çağırır. Veri transfer nesneleri (VTN, Data Transfer Object), bir web servisin kullandığı JSON veya XML yapısını istek ve cevap içerisinde aramaktadır. VTN, bir web servisine ulaşan verinin ve döndürülen verinin nasıl tanımlanması gerektiğini ifade etmektedir. Şekil 2’de bir VTN’nin operasyon süreci gösterilmektedir. VTN, üzerindeki nesnenin kopyasını oluşturarak o kopyayı kolayca ulaşılabilir bir forma çevirme ve nesnenin o anki durumunu koruma işlemlerini gerçekleştirir. Bu da verilerin farklı uygulamalarda kullanılmasına, verilerin ağ üzerinde kolayca dolaşmasına ve paylaşılmasına imkân verir.

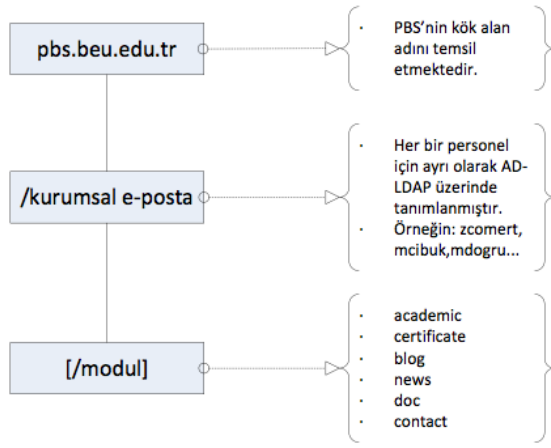
3. Personel Bilgi Sistemi

Personel Bilgi Sistemi (PBS), üniversite personellerinin web ortamında kendilerine ait bir alan adı üzerinden modüler bir yapı ile bilgi paylaşmasını sağlayan ve personel ile site ziyaretçileri arasındaki iletişimi ve etkileşimi arttıran web tabanlı bir otomasyondur [7]. PBS kişisel bilgiler, eğitim bilgileri, özgeçmiş, akademik faaliyetler, sertifikalar, blog, içerik yönetimi, belge yönetimi, iletişim ve mesaj servis modüllerinden oluşmaktadır. Akıllı URL yapısı, zengin dokümantasyonu, tema desteği, çoklu dil tercihleri (Türkçe/İngilizce) ve uygulama ekranlarına benzer nitelikli yönetim paneli PBS’nin diğer önemli özelliklerini oluşturmaktadır.

4. Teknik Bilgiler

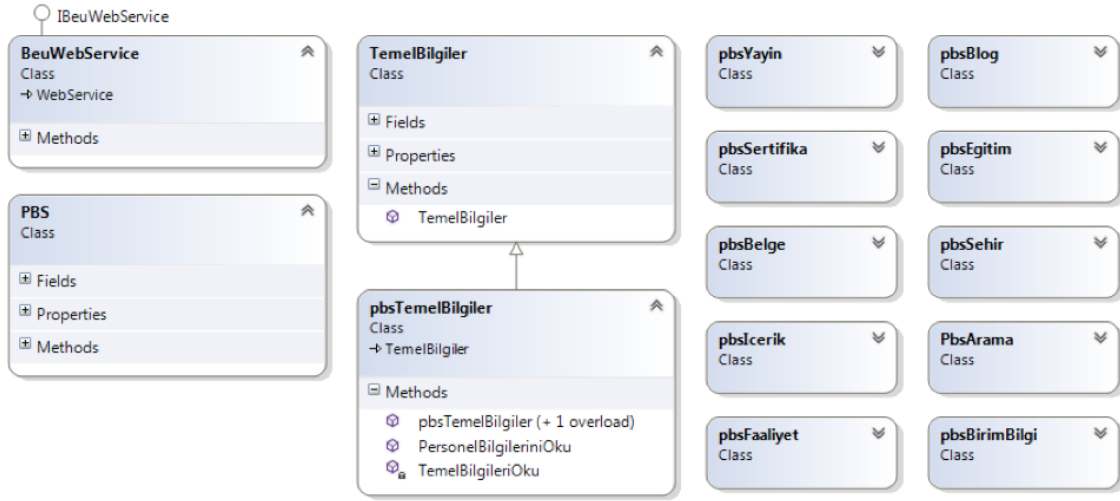
Sistem, ASP.NET C# dili kullanılarak Net Framework 4.5 çatısı üzerine inşa edilmiştir. Veri alış-verişi için WCF (Windows Communication Foundation) platformu tercih edilmiştir. WCF, Net Framework ile yazılan uygulamaların iletişim katmanını olmayı amaç edinen; Net Remoting, XML Web Servisleri, MSMQ, Net Pipe, TCP/IP gibi pek çok farklı yöntemin tek bir çatı altında topladığı bir kütüphanedir [8]. WCF hizmetiyle bir servisin adresi, veri bağlama tekniği ve servisin içerdiği fonksiyonlar tanımlı hale gelmektedir [9]. Sistemde, SOAP (Simple Object Access Protocol) web servisleri senkron ve asenkron şekilde kullanılmıştır. SOAP web servisleri ile Bitlis Eren Üniversitesi web sitesi (BEU-WEB), üzerindeki yerel nesnelerin PBS üzerinde kullanılması sağlanmıştır. Uygulamanın AD (Active Directory) – LDAP (Lightweight Directory Access Protocol) entegrasyonu gerçekleştirilmiştir. AD-LDAP yapısı ile kurum personellerine kurumsal e-posta adresleri ve çeşitli hizmetler sunulmaktadır. Bu entegrasyon sayesinde kullanıcıların kurumsal e-posta adresleri ve şifreleriyle sisteme giriş yapmaları sağlanmış ve aynı zamanda kurumsal e-posta adreslerinin alan adı olarak kullanılması sağlanmıştır. Bu entegrasyonun diğer avantajları ise e-posta hesabı kapatılan kullanıcıların sisteme girişlerinin otomatik olarak engellenmesi, sistemden otomatik olarak kaldırılması ve şifre karmaşasının önüne geçmesidir. XmlReader nesnesi ile RSS (Really Simple Syndication) standartlarında çıktı üreten sistemlerin (blog, kişisel web sayfası gibi) PBS üzerinde içerik olarak sunulması sağlanmıştır. Yine aynı standart kapsamında akademik faaliyetler ve içerik modüllerinin RSS üreterek tercihe bağlı olarak farklı sistemlerle konuşabilmesi sağlanmıştır.

Sistemin bir diğer bileşeni haritalardır. Google Maps API kullanılarak sistemin haritalarla birlikte otomatik olarak içerik üretebilmesi sağlanmıştır.

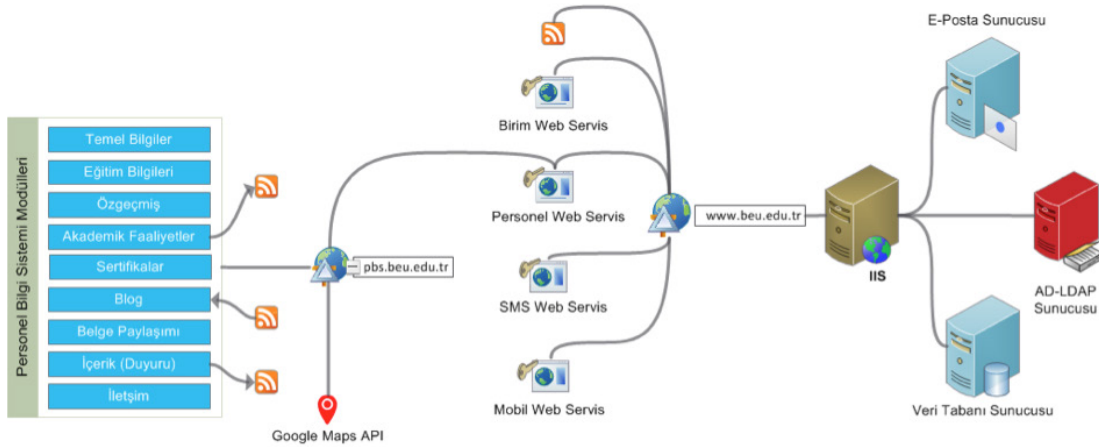


Şekil 3 PBS URL Yapısı

Akılda kalıcı, kolay okunabilir, anlamlı ve arama motorları tarafından kolayca endekslenilebilir bir URL yapısı Şekil 3'de gösterildiği gibi tasarlanmıştır. İçeriklerin tercihe bağlı olarak sosyal ağlar üzerinden paylaşılması için bir eklenti hazırlanmıştır. Erişilebilirliği arttırmak için içeriklerin daha rahat okunmasını sağlayan Reader modülü tasarlanmıştır. Ayrıca içeriklerin bir plan dahilinde (başlangıç-bitiş tarihleri önceden belirlenerek) yayımlanabilmesi sağlanmıştır. Daha önce BEU-WEB üzerinde bir kaç sayfadan oluşan PBS, ihtiyaç analizi ile başlayan ve kendine ait bir alan adı üzerinden pek çok teknolojinin bir araya getirilerek hazırlandığı bir sisteme dönüşmüştür. Sistemin sınıf diyagramına ve genel mimarisine sırasıyla Şekil 4'te ve Şekil 5'de yer verilmiştir.



Şekil 4 PBS Sınıf Diyagramı



Şekil 5 Sistemin Genel Mimarisi

5. Sonular ve Gelecek alıřmalar

Yazılımlar, günümüz kořullarında geniş bir yelpazede kullanılmaktadır. Bununla birlikte özelleřtirilmiş pek ok bilgi internet ortamında farklı platformlar üzerinde farklı veri tabanlarında saklanmaktadır. Yazılımların ihtiya duyduėu bilgileri saėlıklı web servisleri ile dıř ortamdan alabilmesi ve kendi üzerindeki verileri önceden belirlenmiř kurallar dahilinde dıř ortamlarla paylařabilmesi, yazılım geliřtirme maliyetlerinin düşmesi, tutarlı bilgilerin elde edilmesi ve veri giriři tekrarlarının engellenmesi hususunda pozitif katkılar sunmaktadır. Dahası bu veri paylařımının teknolojik tercihlerden soyutlanarak yapılması amacına göre pek ok farklı teknolojinin kullanılabilmesine imkan tanımaktadır.

Bu anlamda PBS, BEU-WEB üzerinde hazırlanmıř nesneleri bir web servisi üzerinden alıp kullanılmaktadır. Bu durum PBS'nin geliřtirilme süresi ve maliyetlerini ciddi derece azaltmıřtır. Ayrıca oturum açma ve yetkilendirme iřlemleri için AD-LDAP entegrasyonun yapılması kullanıcıların sisteme daha rahat girebilmesini ve katılımcı bir yapının oluřmasını saėlamıř ve olası řifre karmařıklıėını engellemiřtir. Kullanıcıların sisteme eklenmesi, silinmesi ya da řifre deėiřtirme/unutma durumları gibi pek ok operasyon e-posta sunucusu üzerinden yürütüldüėünden PBS için bu modüllerin yeniden yazılması söz konusu olmamıřtır. AD-LDAP üzerinde açılan hesapların alan adı olarak kullanılması ve uygulamanın farklı sistemlerle konuřarak ierik üretebilmesi saėlanmıřtır. Böylelikle kiřisel bir web sayfası ya da bir blog üzerinden üretilen ieriklerin doğrudan PBS uygulama ekranlarında gösterilebilmiřtir. Ayrıca bu yapıyla verilerin farklı řekillerde sunulmasına mümkündür. PBS-Google Maps API entegrasyonu ile kullanıcıları doğrum yerleri ve bulundukları kampüs alanına göre bir harita üzerinde gösterilebilmeleri saėlanmıřtır.

Özelde üniversite personelleri için tasarlanan bu sistemin gelecekte üniversite personellerini yakından ilgilendiren Yükseköėretim Bilgi Sistemi (YÖKSİS) ve TUBİTAK Arařtırmacı Bilgi Sistemi (ARBİS) ile birlikte alıřmasını saėlayacak servisler konusunda arařtırmalar yapılacaktır.

Kaynaka

- [1] R. Daigneau, Service Design Patterns, Westford, Massachusetts: Pearson Education, 2012.
- [2] J. Löwy, Programming WCF Services, Sebastopol: O'Reilly Media, 2010.
- [3] J. Purdum, Beginning Object Oriented Programming with C#, Indianapolis: John Wiley

& Sons, 2013.

- [4] K. Elgazzar, A. E. Hassan ve P. Martin, «Clustering WSDL Documents to Bootstrap the Discovery of Web Services,» içinde IEEE International Conference on Web Services, 2010.
- [5] F. Satoh, M. Tatsubori, Y. Nakamura ve N. Mukhi, «Methodology and Tools for End-to-End SOA Security Configurations,» içinde IEEE Congress on Services, 2008.
- [6] J. P. Thomas, M. Thomas ve G. Ghinea, «Modeling of Web Services Flow,» içinde Proceedings of the IEEE International Conference on E-Commerce, 2013.
- [7] M. ıbuk, «AKADEMİK PERSONEL BİLGİ SİSTEMİ,» e-Journal of New World Sciences Academy, cilt 4, no. 4, pp. 601-611, 2009.
- [8] A. M. Riad, A. E. Hassan ve Q. F. Hassan, «Design of SOA-based Grid Computing with Enterprise Service Bus,» International Journal on Advances in Information Sciences and Service Sciences, cilt 2, no. 1, pp. 71-82, 2010.
- [9] F. Boy, «WCF'e Giriř,» 2 12 2010. [evrimii]. Available: <http://www.enterprisecoding.com/post/wcf-giris>. [tarihinde eriřilmiřtir 10 12 2014].