



Erciyes University Journal of the Institute of Science and Technology

Erciyes Üniversitesi Fen Bilimleri Enstitüsü Dergisi

ISSN 1012-2354

Cilt (Volume): 30, Sayı (Issue): 4, Eylül/September-2014

<http://fbe.erciyes.edu.tr/>



Yeni bir web-tabanlı dinamik portal ve portlet mimarisi

Mustafa Ulvi¹ ve Ahmet Sayar¹

¹Bilgisayar Mühendisliği Kocaeli Üniversitesi Umuttepe-İzmit,41380, Kocaeli-Türkiye

ÖZET

Portal mimarileri çeşitli sistemlerden gelen uygulamaların ve içeriklerin bir araya toplanıp kullanıcıya tek web arayüzünde sunulmasını sağlar. Portlet ise portaldaki paketlenmiş çalışan uygulamadır. Bu makalede JAVA teknolojileri kullanarak, MVC'ye dayalı küçük ölçekli portal/portlet çatısı önermekteyiz. Bu çatı birçok programın tek yönetim domaininde çalışmasına imkan verir. Ayrıca, farklı erişim haklarına sahip birçok kullanıcı guruplarının yaratılması/yönetilmesi ve tek kişilik oturum açma (single sign-on) prensibiyle çalışır.

Anahtar

Kelimeler:

Portal, Portlet,
JSF, JPA.

A novel web-based dynamic portal-portlet architecture

ABSTRACT

A portal provides a solution for aggregating content and applications from various systems into a single unified web presentation to the end user. A portlet is a relatively small wrapped application running in a portal container. In this paper, we propose a small scale MVC based portal/portlet framework by using JAVA technologies. The framework enables managing and running multiple programs under the same administrative domain, and enables creation of multiple user groups with different access rights and single sign-on.

Key Words:

Portal, Portlet,
JSF, JPA

1. Giriş

Küreselleşmenin etkisi ve teknolojinin yaygın kullanımı sayesinde bilgiye olan gereksinim artmaktadır. Bilgisayar-insan-işletme birlikteliğinin artması, web teknolojilerinin yaygınlaşması, bilgiyi üreten ve tüketen tarafların çoğalması ve karar vermeyi destekleyen süreçlerin karmaşıklığı, bilgiye olan ihtiyacın artmasının ana nedenleridir (Ndou, 2004).

Bilgi yoğun olarak web ortamında üretilmekte ve üretilen bilgiye web üzerinden erişilmektedir. İnternet kullanımındaki artışlar ve dünyanın her yerindeki bilgi üreticileri sayesinde, web üzerinde farklı biçim, dil ve içerikte milyonlarca sayfalık doküman varlığını sürdürmekte, yararlanıcısı tarafından kullanılmayı beklemektedir. Bu süreç bilginin önemi artırırken, aynı zamanda bilginin miktarının da artması ile sonuçlanmaktadır (KURTEL, 2008).

Bilgiye her yerden erişilebilen, veri tabanına dayalı ağ (web) siteleri ile devingen bilgi bankalarından bilgi sunulan, tarayıcı tabanlı; söyleşi odaları, forum alanları, kataloglar, canlı haberler, e-posta ve kişiselleşme gibi birçok özelliği ve etkinliği kapsayan, daha ucuz, hızlı ve güvenli olarak bilgiye ve hizmete erişimi ve entegrasyonu sağlayan ağ hizmetleri, kişilerin tek bir noktadan İnternet üzerinde dağıtık ve düzensiz bir biçimde yayınlanan bilgi yığınlarına, bütünlük bir yapıda, hızlı ve kolayca ulaşılmasını sağlayan portal (Medianero-Pasco, Bonilla-Morales, & Vargas-Lombardo, 2010) yapıları öne çıkmaktadır. Bilgi erişimindeki kolaylık yanında, portal teknolojisi ile bilgi tek bir elden manipüle edilip yönetilebilir.

Kollaborasyona dayalı çeşitli kullanıcı guruplarının oluşturulması ve etkili yetkinleştirme protokolleri ile büyük projeler için amaca uygun iş birlikleri ortamı sağlanmış olur. Önerilen mimari bilginin güvenli ve kolay bir şekilde ortak kullanımında ortaya çıkabilecek sorunlara potansiyel bir çözüm de sunmaktadır. Portallarda Java tabanlı web bileşeni olan portletler ile geliştirilmektedirler ve portlet konteynırlar tarafından yönetilmektedir (Minter & Linwood, 2004). Portletler gelen istemleri işletir ve dinamik bir içerik üretirler. Portallar portletleri bağlanılabilir bir kullanıcı ara yüz bileşeni olarak kullanırlar. Bu da veri sistemleri için bir sunum katmanı sağlar. Web tabanlı uygulama programı geliştirirken servletlerden bir sonraki aşamada portletler kullanıcı merkezli web uygulamalarına olanak sağlar.

Biz bu çalışmamızda dinamik portal ve portlet mimarisini Java Server Faces (JSF) (Burns & Kitain, 2009), Java Persistence API (JPA) (DeMichiel, 2009) ve Apache teknolojilerini (Apache, 2008) kullanarak geliştirmeye çalıştık. JSF, portal ve portlet arayüzlerinin MVC (Model, View, Controller) (Leff & Rayfield, 2001) yaklaşımıyla gerçeklemeye olanak sağladı. JPA ise verilerin ilişkisel veri tabanlarında depolanması ve işlenmesinde etkin altyapı sağladı. Apache'nin Tomcat teknolojisi ile web sunucuların oluşturulması sağlandı.

Makalenin takip eden bölümleri şu şekilde özetlenebilir. 2. Bölümde portal ve portlet teknolojileri hakkında bilgi verilmiştir. 3. Bölüm bu alanda yapılan çalışmaları özetlemektedir. 4. Bölüm de önerilen mimarinin detayları sunulmaktadır. Son bölüm ise yapılan çalışmanın sonuç ve yorumlarını içermektedir.

2. Portal ve portlet

"Portal" sözcüğüne "liman", "ana kapı" gibi Türkçe karşılıklar gösterilse de, diğer pek çok teknolojik terimde olduğu gibi bu kavram da olduğu gibi dilimize yerleşmiştir. Bu terim, genelde pek çok içeriği bir arada bulunduran internet siteleri için kullanılmaktadır. İnternet portallarında canlı haberler, söyleşi odaları, e-posta, alışverişi, çeşitli rehberler ve bunlar gibi birçok etkinlik yer almaktadır. Portallar, "geçit (gateway)" sözcüğü ile genel olarak es anlamlı olarak kullanılan yeni bir terim olup, web'e bağlanan kullanıcılar için ana başlangıç sitesi görevi üstlenen bir web sitesi; ya da kullanıcıların diğer sitelere erişmek için ziyaret ettikleri sitelerdir. Bir diğer tanıma göre ise (<http://Webopedia.com>), çok geniş kaynaklar ile e-posta, forumlar, arama motorları ve çevrim-içi alışveriş mağazaları gibi çeşitli hizmetler sunan bir web sitesidir. İlk web portalları, AOL gibi, web üzerinde erişimli çevrim-içi hizmetler sağlayan sitelerdi. Ancak, bilinen arama motorlarının çoğu daha büyük bir kitleyi çekmek ve korumak için portal şeklinde dönüşüm geçirmişlerdir. İnternet'teki çok geniş bilgi kaynakları ve servislerine anlaşılabılır, kolay giriş noktası olarak hizmet veren süper sitelerdir. Diğer bir tanıma göre de portallar, pek çok içeriği bir arada bulunduran internet siteleridir.

İnternet portallarının web tüketicileri üzerindeki dönüştürücü etkisi gözlemlendikten kısa bir süre sonra kurumların gereksinimlerine yönelik portal uygulamaları gündeme gelmeye başlamıştır. Portallar, çevrim-içi evrenin (İnternet) derinliklerinde kaybolmak istemeyen sıradan kullanıcılar için, karmaşık işlerin basitleştirildiği, bağlantıların organize edilip sınıflanarak sunulduğu çekici giriş noktaları gibi hizmet görmektedirler. Bir başka açıdan bakıldığında, bir portali çok geniş ve değişik profillerden insanlara yayın hizmeti sunan bir TV kanalına benzetebilmek mümkündür. Portallar, sadece bir başlangıç noktası olmasının ötesinde ücretsiz e-posta, web sitesi alanı, sohbet kanalları veya odaları, tartışma listeleri veya gurupları, alışveriş mağazaları ve arama motorları vb. web tabanlı hizmetleri de sunabilmektedirler. Portallar, yukarıdaki ana özellikleri yanında hava durumu, seyahat, turizm, eğlence, yatırım ve finans bilgileri, çevrim-içi oyunlar, borsalar, hava durumu, e-posta, telefon, harita vb. konularda da hizmetler sağlayan siteler durumundadırlar.

Portal ve portletler (1) hedef kitleye göre (2) ağ yapısına göre (3) sektörlere (Kaynağa) göre ve (4) konularına göre olmak üzere 4 ana başlık altında guruplandırılabilir. Konularına göre portallar, eğitim, kültür-sanat, sağlık, turizm, eğlence vb şeklinde çeşitlendirilebilir. Sektörlere göre ise şirket portalları, sivil toplum portalları, kişisel portallar ve devlet-hükümet portalları şeklinde çeşitlendirilebilir. Hedef kitleye göre portallar ise dikey ve yatay olmak üzere iki guruba ayrılır. Dikey portallar bir sektöre yönelik veya özel amaçlı olan portallardır. Yatay portallar ise tüm sektörlere yönelik hizmet gören daha doğrusu İnternet dünyasındaki tüm konuları kapsayan ve yine İnternet toplumunun geneline hizmet veren çok amaçlı portallardır. Portal özellikleri ve nitelikleri üzerinde yapılmış ayrıntılı çalışmaların çoğunda başlıca; zenginlik, erişilebilirlik, etkileşim, süreklilik, güncellik, güvenilirlik, dizin yapısı ve arama, kişiselleştirme, uygulama olanakları, toplumsallık gibi kriterler ön plana çıkmaktadır. Bir portal, mümkün olduğunca aranan bilgi ve/veya kaynaklara ait özetler ve erişim bilgilerini bulmayı sağlayabilecek bir birikime/toplamıma sahip olabilmelidir. Yani, içerik zenginliği portalın en temel özelliklerinden biri olmalıdır.

Diğer yandan portletler ise Java tabanlı web bileşenleridir ve portlet konteynırları tarafından yönetilirler. Portletler gelen istemleri işletir ve dinamik bir içerik üretirler. Portallar portletleri bağlanılabilir bir kullanıcı ara yüz bileşeni olarak kullanırlar. Bu da veri sistemleri için bir sunum katmanı sağlar. Herbir ayrı uygulama portlet temsilleri oluşturularak, aynı web arayüzünde farklı portletler olarak çalıştırılabilir.

Portal oluşturma girişimlerinin artmaya başlamasıyla birlikte birçok satıcı portal bileşenleri için birbirinden farklı yapılar da birçok API'ler oluşturmaya başladılar ve bunların genel adına da portlet denildi. Doğal olarak ortada bu kadar çok çeşitli ve birbirine uymayan arayüzler olunca uygulama geliştiren insanlar, müşteriler ve portal server (sunucu) satıcıları için ciddi sorunlar çıkmaya başladı. Bu problemin üzerinden gelmek için JSR168 (Abdelnur & Hepper, 2003) çalışma gurubu, portallar hakkında servis ve arayüzlerin standartlaştırılması için gerekli olan kurallar ve kısıtlamaları tanımlamışlardır.

Standartlaştırma ve açık kaynak olma sayesinde, yeni bir teknoloji de olsa portal ve portlet uygulamalarının yaygınlaşması çok hızlı oldu. Nesne yönelimli programlamada olduğu gibi iş paketleri (portlet olarak) farklı vendörler tarafından geliştirilmiş farklı portallarda kolayca kullanımı mümkün kılıyor. Bu tekrar kullanım kolaylığı (Reusability) sayesinde, bir çok projenin çok daha kısa bir şekilde gerçekleşmesini ve farklı vendörler arasında işbirliklerinin oluşturmasını kolaylaştırdı.

3. İlgili çalışmalar

Portal kavramı ilk kez, MyYahoo daki kişiselleştirilmiş portal hizmetinin devreye sokulması ile hayatımıza girmiştir. Bu sitede, kullanıcılar kendi ilgi alanlarına göre profil oluşturmakta ve bu profil doğrultusunda kendisine en güncel bilgiler aramaya gerek kalmadan ulaştırılmaktadır. MyYahoo! portalı internet kullanıcıları tarafından heyecanla karşılanmıştır. Haberler, spor, hava durumu ve finansal bilgilerin tek bir sayfada izlenebilmesi ve kişisel profillerin saklanarak, web sayfasının her açılışında kullanıcıya ilgi alanına yönelik en son bilgilerin verilmesi özellikleri sayesinde kullanıcıları kendisine çekmiştir.

Pluto mimarisi (Pluto, 2013), portlet API'lerin nasıl çalıştığını göstererek ve program geliştiricilere çalışan bir platform örneği sunarak (ki programcılar bu ortamda kendi portletlerinin çalışıp çalışmadığını deneyebilirler) hizmet verir. Fakat işin sıkıcı tarafı plutoda portletler sürücüsüz ortamda çalıştırılıp test edilmektedir. Yani portal olmadan sadece portletler test ediliyor. Plutonun basit bir portal bileşeni vardır ama bu bileşen sadece portal konteyner oluşturmak ve JSR168 (JSR168, 2013) in gereksinimlerini yerine getirecek yeteneklere sahiptir. Buna karşılık daha karmaşık ve açık kodlu Apache Jetspeed projesi (JetSpeed, 2013) portlet konteynırlardan daha ziyade portalı kendi üzerinde bulundurma konusuna odaklanmıştır ve diğer grupların gereksinimlerini de dikkate almıştır.

Portallar ile uygulama açısından en çok Grid hesaplama (Atkinson et al., 2005; F. Berman, G. Fox, & T. Hey, 2003b) alanında karşılaşmaktayız. Portallar Grid hesaplamalar için gateway görevi görür ve sanal organizasyonun coğrafi olarak dağıtık varlıklarına ortak kullanım arayüzleri sağlarlar.

Bu alandaki en güzel örnekler GridPort (F. Berman, G. Fox, & A. J. G. Hey, 2003a) ve GridSphere (Novotny, Russell, & Vehrens, 2004; Novotny, Russell, & Wehrens, 2004) olarak verilebilir. Bunların yanında Bogdanski ve arkadaşları (Bogdanski, Kosiedowski, Mazurek, Rabięga, & Wolniewicz, 2007) geliştirdikleri progress adlı projelerinde Grid portalların kullanıcı dosluğunu ne ölçüde sağlayabileceğini araştırmışlar ve analiz etmişler. Paul (Fodor, 2007) JavaScript kullanarak portlet wrapper mimarisi geliştirmiştir. Alemada (Alameda et al., 2006) ve arkadaşları ise Open Grid Computing (OGC) ortamı için bilim gateway adını verdikleri portal uygulaması geliştirdiler. Bunun için Apache açık kaynak portal projelerinden Velocity ve Jetspeed (JetSpeed, 2013) kullandılar.

Bizim geliştirdiğimiz uygulamada klasik portal yapısında kullanılan veri tabanı işlemlerini kodlamada karmaşıklığa neden olan SQL sorgularından kurtarıp veri tabanı işlemlerini JPA ile gerçekleştirilmesi hedeflenmiştir.

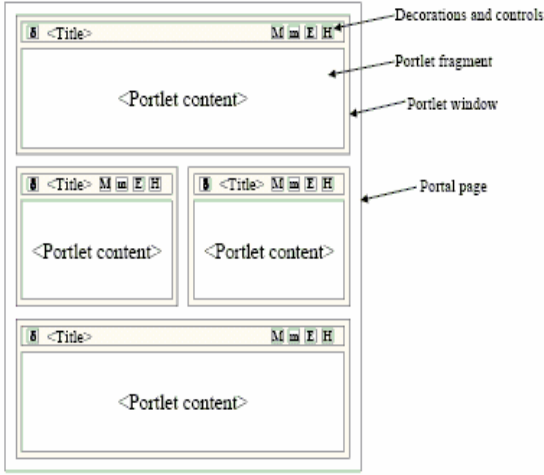
4. Mimari

Dinamik portal portlet projesinde geliştirilebilir ve esnek bir web sitesi içerik elde etmek için JSF, JPA, MySQL ve Apache web sunucuları kullanımı tercih edilmiştir. JSF esnek kodlama imkanı sağlarken MySQL veri tabanı güvenilirliği ile JPA kullanım kolaylığı, hızlılığı ve veri tabanı işlemlerinin basite indirilmesi sebebiyle seçilmiştir.

Java Persistence API ilişkisel verilerin Java nesneleriyle nasıl ilişkilendirileceği, bu nesnelerin ilişkisel veri tabanında daha sonra erişilmeye elverişli şekilde nasıl tutulacağı ve kullanan uygulama sona erdikten sonra da varlığın devamlılığının nasıl sağlanacağı ile ilgilendir.

Portal ve portletlerin geliştirilmesinden web tabanlı kullanıcı arayüzleri JSP kullanılarak geliştirilmiştir. JSF, Java Enterprise Edition (JEE) uygulamalarının web ara yüzünü inşa etmek için oluşturulan, Java'nın standart web çatısıdır (framework). JEE uygulamalarında web arayüzleri oluşturmak için Java servlet, Java server pages (JSP) , Struts, Apache Wicket gibi 50'nin üzerinde çatı bulunur ve JSF de bunlardan bir tanesidir. JSF'nin ortaya çıkmasında ki en büyük amaç basitçe MVC yapısını kullanarak iş ve sunum mantığının birbirinden ayrılması olarak düşünülebilir.

Şekil 1'de bir portalın temel mimarisi verilmiştir. Buna göre, Portal web application kesimi istemci istemlerini alarak işleme koyar ve kullanıcının o an kullanmakta olduğu sayfadaki portletleri bulup getirir. Sonra bu portletlerin içeriklerinin doldurulması için portlet konteynırları çağırır. Portal, portlet konteynıra Portlet-Container-Invoker API aracılığıyla ulaşır. Portlet-Container-Invoker, portlet konteynırın ara yüzü durumundadır ve o katmana ait istemci ana metotlarını bulundurur ve bu metotlar portal açısından bakıldığında portletleri çağırarak için kullanılan fonksiyonlardır. Portlet konteynır portalın kendisinden istediği bilgileri göndermek için Container Provider SPI (Service Provider Interface)'i kullanmak zorundadır. Son olarak portlet API aracılığıyla bütün portletleri çağırırlar.



Şekil 1: Portal portlet genel mimari yapısı

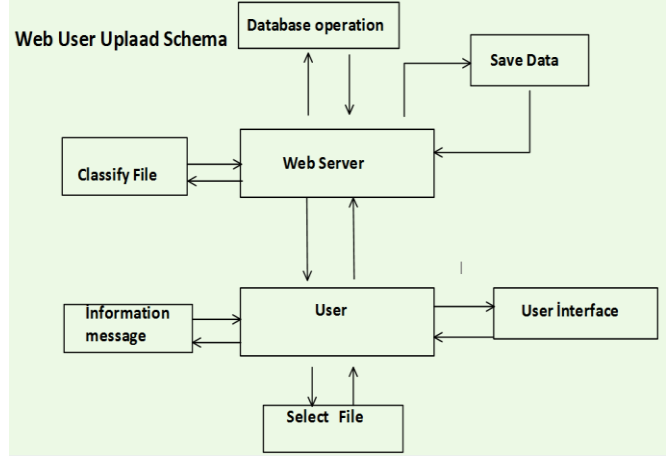
Dinamik portal portlet kullanıcılar ve geliştiriciler için karmaşık web sayfalarını yönetmek için dinamik şablon-bileşenler kullanımıyla kolay metotlar sunmaktadır. Projede JSR168 birçok alanda spesifikasyonları tanımlamaktadır. Bunlar arasında sayılabilecek şu şekilde özetlenebilir: (1) Sayfayı alanlara bölmek ve her portlet için izlenebileceği ve yönetilebileceği bir ekranın tanımlanması, (2) login işlemleri ve kullanıcı bazlı farklı ekranlara sahip olabilmek, (3) Normal kullanıcı ile admin arasında yetki farklılıkları oluşturma ve yetkileri tanımlama, (4) Her bir portlet için gizleme, gösterme, silme gibi fonksiyonları yerine getirme, (5) Veri sistemlerinde kişiselleştirmeyi (personalization), tek kişilik oturum açmayı (single sign on) ve değişik bilgi içeriğinin bütünleştirilmesini sağlayan web tabanlı bir uygulama geliştirme. Bütünleştirme değişik kaynaklardaki bilgi içeriklerinin bir web sayfasına toplanması işlemidir. Bu işlemde klasik portal yapısında farklı olarak kişilerin geliştirdiği uygulamaların web sayfasına yüklenip tek bir çatı altında yayımlamak hedeflenmiştir. Kullanıcılara içeriği kendilerine göre yüklenen sayfaları özelleştirme imkânı sunmaktadır. Portal sayfaları kullandıkları portlet kümelerinin çeşidine göre sayfanın içeriğini değiştirebilir. Bu da kullanıcılara özgü sayfa yapısı tasarlamayı sağlar.

Bunların dışında tek portletin tüm sayfayı kaplaması, admin dışında kullanıcı çeşitliliği, portletlerin ekranda yer değişikliklerinin gerçekleşmesi gibi benzer konularda standart yakalanamamıştır. Standardın dışı olarak genelde web sayfası geliştirme işlemi web sayfası içinde alanlar tanımlayarak onlar üzerinden değil doğrudan kullanıcının kendi geliştirdiği dosyaların sisteme yüklemesi ile gerçekleştirilme işlemi klasik portlet yapısının dışında bir özellik sağlamaktadır.

A. Kullanıcı Mimarisini

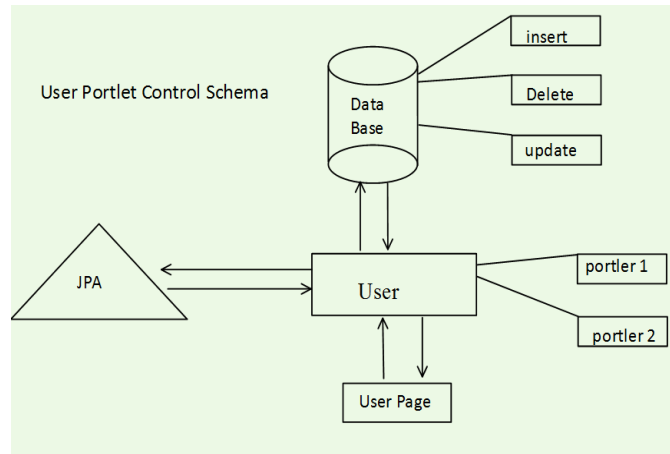
Kullanıcı için geliştirilen iki mimari vardır. Bunlardan ilki Şekil 2'de görüldüğü üzere kullanıcının geliştirdiği uygulamaları sunucu tarafına yükleme işlemidir. Bu ise iki temel yapı üzerine kurulmaktadır. Kullanıcı tarafı olan yapıda kullanıcıya bir ara yüz sunularak dosyalarını yükleme imkanı upload sayfasında gerçekleşmektedir. Bu sayfa aracılığıyla kullanıcı geliştirdiği sayfaları ve bunların kullandığı Java sınıflarını upload etmesini sağlamaktadır.

Burada kullanıcının başka dosya upload etmesi engellenmekte sadece Java sınıfları ve geliştirdiği web sayfasının yüklenmesine izin verilmektedir. Kullanıcı sayfası yükledikten sonra sunucu tarafı işlemeye başlamaktadır. Burada yüklenen dosyanın türüne göre bir sınıflandırma işlemi yapılarak sınıflar ayrı dizine sayfalar ayrı dizine yüklenmektedir. Bu yükleme işlemi JSF çalışma hiyerarşisi gözetilerek gerçekleşmektedir. Kullanıcı yükleme işlemi gerçekleştikten sonra yüklenen sayfalar admin onay sayfasına düşmektedir. Bura da admin onaylama işleminden sonra kullanıcı sayfaları kullanabilmektedir.



Şekil 2:Kullanıcı mimarisini

Kullanıcı için geliştirilen ikinci mimari ise yüklü olan uygulamaları kendine özelleştirilen sayfalarda yayınlama işlemidir. Şekil 3'de görüldüğü üzere kullanıcı için geliştirilen web sayfası aracılığıyla yüklü olan sayfaları portlet içerisinde listelenmektedir. Kullanıcı bu listeden eklemek istediği uygulamayı ekle diyerek JPA'yı tetiklemekte JPA da kullanıcı Id si ile kullanıcının hesabına eklemek istediği sayfayı insert etmektedir. Bu şekilde kullanıcı hesabına sayfa eklenmekte ve bir portlet içerisinde kendisine özelleştirilmiş sayfada yayınlanmaktadır.



Şekil 3: Kullanıcı portlet mimarisini

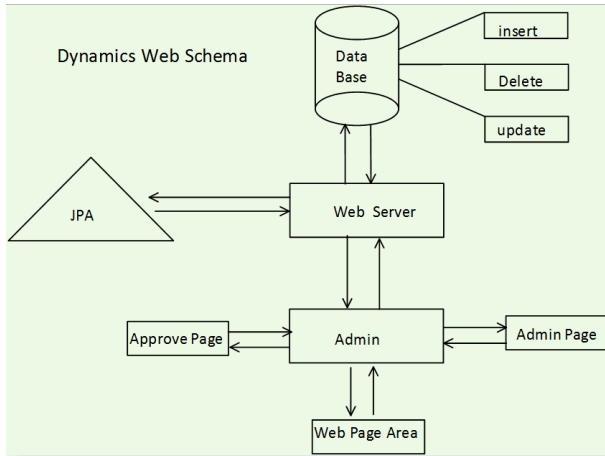
B. Yetkili (Admin) Mimarisini

Admin için geliştirilen mimari genel olarak Şekil 4 deki gibi admin için geliştirilen sayfa üzerinden gerçekleştirilmektedir.

Admin normal kullanıcılar gibi portal ve portletleri kendisine göre özelleştirebilmektedir. Bunun yanında admin için kullanıcılardan farklı birçok yetkisi vardır. Bunların en önemlilerinden biri kullanıcıların yüklemiş olduğu web sayfalarının onaylama işlemi admin tarafından yapılmaktadır. Admin isterse onayladığı web sayfasının onayını geri alarak yayından kaldırma gibi yetkileride mevcuttur. Burda admin aynı zamanda portalları da değiştirme yetkisine sahiptir. Admin isterse tüm portalların isimlerini linklerini değiştirebilmekte ve yeni linkler ekleyip bu linkleri yüklü olan sayfalardan birine ayarlama yapabilmektedir.

5. Sonuçlar

Birçok web tabanlı uygulamanın tek sunucu içinde ve tek arayüzden kontrol edilmesini (single administrative domain) sağlayarak birçok kullanım ve yönetim kolaylığı sağlayan portal mimarileri birçok bilimsel projelerin, özellikle grid (Berman et al., 2003b) ve bulut (Velte, Velte, & Elsenpeter, 2010) hesaplamalı, erişilmesi ve kullanılmasında anahtar rol oynamıştır. Tek yönetim alanı oluşturduğu için, portal mimarileri kullanıcı gruplarının oluşturulması ve bu gruplara uygun erişim haklarının (read, write etc.) ayarlanmasını sağlarlar. Kaynakların yönetilmesi tek elden sağlandığı için bu bir avantaj sağlar.



Şekil 4: Admin Mimar

Her ne kadar çeşitli standartlarda ticari amaçlı yada çeşitli projelerde araştırma amaçlı olarak geliştirilen portal paketleri olsa da, bu çalışmada bu tür mimarilerin Java teknolojileri kullanarak geliştirilebileceğini göstermeye çalıştık. Ayrıca, küçük çaplı olduğu için üniversitelerde herhangi bir derse ait dönem projeleri ve tezlerin uygulamalarının tek sunucudan kurulum ve yönetimini sağlayacağı için, örnek projelerin paylaşımını ve böylece öğrenme sürecini iyileştireceğini öngörmekteyiz.

KAYNAKLAR

1. Abdelnur, A., & Hepper, S. Java Portlet Specification.

2. Alameda, J., Christie, M., Fox, G., Futrelle, J., Gannon, D., Hategan, M., et al. The Open Grid Computing Environments collaboration: portlets and services for science gateways. *Concurrency - Practice and Experience*, 19(1), 921-942, 2006.
3. Apache. (2008). Apache Tomcat Project. Retrieved 03/10/2008, from <http://tomcat.apache.org/>, 2013.
4. Atkinson, M., DeRoure, D., Dunlop, A., Fox, G., Henderson, P., Hey, T., et al. Web Service Grids: An Evolutionary Approach *Concurrency & Computation: Practice&Experience*, 17(2), 377-389, 2005.
5. Berman, F., Fox, G., & Hey, A. J. G. GridPort. In *Grid Computing, Making the Global Infrastructure a Reality* (pp. 1012). Chichester, England: John Wiley & Sons Ltd., 2003a.
6. Berman, F., Fox, G., & Hey, T. *Grid Computing: Making the Global Infrastructure a Reality*. Chichester, England: John Wiley & Sons., 2003b.
7. Bogdanski, M., Kosiedowski, M., Mazurek, C., Rabiega, M., & Wolniewicz, M. Flexibility and user-friendliness of Grid portals: the PROGRESS approach. *Concurrency - Practice and Experience*, 19(1), 827-838., 2007.
8. Burns, E., & Kitain, R. *JavaServer Faces Specification*: Sun Microsystems. (S. Microsystems), 2009.
9. DeMichiel, L. *Java Persistence API Implementation specification*: Sun Microsystems), 2009.
10. Fodor, P. Portlet Wrappers Using Javascript. Paper presented at the Proceedings of the IADIS International Conference on WWW/Internet, Vilreal, Portugal., November 2007.
11. JetSpeed. (2013). Apache Jetspeed. Retrieved January 15, 2013, from <http://portals.apache.org/jetspeed-2/>
12. JSR168. Portal Specification. 2013, from <http://jcp.org/aboutJava/communityprocess/final/jsr168/index.html>, 2013.
13. KURTEL, K. Web'in Geleceği: Anlamsal Web Ege Akademik Bakış. *Ege Academic Review*, 18(1), 205-213., 2008.
14. Leff, A., & Rayfield, J. T. Web-application development using the Model/View/Controller design pattern. Paper presented at the 5th IEEE International Enterprise Distributed Object Computing Conference., 2001.
15. Medianero-Pasco, X., Bonilla-Morales, B., & Vargas-Lombardo, M.. Grids Portals: Frameworks, Middleware or Toolkit. *International Journal of Computer Science Issues (IJCSI)*, 7(3), 6-11., 2010.
16. Minter, D., & Linwood, J. *Building Portals with the Java Portlet API*., 2004.

17. Ndou, V. D.. E-Government for Developing Countries- Opportunities and Challenges. The Electronic Journal on Information Systems in Developing Countries, 18(1), 1-24., 2004.
18. Novotny, J., Russell, M., & Vehrens, O. GridSphere: a portal framework for building collaborations. Concurrency and Computation: Practice and Experience, 16(5), 503-513., 2004.
19. Novotny, J., Russell, M., & Wehrens, O. GridSphere: An Advanced Portal Framework. Paper presented at the 30th EUROMICRO Conference. August 31, 2004.
20. Pluto. (2013). Apache Pluto. Retrieved January 15, 2013, from <http://portals.apache.org/pluto/>
21. Velte, A. T., Velte, T. J., & Elsenpeter, R. Cloud Computing, A Practical Approach. USA: McGraw-Hill Osborne Media., 2010.