T.C.
DOKUZ EYLÜL ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
YÖNETİM BİLİŞİM SİSTEMLERİ ANABİLİM DALI
YÖNETİM BİLİŞİM SİSTEMLERİ PROGRAMI
YÜKSEK LİSANS TEZİ

YENİ NESİL İÇERİK YÖNETİM SİSTEMİ MODELİ

Samed Cansın ESKİOĞLU

Danışman
Prof.Dr. Vahap TECİM

İZMİR - 2017

YÜKSEK LİSANS
TEZ ONAY SAYFASI

Üniversite	: Dokuz Eylül Üniversitesi
Enstitü	: Sosyal Bilimler Enstitüsü
Adı ve Soyadı	: SAMED CANSIN ESKİOĞLU
Öğrenci No	: 2014800649
Tez Başlığı	: Yeni Nesil İçerik Yönetim Sistemi Modeli
Savunma Tarihi	: 03.07.2017
Danışmanı	: Prof.Dr.Vahap TECİM

JÜRİ ÜYELERİ

Ünvanı, Adı, Soyadı	**Üniversitesi**	**İmza**
Prof.Dr.Vahap TECİM	-DOKUZ EYLÜL ÜNİVERSİTESİ	.....
Yrd.Doç.Dr.Can AYDIN	-DOKUZ EYLÜL ÜNİVERSİTESİ	.....
Doç:Dr.Murat KOMESLİ	-YAŞAR ÜNİVERSİTESİ	.....

SAMED CANSIN ESKİOĞLU tarafından hazırlanmış ve sunulmuş olan bu tez savunmada başarılı bulunarak oy birliği(✓) / ~~oy çokluğu( )~~ ile kabul edilmiştir.

Prof. Dr. M. Banu DURUKAN SALI
Müdür

YEMİN METNİ

Yüksek Lisans Tezi olarak sunduğum "Yeni Nesil İçerik Yönetim Sistemi Modeli" adlı çalışmanın, tarafımdan, akademik kurallara ve etik değerlere uygun olarak yazıldığını ve yararlandığım eserlerin kaynakçada gösterilenlerden oluştuğunu, bunlara atıf yapılarak yararlanılmış olduğunu belirtir ve bunu onurumla doğrularım.

Tarih

01/06/2017

Samed Cansın ESKİOĞLU

ÖZET

Yüksek Lisans Tezi

Yeni Nesil İçerik Yönetim Sistemi Modeli

Samed Cansın ESKİOĞLU

Dokuz Eylül Üniversitesi

Sosyal Bilimler Enstitüsü

Yönetim Bilişim Sistemleri Anabilim Dalı

Yönetim Bilişim Sistemleri Programı

Tez kapsamında İçerik Yönetim Sistemleri'nin; işletmelerin pratik hayattaki insani süreçlerine uyum sağlayamamasının, bunu sağlayabilen İş Akış Yönetimi yazılımlarının bile yeterli düzeyde esnek ve her örgüt katmanınca kullanılabilir yapıda olmamasının temel sebebi araştırılmıştır.

Kullanıcı dostu Kurumsal İçerik Yönetim Sistemleri için iş akış destekli, genişletilebilir yeni nesil bir model geliştirme ihtiyacı olduğu gerçeği İçerik Yönetim Sistemleri ve türevlerinin yeterli esnekliği ve kullanıcı dostu işlevselliği sağlayamayan geleneksel modeller kullanmasından doğmaktadır. Bu durum, farklı karmaşık seviyelerdeki ve yönetimsel yapılardaki örgüt faaliyetlerine gerçekçi bilgi döngüsü sağlayamamasından da kaynaklanır.

Yapılan çalışmada, ortaya konulan sorun disiplinler arası yaklaşım ile analiz edilmiş ve ilkel kullanıcı yaşam döngüleri geliştirilerek günümüz veri saklama yaklaşımlarından daha farklı bir model geliştirilmiştir. Uygulama yazılımı geliştirildikten ve örnek veriler girilerek alana entegre edildikten sonra da; veri istifinde gözlemlenen verimliliklerin yanı sıra, uygulama eklentilerinin ve tema yapılarının yerini alan kontrolör teknolojisi ile tatmin edici düzeyde teknolojik ve mobil destekli çıktılar alınması modelin oldukça doğru şekilde işlev gördüğünü destekler niteliktedir.

Anahtar Kelimeler: İçerik Yönetim Sistemi, İş Akış Yönetimi, Kitle Veri.

ABSTRACT

Master's Thesis

A New Generation Content Management System Model

Samed Cansin ESKIOGLU

Dokuz Eylül University

Graduate School of Social Sciences

Department of Management Information Systems

Management Information Systems Program

Fundemental reason of why content management systems may not assort with corporations' humanistic processes in practical life and why workflow management softwares which corresponds but may not be built in a structure flexible enough and useable by most of the organization layers; has been researched.

Non friendly functional and inflexible traditional model usage in content management systems or similar developments, results the need of a workflow supported expendable new generation model development. This situation also arises from the inability of prividing realistic information lifecycle to organization activities in different complex levels and managerial structures. In the study work, exposed problem has been analysed by interdisciplinary approach and new generation data life cycles has been developed that is far away from present data storage approaches.

In the study work, the present problem has been analysed by interdisciplinary approach and a model has been designed different than modern day data storage approaches with improving primitive user life cycles. After software has been developed and example data has been input for integration to the research area; technological and mobile supported outputs in satisfactory levels taken, thanks to the controller technology that took software plugin and theme structures, and it supports the model's right functioning fact; besides the productivity in data stowage.

Keywords: Content Management System, Workflow Management, Mass Data

İÇİNDEKİLER

TEZ/PROJE ONAY SAYFASI i
YEMİN METNİ iii
ÖZET v
ABSTRACT vi
İÇİNDEKİLER vii
KISALTMALAR ix
TABLOLAR LİSTESİ x
ŞEKİLLER LİSTESİ xi

GİRİŞ 1

BİRİNCİ BÖLÜM
İÇERİK YÖNETİM SİSTEMLERİ

1.1. İÇERİK KAVRAMI 4
1.2. İÇERİK YÖNETİM SİSTEMİNİN TANIMI 6
1.3. TEMA VE EKLENTİ SEKTÖRÜ 9
1.4. ÇEŞİTLİLİK VE TERCİH YÖNELİMİ 10

İKİNCİ BÖLÜM
İÇERİK YÖNETİM SİSTEMLERİNİ GELİŞTİRMEYE YÖNELİK MODEL TASARIMI

2.1. TEMEL YAPININ OLUMSUZLUKLARI 12
2.1.1. Veri Saklama ve İşleme Yöntemleri 12
2.1.2. Eklenti Geliştirme Sürecindeki Zorluklar 13

2.2. HARPA İÇERİK YÖNETİM SİSTEMİ MODELİ 14
2.2.1. Amaç 14
2.2.2. Tanım 14
2.2.3. Metodoloji 15
2.2.4. Modelin Teknik Altyapısı 16
2.2.4.1. Hiyerarşik Modelin Temeli 16
2.2.4.2. Tasarlanan Model 18
2.2.4.3. Dağıtık Küme Yapısı 27
2.2.4.4. Veri Atama Kolaylığı 27
2.2.4.5. Alternatif Kontrolör Desteği 30
2.2.4.6. İçerik Yönetiminde İş Akış Desteği 37

ÜÇÜNCÜ BÖLÜM
UYGULAMA

3.1. GELİŞTİRİLEN UYGULAMA PLATFORMU 41
3.2. UYGULAMA ALANI (ULUSLARARASI ÖĞRENCİ KAYIT SİSTEMİ) 54
3.3. ÖN TANIMLI KONTROLÖRLER 59
3.4. KISITLAR 61

SONUÇ 63
KAYNAKÇA 65

KISALTMALAR

İYS	İçerik Yönetim Sistemi
YBS	Yönetim Bilişim Sistemleri
CMS	Content Management Systems
San.	Sanayi
Tic.	Ticaret
Ltd.	Limited
Şti.	Şirketi
HARPA	Harvestable Paginal
B2C	Business to Consumer
MİY	Müşteri İlişkileri Yönetimi
HTML	Hypertext Markup Language
PHP	Hypertext Preprocessor
SSI	Server Side Includes
DIKW	Data Information Knowledge Wisdom
S.	Sayfa

TABLOLAR LİSTESİ

Tablo 1: İYS Dağıtımlarında Kullanıcı Taraflı Tercih Sebepleri s. 11

Tablo 2: HARPA İYSde Tanımlı İçerik Türlerinin ve Kontrolörlerinin Sahip Olabilecekleri Seviyeler s. 19

Tablo 3: HARPA İYS Ve Öğrenci Kayıt Sistemi Üzerindeki Kullanıcı Grupları Ve Yetkileri s. 50

ŞEKİLLER LİSTESİ

Şekil 1: DIKW Piramidi Ve Açılımı Üzerinde İçerik Ve Bağlam Kavramlarının Gösterimi s. 5

Şekil 2: IBM Müşteriye Dönük Çözümler Bilgi Grafiği s. 6

Şekil 3: İçerik Yönetimi Kavramının İlkel Yaşam Döngüsü s. 7

Şekil 4: Wordpress Veritabanı İlişki Diyagramı s. 12

Şekil 5: HARPA İYS Konsept Diyagramı s. 16

Şekil 6: İlkel Hiyerarşik İçerik Saklama Yaklaşımı s. 17

Şekil 7: HARPA İYS Modelinin Aktör Diyagramı s. 18

Şekil 8: HARPA İYS Veritabanı - İlişki Diyagramı s. 20

Şekil 9: HARPA İYS Veri Ekleme Motoru Akış Diyagramı s. 22

Şekil 10: HARPA İYS Sayfa Yetki Kontrolü - Akış Diyagramı s. 23

Şekil 11: HARPA İYS Kullanıcı Arayüzü Veri Akış Diyagramı s. 24

Şekil 12: HARPA Veritabanında Farklı Türlerdeki Ham Verilerin Farklı Kümelerde Tutulması s. 28

Şekil 13: HARPA Veritabanında Oluşan Form Aksiyonlarının Kayıt Dökümü Halinde Tutulması s. 29

Şekil 14: HARPA İYS Kontrolör Bileşeni Örnek Klasör Ve Dosya Yapısı s. 31

Şekil 15: HARPA İYS Aksiyonları Klasör Dizini s. 32

Şekil 16: HARPA İYS Kontrolör Dosyası Control.Tpl Örnek Ekran Görüntüsü s. 33

Şekil 17: HARPA İYS Kontrolör Dosyalarının Yükleme Zamanında Birbirlerini Kapsaması s. 34

Şekil 18: HARPA İYS Dizayn Sayfalarında İçerik Görüntülenme Yapısı s. 35

Şekil 19: HARPA İYS Tarafından Üretilen 3 Boyutlu Çember Tablo Bileşeni s. 36

Şekil 20: HARPA İYS Tarafından Üretilen Formal Dijital İmza Kontrolörü s. 36

Şekil 21: HARPA İYS Kontrol Paneli Üzerinde Proje Sayfalarının İş Akışı Görünümü s. 38

Şekil 22: HARPA İYS Tarafından Üretilen Formal Sayfalardaki İş Akışı Sağ Kısım Menüsü Ekran Görüntüsü s. 39

Şekil 23: HARPA İYS Kullanıcının Yetkili Olduğu Projeler Listesi s. 40

Şekil 24: HARPA İYS Kontrol Paneli Üzerinden Mobil Destekli Form Yaratma İşlemi s. 42

Şekil 25: HARPA İYS Kontrol Paneli Üzerinden Statik Dizayn Sayfası Geliştirme s. 43

Şekil 26: HARPA İYS Kontrol Paneli - Gelişmiş Veri/Fotoğraf Bağlama Araçları s. 44

Şekil 27: HARPA İYSde Tanımlanan Form Elemanına Tanımlı Aksiyon Atama s. 45

Şekil 28: HARPA İYS Kontrol Paneli Üzerinde Web Sayfalarının İş Akışı Olarak Projeye Dâhil Edilmesi s. 46

Şekil 29: HARPA İYS Uygulamasında Kayıtlı Kullanıcı Girişi s. 47

Şekil 30: HARPA İYS Ürün Sitesi (www.harpacms.com) Ekran Görüntüsü s. 48

Şekil 31: HARPA İYS Kontrolör Bileşen Yükleme Sayfası Ekran Görüntüsü s. 49

Şekil 32: HARPA İYS Üzerinde Kayıtlı Form Verilerinin Rapor Dökümünde Görüntülenmesi s. 51

Şekil 33: HARPA İYS Rapor Dökümünde Formal Veri Detayı Görüntüleme İşlemi s. 52

Şekil 34: HARPA İYS ile Tasarlanmış Mobil Destekli Öğrenci Kayıt Sistemi Çıktısı Kullanıcı Arayüzü s. 54

Şekil 35: HARPA İYS Tarafında Üretilen Uygulama Alanı Çıktısı (Öğrenci Kayıt Sistemi Tanıtım Sayfası) s. 58

GİRİŞ

İçerik yönetimi, küresel çapta internet teknolojisinin önemli yapı taşlarından biri olarak kabul edilmektedir (Mescan, 2004; Innovative Architects). Günümüzde çoğunlukla internet sitesi kurucusu olarak algılanan İçerik Yönetim Sistemi (İYS); arşivleme ve yayıncılık amaçlarıyla kullanılabildiği gibi, elektronik ticaret ve sosyal medya adaptasyonları ile çok farklı alanlarda da altyapı niteliği taşır. İYSler, çok sayıda insanın farklı noktalardan ortak paylaşımlı veriye ulaşmasını ve düzenleme yapmasını sağlayan, hedef kitlelere yayın yapma imkânı verirken farklı yetkiler verilmiş aktörler arasındaki interaktif iletişimi sağlayan yazılım, tasarım ve veritabanından oluşan paket çözümlerdir. Kendi içinde farklı anlayışlar barındırırken, diğer bilgi işlem yazılımı türleri ile melez yapılarda olan türevleri de bulunmaktadır (Kızılboğa, 2014: 25).

İki binli yıllarda eklenti destekli interaktif portal yazılımlarının popülaritelerini kaybetmelerinden bu yana, alternatif niteliğindeki internet makalesi oluşturma sistemleri insani süreçlere adapte olmaya devam ederek, günümüzün yaygın tema ve eklenti entegrasyonlarına ulaşıncaya dek oldukça uzun bir gelişim dönemi geçirmişlerdir (Targowski ve Tarn, 2007: 84; The University of Hong Kong, 2013; Hart, 2017; LeBlanc, 2011; Akçakaya, 2006: 88; Kızılboğa, 2014: 4; North Patrol: 2013). Teknik özelliklerinin çeşitliliği ile kullanıcılara, geliştiricilere ve sistem yöneticilerine yönelik siber çözüm niteliğinde olmakla birlikte, her kesimden iş sektörünün ekonomik boyuttaki taleplerine yanıt verebilmektedir. Ancak mevcut sistemdeki, insani süreçlere uyum sağlamayan kullanım mekanizmaları, genişletme noktasında yazılım tecrübesi gereksinimleri, verimsiz istif yöntemleri, form tabanlı işlemlerde karmaşık eklentilere bağımlılık ve sunucular arası taşıma sonrasındaki problemler (WebHostingMedia, 2016) gibi dezavantajları sebebiyle karar verici noktalardaki kurumsal hedef kullanıcıları, özel sipariş üzerine hazırlanan yazılımlar gibi farklı alternatiflere yöneltebilmektedirler (McIlwain, 2016).

Bu tez, web geliştiricilere yeni bir anlayış önermek; İYSler, MİY (Müşteri İlişkileri Yönetimleri), otomasyon uygulamaları ve benzeri yazılımların geliştirme süreçlerine yeni bir alternatif geliştirme modeli sunarak, daha kullanıcı dostu geliştirilebilir platformlar üretilmesine katkıda bulunma amacı taşır.

Kullanıcı dostu kurumsal İYSler için iş akış destekli, genişletilebilir yeni nesil bir model hipotezi üzerine kurulan çalışma, mevcut sistemlerin pratik hayattaki insani süreçlere uyum sağlayamamasının, bunu sağlayabilen iş akış yönetimi, yazılımlarının bile yeterli düzeyde esnek ve her örgüt katmanınca kullanılabilir yapıda olmamasının temel sebebini

araştırmıştır. Bu durumu raporda incelenen teknik gerekçelere bağlarken, yeterli esnekliği ve kullanıcı dostu işlevselliği sağlayamayan geleneksel modeller kullanmasından doğan; farklı karmaşık seviyelerdeki ve yönetimsel yapılardaki örgüt faaliyetlerine kullanıcı deneyimi yüksek geliştirme ortamı ve arayüz çıktıları sağlayamaması gözlemine dayandırmıştır.

Tez kapsamında, İYSlerin gelişmesindeki zorluklar incelenip yeni bir model oluşturma gereksinimi ortaya konulduktan sonra, ihtiyaçları karşılayabilecek ve tema bağımlılığını ortadan kaldıran geliştirilebilir melez bir enformatik yapı olan HARPA (Harvestable Paginal) modeli planlanmış, daha sonra bu modelde çalışan web tabanlı bir yazılım ile onlarca bileşeni geliştirilmiş, uygulama dâhilinde olan kontrol paneli üzerinden alan uygulaması için internet sayfaları üretilmiştir. Bu esnada performans ve kararlılık gibi konular da yakından takip edilmiş, formal sayfalar üzerinden örnek başvuru işlemleri yapılarak raporlama kalitesi incelenmiştir. Yazılım Hypertext Preprocessor (PHP) 5.6.26 sürümünde geliştirilmiş, veritabanı ve saklı yordamlar MYSQL 5.6.34 üzerinde kurulmuş, arayüz geliştirmede jQuery ve jQuery Mobile kütüphaneleri kullanılmış, 3. Parti olarak çeşitli script yazılımlar uygulamaya dâhil edilmiştir. Örnek uygulama ve çıktıları www.harpacms.com web adresinde barındırılmaktadır.

HARPA İYS modeli, günümüz bilişim sorunlarına çözüm noktasında bir yol haritası tanımlarken, aynı zamanda yeni bir anlayış ve algıyı da alternatifler arasına yerleştirme amacı taşır. İYSler gelişim sürecinde, alışılagelmiş internet makalesi mantığını güncelleme yönelimi üzerinden ilerlediklerinden, içeriğe ayrık ve nesnel perspektiflerden bakma yönelimi gösterememişlerdir. Oysaki istif edilen ham veriye esneklik katabilmenin tek yolu nesne yönelimli veritabanı modelleri değildir; geleneksel hiyerarşik kullanımlar geliştirilerek te form tabanlı ve kullanıcı etkileşimi yüksek mobil destekli bilişim sistemleri geliştirilebilir. HARPA İYS, içeriği ve son kullanıcı verilerini yaşam döngüsü boyunca aynı hiyerarşik yapıda tutarak, hem geliştirme sürecini kısaltmış hem de yönetimde kullanım kolaylığını arttırmıştır. Ayrıca kümeleme tekniği ile verilerin istifinde tasarruflu bir yöntem ile kitle veride hiyerarşinin getirebileceği olası performans sorunlarının önüne geçmiştir.

Tez, 3 bölümlük rapor, web uygulama yazılımından ve örnek veriler girerek hazırlanan öğrenci kayıt sistemi çıktısından oluşur. Bölüm 1'de, içerik kavramının disiplinle arası yaklaşım ile incelenmesi, İçerik Yönetim Sistemi'nin tanımlanması, kabul gören yapı, tema ve eklenti sektörü, çeşitlilik ve tercih yönelimi gibi konular ele alınır. Bölüm 2'de, kuramsal bir çalışma ile yeni nesil içerik yönetim sistemi modeli önerisi yapılmaktadır. Bu kısım çeşitli alt başlıklar ile ele alınırken, ortaya konulan çalışma, farklı katmanlarda analiz edilerek isimlendirilir. Bölüm 3'te, geliştirilen içerik yönetim sistemi uygulaması platformu tanıtılır,

alan uygulaması verileri girilerek oluşturulan öğrenci kayıt sistemine ilişkin tanımlamalar, veriler, çıktılar ve incelemeler ele alınır.

Sonuç olarak, içerik ve içerik yönetimi konuları literatürde detaylı şekilde incelenmiştir ve içerik tabanlı sistemlere yeni bir tarz ve anlayış getirilerek günümüzün teknoloji beklentilerine uygun nitelikte bir model tasarlanmıştır. Model üzerinde çalışır bir uygulama geliştirildikten sonra, seçilen pilot departmanın siber ihtiyaçları örnek bir iş akışına adapte edilerek uygulama üzerinde tanımlanmıştır. Elde edilen web tabanlı çıktılar ve kalite unsurları değerlendirilmiş, kullanılan teknik unsurlar ve detaylar paylaşılmıştır.

BİRİNCİ BÖLÜM
İÇERİK YÖNETİM SİSTEMLERİ

1.1. İÇERİK KAVRAMI

Türk Dil Kurumu sözlüğüne göre içerik sözcüğü, 'İletişiminde kaynaktan alıcıya iletilen bilgi bütününün veya yığın iletinin taşıdığı duygu ve anlam yükü' şeklinde tanımlanmıştır (TDK, 1981:1). Bazı kaynaklar ise, bir mesajın içinde bulunanların bütününü veya muhtevasını içerik olarak nitelendirmişlerdir (Ozan, 2009:1).

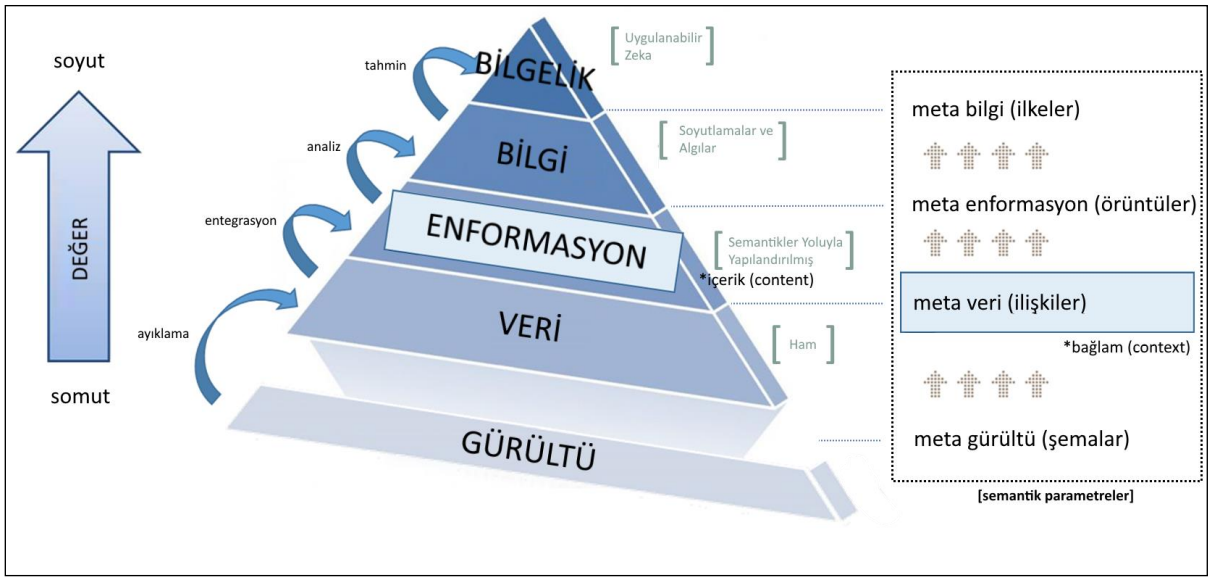
İçerik kavramını ele almak için bilgi kavramının en geçerli hiyerarşisi olan Bilgi Piramidi'ni (DIKW) disiplinler arası kapsamda açıklamak gerekir. DIKW (Data Information Knowledge Wisdom) birçok sisteme uygulanabilir bir bilgi sistemleri modelidir (Mercier, Stevens ve Toia, 2012:3). Söz gelimi bir veri ya da veriler (data), gürültü (noise) katmanından ayıklama metotlarına meta gürültü (meta noise) parametresi geçirilerek elde edilir. Enformasyon (information) katmanına ise veri ile meta veri (meta data) parametrelerinin entegrasyon (integration) metotlarına geçirilmesi ile ulaşılır. Bilgi (knowledge) yapısı, bilgi ediniminin meta enformasyon (meta information) parametreleri ile analiz (analyse) metotları üzerinden işlenmesi sonucu oluşur. Bilgelik (wisdom) katmanı ise bilgi unsurunun meta bilgi (meta knowledge) parametreleri ile tahmin (prediction) metotları ile işleme sokulması ile elde edilir. Buradaki katmanlar aşağıdan yukarıya doğru çıktıkça somut değerden soyut değer niteliğine dönüşür; maddi karşılık değeri artar ve geçmiş perspektifinden gelecek perspektifine değişime uğrar. Tüm meta unsurlar semantik parametre olarak değerlendirilir.

Bu anlatımdaki veri (data) yapısı çoğul nitelikte kullanılmıştır. Söz gelimi herhangi bir veri; bir veya birden fazla veri parçası (datum) unsurundan oluşur. Her bir veri parçası ise işaret (signal) unsurlarından oluşmaktadır. Ancak bunlar farklı disiplinlerin alanlarına girdiklerinden Şekil 1'de gösterilmemiştir. Neticede bir çoğul veri gürültü katmanından ayıklanarak elde edilir. Gürültü katmanı bazı disiplinlerde çevre (environment) olarak ta betimlenmiştir.

İçerik (Content) ve Bağlam (Context) kavramları bilgi sisteminin değer kattığı katmana göre yer değiştirilebilir. İçerik Yönetim Sistemleri üzerine yapılan model değişikliği çalışması, 'Data' ve 'Information' katmanlarının verimli kullanımı üzerine olduğundan; 'Content', 'Information' eşdeğeri sayılabilirken; 'Context' ise meta data eşdeğeri olarak kabul edilebilir.

Somut bir örnek ile ‘Context’ cam bir bardak olarak nitelendirilebilirken; ‘Content’ ise bardağa doldurulan içeriktir yani sıvıdır. Bardak örneğindeki gibi, dijital sistemlerde de ‘content’ yapısı, ‘context’ ile şekillenir ve sınırları belirlenir. Ayrıca her bir katmanın kendi içinde alt katmanları da olduğu kesindir, örneğin ‘Wisdom’ katmanının ‘Insight’, ’Decision – Action Plan’ ve ’Vision’ alt katmanları olduğu gibi. Lakin araştırmanın kapsamı ‘Information’ yapısına değer kattığından bu unsurlar incelenmemiştir.

Şekil 1: DIKW Piramidi Ve Açılımı Üzerinde İçerik Ve Bağlam Kavramlarının Gösterimi

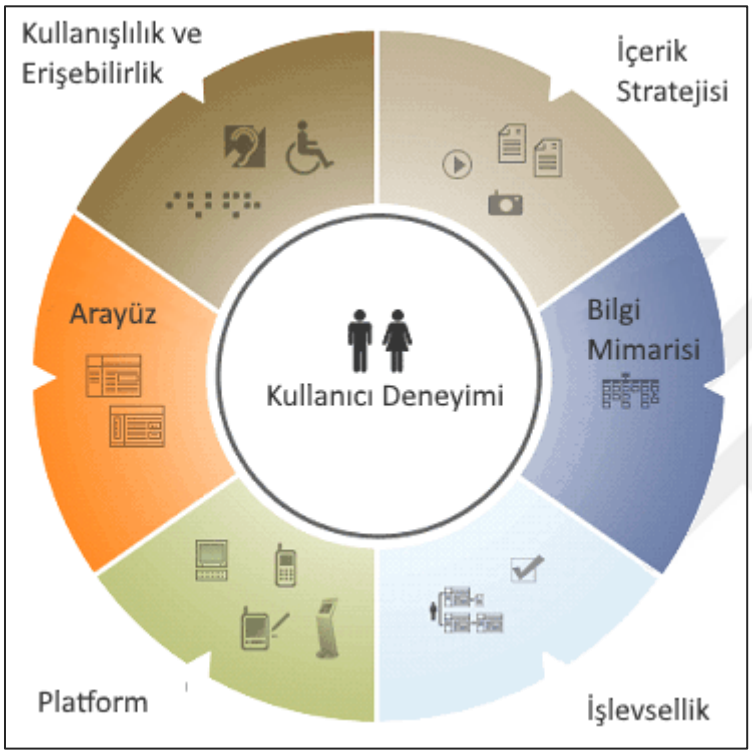


Kaynak: Ackoff, 1989: 3-9; Excolo, 2016: 1; Henriques, 2014: 1; McLean, 2012: 296; Riley, 2017: 7; Marasco, 2005: 1-330; Hey, 2004: 3; Jifa, 2013: 714-715

Uygulamanın, kullanıcıların gerçek hayat tecrübeleri ile örtüşen çıktılar üretmesi durumu kullanıcı deneyimi (user experience) ifadesi ile açıklanır (Garret J., 2010: 6). Kullanıcı kitleleri; üye, ziyaretçi ve aktör gibi farklı yetki düzeylerindeki kavramlar ile ortaya konabildiğinden her düzeyde kullanıcının, kullanımda farklı beklentileri oluşabilmektedir. Ancak temelde kullanıcı deneyiminde, kullanıcının doğal iş ve yaşam süreçlerine benzer erişimler üzerinden dijital işlemlerini gerçekleştirebilmesi beklenir. Bu konu psikolojik ve sosyolojik etmenler içerse de, içerik barındıran bilişim sistemlerinde kullanıcı deneyimi temelde altı farklı unsur üzerinden değer kazanır. Şekil 2’de IBM Müşteriye Dönük Çözümler Bilgi Grafiği (IBM's Customer Facing Solutions Infographic) görülmektedir (UX Magazine, 2011:5). Buradaki kullanışlılık ve erişilebilirlik, içerik stratejisi, işlevsellik, platform (yaygın platform desteği) kavramları yapısal olmayan ancak stratejik anlamda öneme sahip unsurlardır. Bunlar içerik tabanlı bilgi sistemlerinde kullanıcı deneyimini oluşturan faktörlerdir. Arayüz (kullanıcı arayüzü) ve bilgi mimarisi faktörleri ise somut ve yapısal

olarak ortaya konabilen başlıca unsurlardır. Kurulu bir bilişim sisteminde, bu unsurlardan herhangi birisinin üzerine yapılacak bir iyileştirme veya yükseltme işlemi toplamda kullanıcı deneyimi kalitesini arttıracaktır. Çalışmanın sonraki kısımlarında adı geçen tasarlanan yeni nesil model, yapısal iki unsura da değer katan kullanıcı deneyimini arttırmaya yönelik özelliklere sahiptir.

Şekil 2: IBM Müşteriye Dönük Çözümler Bilgi Grafiği



Kaynak: UX Magazine, 2011:5

1.2. İÇERİK YÖNETİM SİSTEMİNİN TANIMI

Kaynaklar, 'İçerik Yönetim Sistemi' tanımını temelde belge ya da benzeri içeriklerin çeşitli yapılandırmalar yoluyla yaratılmasına ve düzenlenmesine yardımcı olan yazılım dizgeleri olarak betimlemişlerdir. Çoğu zaman web sitesi hazırlayıcı yazılımlar veya günlük servisi olarak dar anlamlı bir tanımla değinilen İYS, uluslararası platformda birçok organizasyon için zorunluluk hâline gelen bilişim sistemlerinin ayrılmaz bir parçası niteliğinde olan modern bir varlık (asset) yönetimi tekniği olmuştur (Çelik, 2011:2). Bazı kaynaklar ise İYSyi çok fazla sayıda ya da çok geniş kapsamlı bir içeriğin kolay ve etkin

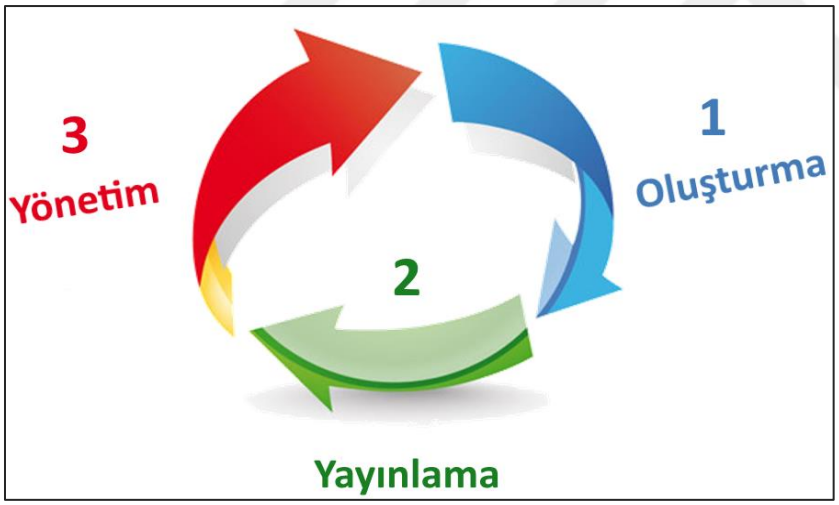
şekilde yönetilmesi için geliştirilmiş uygulama paketleri olarak tanımlar (Yorulmaz, Yasemin ve Şimşek, 2009:1).

İYS, ortak çalışma ve farklı aktörlerin katılımını gerektiren ortamlardaki iş akışını işlemler manuel veya bilgisayar tabanlı olması fark etmeksizin yönetmeyi sağlayan işlemler bütünüdür (Altıntaş, 2016:3).

Bu işlemler temelde aşağıda bahsedilen faydaları sağlamak için tasarlanmıştır (Altıntaş, 2016:3);

- Çok sayıda insanın farklı noktalardan ortak paylaşımlı veriye ulaşması ve düzenleme yapmasına olanak sağlamak,
- Tanımlanan veri için farklı aktörlere farklı yetkiler tanımlamak,
- İçeriğin kolay saklanmasını ve okunmasını sağlamak,
- Tekrar verilerin miktarını azaltmak,
- Dokümantasyon yazımını kolaylaştırmak,
- Aktörler arasındaki interaktif iletişimi sağlamak

Şekil 3: İçerik Yönetimi Kavramının İlkel Yaşam Döngüsü



Kaynak: Yazar tarafından derlenmiştir.

Şekil 3'te grafikte basit bir içerik yönetim sisteminin sahip olması gereken adımlar asgari ölçüde sırasıyla tanımlanmıştır. Söz konusu üç adım ilkel bir yaşam döngüsünü ortaya koyar. Bu nitelikteki herhangi bir bilgi sistemi, içerik yönetimi kapsamında değerlendirilebilir.

Bir İYS'inde veri herhangi bir şey olabilir. Dokümanlar, filmler, resimler, telefon numaraları, bilimsel veri vb. İYS genelde veriyi kaydetmek, kontrol etmek ve gözden geçirmek (zenginleştirmek) ve yayınlamak için kullanılır. Ancak çeşitli kaynaklar içerik

yönetim sürecini farklı adımlar ile tanımlarken, farklı konsept yapıları da ortaya koymuşlardır. Bunlar yorum farklılıkları ve sektörlere yönelik beklentiler ile oluşan arz nitelikleri neticesinde şekil kazanabilen durumlardır. Bazı kaynaklar, kurumsal bilginin bütün oluşturma, yayınlama, paylaşma ve depolama sürecini içerik yönetim kapsamında tutarken (Ozan, 2009:1), farklı perspektiflerden yapılan çalışmalar içerik yönetim adımlarına 'İş Akışı' adımı da eklemişlerdir (CMS Review, 2006:1).

Günümüz teknolojisinde en çok kabul gören İYS sayfa veya makale oluşturma mantığı ile çalışır. Blog oluşturma sitelerinin gelişmiş versiyonu niteliğinde olan İYS, basit teknolojik altyapılarına rağmen eklenti ve tema desteği sayesinde zengin çıktılar vererek, günümüz internet teknolojisi ihtiyaçlarının birçoğunu karşılayabilmektedir. En gelişmiş ve yaygın eklenti/tema ağı olan İYS yazılımları; Wordpress, Joomla, Drupal'dır.

Bu tarz İYSler son kullanıcıya makale formunda yeni içerik tanımlamalarına imkân verirler. Bu makaleler düz metin formatında olsalar da, fotoğraf gibi farklı türde kaynaklara işaretlemelerle yer verirler. Sistem ön tanımlamalara göre makale metnine stil ayarları yapar, ancak içeriği ve görselliği yapısal anlamda ayrı katmanlarda tutar. Bu yapının kendi içinde çok sayıda avantajı vardır, özellikle çok sayıda makaleyi sistemde yayınlarken 'bak ve hisset' (look and feel) terminolojisini her sayfada sürdürebilmek gibi. Bu sayede makale havuzu kullanıcının eklediği içerik ile daha da genişleyecektir. Zaman zaman sistemler belli kesim kullanıcılara iş akışı konsepti de uygularlar, bu yeni eklenmiş bir içeriğin sistemde nasıl döndürüleceği ile ilgili bir tanımlamadır (Çatal, 2006:14).

Bunların dışında, İYS kategorisinde değerlendirilebilecek farklı ürünler de mevcuttur. Wiki (Ansiklopedik Bilgi Paylaşım Ağı), Uzaktan Eğitim Sistemleri, Elektronik Ticaret Paket Siteleri de genişletilebilir oldukları sürece içerik yönetiminin farklı konseptlerini oluştururlar.

Tür kapsamında değerlendirilen öğrenme içerik yönetim sistemleri (Learning Content Management Systems) ise elektronik eğitim ve öğretim alanında içerik oluşturulması, yönetimi, dağıtımı ve öğrenme materyallerinin yeniden kullanılabilmesi gibi amaçlarla katkı sağlar ve SCORM standartları gibi yetkinlikleri taşır (Ozan, 2009:5).

1.3. TEMA VE EKLENTİ SEKTÖRÜ

İYSlerin yaygınlaşması ile küresel çapta yüz binlerce geliştirici ve tasarımcı gelişim platformlarına ortak olarak, İYSlerin market sayfalarında kendi tasarladıkları tema paketlerini ve geliştirdikleri eklenti ve modül genişletmelerini ücretsiz ya da ücret karşılığı olarak dağıtmaktadırlar.

Bazı kaynaklara göre, uluslararası platformlarda faaliyet yürüten organizasyonlar açısından İYSler modern ve gerekli bir varlık yönetimi kanalı haline dönüşmüştür. Günümüz bilişim dünyasında İYSler kendi alında bölünmüş bir alt pazar ya da kendi başına bir yazılım endüstrisi haline gelmiş durumdadırlar. Birçok yazılım firması ürünleri ile birçok geliştirici de açık kaynak kod projeleri ile bu pazara katkıda bulunmaktadır ve İYS kavramının ölçeği büyüyerek terimin anlamı parçalanmaya başlamıştır (Kırbaş, 2007:7).

Modül ve temalar, standart yazılım geliştirme süreçlerinden farklı olarak daha karmaşık yapıda olduklarından çökme ve yeni versiyonlarla uyumsuzluk gibi sorunlar ile karşı karşıyadırlar. Buna rağmen market alanları çok geniş kitlelere ve ihtiyaç sahibi internet sitesi kullanıcılarına hitap ettiklerinden, geliştiriciler büyük cirolar sağlayabilmektedirler. Bu sebeple genişletme sektöründeki katılımcılar, yazılım sektörüne de ciddi katkılar sağladığı varsayılabilir. Örnek olarak, Envato Market kanalı üzerinden en çok satılan Wordpress içerik yönetim sistemi teması; ‘Good Space - Responsive Minimal WP Theme’ satın alma başına 52usd ücret talep ederek, 3 Şubat 2017 gözlemlemelerine göre tam 6,491 adet kullanım lisansı satmıştır. Toplamda 337,532 USD brüt (platform komisyonu ve elektronik para transfer komisyonları giderleri, inceleme tarihinden sonra yapılan iade tutarları hariç) gelir elde eden geliştirici, uzaktan teknik destek süresini 6 ay dan 12 aya çıkaranlardan da ek olarak 15 USD ücret talep etmektedir. Bu bilgi ‘Good Space Responsive Minimal WP Theme’ isimli ürünün incelenmesi ile elde edilmiştir (Envato Market, 2009).

Bu örnek ürün değerlendirmesi sadece Wordpress altında çalışan tek bir tema paketi için yapılmıştır. Market alanındaki diğer temaları, eklentiler ve genişletme paketleri bile hesaba katıldığında sadece Wordpress altında geliştirilen ücretli dağıtımlar ele alınmış olunur. Piyasada dağıtılan diğer İYSler ve elektronik ticaret gibi benzer yapılardaki açık kaynak kodlu ürünlerin genişletme paketlerinin piyasa değerleri de analiz edilirse, sektörün ne kadar büyük bir kitle desteği ile ayakta kaldığı net olarak ortaya konulabilir.

1.4. ÇEŞİTLİLİK VE TERCİH YÖNELİMİ

İçerik Yönetim Sistemleri, farklı isim ve platform destekleri ile piyasaya sürülürken, sistemin hangi öğeyi içerik anlamında baz aldığı oldukça önemlidir. Sayfa oluşturma ve web sitesi yaratma bazında çalışan İYSler içeriği makale ya da sayfa bazında tutarken, B2C platformu bazında çalışan elektronik ticaret paketleri içeriği ürün bazında tutar. Wikipedia benzeri dağıtımlar içeriğe ansiklopedik bilgi bazında yaklaşırken, bazı sistemler proje tabanlı çalışarak içeriğe iş sürecinin bir parçası olarak yaklaşırlar. Her sistemin içeriği referans aldığı somut karşılık birbirinden farklı olabilirken, benzer yaklaşımlara göre birbirine ikamesi olan ürünlerin belli konseptler altında toplanmıştır.

Aşağıdaki listede, içerik kavramını birbirinden farklı yapılarda ortaya koyan İçerik Yönetim Sistemleri sıralanmıştır (my2tech, 2016:1);

- DotNetNuke
- Drupal
- e107
- eBilge CMS
- Elgg
- Elxis CMS
- eZ Publish
- GBAK
- IBM Lotus Domino
- IBM Websphere
- Joomla
- Kentico
- Mambo
- Moodle
- OSCommerce
- PHP-Nuke
- phpKF-CMS
- Pligg
- Plonfff
- PostNuke
- Simple Machines Forum

- SiteFrame
- Sitecore
- SolidSite
- TYPO3
- Webrender
- WordPress

İçerik Yönetim Sistemleri dinamik ve geliştirilebilir altyapıları ile benzer yapılarda piyasaya sürülseler de, nihai ürün bazında taşıdıkları farklılıklar sebebiyle kullanıcı kitleleri bu ürünler arasında spesifik tercihleri doğrultusunda seçim yapmak durumunda kalırlar. Yazılım paketlerinin güçlü yönleri farklı beklentilerdeki kullanıcıların tercihlerine hitap ederken, bu dağıtımlar birbirinden farklı özellik ve destekleriyle İYS sektöründe çeşitlilik niteliği taşır. Tablo 1'de popüler ürünlerin hangi tercihlere göre talep gördüğü incelenmiştir.

Tablo 1: İYS Dağıtımlarında Kullanıcı Taraflı Tercih Sebepleri

Ürün	**Tercih Sebebi**
Wordpress	En geniş eklenti/tema çeşitliliği
Joomla	Güçlü sosyal ağ ve e-ticaret adaptasyonları
Drupal	Performans, modifikasyon imkânları
Moodle	Öğrenme yönetim sistemi

Kaynak: Yazar tarafından derlenmiştir.

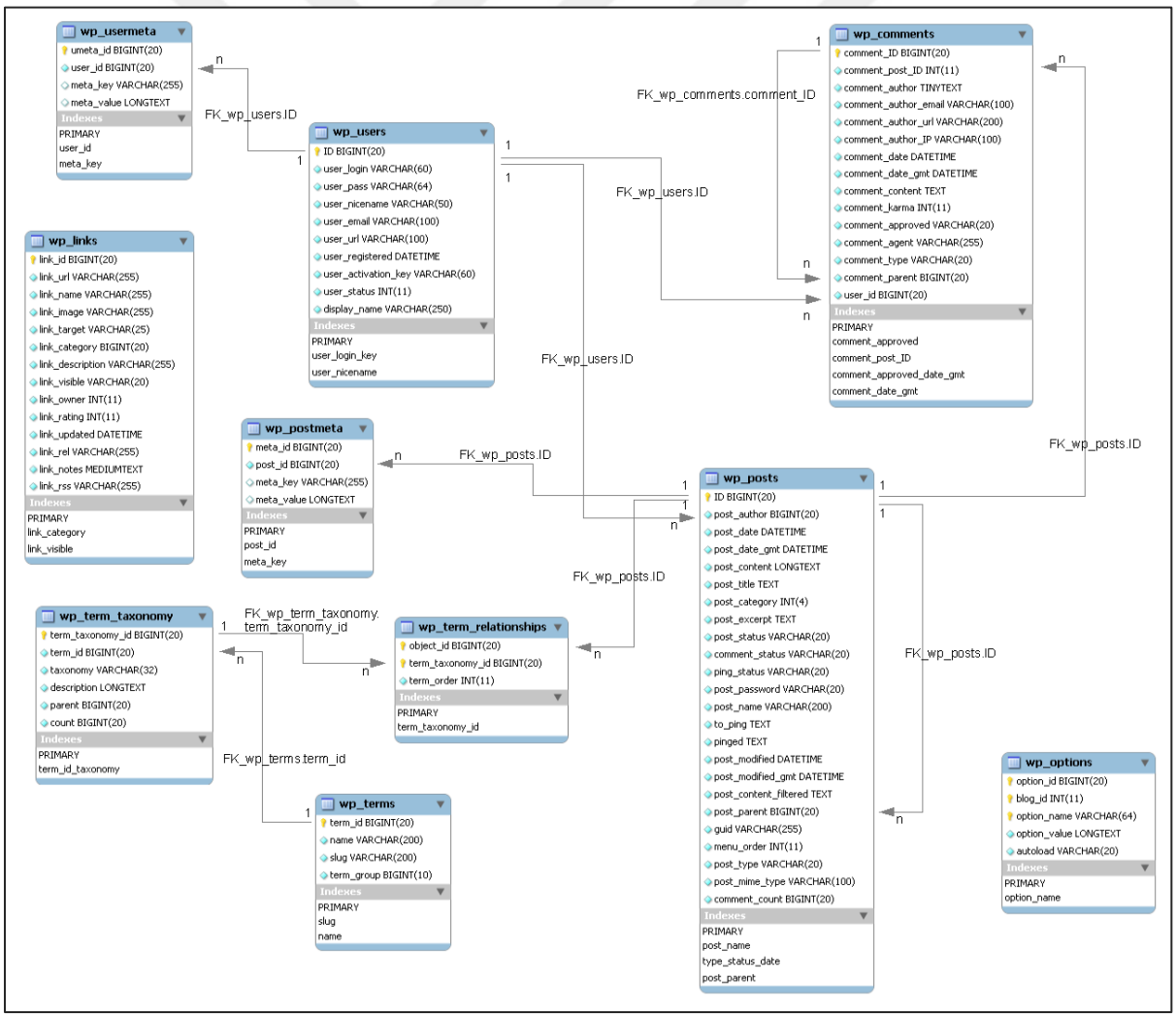
İKİNCİ BÖLÜM
İÇERİK YÖNETİM SİSTEMLERİNİ GELİŞTİRMEYE YÖNELİK MODEL TASARIMI

2.1. TEMEL YAPININ OLUMSUZLUKLARI

2.1.1 Veri Saklama ve İşleme Yöntemleri

İYSler çok farklı veritabanları üzerine çalışıyor olsa da, temelde benzer prensipler üzerine verileri saklarlar. Günümüzde kabul gören ve yaygınlaşan yapı, blog ve internet makalesi oluşturma mantığı üzerine kurulduğundan, içerik katmanına yaklaşım da temel olarak sayfa ya da sayfanın yapısına gömülü olan metin bazında gerçekleşir.

Şekil 4: Wordpress Veritabanı İlişki Diyagramı



Kaynak: Wordpress Codex, 2010:2

Şekil 4'te görülen Wordpress veritabanı incelendiğinde, temelde blog veya makale prensibi ile verilerin saklandığı ve işlendiği değerlendirilebilir. Modül bazında yüklenen eklentiler bu verilere erişebildiği gibi kendi özel veri saklama alanlarını da oluşturur ve kullanırlar. Bu anlamda İYSlerde verimsiz istif alanı, istemsiz veri tekrarı gibi olgulardan bahsedilebilir.

Bir diğer problem ise, veri tutarlılığı ve dağıtımdaki sorunlardır. Örneğin, temsili X alan adı üzerinde çalışan bir web alanına kurulacak Wordpress içerik yönetim sistemi, içerik kayıtları oluşturulup eklentiler kurulduktan ve temalar yüklendikten sonra, bir sebepten internet sitesi temsili Y alan adına bağlı bir web alanına transfer edilmek isteniyor. Diğer bir deyişle, internet sitesinin erişim adresi değiştiriliyor. Bu durumda, tüm dosyalar ve veritabanı taşındığı halde internet sitesindeki birçok içerik, fotoğraf ve medya dosyası kullanım dışı kalacaktır. Bunun temel sebebi, bazı eklentilerin iç sayfalar arası erişim adresini saklarken tam mutlak adres (absolute url) formatında istiflemesi olarak kabul edilebilir. Bu gibi sorunların çözümüne ilişkin arama motorlarında onlarca makale bulunsa da, temelde benzeri sorunun çözümü angarya yöntemler ve veritabanındaki kayıtlı adresleri teker teker müdahale ile düzeltmek olacaktır. Bu anlamda, konfigürasyon dosyalarının düzenlenmesi de yeterli olmayacaktır.

2.1.2. Eklenti Geliştirme Sürecindeki Zorluklar

Her ne kadar, genişletme yapılarının pazarlama ve yaygınlaştırma ortamları küresel anlamda yeterliliği korusa da, eklentilerin ve temaların fonksiyonel yapıları karmaşıklaştıkça geliştirme süreçleri de standart web uygulamalarının iyileştirme süreçlerine kıyasla daha karmaşık hal alabilir. Diğer taraftan, eklenti ve temaların birbirleri ile çakışma, yazılım paketinin versiyonu ile uyumsuzluk sorunları sektörün en güncel problemlerinden sadece birkaçıdır.

Güvenlik sorunu ise bu başlık altında incelenmesi gereken önemli noktalardan biridir. Ticari amaçlarla piyasaya sürülen genişletme eklentileri çok sayıda web sitesinde gerçek zamanlı olarak kullanılmakta ve ticari amaçlarla kurulmuş İYSlerde hassas bilgilere teknik anlamda erişim sağlayabilmektedir. Bu eklentileri satın alan ve internet sitelerine kuran geliştirici ve kullanıcılar güvenlik kaygılarında ne derece hassas olurlarsa olsunlar satın aldıkları eklentilerin geliştiricileri aynı hassasiyette ya da uzmanlıkta olmayabilirler. Bu gibi durumlarda online ödeme entegrasyonuna sahip ya da kişisel bilgilerin kayıt altına alındığı sistemlerde sonradan kurulan çeşitli eklentiler, sistemde güvenlik zafiyetine sebep

olabilmektedir. Günümüzde saldırganların, güvenlik açığı bilinen popüler eklentileri kullanan hedef siteleri arama motorları üzerinden örümcek yazılımları sayesinde tespit edebildiği ve basit işlemler ile toplu saldırılar yaparak yönetici erişimi sağlayabildiği ya da web sitelerinin ana sayfalarını otomatik olarak tahrip edebildiği bilinmektedir. Gün geçtikçe bu tarz siber saldırı araçlarının sayıları artarken, yeni eklentiler de küresel pazarda yerini almaktadır. Bu durum, yasal standartlar ya da kurumsal kaygılardan uzak eklenti pazarının küresel bazda siber güvenlik sorunsalı oluşturduğunun en güçlü dayanağıdır. Eklenti genişletmelerinin artık siber güvenlik konusunda uzmanlık gerektirmeyen daha güvenli ve yapısal bir platformda geliştirilmesi kaçınılmazdır.

2.2. HARPA İÇERİK YÖNETİM SİSTEMİ MODELİ

2.2.1. Amaç

Profesyonel amaçlarla piyasada kullanılan İYSlerdeki mevcut verimsizlikler ve uygulamaların gelişim hızını azaltan mimari sebeplerin ve teknik paradoksların aşılması; sıkça karşılaşılan esneklik sorunları, performans verimsizlikleri ve web hizmetlerinin süreçlere adapte edilememesi gibi sorunların aşılması; tüm bu avantajları sağlayacak bir bilgi sistemi modelinin yazılım geliştirme alanına entegre edilmesi amaçlanmıştır.

Ayrıca İYS kapsamına giren bilgi sistemi türlerinin kapsamını genişletmek amacıyla yeni bir konseptin ortaya konması hedeflenmiştir.

Geliştirilen yeni nesil model, kullanıcı dostu çıktılar alabilen kurumsal İYSler için iş akış destekli bir platform oluşturma amacı taşır. Bu anlamda, yazılım geliştiricileri ve proje yöneticileri için alternatif bir yapı örneği teşkil eder.

2.2.2. Tanım

Model, nitelikleri sebebiyle; Harmanlanabilir Sayfalama ifadelerinin İngilizce karşılıkları olan ‘Harvestable’ ve ‘Paginal’ sözcüklerinin ilk üç ve iki harflerinden oluşan ‘HARPA’ ismini taşır. Buradaki harmanlanabilir ifadesi, yapının veritabanı tarafındaki veri saklama ilkesinin özellikleri ifade ederken; sayfalama sözcüğü ise kullanıcı arayüzü oluşturulurken kendine özgü hiyerarşik yapının korunarak iç içe geçen içeriklerin sayfayı

oluşturması ilkesini ifade eder. Temel olarak; 'Harvestable' veritabanı yapısını, 'Paginal' ise kullanıcı arayüzü yapısını ifade eder.

Bu yapılar aynı zamanda kullanıcı deneyimi faktörlerinden 'Arayüz' ve 'Bilgi Altyapısı' unsurlarını temsil ettiğinden enformatik model, kullanıcı deneyimi destekli bir yapıdır.

Söz konusu modelin yazılım geliştirme süreçlerinde kullanılmasının, çeşitli avantajları beraberinde getireceği iddia edilmektedir;

- Dağıtım aşamasında bilinen sorunların yaşanmaması
- Daha yapısal veriler ve kitlesel veri
- Tema ve Eklenti kavramlarını tek çatı altında birleştiren kontrolör yapısı
- Daha kaliteli çıktılar (UI) ve kullanıcı etkileşimi (UX)
- Kontrol panelinde 'shortcode' ve 'parameter' gibi kullanımlar yerine kolay sürükle-bırak işlevler
- Proje Kapsamlı kullanım ve Form desteği

2.2.3. Metodoloji

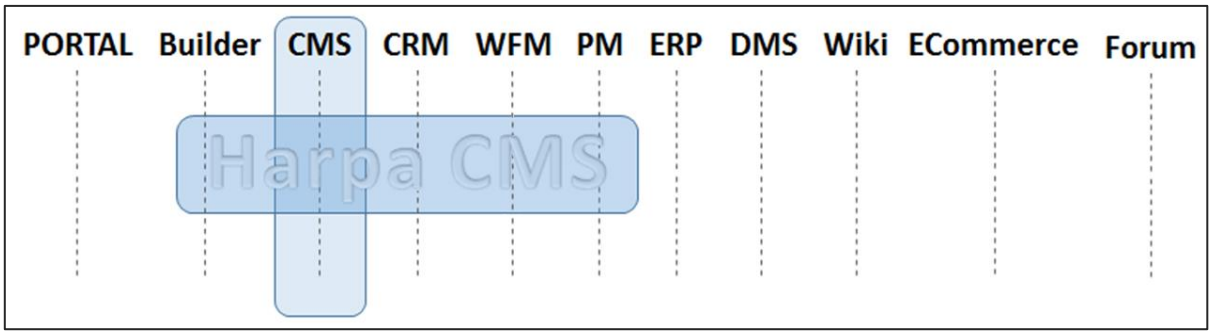
Modelin işlevselliği ve yapılabilirliği teknik anlamda ispatlanmıştır ve örnek işletme faaliyetlerine entegre edilmiştir. Uygulama bazında henüz tamamlanmamış kısımlar mevcut olsa da, formdan türeyen bilginin yaşam döngüsünü tamamlayabilmesi için asgari koşullar sağlanmıştır. Yönetim Bilişim Sistemlerine en önemli katkısı, kolay kullanımın ve esnekliğin ötesinde, gelecek çağın Bilgi Teknolojilerine ışık tutmasıdır. Model geliştirildikçe, İş Akış Yönetimi uygulamalarının yerine geçebileceği ve MİY türevi alternatiflerin yerine de ikame olabileceği iddia edilmektedir.

Geliştirilebilir eklenti/modül destekli tüm yazılım grupları; geliştirme modeli, arayüz, çıktılar ve kullanıcı etkileşimi bazında farklılıklar içerir. Birden çok gruba kayan uygulamalar konsept yapısını oluşturur. Böylelikle Harpa İçerik Yönetim Sistemi yapısı da, kendine özgü modeli ve nitelikleri ile benzersiz bir konsept oluşturarak çok daha geniş bir kullanıcı kitlesine hitap eder.

HARPA İYSnin hangi konseptleri kapsadığı ve hangileri üzerine katkı sağlamadığı Şekil 5'te görülmektedir. Platform ve model; kolay web sayfası yapıcı araçlar (Builder), web günlüğü (Blog), müşteri ilişkileri yönetimi (CRM), iş akışı yönetimi (WFM), proje yönetimi (PM) gibi konseptlerin özelliklerini barındıran hibrit bir yapı taşır. Yapısal anlamda, portal, kurumsal kaynak planlaması (ERP), doküman yönetim sistemi (DMS), ansiklopedik bilgi

(Wiki), elektronik ticaret (ECommerce), tartışma platformu (Forum) gibi alanlarda özellikler barındırmasa da, sanal pos gibi entegrasyonlu eklentiler geliştirildiği takdirde elektronik ticaret amacıyla da kullanılabilir.

Şekil 5: HARPA İYS Konsept Diyagramı



Kaynak: Yazar tarafından derlenmiştir.

Genel kapsamda İYSler; WCMS (Web İçerik Yönetim Sistemi), CCMS (Bileşen İçerik Yönetim Sistemi), ECMS (Kurumsal İçerik Yönetim Sistemi), DMS-EDM (Doküman Yönetim Sistemi veya Elektronik Doküman Yönetimi), LMS-LCMS (Öğrenim İçerik Yönetim Sistemi), EDRMS (Elektronik Belge ve Kayıt Yönetim Sistemi) ve PCMS(Yayın İçerik Yönetim Sistemi) gibi türlere de ayrılırlar (Karataş, 2013: 156; Samuels, 2013). Web tabanlı HARPA İYS, WCMS kapsamına girmektedir.

2.2.4. Modelin Teknik Altyapısı

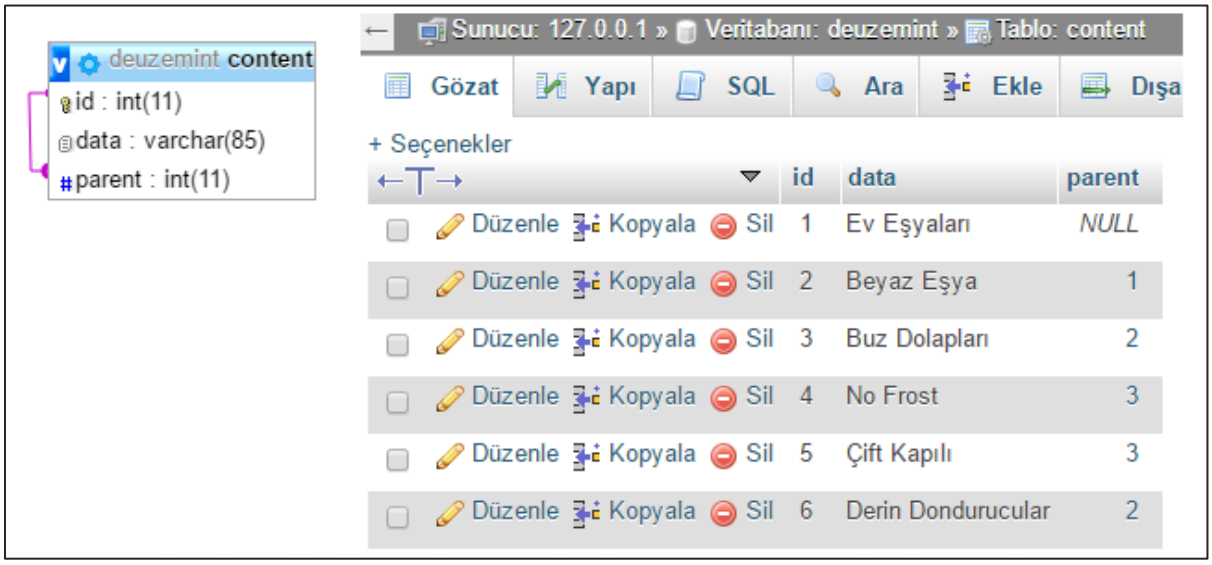
2.2.4.1. Hiyerarşik Modelin Temeli

İçerik Yönetim Sistemlerine hiyerarşik tabanlı yeni bir model önerisinde bulunabilmek için öncelikle günümüzde kullanılan hiyerarşik veritabanı sistemini ve kullanım alanlarını doğru şekilde ifade etmek gerekir.

İlkel ya da minimal hiyerarşik model temelde tek bir sabit tablodan oluşur. İçerik ya da verilerin tutulduğu alanda bir veri türü tanımlanmıştır. Aynı tür altında, web sitesindeki tüm menü etiketleri, linkler, metinler ve HTML içerikler barındırılır. Veriler hiyerarşik olarak birbirlerine ast üst ilişkisi ile bağlanırlar. Bu yapıda nümerik veriler dâhil her türlü veri aynı alan altında tutulamayacağından geniş kullanım alanlarında hem verimsiz hem de SQL sorguları bakımından performans düşmanı bir yapıdır. Bu sebeple, minimal hiyerarşik model internet sitelerinde kısıtlı ve belli amaçlar amacıyla kullanılır.

Şekil 6'da ki ekran görüntüsü, hiyerarşinin doğru amaçla kullanımında tutulan verilerin bir örneğini yansıtır, elektronik ticaret sitelerindeki basit bir kategori ve alt kategoriler yapılandırılması bu veri modeli kurularak gerçekleştirilir. Çeşitli otomasyon uygulamalarında da muhtelif amaçlar ile kullanıldığından bu saklama yöntemine sıkça rastlanabilir. Tek tip veri saklama amacıyla kullanılması şartıyla; ihtiyaçları karşılayan ve performans bakımından sorun teşkil etmeyen bir yapıdır.

Şekil 6: İlkel Hiyerarşik İçerik Saklama Yaklaşımı Veritabanı Yapısı



Kaynak: Yazar tarafından derlenmiştir.

Basit hiyerarşik modelin güncel kullanım alanları aşağıdaki gibi tanımlanabilir;

- Forum: Yetki/Rütbe Düzeyi Tanımlamaları
- E-Ticaret: Kategori / Alt Kategori Tanımlamaları
- Kargo Sistemleri: Şehir/Cadde / Sk. Alt Tanımlamaları
- İYS: Menü Tanımlamaları

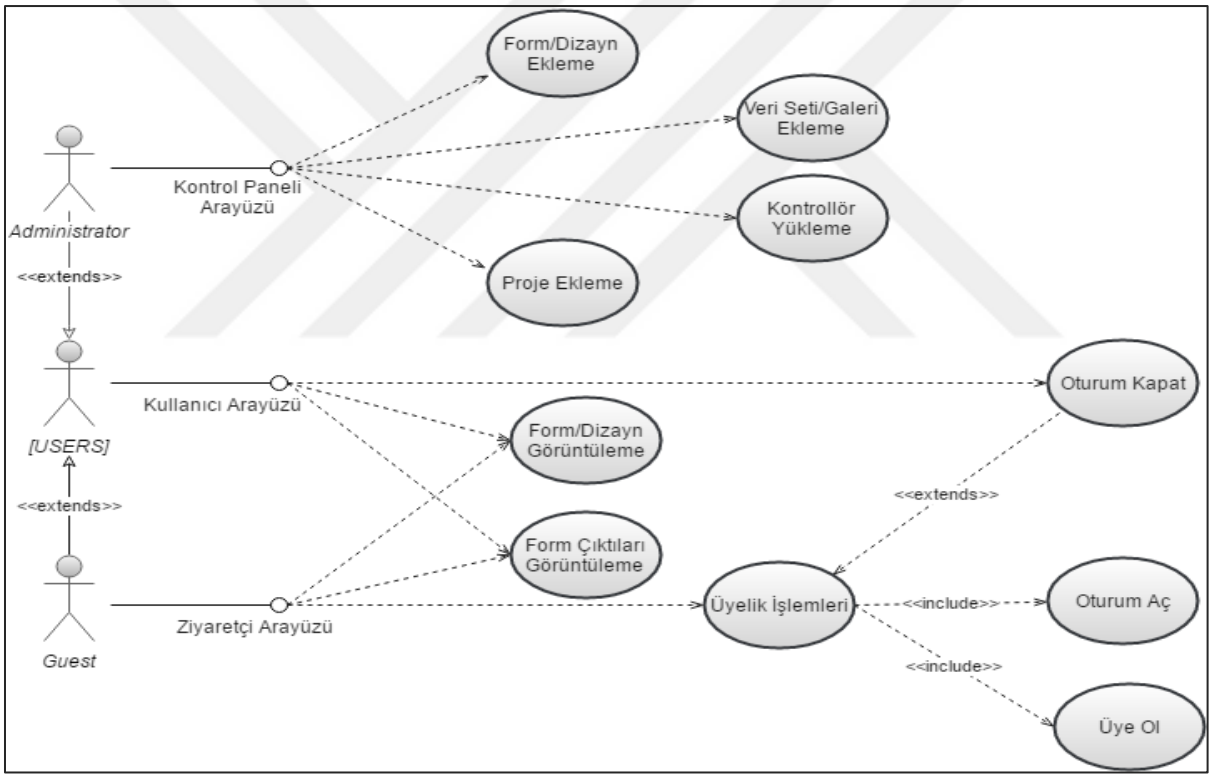
Bu yapının içerik yönetim sistemi veri saklama altyapısında sağlıklı olarak kullanılabilmesi mümkün olmamaktadır. Bunun sebebi tek bir veri alanında (field) farklı türde veri ve içeriklerin tutulamamasıdır. MYSQL ve diğer veritabanı sistemlerinde her sütun için yalnızca bir veri türü tanımlanabilir ve sistem bu türe göre bellekte ilgili saklama alanını tahsis eder. 'LONGTEXT' ve benzeri türlerdeki içerikleri saklayabilecek büyük bir alan tanımlandığı takdirde de sistemin diğer katmanlarını etkileyebilecek performans sorunları ortaya çıkabilir. Bu sebeple; yapısal ve hiyerarşik desteği olan bir model türetilmesi için veri alanlarının türlerine göre kümelere ayrılması gerekliliği ortaya çıkmıştır.

Hiyerarşik modelin gelişmiş yapısı ‘Ağaç Modeli’ olarak bilinir. Tablolar birbirlerine ast üst ilişkisi ile bağlanır. Otomasyon sistemlerinde kullanılabilen ancak popüler olmayan hiyerarşik yapıdır.

2.2.4.2. Tasarlanan Model

HARPA İçerik Yönetim Sistemi, dinamik otomasyon ve içerik sayfalama yapılarının karma hali olarak değerlendirilebilir. Modelin kapsamlı nitelikte ortaya konabilmesi için farklı katmanlarda işleyişinin diyagramlar üzerinden incelenmesi gerekir. Ancak bu gösterimlerin sadece yaygın ‘blogger’ yani günlük tabanlı modele olan farklılıkları ele alınacaktır.

Şekil 7: HARPA İYS Modelinin Aktör Diyagramı



Kaynak: Yazar tarafından derlenmiştir.

HARPA veritabanında her bir içerik kaydı ve adı ‘content’ tablosunda merkezi olarak tutulur. İlişkisel olarak tüm diğer tablolar içeriğe ilişkin her türlü bilgiye ‘content.id’ birincil anahtarı ile ulaşır. Her bir içerik kaydının tür id bilgisi ‘properties’ tablosunda sakladığından, bir içeriğin ‘data’ kaydına ‘properties.fkContent’ üzerinden ‘properties.fkType’ verisi alınarak, daha sonra bu veri üzerinden ‘types’ tablosundaki ilgili tür kaydına ‘types.id’ üzerinden ulaşarak ‘types.out_cluster’ yani içerik ham verisinin tutulduğu hedef veri kümesi

öğrenilir. Bu hedef küme tablosu MYSQL tarafındaki saklı yordamlar ile dinamik sorgularla çalıştırılarak içindeki ‘fkContent’ dış anahtarı ile ilgili içerik ‘id’ numarası eşleştirilerek ‘data’ tablo alanına ulaşılır. Ham veri kaydına ulaşıldıktan sonra ilgili sorgu işlemi bu alan üzerinden gerçekleştirilir. Bu sayede veriler yapısal türlerine göre ayrılırken, ham veriler ve faal olarak oluşturulan veriler birbirinden ayrılmış olur. Bu verimli yapıda küme kayıtları ‘types’ tablosunda saklanırken, küme ön ekleri ‘cluster_’ ile başlayarak, 001’den 0016’ya kadar artan rakamlar son ek olarak tablo adına eklenmiştir. Her bir küme tablosunda üç alandan fazla sütun olmayacağından sorgular performanslı şekilde gerçekleştirilir. Ancak burada dikkat edilmesi gerek husus, bir içeriğin ismi var ise bu isim bir ham veri olacağından ve her durumda ‘varchar’ türünde tutulmak zorunda olduğundan bu isimler her zaman ‘cluster_009’ tablosunda tutulur. Bu bilgiye, ilgili ‘content’ tablosu kaydındaki ‘content.fkClusterAlias’ içinde tutulan yabancı anahtar, ‘cluster_009.id’ birincil anahtar alanı ile eşleştirilerek ulaşılır. Geri dönen ‘data’ alanı kaydı ilgili içerik kaydının isim yani ‘Alias’ bilgisidir; formal sayfalarda ve sistemdeki birçok sayfada bu isim ilgili kullanıcıya gösterilir.

Hiyerarşik olarak tutulan her türlü statik ve formal bilginin ve bu formlardan üretilen form girdilerinin durumlarını yorumlayabilmek için HARPA sistemi bilginin aşamasına göre bir seviyelendirme tanır ve bunlar tanımlı türler tablosu ‘types’ içerisinde ‘level’ alanında tutularak o türdeki tüm bilgilere ait bir durum niteliği taşır.

Tablo 2: HARPA İYSde Tanımlı İçerik Türlerinin ve Kontrolörlerinin Sahip Olabilecekleri Seviyeler

Seviye	Durum
0	Bilgi Aktaramayan Yapı (form, section, image)
1	Bilgi Aktarabilen Yapı (textbox, checkbox, textarea)
2	Bilgi Tutan, Aktaramayan Küme (Veri Seti, ID Köprüsü, Sayaç, Statik Bilgi)
3	Zaten Aktarılmış - Form Üzerinden Kaydedilmiş- Bilgi Kümesi

Kaynak: Yazar tarafından derlenmiştir.

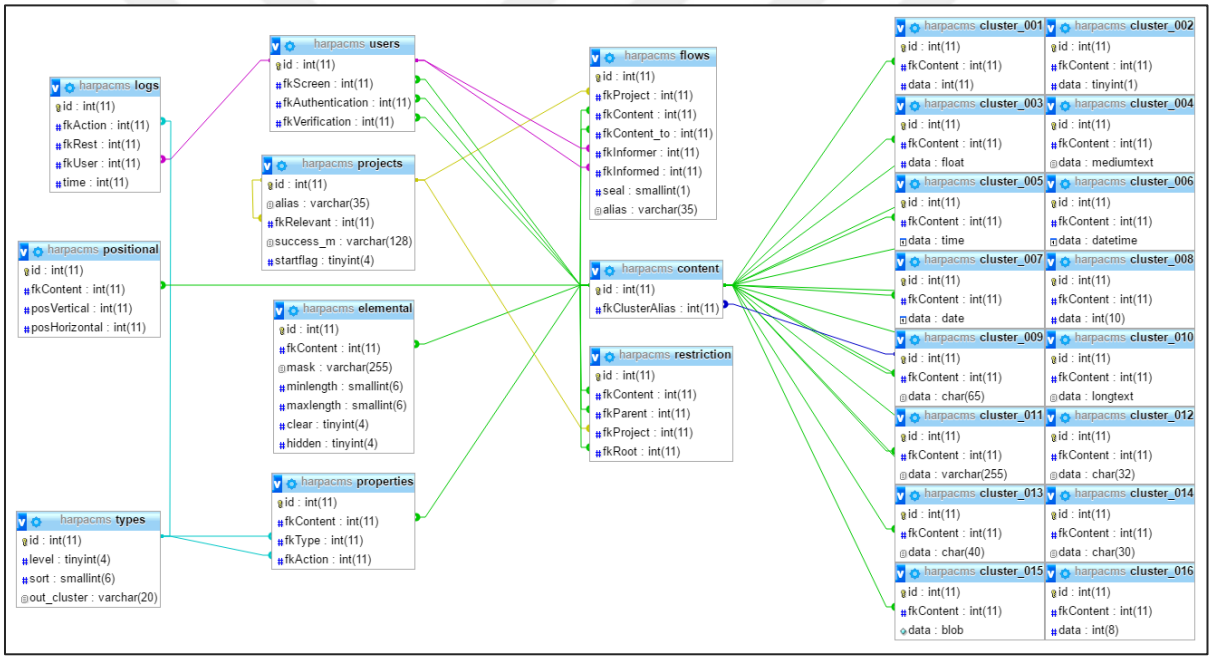
Bir içeriğin hiyerarşik olarak alt verisi olduğu bilgisi ‘restriction.Parent’ ve ‘restriction.Root’ alanlarında tutulurken, o içerik formal olarak oluşmuş ise hangi proje altında oluştuğu bilgisi ‘restriction.fkProject’ alanında saklanır.

Formal olarak tetiklenen her türlü hareket ‘logs’ tablosunda tutulur. Hangi kullanıcı tarafından, hangi içeriğin hangi aksiyon ile hangi zaman mührü ile gerçekleştirildiği bilgileri bu tabloda ayrı ayrı saklanır. Böylelikle formal hareketler ile kütük bilgileri aynı yerde verimli şekilde tutulmuş olur.

‘Elemental’ tablosu sadece formal içerikler için maskeleme, alabileceği minimum ve maksimum karakter sayısı, kullanıcıya tek tık ile form alanını temizleyebilme olanağı verilip verilmediği, gizli alan olup olmadığı durumu bilgilerini taşır.

Her kullanıcıya ait bilgiler ‘users’ klasöründe saklanır. ‘users.fkScreen’ alanı kullanıcı adı ham verisi, ‘users.fkAuthentication’ alanı kullanıcı parolası, ‘users.fkVerification’ alanı ise kullanıcı doğrulama bilgisi dış anahtarlarını tutar. Bu dış anahtarlar ile yine ‘content’ tablosu üzerinden ilgili kümeden ilgili ham veri çekilir. Buradaki ‘fkVerification’ alanı mevcut sistemde kullanılmamakta ve ileride yapılabilecek geliştirmeler için pasif olarak tutulmaktadır. Sistem ilk kurulduğunda ‘Guest’ ve ‘Administrator’ kullanıcı kayıtları hazır olarak gelir.

Şekil 8: HARPA İYS Veritabanı - İlişki Diyagramı



Kaynak: Yazar tarafından derlenmiştir.

‘Positional’ tablosundaki ‘posVertical’ alanı, formal olan ve olmayan tüm içeriğin dikey olarak sıralama bilgisini tutar. Aynı tablodaki ‘posHorizontal’ alanı ise sadece formal olmayan sayfalarda yatay düzlemdeki sayfa kolonunu belirtir. Bu bilgiyi tema niteliğindeki kontrolörler, ilgili içeriğin ilgili kolona yerleştirirken kullanır. İçeriklerde proje yapısını, ‘Flows’ ve ‘Projects’ isimli iki tabloda tutulan veriler belirler.

‘Project’ tablosundaki ‘Alias’ alanı, projenin adını direk olarak kendi içerisinde tutar. Bu tablodaki ‘fkRelevant’ alanı ise sadece projeleri birbirlerine bağlayabilmek adına alakalı diğer proje için dış anahtar görevi görür. ‘success_m’ alanı, formal sayfalardan oluşan bir

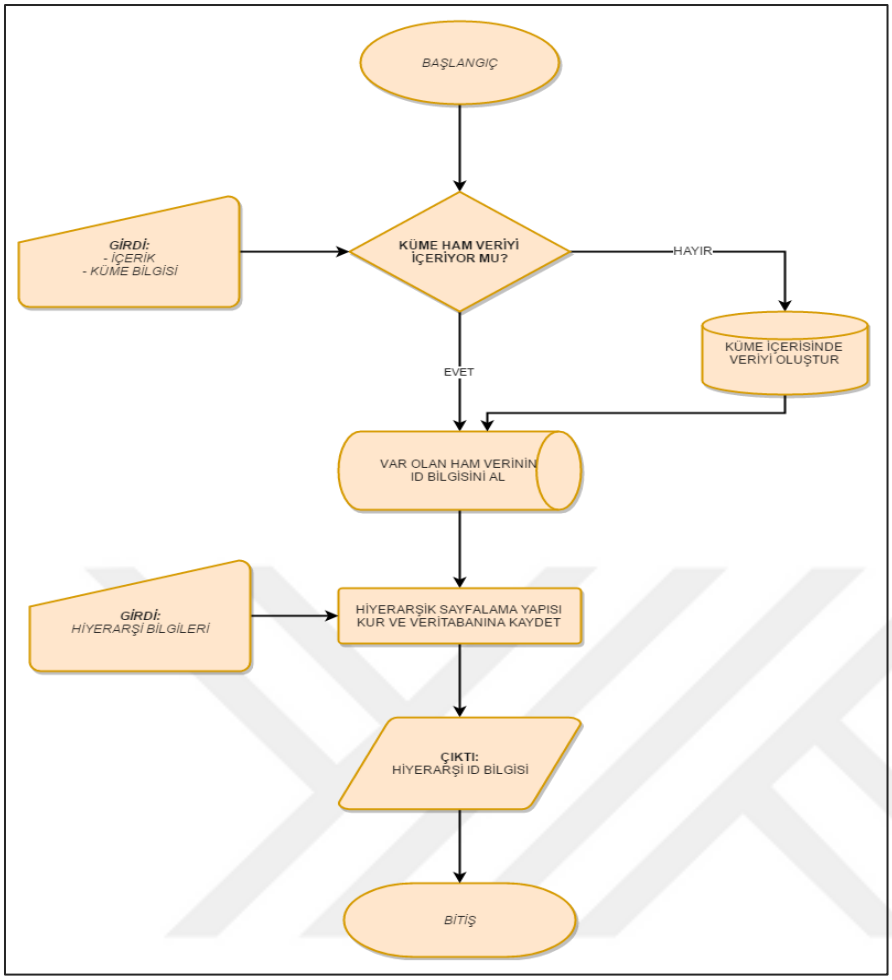
projenin her adımı geçilerek proje tamamlandıktan sonra kullanıcıya gösterilecek onay mesajını ‘varchar’ değerinde tutar. ‘startflag’ alanı binary değer olarak, hangi projenin web sitesine bağlanıldığında otomatik olarak açılacak başlangıç projesi olacağını belirtir. Hangi kaydın ‘startflag’ alanı nümerik olarak 1 değerini alırsa, o proje internet sitesinin başlangıç projesidir ve ilk iş akışı çalıştırılacaktır. Sistematik olarak, bir projenin ‘startflag’ alanı 1 ise diğer alanlar 0 değeri olarak değiştirilir, zira bir internet sitesi ilk açılışında ancak bir projenin içindeki bir iş akışını yani bir sayfayı çalıştırabilir. Bu alandaki binary veri TINYINT değerinde tutulur.

İş Akışları (Workflows), ‘flows’ tablosunda kayıtlı tutulmaktadır. ‘fkProject’ o iş adımının hangi projeye bağlı olduğu, ‘fkContent’ o iş akışı kaydının hangi formal ya da durağan sayfayı temsil ettiği, ‘fkContent_to’ bir sonraki iş akışı, ‘fkInformer’ o iş akışını hangi kullanıcının görüntüleyebileceğini, ‘fkInformed’ ise o iş akışı eğer dinamik bilgi aktaran formal bir sayfa ise hangi kullanıcının bu form girdilerini raporlama kısımlarında görüntüleyebileceği bilgisini dış anahtar niteliğinde barındırır. ‘seal’, eğer iş akışı formal bir sayfayı barındırıyor ise her bir erişim izni olan kullanıcının kaç defa formu doldurabileceği (mühür) bilgisini nümerik olarak taşır. ‘Alias’ ise bu iş akışının ismini sayfa isminden farklı olarak ‘varchar’ değerinde barındırır.

Veritabanının tamamını ele alırsak, sadece proje ve iş akışı isimleri ham veri olarak aynı tabloda tutulur ve kümelerden çekilmez. Bu durum elbette modelin verimsizliği niteliğinde ele alınabilse de; model ve veritabanı geliştirildikçe bu bilgiler kümelere taşınabilir, yapılacak bu çalışma ile tüm ham veriler ile parametre bazındaki değerler birbirinden kısmi anlamda izole edilebilir.

Şekil 9’da ki akış diyagramı, HARPA CMS platformunda oluşturulan formal alanlar üzerinden veri ekleme işlemi prosedürünü basit anlamda projekte eder. İşlem, formal bir sayfada bulunan ve ön tanımlı aksiyonlardan ‘Yeni Veri Ekle’ işlemi atanmış tüm form alanları için sayfadaki taranma sırasına göre ayrı ayrı tetiklenir. Bu durum örnek bir yeni veri ekleme işlemini açıklar. Farklı aksiyonlar tanımlanmış form elemanları da aynı şekilde sayfanın tarama sırasına göre ayrı ayrı tetiklenir ve aksiyon çağrılır.

Şekil 9: HARPA İYS Veri Ekleme Motoru Akış Diyagramı



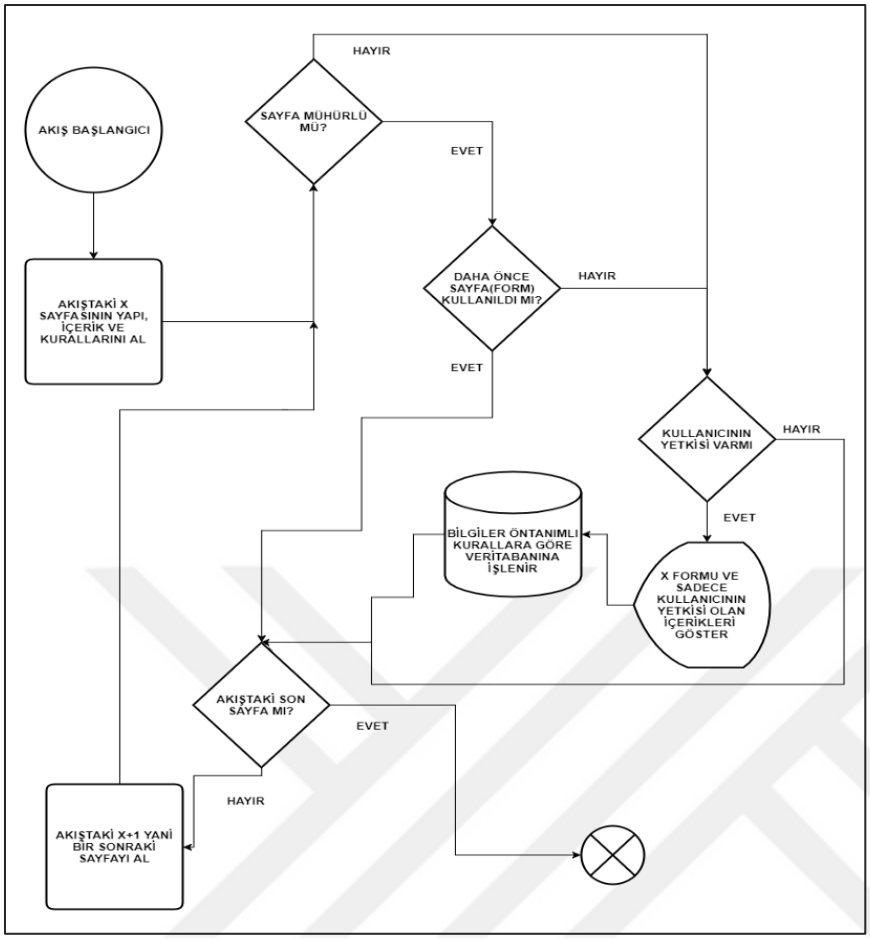
Kaynak: Yazar tarafından derlenmiştir.

Sistemde yüklü olan diğer aksiyon işlemleri şunlardır;

- ‘Tekil Veri Sil’
- ‘Güvenlik Kutucuğu Kontrolü Yap’
- ‘Onay Alanını Kontrol Et’
- ‘Kaydı Yenile’
- ‘Dosyaları Sunucuya Yükle’

Sistem dili İngilizce olduğu için ayar dosyalarında aksiyonların İngilizce karşılıkları kayıt edilmiştir.

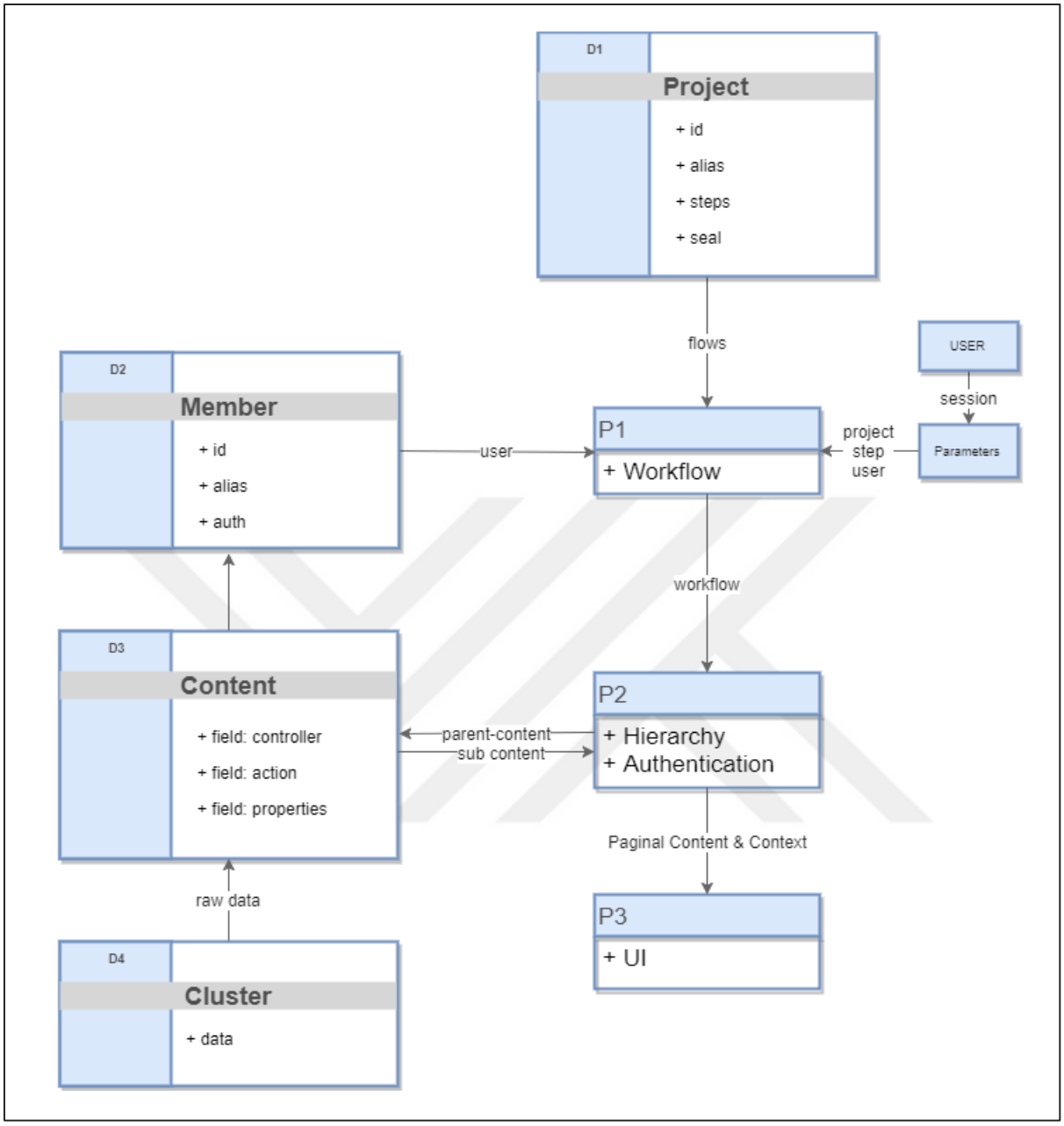
Şekil 10: HARPA İYS Sayfa Yetki Kontrolü - Akış Diyagramı



Kaynak: Yazar tarafından derlenmiştir.

Şekil 10'da ki akış diyagramı, HARPA CMS üzerinden oluşturulmuş bir sayfanın kullanıcıya özgü görüntüleme yetkisini denetleyen mekanizmayı temsil eder. Yetki kontrolünden geçen her sayfa, içeriği ekrana basar. Ancak veritabanı tarafında yetki kontrolü sayfa üzerinde değil, sayfayı çağıran iş akışı üzerinde gerçekleştirilir. Yani yayımlanmış her bir sayfa, en az bir proje içinde tanımlı bir iş akışına bağlı olduğu için; kullanıcının bir sayfayı görüntülerken sayfanın bağlı olduğu iş akışına erişim hakkı olması gerekir. Söz konusu denetim yapısı; kullanıcının iş akışına bağlı sayfayı izleme yetkisi olup olmadığını, eğer form içeren dinamik bir sayfa ise bu sayfayı daha önce kullanıp kullanmadığını, sayfayı kullanma limitini doldurup doldurmadığını kontrol eder. Form sayfalarındaki kullanıcı başına kullanım limiti, iş akışındaki mühür bilgisi ile tanımlanır. Yönetici oluşturmakta olduğu proje içerisindeki iş akışını tanımlarken, nümerik rakam ile kullanıcı başına limit girebilir. Bu sayede, iş akışına bağlı bir form ilgili kullanıcı tarafından sadece bir defa, tanımlı kez ya da sınırsız sayıda doldurulabilir. Kullanıcı, ilgili form sayfasını kullanma hakkını doldurduğunda, kullanıcıyı projedeki bir sonraki iş akışına bağlı olan sayfaya yönlendirilir.

Şekil 11: HARPA İYS Kullanıcı Arayüzü Veri Akış Diyagramı



Kaynak: Yazar tarafından derlenmiştir.

Şekil 11'de ki veri akış diyagramı Seviye-0 (Zero Level) bir gösterimdir. Bu projeksiyon sistemin veri akışını düşük ölçekli olarak gösterim amacı taşır, bağlam diyagramı (Context Diagram) olarak ta bilinir (Lucid Charts, 2017:6). 'Gane and Sarson' notasyonu kullanılmıştır.

Görsel, yetki kontrolünden geçen sayfanın içerik yapısının oluşturulma sürecini grafiksel projekte eder. P1, P2, P3 blokları işlem (Process) gruplarını temsil eder. D1, D2, D3, D4 ise veri (Data) gruplarını temsil eder. Oklar üzerinde etiketlenen bilgiler, parametreler ve sorgular sonucunda geriye dönen bilgi gruplarıdır.

Sistemin modüler ve geliştirilebilir yapısı, veri akışlarının tutarlılığı gibi konular katmanlı mimari gereksinimleri doğurur. Günümüzde yazılım mühendisliği için fiili zorunluluk anlamında bir standart haline gelen obje tabanlı programlama, problemler dizisine geliştirilen çözüm alanındaki varlıkların yazılım tabanlı nesnelerle modellenmesini öngören bir teknik' olarak tanımlanır (Taşdelen, 2005:1). Bu sebeple sistem obje tabanlı programlama (object oriented programming) niteliklerine göre çok katmanlı (n-tier) mimaride geliştirilmiştir. Bu sayede, içerik yönetim sisteminin yapısal olarak genişletilmesi (upgrading) ve kodların iyileştirilmesi (re-factoring) işlemleri verimli şekilde yürütülebilir.

Veritabanı tarafındaki sorgu işlemleri aşağıdaki saklı yordam ve fonksiyonlar ile otomatik hale getirilip MYSQL veritabanında çalışır halde tutulmaktadır.

Saklı Yordamlar:

- addContext: Yeni içerik ekler.
- addData: İçeriğe veri ekler.
- addElemental: İçeriğe biçimlendirme özellikleri ekler.
- addField: Sayfa içeriği içerik ekler.
- addFlow: Projeye akış ekler.
- addLog: Olay ekler.
- addPositional: Sayfa içeriğine yatay ve dikey pozisyon bilgisi ekler.
- addProject: Yeni proje ekler.
- addProperty: Alana tür bilgisi ekler.
- addPropertyWithAction: Alana tür ve aksiyon bilgisi ekler
- addRestriction: İçeriğe hiyerarşi bilgisi ekler.
- changeData: Kümedeki ilgili veriyi değiştirir.
- checkFilled: Boş alan kontrolü yapar.
- checkIfContentByUser: Kullanıcı daha önce içeriği
- checkIfContentExists: İçeriğin mevcut olup olmadığını kontrol eder.
- checkIfElementIsDataContainer: İçeriğim formal bir alan olup olmadığını kontrol eder.
- checkIfElementIsEntry: İçeriğin kullanıcı girdisi olup olmadığını kontrol eder.
- checkIfElementIsIDBridge: İçeriğin bir köprü olup olmadığını kontrol eder.
- checkIfElementIsVirtualDataContainer:
- checkIfHasChild: İçeriğin alt içeriği olup olmadığını kontrol eder.
- checkSeal: Form sayfası kullanım limitini verir.

- checkUserAuthFlow: Kullanıcının iş akışına olan iznini kontrol eder.
- checkUserAuthRep: Kullanıcının rapora olan iznini kontrol eder.
- createNewUser: Yeni üye yaratır.
- getActionFileName: Formal alanın tanımlı aksiyon dosyasını verir.
- getActionId: Formal alanın tanımlı aksiyon kimlik numarasını verir.
- getAllFlows: Projedeki tüm akışları getirir.
- getAllProjects: Kayıtlı tüm projeleri getirir.
- getAllSubContext: Alt içerikleri getirir.
- getAuthorizedProjects: Yetkili projeleri getirir.
- getAuthorizedReports: Yetkili raporları getirir.
- getAuthorizedStepsNum: Yetkili akışların sayısını getirir.
- getContentData: İçeriğin ham verisini getirir.
- getContentUserID: Girdiyi yapan kullanıcının kimlik numarasını getirir.
- getContentUserName: Girdiyi yapan kullanıcının kullanıcı adını getirir.
- getContext: İçeriği getirir.
- getContextMin: Alanın en az girilebilir karakter sayısını getirir.
- getContextPos: İçeriğin yatay pozisyonunu getirir.
- getContextSeq: İçeriğin dikey pozisyonunu getirir.
- getDataSetContent: Verisetini getirir.
- getDefinedActions: Sistemde tanımı tüm kullanılabilir aksiyonları getirir.
- getElementalProperties: İçeriğin biçimlendirme özelliklerini getirir.
- getEnabledContext: Aktif içeriği getirir.
- getEnabledContextAndEntry: Aktif içerikleri ve altındaki kullanıcı girdileri getirir.
- getEnabledContextByUserProject: Projedeki kullanıcının yetkili olduğu içerikleri getirir.
- getEnabledContextReversed: Aktif içeriği ters listelemeyle getirir.
- getFlows: Projedeki tüm iş akışlarını getirir.
- getFlowsIntoProjectByUserInformed: Projede kullanıcının yetkisi olan raporları getirir.
- getFlowsIntoProjectByUserInformer: Projede kullanıcının yetkisi olan formları getirir.
- getLastProjects: Sistemdeki en yeni beş projeyi getirir.
- getNextFlow: Projedeki bir sonraki iş akışının kimlik numarasını getirir.
- getParentContext: Hiyerarşik olarak bir üst içeriği getirir.
- getPhotoContent: Sistemdeki fotoğrafları getirir

- getPositional: Alanın sayfada tanımlanmış yatay ve dikey pozisyonlarını getirir.
- getProjectAndPageByFlow: Akış kimlik numarasından proje ve tanımlı sayfayı getirir
- getRootPages: Sistemde kayıtlı tüm sayfaları getirir.
- getStartFlag: Web sitesinde tanımlı başlangıç sayfasını getirir.
- getStartFlow: Projedeki ilk iş akışını getirir.
- getTheme: İlgili içeriğin tema bilgilerini getirir.
- getUserControls: Sistemde tanımlı tüm kullanıcı kontrolör bileşenlerini getirir.
- getUserNameById: Kimlik numarası ile kullanıcı adını getirir.
- getUsers: Sistemde kayıtlı olan tüm kullanıcı bilgilerini getirir.
- getUsersInProjectLogs: Projeye form üzerinden bilgi giren tüm kullanıcıları getirir.
- listContentMatches: İlgili kümede ilgili ham verinin mevcudiyetini kontrol eder.
- removeLog: Alan üzerinde oluşan kullanıcı olay kayıtlarını siler.
- removeProperty: Alanın tür ve aksiyon bilgilerini siler.
- removeRestriction: İçeriğin hiyerarşi bilgilerini siler.
- setStartFlag: İnternet sitesi için başlangıç projesi tanımlar.

Saklı Fonksiyonlar:

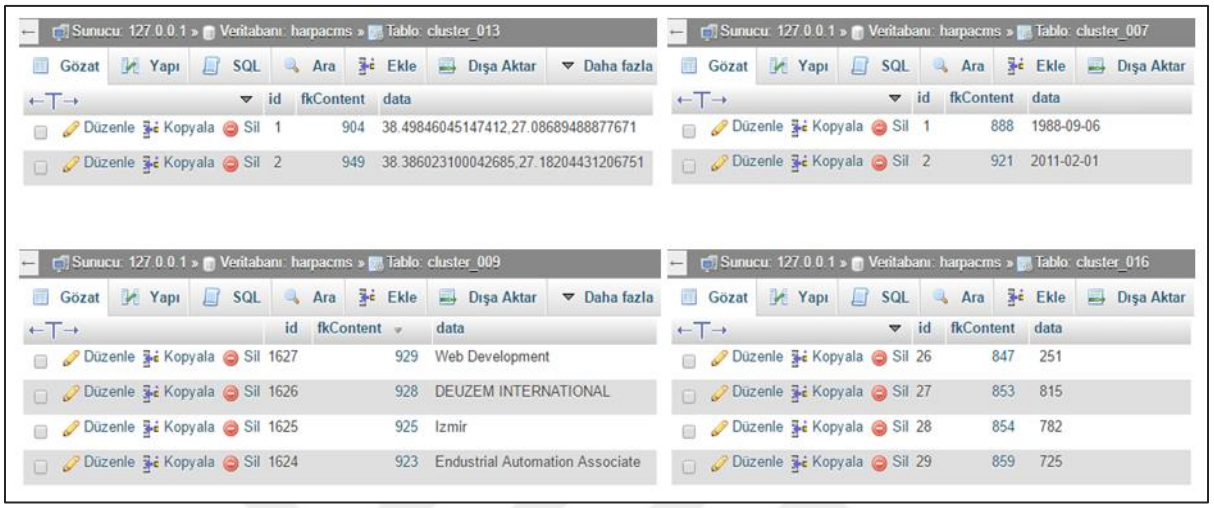
- getAlias: İçeriğin adını getirir.
- getContentClusterName: İçeriğin kayıtlı olduğu küme tablosunun adını getirir.
- getContentLogTime: İçeriğin kayıtlı olduğu zaman mührünü getirir.
- getContentType: İçeriğin tür kimlik adını getirir.
- getProjectName: Projenin adını getirir.
- getProjectRel: Kardeş projenin kimlik numarasını getirir.
- getProjectSuccessMessage: Formlar doldurulduktan sonra ekrana basılacak mesajı getirir.
- getUserIdIfExists: Mevcut ise kullanıcının kimlik numarasını getirir.

2.2.4.3. Dağıtık Küme Yapısı

Ancak farklı türlerdeki içeriğin tek bir sütunda aynı tür altında tutulması normalizasyon kurallarına dahi uymayacağından, bu ham veriler dinamik olarak türlerine göre farklı tablolara dağıtılır. Hangi tabloya dağıtılacağı, hangi tablodan çekileceği; o verinin ‘properties’ tablosunda belirtilen türünün ‘types’ tablosundaki tür karşılığı üzerinden ‘out_cluster’ hedef değeri çekilerek ulaşılır. Neticede, ‘int’ değerler hususi bir tabloda, ‘text’

değerler farklı bir tabloda, farklı uzunluklardaki ‘varchar’ değerler de başka tablolarda tutulur. Bu yapı performans ve kapasite kullanımında son derece yüksek düzeyde verim sağladığı gibi Büyük Veri konseptine önemli bir adım atar.

Şekil 12: HARPA Veritabanında Farklı Türlerdeki Ham Verilerin Farklı Kümelerde Tutulması



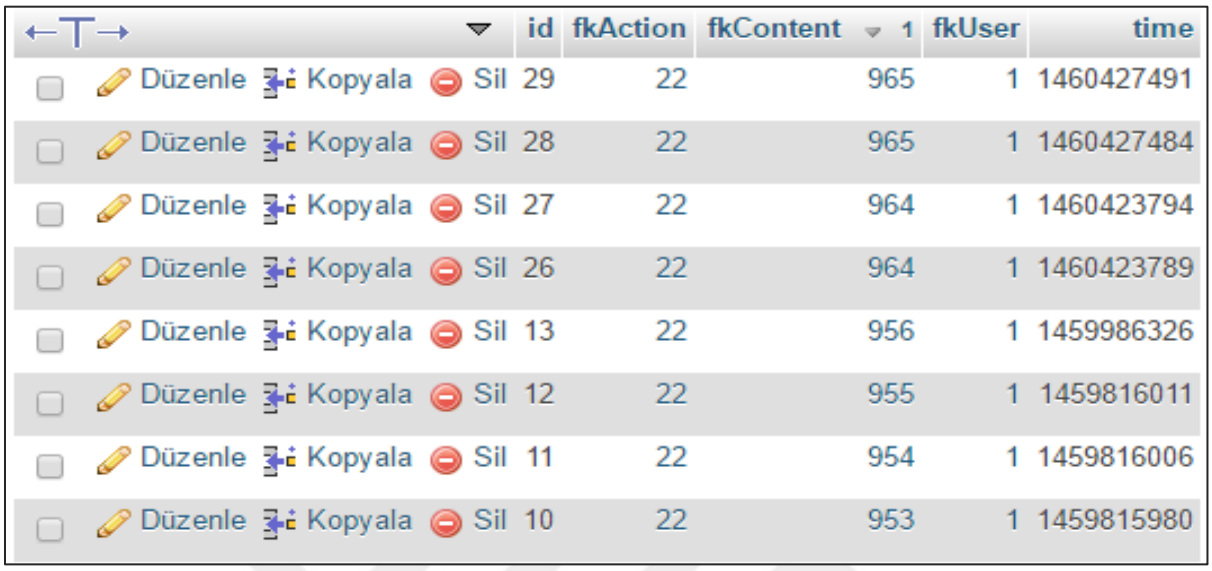
Kaynak: Yazar tarafından derlenmiştir.

Oluşturulan web sitesindeki tüm hareketler log tablosunda tutulur. Tüm kullanıcı girdileri dâhil tüm bilgi parçaları ayrı kayıtlar halinde ve hangi aksiyona tabi olarak ortaya çıktığı yine bu kayıt atama altında tutulur. Bu bilgiler aynı zamanda kullanıcı girdisi raporlamada da kullanılır. Time isimli integer türündeki alan, linux datetime türünde anlık zaman değerini tutar, bu alan türü date-time a göre bellekte daha az yer kaplar.

İster yönetici tarafından geliştirilen web sitesi unsurları ve statik içerikleri olsun, ister oluşturulan formlar aracılığı ile ziyaretçinin girdiği kullanıcı verileri olsun; HARPA İYS modelinde tüm veriler aynı veri havuzuna aittir. Her veri, bağlı olduğu kontrolör türünün ilişkili olduğu kümede tutulur. Form elemanlarına yapılan kullanıcı girdileri de aynı şekilde hangi form elemanından üretildiyse, o kontrolörün bağlı olduğu küme tablosunda fiziksel olurlar tutulurlar. Ancak her içerik hiyerarşik olarak birbirinin altında bağlı durumdadır ve ‘restriction’ tablosunda bu ast-üst ilişkisi olarak tanımlıdır. Örneğin, bir bilgi aktarım formu hiyerarşik olarak bir sayfa içeriğinin altında ise, bir metin alanı bu bilgi aktarım formu içeriğinin altında tanımlıdır. Buraya kadarki ifadeler, HTML işaretleme dili ile benzerlik gösterir. Ancak HARPA İYS modelinin gereği olarak; herhangi bir yetkili kullanıcının söz gelimi bu yazı alanına girdiği formal veriler de, bu metin alanının hiyerarşik olarak altına eklenir. Ancak kullanıcılar tarafından doldurulan bu formların kullanıcı verileri sayfalama kısımlarında görünmez ve sadece birleştirilmiş raporlama kısmında yetkili kullanıcı tarafından

görüntülenebilir. Arka planda yani veritabanında tüm veriler bir havuzda hiyerarşik olarak harmanlandığı için model de ismini bu mantıktan alır.

Şekil 13: HARPA Veritabanında Oluşan Form Aksiyonlarının Kayıt Dökümü Halinde Tutulması



	id	fkAction	fkContent 1	fkUser	time
Düzenle Kopyala Sil	29	22	965	1	1460427491
Düzenle Kopyala Sil	28	22	965	1	1460427484
Düzenle Kopyala Sil	27	22	964	1	1460423794
Düzenle Kopyala Sil	26	22	964	1	1460423789
Düzenle Kopyala Sil	13	22	956	1	1459986326
Düzenle Kopyala Sil	12	22	955	1	1459816011
Düzenle Kopyala Sil	11	22	954	1	1459816006
Düzenle Kopyala Sil	10	22	953	1	1459815980

Kaynak: Yazar tarafından derlenmiştir.

İçerik tabanlı hareket gereksinimi, piyasadaki popüler platformlarda kullanıcıların hangi sayfaları incelediği bilgisine gerektiği zaman 'Hareket Dökümü' benzeri isimler altından ulaşabildiği gerçeğinden anlaşılabilir. Gelecekte web sitelerinin, kullanıcıların fare işaretçi hareketlerini/pozisyonlarını bile saniyesi saniyesine kayıt altında tutacağı ve sayfa içi istatistiklerde kullanılacağı tahminler arasındadır.

2.2.4.4. Veri Atama Kolaylığı

Modele bağlı kalınarak oluşturulan bir web sitesindeki her türlü içerik, sayfa, form, hatta kullanıcı girdileri bir başka bir içeriğin değer ya da içerik materyali olarak kullanılabilir. Bu işlem tek bir ID bilgisi bağlayarak yapılabildiği gibi tüm alt hiyerarşik elemanlar da bu kurala tabi olur. Ayrıca 'Veri Köprüsü' adında tanımlanmış tür; liste vb. kontrolörlere değer atamada kullanılır.

Bu sayede BLOB gibi büyük boyutlu içerikler ve gömülü dosyalar dâhil her türlü veri sadece bir defa tanımlanmış olduğundan performansın yanı sıra veritabanının işgal ettiği disk alanı azalmış olur.

Ancak bu merkezi veri atama sistemi modelin bir getirisidir ve uygulama tabanında işleve uygun eklentiler hazırlanması gerekir. HARPA CMS yazılımında varsayılan olarak bu özelliği kullanan fazla sayıda bileşen kontrolör bulunmamaktadır.

2.2.4.5. Alternatif Kontrolör Desteği

Modül ya da eklenti desteği alternatifi olarak bu konseptin kullanımı, uygulama geliştikçe standart eklentilere göre çok daha esnek kullanıcı kontrolü ve yetki tabanlı içerik kullanımı sağlayacaktır. Tüm kontrolör bileşenler iç içe geçerek sayfa çıktısını oluşturur.

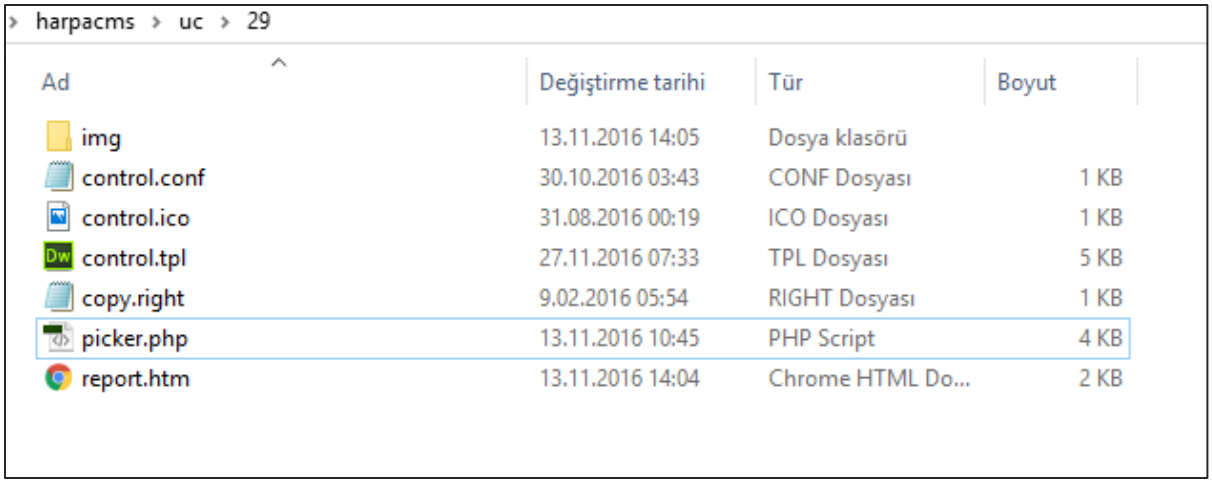
Nitelikleri;

- Tekil aksiyon yüklenebilir.
- Birbirlerinin hiyerarşik olarak altına yerleşebilir.
- Geliştirilebilir/Paylaşılabilir.
- Tüm diğer kontrolörler ile aynı bilgi kümesini paylaşabilir.
- Sayfalara Sürükle/Bırak mekanizmasıyla kolay yerleştirme.

Microsoft .net platformunda kullanıldığı gibi, HTML kontrolleri tekrardan tanımlanmış ve bunlar için toplam otuz kadar kontrolör dosyası oluşturulmuştur. Bunlardan bazıları çağın ihtiyacı olan fakat HTML işaretleme dilinde bulunmayan, JQUERY (JavaScript tabanlı bir yazılım kütüphanesi) destekli hibrit ve karmaşık kontrolörlerdir. Bunlar sayfa içeriğine basit bir yazı ekler gibi, dinamik tablo/liste ekleme mantığı ile çalışır. Bu kontrolörlere değerler, içeriklerin ID bilgisi ile kolayca bağlanır.

Harpa İçerik Yönetim Sistemi'nde kontrolörler ve bağımlı dosyalar, isimlendirilmiş fiziksel klasörlerde tutulurken, her bir bileşen platformun farklı katmanlarında farklı dosyalar tarafından temsil edilirler. Söz gelimi bir yazı alanı bileşeni, varsayılan içeriği yönetici tarafından tanımlanırken kontrol panelinde farklı bir tema ile ekrana yansıtılır. Aynı bileşen üretilen web sitesi çıktısında farklı bir tema ile sayfaya basılır ve kullanıcı bu yazı alanının varsayılan metnini kabul edebilir ya da kendisi değiştirebilir. Neticede önyüzdeki formun onaylanması ile işleme tabi tutulan bu yazı alanının içeriği, veritabanı tarafında kayda geçer. Bu kayıtlı form verisi yetkili kullanıcıya rapor kısmında farklı bir tema ile görüntülenir. Farklı bir örnek ile koordinatı değiştirilebilir bir harita uygulaması kullanıcı arayüzünde kullanıcının adres koordinatını tanımlamasına izin verebilir. Ancak bu veri, rapor kısmında değiştirilmemelidir ve sadece harita üzerinde işaretli olarak gösterilmelidir.

Şekil 14: HARPA İYS Kontrolör Bileşeni Örnek Klasör Ve Dosya Yapısı



Kaynak: Yazar tarafından derlenmiştir.

Şekil 14'te bir kontrolör bileşen klasöründe bulunan tema ve ayar dosyaları gösterilmektedir. Bu dosyaların çoğu bir HARPA bileşeninin barındırması zorunlu unsurlar olsa da, bazı dosyalar opsiyoneldir.

'control.tpl' dosyası oluşturulmuş önyüz sayfalarında, o kontrolörü ve taşıdığı içeriği temsil ederken, kendine özgü basit yapısı ile geliştirilebilir ve farklı dil destekleri de taşır niteliktedir. Kullanıcı arayüzü temasıdır.

'picker.php' dosyası, kontrolör oluşturulurken, varsayılan değer atanacak ise ve bu değer katar değerden farklı değerler olabilecek ise, yöneticiye varsayılan içerik girişinde yardımcı öğeler içerir. Örneğin, seçim listesi kontrolörü önyüzde kullanıcıya bir liste sunup seçeneği girdi yapma imkânı sağlarken, bu kontrolör sayfaya yerleştirilirken içerdiği listenin hangi veri setinde tanımlı olduğunu yönetici işaretlemek durumundadır. Yönetici arayüzü temasıdır.

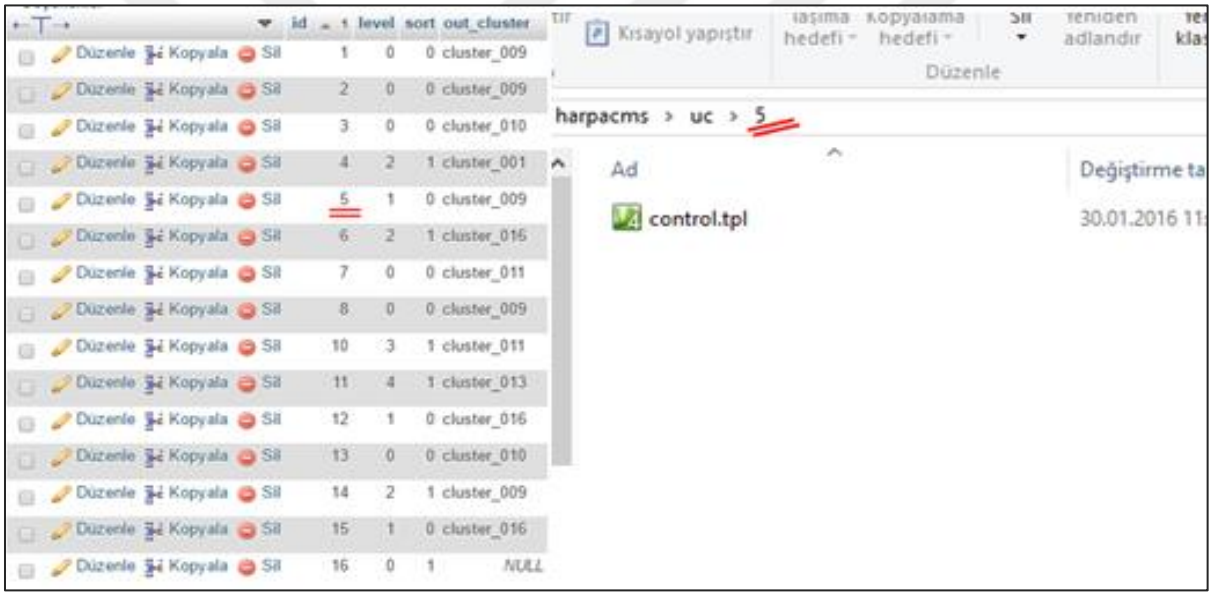
'report.htm' dosyası ise rapor arayüzü temasıdır ve kullanıcılara o içeriğin şablonunu raporlama kısımda temsil amacı taşır. Örneğin bir içerik yazı alanı üzerinden girdi yapılıp kaydedilmişse de rapor kısmında metin olarak gösterilmesi gerekir. Ya da bir dosya, forma çoklu dosya yükleme kontrolörü üzerinden yüklenmişse de, form girdilerini inceleyen bir kullanıcı rapor kısmında bu dosyaya link olarak ulaşmak isteyecektir.

Söz konusu sebepler ile bir içerik web sitesinin her katmanında farklı dosyalarla şablon haline getirilmek durumundadır. Dizindeki diğer dosyalar ve klasörler ise, çalışan bileşenin ihtiyacı olan Javascript ve CSS dosyaları ya da imaj dosyalarından oluşur.

Her modülün sistem tür dosyaları; Şekil 15'te görüldüğü üzere türün anahtar ID numarası ile adlandırılmış klasör içinde farklı uzantılar ile barındırılır, bu dosyalar gerektiği zaman yazılım içine anlık olarak dâhil edilir ve çalıştırılır.

Aynı şekilde, formal işlemler için gerekli olan aksiyon dosyaları da aynı tabloda tür bilgisi olarak tanımlanmıştır ve ID bilgisi ile adlandırılmış klasörden çekilerek PHP uygulaması tarafında çalıştırılır. Örneğin 'Yeni Veri Ekle' işlemini gerçekleştirerek kullanıcı girdisini veritabanına yeni kayıt olarak ekleyen tanımlı aksiyon 'harpacms/actions/22/' altında bulunan 'action.php' dosyasıdır. Bu dosyanın yapısı tipik obje tabanlı bir PHP dosyasıdır. Aynı dizinde aksiyon dosyasına özgü ayarları barındıran, 'app.conf' dosyası da bulunur.

Şekil 15: HARPA İYS Aksiyonları Klasör Dizini



Kaynak: Yazar tarafından derlenmiştir.

Her eklentinin (kontrolör bileşen) toplamda üç farklı katman teması bulunur ve bir HARPA kontrolör eklentisinin sağlıklı çalışabilmesi için bunlardan sadece kullanıcı arayüzü temasını barındırması yeterlidir.

Control.tpl önyüz şablon dosyaları standart HTML kodlama unsurları barındırır. Bağımlı olduğu türdeki her içerik ekrana basılacağı zaman bu dosya çalıştırılır. İçerisindeki Server Side Includes (SSI) dili sözdizimine benzer özel işaretler, çeşitli amaçlar ile harpacms script kodu tarafından her defasında yorumlanır ve şablonun veri girişleri niteliğindedir.

Şekil 16: HARPA İYS Kontrolör Dosyası Control.Tpl Örnek Ekran Görüntüsü

```
<span <!-- templateData[elementhidden] -->>
      » <u><!-- templateData[elementalias] --></u><br>
      <div id="el<!-- templateData[elementid] -->">
      <!-- templateData[startloop] -->
      <label><input class="chkbox" name="el<!-- templateData[subid] -->" id="el<!-- templateData[subid] -->" type="checkbox" value="
<!-- templateData[subid] -->"  /><!-- templateData[subalias] --></label>
      <!-- templateData[endloop] -->
      </div>
      </span>
```

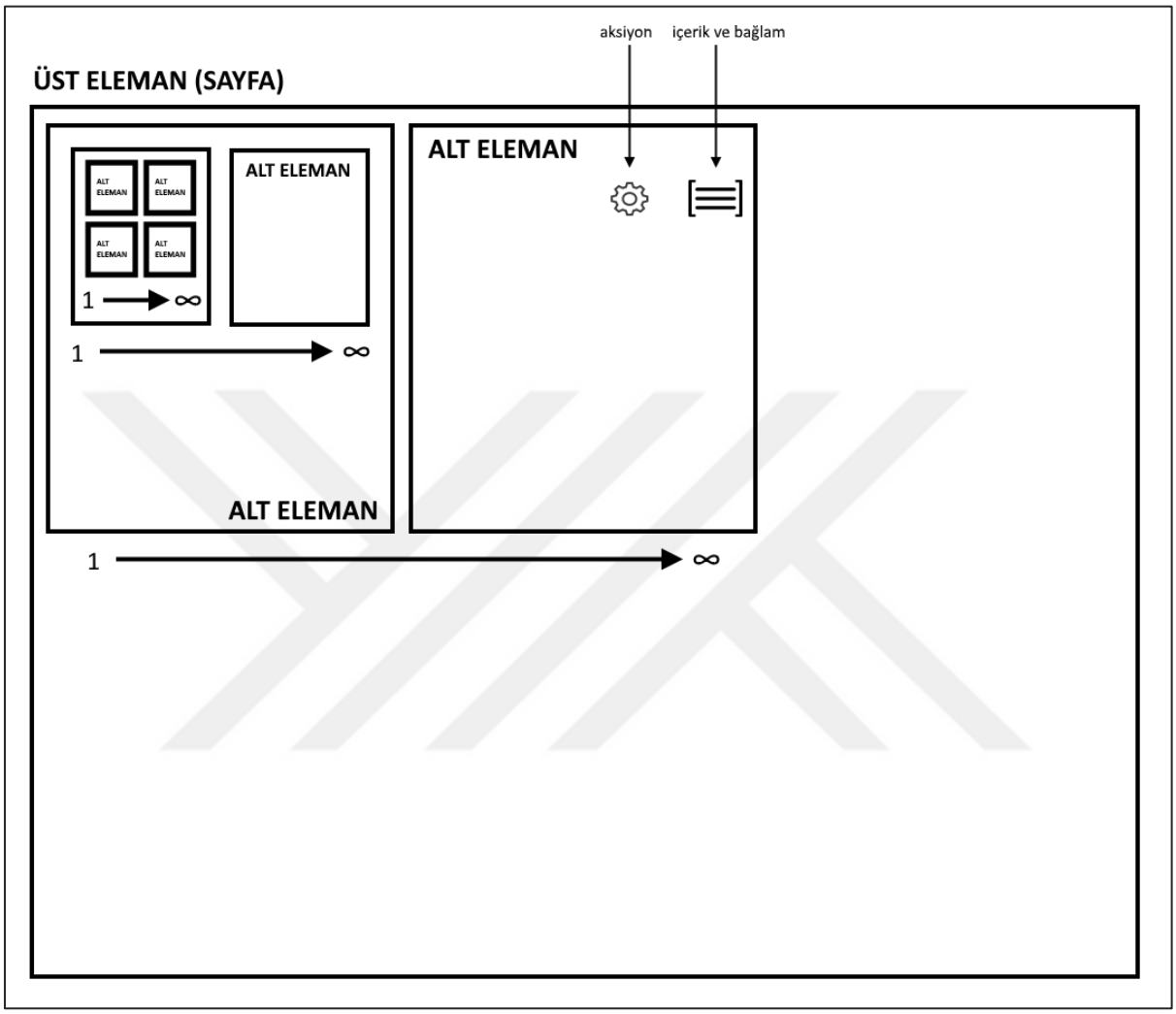
Kaynak: Yazar tarafından derlenmiştir.

Örneğin, birden fazla alt veri barındıran bir seçim kutusu (select box) kontrolör önyüz şablon dosyası yorumlanırken, '<!-- templateData[elementhidden] -->' kod parçası görüldüğü yere, eğer o eleman gizli ise bunu belirten ifade yerleştirilir. '<!-- templateData[elementalias] -->' ifadesi o ana içeriğin adı olan katar veri için girdi yeridir. '<!-- templateData[elementcontext] -->' tanımlaması ana verinin içeriğini, '<!-- templateData[elementid] -->' ise ana verinin ID numarasını ifade eder. '<!-- templateData[startloop] -->' ve '<!-- templateData[endloop] -->' ifadeleri, liste seçenekleri gibi alt verileri olan içerikler için her bir alt veriye uyarlanacak olan kod parçaçığının başlangıç ve bitiş kısımlarını ifade eder. '<!-- templateData[subalias] -->' alt verinin adını, '<!-- templateData[subid] -->' alt verinin sistemdeki benzersiz 'içerik ID' numarasını, <!-- templateData[subcontext] --> ise alt verinin kendisini temsil eder. Bazı kontrolörler, ek parametreler de gerektirir. Bunlar yöneticinin içeriği oluştururken girdiği formal parametreler ile doğru orantılı olarak artar. Örneğin, bir metin alan bileşeninde, kullanıcının en az azgari yazması gereken karakter sayısı ve maksimum yazabileceği karakter sayıları sırasıyla '<!-- templateData[elementminlength] -->' ve '<!-- templateData[elementmaxlength] -->' ile ifade edilir. Tüm bu özel tanımlamalar, sunucu taraflı script kodlarındaki döngü parçaları ile aynı mantıkta çalışır, ancak çok yalın ve geliştirilebilir yapıdadır. '<!-- templateData[elementclear] -->' ifadesi, tek tık ile değeri silinebilen kontrolör bileşenlerinde kullanılır ve mobil kullanımı destek amacıyla tutulan formal verinin temsilidir.

HARPA kontrolör bileşenleri, tarayıcı yüklenme zamanında birbirinin içerisine geçerek ana içeriği oluştururlar. Şekil 17'de görüldüğü gibi, bu yapı standart HTML işaretleme dilinde elemanların iç içe geçerek form yapılarını oluşturması ile oldukça benzerlik gösterir. Kontrol paneli üzerinden sürükle-bırak fonksiyonlar ile oluşturulan her bir nesne, kendi içerik ve bağlam parametrelerini aldığı gibi, eğer içeriyorsa bir alt nesnenin de oluşmuş çıktısını içine alır. Eğer formal bir sayfadan söz ediyorsak, söz konusu alt nesneler kullanıcı

tarafından doldurulduktan ve form onaylandıktan sonra bu nesnelerin (form elemanlarının) kullanıcı tarafından alınan tüm değerleri, o kontrolörlere atanan aksiyon ile işleme tutulur.

Şekil 17: HARPA İYS Kontrolör Dosyalarının Yükleme Zamanında Birbirlerini Kapsaması

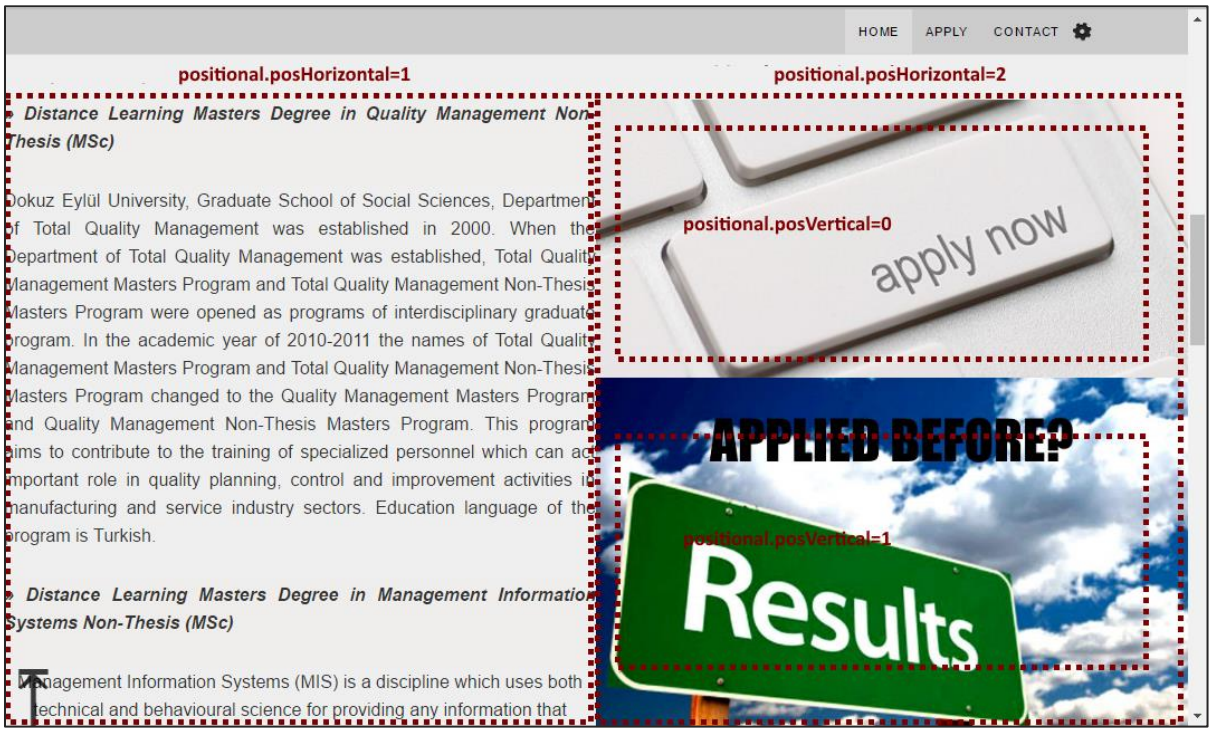


Kaynak: Yazar tarafından derlenmiştir.

Şekil 18'de, 'Parallax Page' isimli tema kontrolörü üzerinde oluşturulan örnek sayfa yapısı görünmektedir. Dizayn sayfası sürükle bırak editörü üzerinden tasarlanan bu yapı, projedeki üç iş akışından birisi (APPLY isimli iş akışı) tarafından temsil edilir. Bu sayfa editörden tasarlanırken, iki bölümlü sayfa (Double Column) olarak tanımlanmıştır. Bir pencerede üç farklı kayar sayfa bulundurulacağından ve birleşik formda kullanılacağından; başvuru sayfasında başlık (banner) ve künye (footer) kısımları boş bırakılmıştır. Sol sütun içerisine düşecek içerik olan 'HTML İçerik' kontrolörü sayfa üzerinden sürüklenerek sol sütuna yerleştirilse de, arka planda bu içerik sayfanın hiyerarşik olarak bir alt bileşenidir. Kullanıcı önyüzünde içeriğin sol kısma düşürülme işlemi, paralaks (parallax) sayfa kontrolörü

dosyalarında konumlandırma amaçlı programlanmış bazı dinamik JQUERY metotları ile gerçekleştirilir. Ancak, içeriğin sol kısma düşmesini sağlayan veritabanı tanımlaması, 'positional' isimli tablodaki 'posHorizontal' isimli alan bilgisidir. Bu tanımlama sayesinde, formal sayfalarda olduğunun aksine hiyerarşik olmayan bir yapı ile dizayn sayfalarında sütun tabanlı konumlandırma yapılır.

Şekil 18: HARPA İYS Dizayn Sayfalarında İçerik Görüntülenme Yapısı



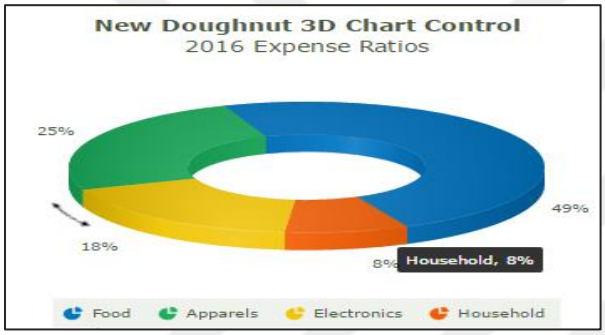
Kaynak: Yazar tarafından derlenmiştir.

Dizayn sayfalarında sütun temsili numaraları 0 ile 5 arasında değişir. Bu durum yönetim paneli tarafında getirilen işlevsel kısıtlama ile, bir dizayn sayfasında maksimum dört dikey sütun olabileceğindendir ancak model açısından bir kısıtlama söz konusu değildir. 0 numaralı sütun her zaman en üstte bulunan başlık kısmıdır ve en son sütun ise künye yani web sayfasının en alt kısmında bulunan bölümdür. Örneğin, iki kolonlu bir web sayfasında 0 numaralı kolon başlık, 1 numaralı kolon sayfanın sol sütunu, 2 numaralı kolon sayfanın sağ sütunu, 3 numaralı kolon ise en alttaki künye kısmıdır. 'positional' isimli tablodaki 'posHorizontal' alanına ilgili içerik için hangi numara atanırsa içerik o bölmede ekrana basılır.

Aynı sütun içinde bulunan içeriklerin dikey sıralama bilgisi, yine ‘positional’ isimli tablodaki ‘posVertical’ isimli alanda tutulur. Bu sıralama bilgileri 0 ile sonsuz arasında değişir ve 0 numaralı dikey sıra bilgisine sahip içerik her zaman sütunun en üstünde ekrana basılır.

Şekil 19’da ki çember kontrolörü, durağan ve formal sayfalarda üç boyutlu çıktı sağlayan gelişmiş bir çizelge bileşenidir. Kontrol panelinden tanımlanmış ikili veri seti üzerinden çektiği verilerdeki sayısal yüzlük değerleri, halka üzerinde bölmeler yaratma amacıyla kullanır. Üretilen çizelge fare işaretçisi yardımıyla kendi ekseni etrafında döndürülebilir. İşaretçi bölmelerin üzerine getirildiği takdirde o dilimin ismi ve yüzde değeri etiket içerisinde görünür. Yıllık istatistikler ve oranları gösterme amacıyla kullanımı uygundur.

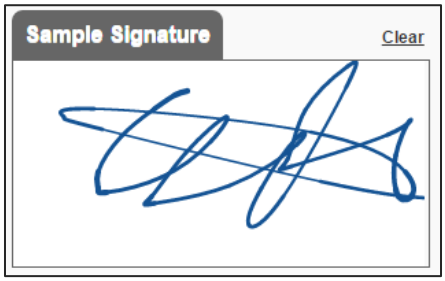
Şekil 19: HARPA İYS Tarafından Üretilen 3 Boyutlu Çember Tablo Bileşeni



Kaynak: Yazar tarafından derlenmiştir.

Şekil 20’de ki kontrolör ekran görüntüsünde, form sayfasındaki bir imza alanına kullanıcı tarafından atılmış örnek bir imza görünmektedir. Bu imza alanı, kullanıcıların form alanını doldurduktan sonra temsili imza örneklerini sisteme yükleyebilmeleri için olanak sağlamaktadır. Gezici ekiplerce doldurulan teslimat formlarında da müşterinin imzası bu yolla sisteme yüklenebilir.

Şekil 20: HARPA İYS Tarafından Üretilen Formal Dijital İmza Kontrolörü



Kaynak: Yazar tarafından derlenmiştir.

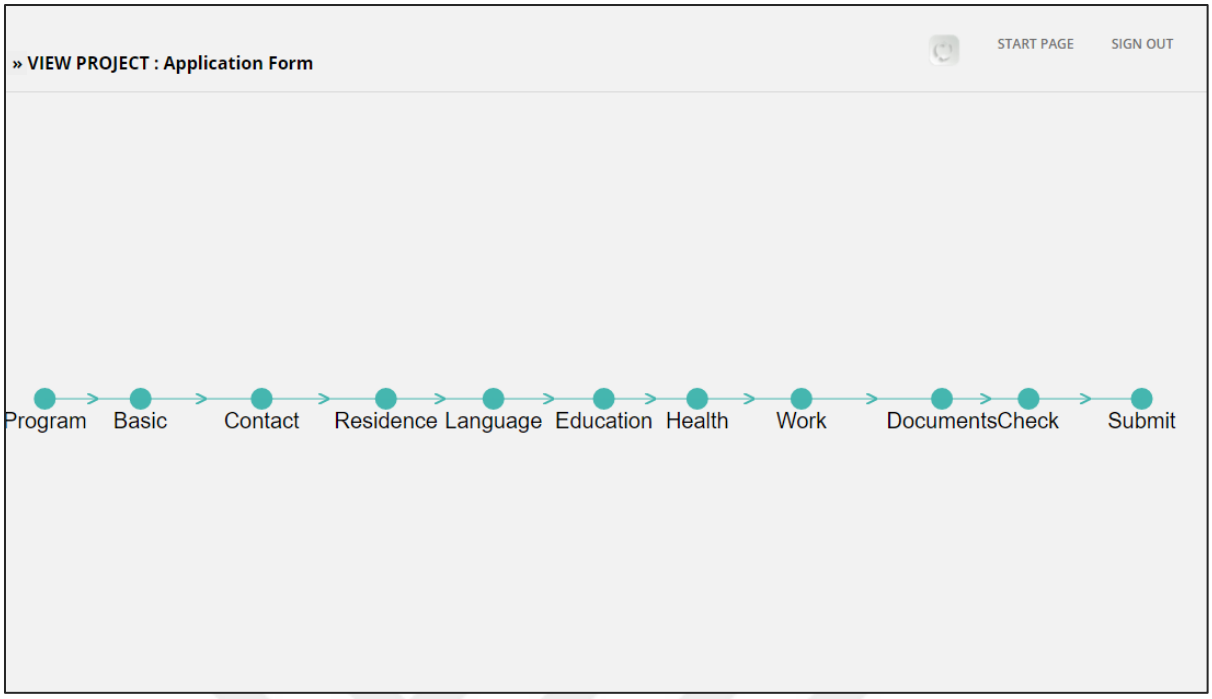
İmza kontrolörü, bilgisayar faresi ile imza atılabilmesine imkân sağlarken mobil desteği sayesinde tablet ve cep telefonu dokunmatik ekranı üzerinden de aynı işlevi sağlar. Teknik açıdan; kullanıcı imzasını attıktan hemen sonra, uygulama imza örneğine yuvarlama efekti vererek imzayı görsel anlamda daha da estetik hale getirir. Ancak sayfa 'POST' edildikten sonra sisteme kullanıcı imzasının orijinal ve efektsiz olan şekli yüklenir.

2.2.4.6. İçerik Yönetiminde İş Akış Desteği

İş akışı, bir iş sürecindeki doküman, bilgi ya da görevlerin bir katılımcıdan diğer katılımcıya tanımlı prosedür tabanlı eylemler neticesinde aktarılması işleminin, kısmen ya da tamamen otomatikleştirilmesidir (Avenoğlu, 2014). Bu tanıma göre, İçerik Yönetim Sistemindeki her türlü veri, bir kullanıcıdan diğer kullanıcıya belli kurallar ile aktarılabilmelidir ve bu işlemler iş akışına adapte edilebilmelidir. Ayrıca önceki bölümlerde de bahsedildiği gibi bazı kaynaklar, iş akışını içerik yönetiminin adımlarından biri olarak ortaya koymuştur (CMS Review, 2006:1). HARPA İçerik Yönetim Sistemi, bu bulgulara dayanarak, statik olarak tanımlanmış ya da formal işlemler ile kullanıcıdan alınan verilerin proje tabanlı iş akışında yürütülebileceğini varsayımını ispatlar niteliktedir. Bu sebeple, başvuru formları gibi veri aktaran dinamik sayfaların yanı sıra, oluşturulan statik dizayn sayfaları da art arda eklenerek basit iş akışı oluşturulup projeye dâhil edilebilmektedir.

Formal işlem yapmayan statik sayfalarda, kullanıcının sayfayı tamamen görmesi bir iş parçacığı olayını yaratır. Dâhili paralaks nitelikli tema sayesinde, aynı projeye eklenmiş iş parçacıkları yani statik sayfalar, mobil destekli ve tek tıklama ile kayar mekanizma ile dikey olarak kayarak, kullanıcıya hızlı bir sayfa gezinti deneyimi yaşatır. Bir iş akışında onlarca sayfa bulunabileceğinden, bu sayfalar kullanıcı tarafından aşağı doğru kaydırıldıkça bir sonraki sayfa dinamik olarak 'Ajax' ile yüklenir ve yüklemelerde performans kaybı yaşanmaz. Buradaki paralaks yapı, projede bulunan tasarım sayfalarını art arda ekleyerek tek bir projeksiyon haline getirir. Kullanıcı sayfayı fare tekerleği yolu ile aşağı doğru kaydırdıkça, ekran bir sonraki sayfaya geçiş yapar. Aynı zamanda gerçek zamanlı olarak üst kısım menüsünde de mevcut konum işaretlenir. Tüm bu işlevler mobil desteklidir ve düşük çözünürlüklü ekranlara uyumlu olduğundan tasarım ekran boyutuna göre dinamik olarak değişebilir.

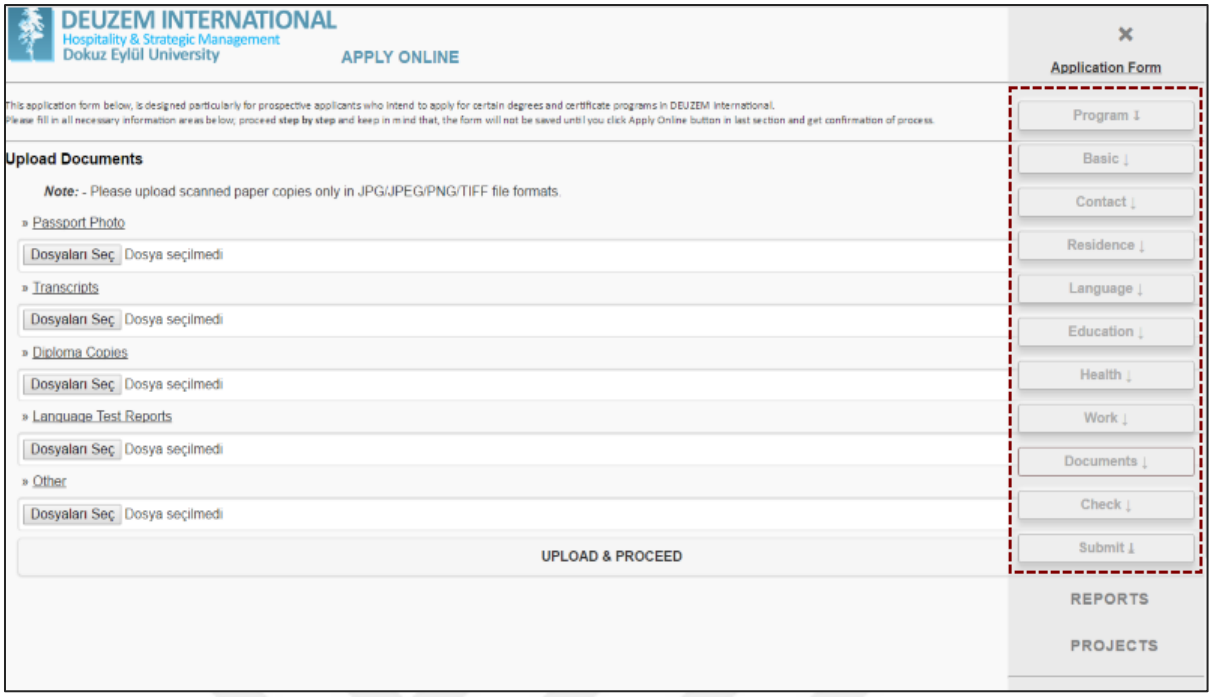
Şekil 21: HARPA İYS Kontrol Paneli Üzerinde Proje Sayfalarının İş Akışı Görünümü



Kaynak: Yazar tarafından derlenmiştir.

Şekil 21'de ki ekran görüntüsünde, HARPA CMS kontrol paneli altında bulunan proje kısmında alan uygulamasının başvuru formunu oluşturan iş akışı adımları görünmektedir. Buradaki iş akışlarının her birisi sistemde tasarlanmış bir sayfayı temsil eder. İlgili proje 'Application Form', kullanıcı tarafında ilk erişim sağlandığında 'Program' adındaki ilk iş akışını yükler. İlk temsil edilen sayfa eğer dizayn sayfası yani durağan tasarım sayfası ise diğer tüm akışlar kayar sayfa şeklinde kullanıcının ekranında alt alta eklenir. Eğer ilk sayfa bilgi taşıyan bir form sayfası ise bu sayfadaki alanlar doldurulup gönderilmeden bir sonraki iş akışının temsil ettiği sayfaya geçilmez. Bu yapı altında hem durağan sayfalar hem de form sayfaları basit iş akışı yapısına adapte edilmiş olur.

Şekil 22: HARPA İYS Tarafından Üretilen Formal Sayfalardaki İş Akışı Sağ Kısım Menüsü Ekran Görüntüsü



Kaynak: Yazar tarafından derlenmiştir.

Şekil 22'de ki ekran görüntüsünde, HARPA CMS üzerinden üretilen formal sayfalarda, kullanıcının akışlar arasında gezinebilmesi ve yetkisi olan raporlara ulaşabilmesi amacıyla yerleştirilmiş sağ kısım menü bulunmaktadır. Şekil 22'de ki işaretlenmiş kısımda ise mevcut projenin her iş adımı yukarıdan aşağı doğru sırasıyla görüntülenmektedir. Kullanıcı hangi bağlantıya tıklarsa ilgili iş akışının bağlı olduğu sayfaya yönlendirilir. İlgili bağlantı butonlarının üzerinde, sayfanın değil iş akışının 'alias' bilgisi yani ismi görüntülenmektedir. Bu isimlerin yanında, ilgili iş akışının ilk adım, son adım ya da ara adımlardan birisi olduğunu simgeleyen kesik ok işaretleri bulunur. Grubun altındaki bölmede ise ilgili projenin raporlarına erişim, yetkili diğer projelerin listesine erişim, ana sayfaya erişim, kullanıcı yetkili ise kontrol paneline erişim ve kullanıcı girişi yapılmışsa kullanıcının sistemden çıkması işlemine erişim bağlantıları bulunur.

Bu açılır menü paneli, sayfadaki mobil üç çizgi ikonuna tıklandığında açılır ve çarpı işaretine tıklandığı zaman yavaşça sağ tarafa doğru yavaşça kapanır; gereksiz yere ekranda yer kaplamadığından oldukça kullanışlı bir yapıdadır.

Şekil 23: HARPA İYS Kullanıcının Yetkili Olduğu Projeler Listesi



Kaynak: Yazar tarafından derlenmiştir.

HARPA İçerik Yönetim Sistemi platformunda, herhangi bir kullanıcının ya da ziyaretçinin projedeki en az bir iş akışına sayfa görüntüleme ya da rapor (form girdilerini okuma) izni var ise, o kullanıcının o projeye izni olduğu varsayılır. Şekil 23'te ki ekran görüntüsünde kullanıcının yetkisi dâhilindeki tüm projeleri basit olarak listeleyen 'listprojects.php' sayfası görünmektedir. Tablodaki her satırın ilk kolonunda projenin adı ve ilk iş akışına link bağlantısı, ikinci kolonunda o projenin içerisinde o kullanıcının yetkisine bakılmaksızın bulunan tüm iş akışlarının toplam sayısı, üçüncü kolonda ise o proje altındaki her iş akışına yapılan form girdilerinin kullanıcı referanslı raporlanması için link bağlantısı bulunmaktadır.

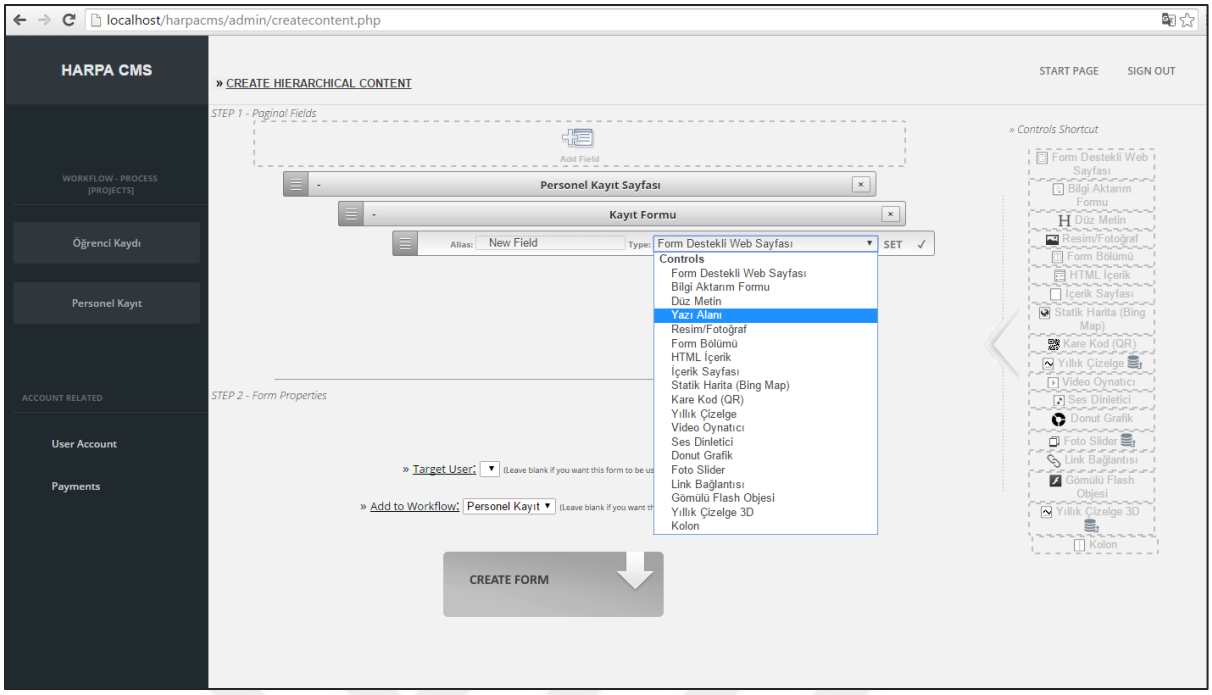
ÜÇÜNCÜ BÖLÜM
UYGULAMA

3.1. GELİŞTİRİLEN UYGULAMA PLATFORMU

Mobil destekli ve kullanıcı dostu arayüze sahip uygulama, kolay kullanımı nedeniyle günümüz popüler sistemler ile yarışabilecek nitelikte çıktı verir. Modelin hiyerarşik veri saklama yapısı sebebiyle, kontrol paneli veri tasarlama ortamı da hiyerarşik yapıya göre şekillenmiştir. Geliştirme ekranlarının sağ kısmındaki kısayol butonları sayesinde form destekli sayfa ve alt bileşenleri tek tıkla tasarlama ortamına eklenebilir.

Şekil 24'te ki ekran görüntüsünde, üç tasarlama aracından birincisi olan hiyerarşik form sayfası tasarlama aracında örnek bir personel kayıt sayfası tasarım aşaması görünmektedir. Sağ kısımdaki menüden tek fare tıklaması ile çalışma alanına eklenen kontrolör bileşenler çubuk formunda görünür. Her bir kontrolör taşıdığı değere göre içerik parçasını temsil eder. Kontrolör çubuğunun sol tarafında bulunan koyu renkteki sürükleme kutucuğu yardımıyla, ilgili içerik başka içeriklerin altına hiyerarşik olarak taşınabilir. Fare işaretçisi ile tıklayıp sürüklemek ve konumlandırmak yeterlidir. Çubuğun üzerindeki etikete tıklandığında, o içeriğin ismi yeniden adlandırılır, ayrıca içeriğin türü anlık olarka yanındaki listeden değiştirilebilir. Çubuğun en sağ kısmında kalan tik işaretine tıklandığında yapılan ayar uygulanmış olur. Yine çubuğun üzerinde bulunan 'SET' butonuna tıklandığı zaman, o içeriğin değeri ve formal ayarları yer alır. Bu formal ayarlar, içerik kontrolörünün değerinin en az ve en çok taşıyabileceği karakter sayısı, alanı hızlı silme butonu, alanı gizleme, yazı alanı maskeleme ve alanın form içerisinde doldurulduktan sonra tetikleyeceği tanımlı aksiyon ayarlarıdır. Tüm bu parametreler girildikten sonra içerik form sayfasında hiyerarşik olarak yada dikey sıralama bazında konumlandırılır. Bazı kontrolörler hiyerarşik olarak iç içe konumlandırılması gerekebilir. Örneğin 'Data Form' yani veri aktaran form eklentisi, tüm veri aktaran yazı alanı, seçim kutusu gibi diğer alanları içinde tutmalıdır. Zira HTML işaretleme dilinde de yazı alanı ve seçim kutusu gibi veri taşıyan elemanlar 'form' tagları içerisinde yer almalıdır. HARPA form tasarım aracındaki yapı da bu mantıkta çalışır. Hiyerarşik olarak aynı hizada yer alan tüm bileşenler ise araçtaki yerleştirilme sırasına göre ön yüz sayfalarında ekrana basılır.

Şekil 24: HARPA İYS Kontrol Paneli Üzerinden Mobil Destekli Form Yaratma İşlemi



Kaynak: Yazar tarafından derlenmiştir.

Form elemanları yerleştirildikten ve form adı yazıldıktan sonra ‘Create Form’ butonuna tıklanır ve form sayfası üretilmiş olur. Ancak bu sayfanın aktif olarak yayımlanabilmesi için proje dizayn aracı içerisinden yaratılan bir projenin alt basamağı olarak tanımlanmalıdır ve sıraya konmalıdır.

Şekil 24’te ki ekran görüntüsünde, üç tasarım aracından ikincisi olan dizayn sayfası tasarlama aracı görünmektedir. Dizayn sayfaları bu raporda statik olarak betimlenmiştir, ancak ifade edilen özellik dizayn sayfalarının başka sayfalara veri aktaramıyor oluşudur. Dizayn sayfaları da kendi içerisindeki kontrolörler üzerinden veritabanı tarafı ile bağlantı kurar ve içerik taşırlar.

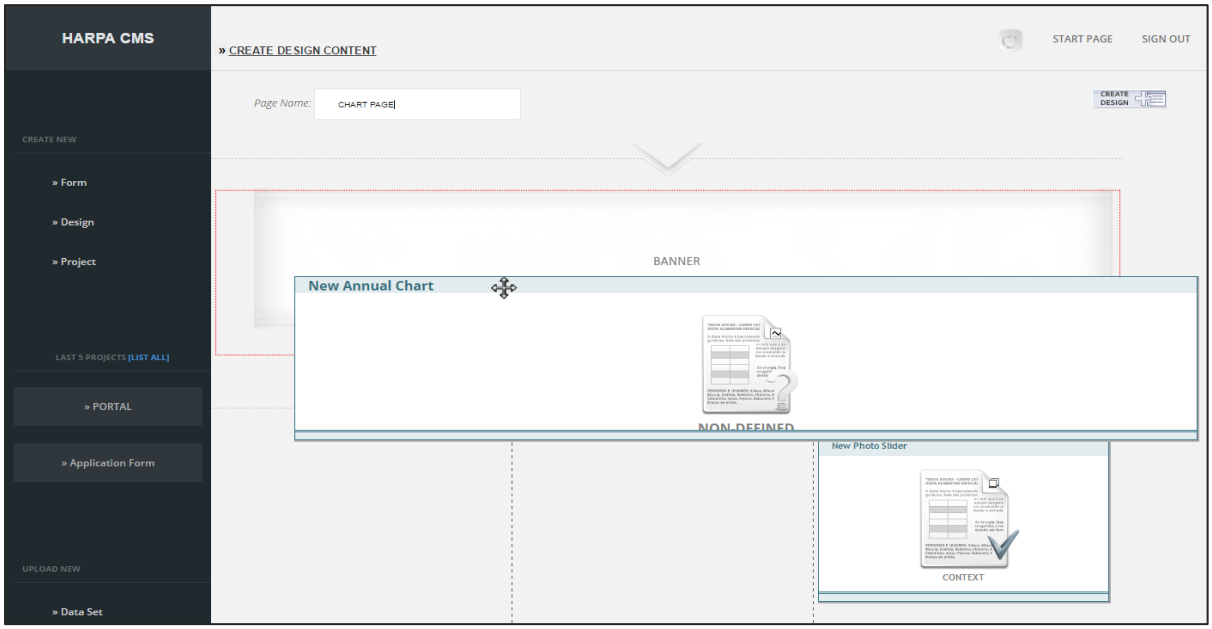
Bu araca kontrol panelinin sol kısım menüsündeki ‘Design’ menü bağlantısına tıkladıktan sonra açılan ek menüden sayfa içerik kısmının kaç parçaya bölüneceği seçeneğine tıklandıktan sonra erişilir. Üç bölümlü bir dizayn sayfasında başlık kısmı üç kolon ve künye kısmı bulunur. Bu kısımlardan herhangi birinin üstünde bulunan ok işaretine tıklayınca, ekranın sağ kısmında içerisinde dizayn sayfasında kullanıma uygun kontrolör bileşenler olan bir araç kutusu menüsü açılır. Buradaki uygun bir kontrolöre tıklandığında ilgili bileşen, ilgili sütuna ya da kısma eklenir. Bileşenler tasarlama aracına ilk eklendiği anda tanımlanmamış içerik, üzerinde soru işareti olan bir doküman ikonu barındıran bir kutucuk şeklinde temsil edilir. Bu ikona tıklandığı zaman ise ekrana diyalog türünde açılır bir ekran penceresi gelir ve

kullanıcının içeriği tanımlanması istenir. İçeriği girip pencere onaylandıktan sonra ilgili içerik temsili ikonunun üzerindeki soru işareti silinir ve bir onay işareti gözükür. İçerik tanımlamasında, eklenen kontrolörün çeşidine göre içeriğin farklı şekillerde girmesi beklenebilir. Bunlar eklenti dosyalarındaki dâhili şablonlar ile belirlenir. Bazı eklentiler varsayılan olarak değer alamıyor ise tanımlama ekranında veri girişine izin vermez. Bazı eklentiler ise verilerin interaktif araçlar ile o içerik bloğuna bağlanmasına izin veriyor olabilir. Ayrıca, ön tanımlı bir liste yoluyla kullanıcının içeriği seçerek tanımlamasına şart koşan kontrolörler de mevcuttur.

Dizayn sayfası tasarlama aracının öne çıkan özelliği sürükle-bırak desteği olmasıdır. Tanımlı olsun olmasın, her içerik bloğu kolonlar arasında veya kısımlar arasında sürüklenerek taşınabilir. Bunu yapma için kutucuğun üst kısmındaki taşıma çubuğundan tutup sürüklemek yeterlidir. Taşınan blok varış kısmına yaklaştığı zaman otomatik olarak hizalama aldığından bu işlem oldukça kolaydır. Ayrıca bu taşıma işlemi kolon içindeki sıralamayı değiştirmek için de kullanılır. Taşıma çubuğunda bulunan ok bağlantıları sayesinde bu bloklar sürükleyerek taşımaya ihtiyaç duymadan sıralamada bir üst veya bir alt hizaya kaydırılabilir.

Sayfa tasarımı oluşturulduktan sonra ilgili sayfanın adı yazılarak ‘Create Design’ butonu ile sayfa oluşturulur. Ancak dizayn sayfaları da form sayfaları gibi, oluşturulduktan sonra yayın dışı durumdadırlar. Bir proje içerisindeki iş akışına eklenerek yayına geçirilirler.

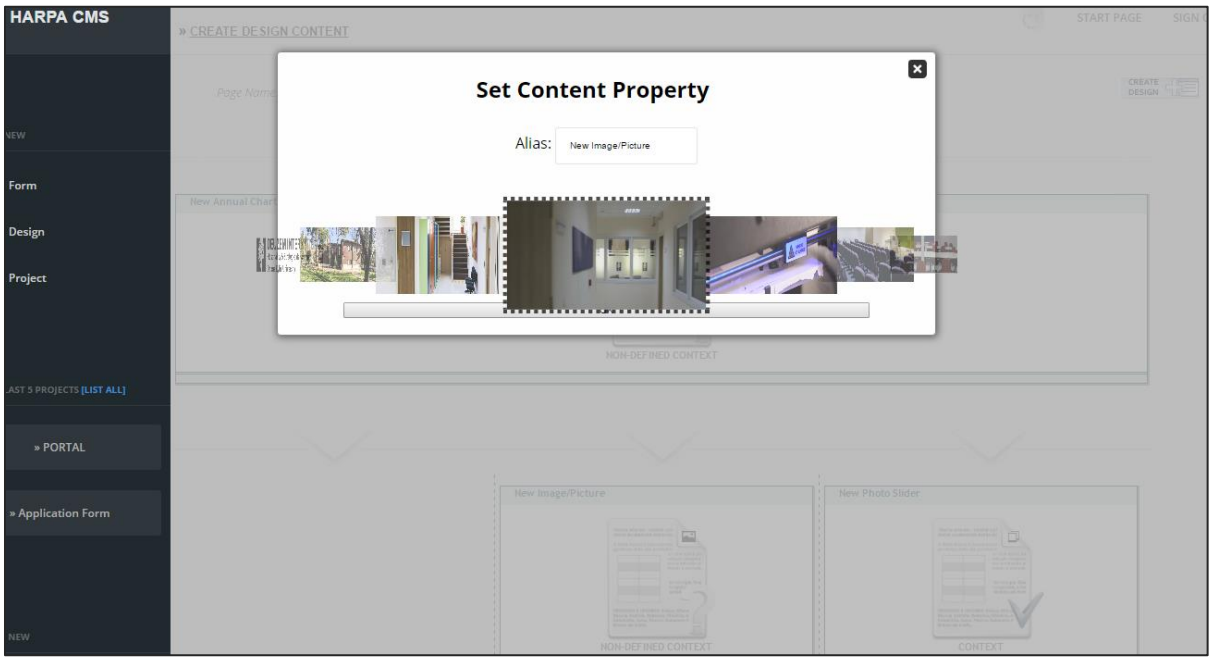
Şekil 25: HARPA İYS Kontrol Paneli Üzerinden Statik Dizayn Sayfası Geliştirme



Kaynak: Yazar tarafından derlenmiştir.

HARPA İYS içerisinde kullanım kolaylığı ve kullanıcı etkileşimini arttıran birçok özellik bulunmaktadır. Bunlardan bir tanesi, fotoğraf kontrolörü (image/picture) bileşeni yönetim paneli tarafında çalışan dâhili şablonu (picker.php) önemli bir özellik barındırır. Sistemde yüklü tüm fotoğrafları 'Atlı Karınca' efekti ile ekranda döndürerek kullanıcının üç boyutlu olarak galeride gezmesine olanak tanır. Kullanıcı bu gezintiyi fare tekerleğini döndürerek gerçekleştirir. Uygun gördüğü fotoğrafı tıklayarak onaylar ve parametreleri tanımlayarak fotoğraf içeriğini sayfaya yerleştirmiş olur.

Şekil 26: HARPA İYS Kontrol Paneli - Gelişmiş Veri/Fotoğraf Bağlama Araçları



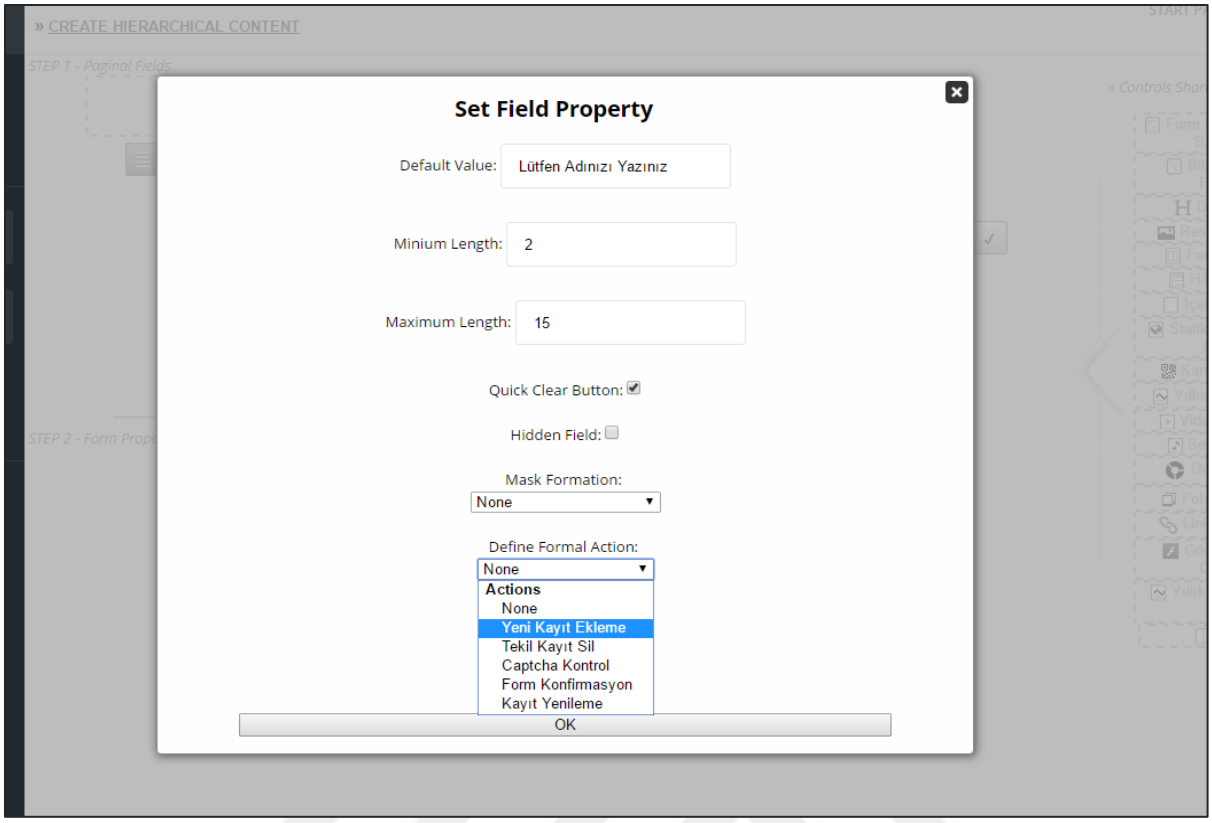
Kaynak: Yazar tarafından derlenmiştir.

Her bir içeriğin öznitelikleri düzenleme aşamasında 'SET' butonuna tıklanarak açılan küçük pencere üzerinden tanımlanabilir. Bunların içerisinde ön tanımlı değer, alanın en az ve en çok taşıyabileceği karakter sayısı (bu sayılara göre kullanıcı çıktısında form boş alan ve alan aşım kontrolleri yapılır), hızlı alan temizleme ikonu, alanı gizleme ayarı, tarih veya telefon numarası gibi klavye girdisi sonrasında belli bir formasyona dizilecek ve şekillenecek bilgiler için maskeleme şeması (Mask Formation), tanımlı aksiyon bulunur.

Maskeleme şeması seçenekleri içinde ön tanımlı olarak şu biçimler mevcuttur;

Uluslararası telefon numarası, gün/ay/yıl tarih formatı, yıl/ay/gün tarih formatı, sadece nümerik, ip adresi formatı, oran, dosya filtresi – sadece fotoğraf, dosya filtresi – sadece medya öğesi, dosya filtresi – sadece evrak dosyası.

Şekil 27: HARPA İYSde Tanımlanan Form Elemanına Tanımlı Aksiyon Atama



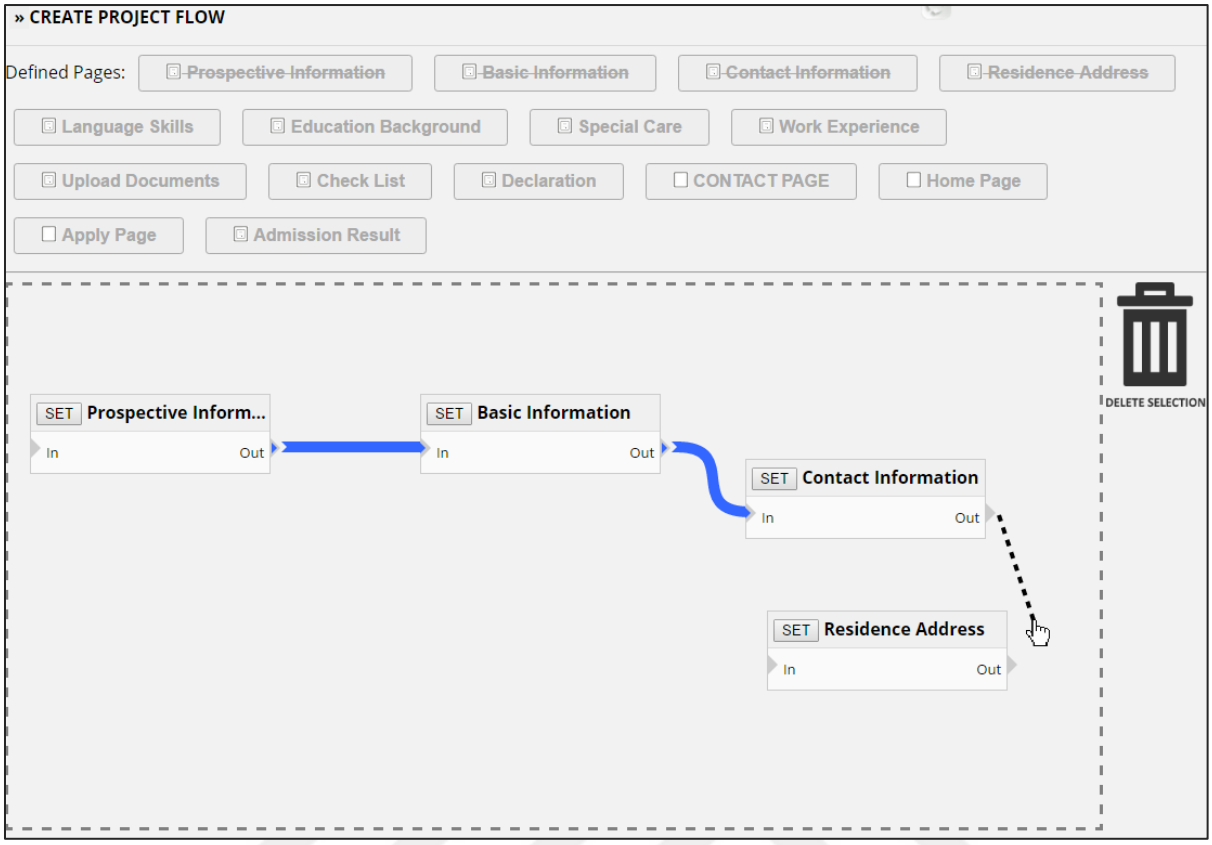
Kaynak: Yazar tarafından derlenmiştir.

Şekil 27'de gösterilen aksiyon tanımlama seçeneği sayesinde, her bir veri alanına ayrı bir aksiyon atayarak özgün formlar oluşturulabilir. Ön tanımlı 'Güvenlik Kodu' (CAPTCHA) adı verilen mekanizmayı taşıyan kontrolör bile sayfaya tasarlama aracı yardımıyla kolayca eklenebilir ve 'Form Konfirmasyon' adlı ön tanımlı aksiyonu buradan atanarak sayfada güvenlik fotoğrafı kontrolü sağlanabilir.

Sayfa çıktıları mobil destekli olduğundan, kullanıcılar çözünürlük problemleri yaşamadan ya da tema arayışına girmeden form sayfalarını oluşturabilirler. Hiyerarşik olarak her bir alt veri tek tek yetki kontrolünden geçtiği için istisnai güvenlik problemleri ya da zafiyetleri söz konusu olmaz. Sistem, kayıtlı kullanıcı doğrulamasını sayfa bazında değil her bir içerik veya form alanı bazında yapar. Bunun yanı sıra; iç ve dış içerik hizmetlerinde kullanıcı denetimi ve üyelik işlemlerinin yapılabilmesi amacıyla geçiş sayfaları da sisteme entegre edilmiştir.

Şekilde 28'de, kontrol paneli içerisindeki proje oluşturma editörünün ekran görüntüsü yer alır. Form ve dizayn sayfaları oluşturulduktan sonra bu sayfa üzerinde sıraya konularak iş akışı oluşturulur ve proje tamamlanır. Sistemde oluşturulmuş sayfalar ekranın üst kısmındaki panelde görünür ve yönetici istediği sayfayı fare ile sürükleyerek çalışma alanına taşır.

Şekil 28: HARPA İYS Kontrol Paneli Üzerinde Web Sayfalarının İş Akışı Olarak Projeye Dâhil Edilmesi

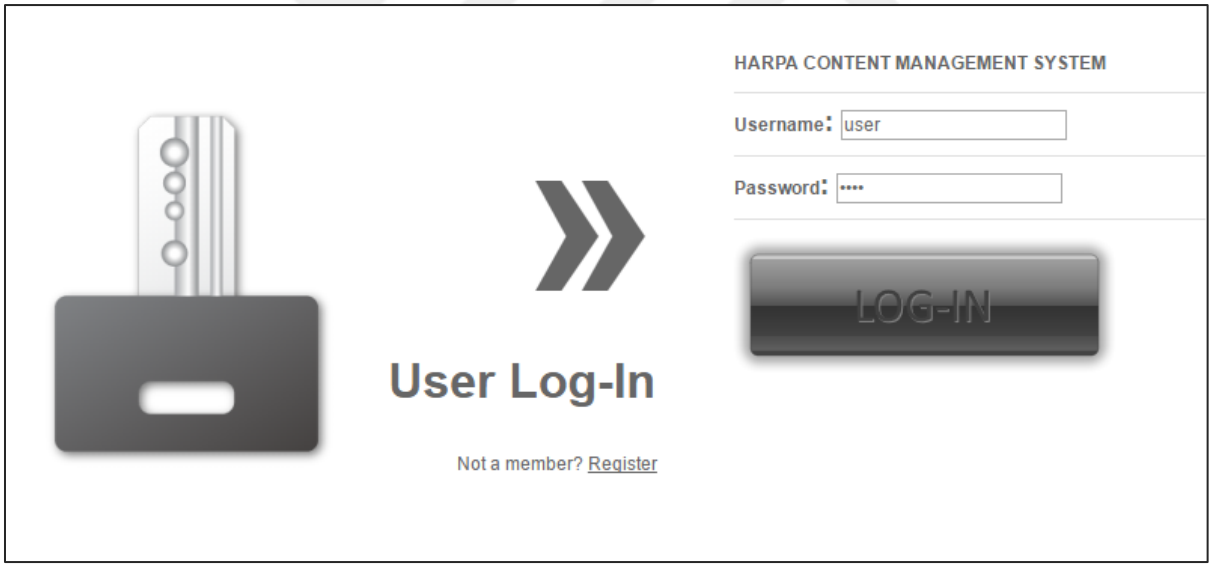


Kaynak: Yazar tarafından derlenmiştir.

Ayrıca Şekil 28'de görüldüğü gibi taşınan sayfa isimlerinin üzeri otomatik olarak çizilir bunun sebebi aynı projede aynı sayfanın bir daha kullanılamamasıdır. Taşınan sayfalar çalışma alanında dikdörtgen iş akışı kutucuğu olarak temsil edilir. Bu kutucuklar sağ ve sol kısmında bulunan ok işaretleri ile birbirlerine bağlanırlar. İki iş akışını birbirine bağlamak için, önce çalıştırılacak iş akışının sağ tarafındaki ok işaretine tıklanır, ardından ikinci iş akışı kutucuğunun sol tarafındaki oka tıklanır ve tesisat kurulmuş olur. Bu işlem ile iki iş akışı arasında köprü kurulmuş olur ve görsel anlamda bu bağlantı düz ya da kıvrımlı çizgi ile temsil edilir. Bu kutucuklar sınırsız sayıda olabilir ve birbirlerine bağlanmak suretiyle akış tamamlanır. Proje sistemi kurulduktan sonra aynı sayfada proje ismi yazılır ve yapı kaydedilir. Bu işlemden sonra tüm sayfalar iş akışı yapısında yayınlanmış olur ve bu aşama İYSlerde gerekli olan 'yayınlama' aşamasıdır. Bu yapının diğer İYSlere göre başlıca farkı onaylanan sayfaların tekil olarak değil proje halinde aktif edilmesidir. Önemli bir husus, kurulan iş akış hattındaki son iş parçacığı, ilk iş parçacığına bağlanamaz; akış hattının kısır döngüye sokulamaması amacıyla sistem korunduğundan sayfa böyle durumlarda uyarı verir. Oluşturulan projenin çalışır biçimde kaydedilebilmesi için ayrıca her bir iş akış kutucuğunun

ayarlanması gerekir. İlgili kutucuğun sol kısmında yer alan ‘SET’ butonuna tıkladıktan sonra açılan diyalog türü pencerede o iş akışına özgü parametreler girilir ve pencere onaylanır. Bu bilgiler; akış ismi, eğer form sayfası içeren bir iş akışı ise aynı kullanıcı tarafından o formun en fazla kaç defa doldurulabileceği sayısı (döngü sayısı), sayfaya erişim izni verilen kullanıcı veya kullanıcı grubu (tüm üyeler, tüm ziyaretçiler), eğer form sayfası içeren bir iş akışı ise kullanıcıların form girdilerini hangi kullanıcının veya kullanıcı grubunun görebileceği bilgisidir. Bu zorunlu bilgiler tüm iş akışlarına uygulanmak zorundadır. Ayrıca sistem üzerinde bütünü ilgilendiren bir kısır döngüye izin verilmese de, herhangi bir iş akışı üzerinde ayar yapılarak akışın o parçacıkta takılı kalması ve kısır döngüye girmesi sağlanabilir. Bu gerçekleştirilerek projenin herhangi bir form sayfasının kullanıcı tarafından sınırsız sayıda doldurulabilmesi sağlanabilir. Bunun için o iş akışının döngü sayısı -1 olarak girilebilir. Eğer bu alan 0 değeri alır ise kullanıcının bu form sayfasını doldurması beklenmeden baypas edilmesi sağlanır.

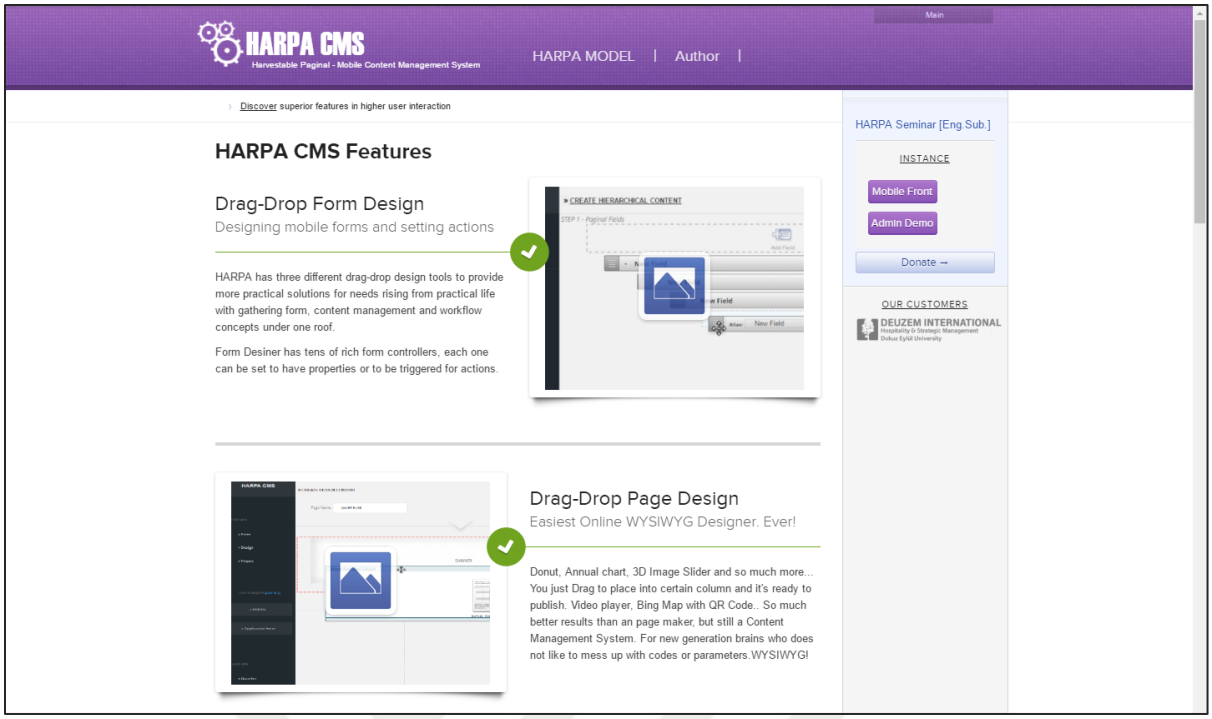
Şekil 29: HARPA İYS Uygulamasında Kayıtlı Kullanıcı Girişi



Kaynak: Yazar tarafından derlenmiştir.

Şekil 29’da ki sayfa, ziyaretçiye üyelik gerektiren bölümlere erişim sağlamak istediği zamanlarda görüntülenir. Ziyaretçi eğer daha önce üyelik yapmamış ise üye olması gerektiğini hatırlatan bir bağlantı ekranın ortasında mevcuttur. Eğer ziyaretçi sistemde kayıtlı bir üyeliğin sahibi ise ekranın sağ kısmından kullanıcı adı ve parolasını girer ve sayfaya erişim sağlar.

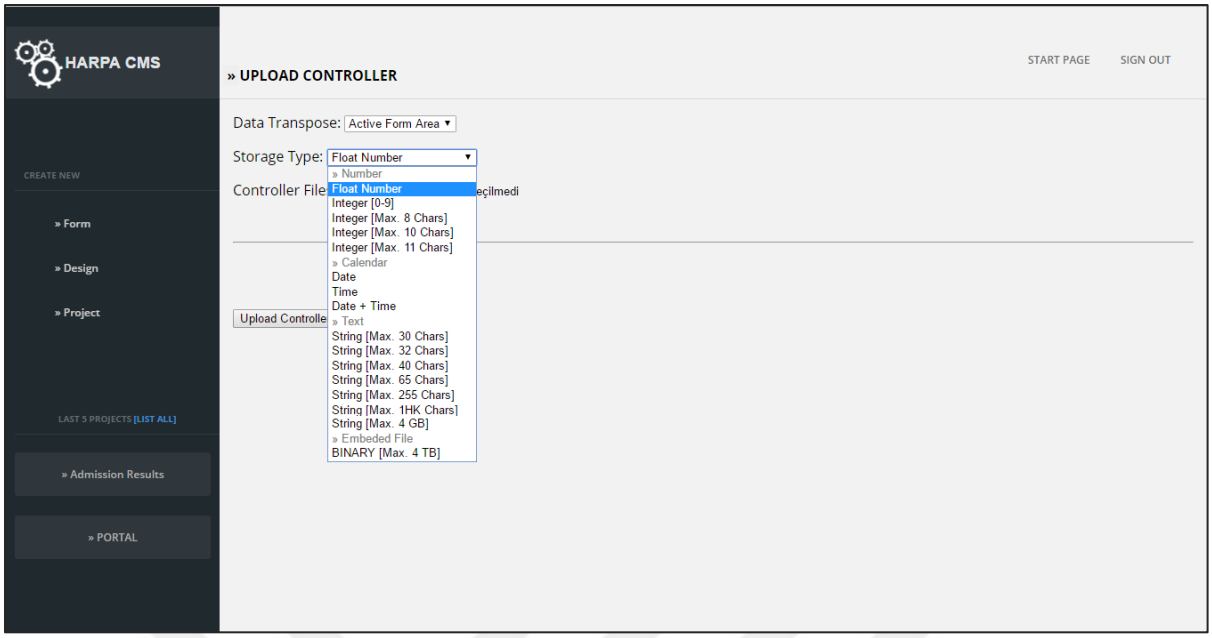
Şekil 30: HARPA İYS Ürün Sitesi (www.harpacms.com) Ekran Görüntüsü



Kaynak: Yazar tarafından derlenmiştir.

Şekil 30'da ki ekran görüntüsü, ön yüz ve kontrol paneli demo (tanıtım), dokümantasyon, seminer video kaydı (İngilizce altyazılı) ve ekran görüntülerini içeren ürün sitesine aittir. Ancak ürün sitesi, HARPA içerik yönetim sistemi ile oluşturulmamıştır. Tez uygulaması kapsamının dışındadır; dağıtım ve ön bilgilendirme amaçlı kullanılır.

Şekil 31: HARPA İYS Kontrolör Bileşen Yükleme Sayfası Ekran Görüntüsü



Kaynak: Yazar tarafından derlenmiştir.

Şekil 31'de, kontrol paneli kontrolör yükleme ekran görüntüsü yer alır. Kullanıcılar kendileri geliştirdikleri kontrolör bileşenleri bu sayfa üzerinden sisteme yükleyerek platformu genişletirler. Söz konusu sayfada sistemin işleyişi açısından önemli iki ayar kısmı bulunmaktadır. Bunlardan birincisi (Data Transpose), yüklenen bileşen kontrolörün aktif veri taşıyan ve aktaran bir form kontrolörü olup olmadığıdır. İkinci liste (Storage Type) ise, bu kontrolörün sisteme kaydedeceği ya da temsil edeceği verilerin türü ve uzunluğudur. Bu tanımlama sayesinde, bileşenin arka planda hangi veri kümesini kullanacağı da belirlenmiş olur. HARPA İYS yapı itibarı ile kümelenmiş veri teknolojisi ile çalıştığından, bu tanımlama oldukça önemlidir ve sistemin kitlesel düzeyde veri ile çalışarak sistemin performanslı bir ziyaret deneyimi sunmasını sağlar.

Her HARPA İYS kontrolör bileşen eklentisi, aynı türde veriyi üzerinde barındırır ve eğer dinamik veri aktaran formal bir bileşen ise aynı türde veriyi sisteme taşır, yani tuttuğu ve aktardığı veri türü aynıdır. Bu sebeple, her bileşen MYSQL veritabanındaki 16 kümeden birisine tanımlanmak durumundadır. Birden fazla bileşen aynı kümeyi paylaşabilir ancak platformdaki performans standartlarını yüksek tutmak isteyen bir yönetici, veri saklama yoğunluğunu tüm kümelere dağıtmak durumunda olduğunu bilmelidir. Bu sayede sistem ve üzerinden oluşturulacak olan internet siteleri optimum kaynakları kullanabilir. Unutulmamalıdır ki, MYSQL tablolarında toplam veri tutma kapasitesi sınırlıdır.

Tablo 3: HARPA İYS Ve Öğrenci Kayıt Sistemi Üzerindeki Kullanıcı Grupları Ve Yetkileri

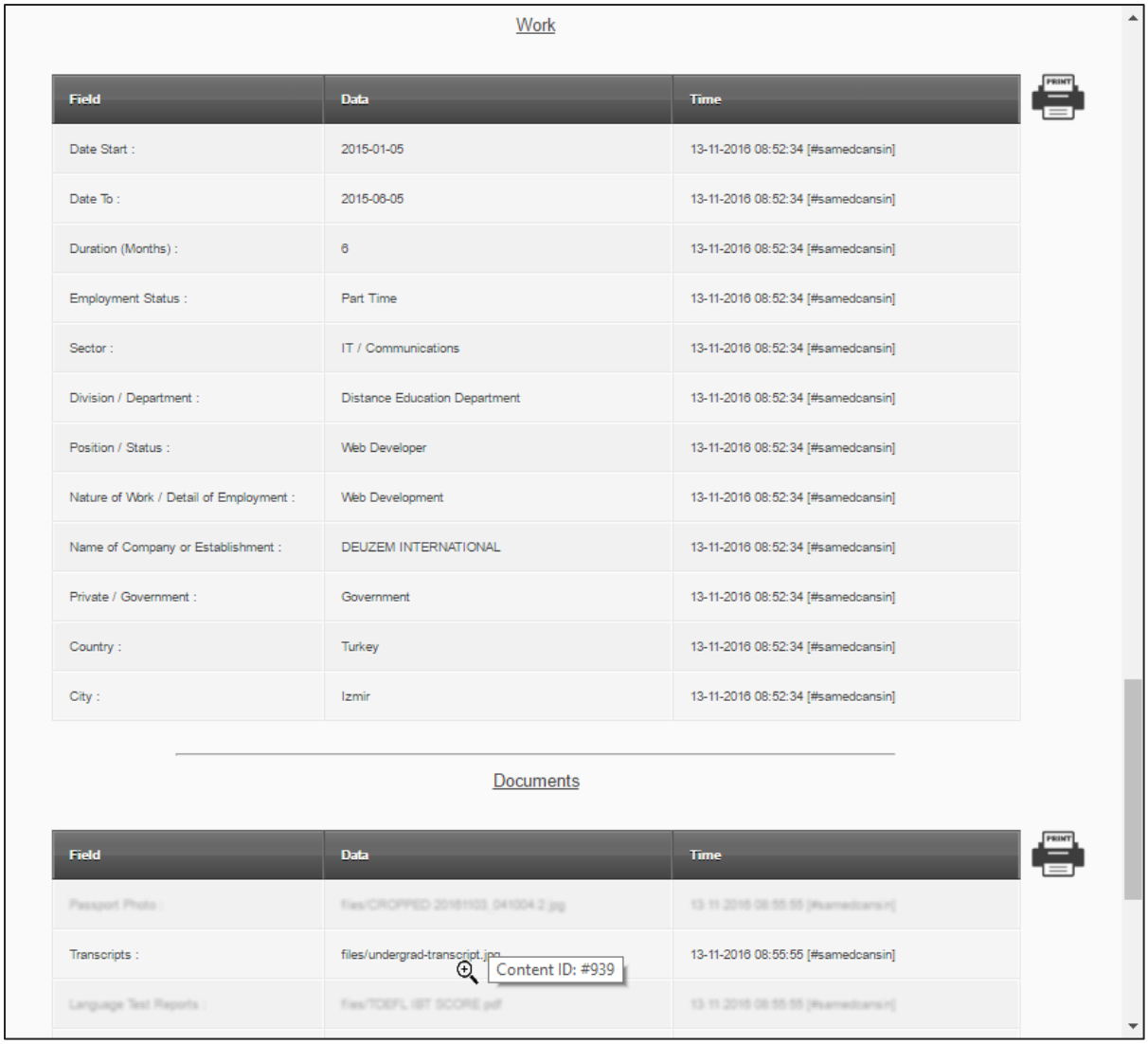
Yetki	Tanımlı Yönetici	Ziyaretçi	Tüm Üyeler
Site içerisinde kullanılmak üzere fotoğraf galerisi ve liste verileri yükleme.	✓	X	X
Site içerisinde kullanılmak üzere fotoğraf galerisi ve liste verileri yükleme.	✓	X	X
Başvuru sistemini oluşturan form sayfaları oluşturma.	✓	X	X
Tüm sayfaları iki farklı iş akışına oturtarak bağlantılı iki proje oluşturma.	✓	X	X
Katalog sitesini ziyaret etme.	✓	✓	X
Üyelik kaydı yapma.	X	✓	X
Üye girişi gerçekleştirme.	X	✓	X
Üye çıkışı gerçekleştirme.	✓	X	✓
Form sayfalarını doldurarak başvuruyu gerçekleştirme.	✓	X	✓
Yapılan başvuruları inceleme ve çıktısını alma.	✓	X	✓

Kaynak: Yazar tarafından derlenmiştir.

İçerik Yönetim Sistemi ve platformun çıktı niteliğindeki öğrenci kayıt sistemi bazında erişim yetkileri, üç grup üzerinden incelenebilir; yönetici (administrator), ziyaretçi (guest) ve üye kaydı oluşturarak sisteme kayıt olan tüm diğer kullanıcılar. Bunlar kullanıcı grupları olarak tanımlanabilse de, forum benzeri sistemlerdeki gibi arttırılabilir dinamik yetki grupları bulunmaktadır. Yönetim paneli üzerinden oluşturulan öğrenci kayıt sisteminde, aday öğrencilerin ön kayıt yaptırabilmesi için bir sistem olduğundan öncelikle sisteme üye olmaları gerekmektedir.

HARPA İçerik Yönetim Sistemi raporlama kısmı, kullanıcıların form sayfaları üzerinden girdikleri verilerin yine kullanıcılara görüntülenmesini sağlayan sayfadır. Bu bölüme ilgili form sayfasının sağ kısmındaki açılır menüden 'REPORTS' bağlantısına tıkladıktan sonra, ekranda görünen kullanıcı listesinden hangi kullanıcının form girdileri görüntülenecekse listedeki o kullanıcı bağlantısına tıklandıktan sonra ulaşılır. Aynı kullanıcı listesi sayfasına, sağ kısım menüdeki 'PROJECTS' bağlantısına tıkladıktan sonra, açılan pencereden ilgili projenin yanındaki 'Reports' linkine tıklayarak ta ulaşılır.

Şekil 32: HARPA İYS Üzerinde Kayıtlı Form Verilerinin Rapor Dökümünde Görüntülenmesi



Work

Field	Data	Time
Date Start :	2015-01-05	13-11-2016 08:52:34 [#samedcansin]
Date To :	2015-06-05	13-11-2016 08:52:34 [#samedcansin]
Duration (Months) :	6	13-11-2016 08:52:34 [#samedcansin]
Employment Status :	Part Time	13-11-2016 08:52:34 [#samedcansin]
Sector :	IT / Communications	13-11-2016 08:52:34 [#samedcansin]
Division / Department :	Distance Education Department	13-11-2016 08:52:34 [#samedcansin]
Position / Status :	Web Developer	13-11-2016 08:52:34 [#samedcansin]
Nature of Work / Detail of Employment :	Web Development	13-11-2016 08:52:34 [#samedcansin]
Name of Company or Establishment :	DEUZEM INTERNATIONAL	13-11-2016 08:52:34 [#samedcansin]
Private / Government :	Government	13-11-2016 08:52:34 [#samedcansin]
Country :	Turkey	13-11-2016 08:52:34 [#samedcansin]
City :	Izmir	13-11-2016 08:52:34 [#samedcansin]

Documents

Field	Data	Time
Passport Photo :	files/CROPPED 20161103_041004 2.jpg	13 11 2016 08:55:55 [#samedcansin]
Transcripts :	files/undergrad-transcript.jpg	13-11-2016 08:55:55 [#samedcansin]
Language Test Reports :	files/TOEFL IBT SCORE.pdf	13 11 2016 08:55:55 [#samedcansin]

Content ID: #939

Kaynak: Yazar tarafından derlenmiştir.

Neticede, bir raporlama sayfasında; ilgili projenin içindeki tüm iş akışlarına bağlı form sayfaları üzerinden, hedef kullanıcının tüm form girdileri birleştirilmiş halde izlenir. Bu sebeple ilgili kısım, hedef kullanıcı referanslı birleştirilmiş form olarak ele alınabilir.

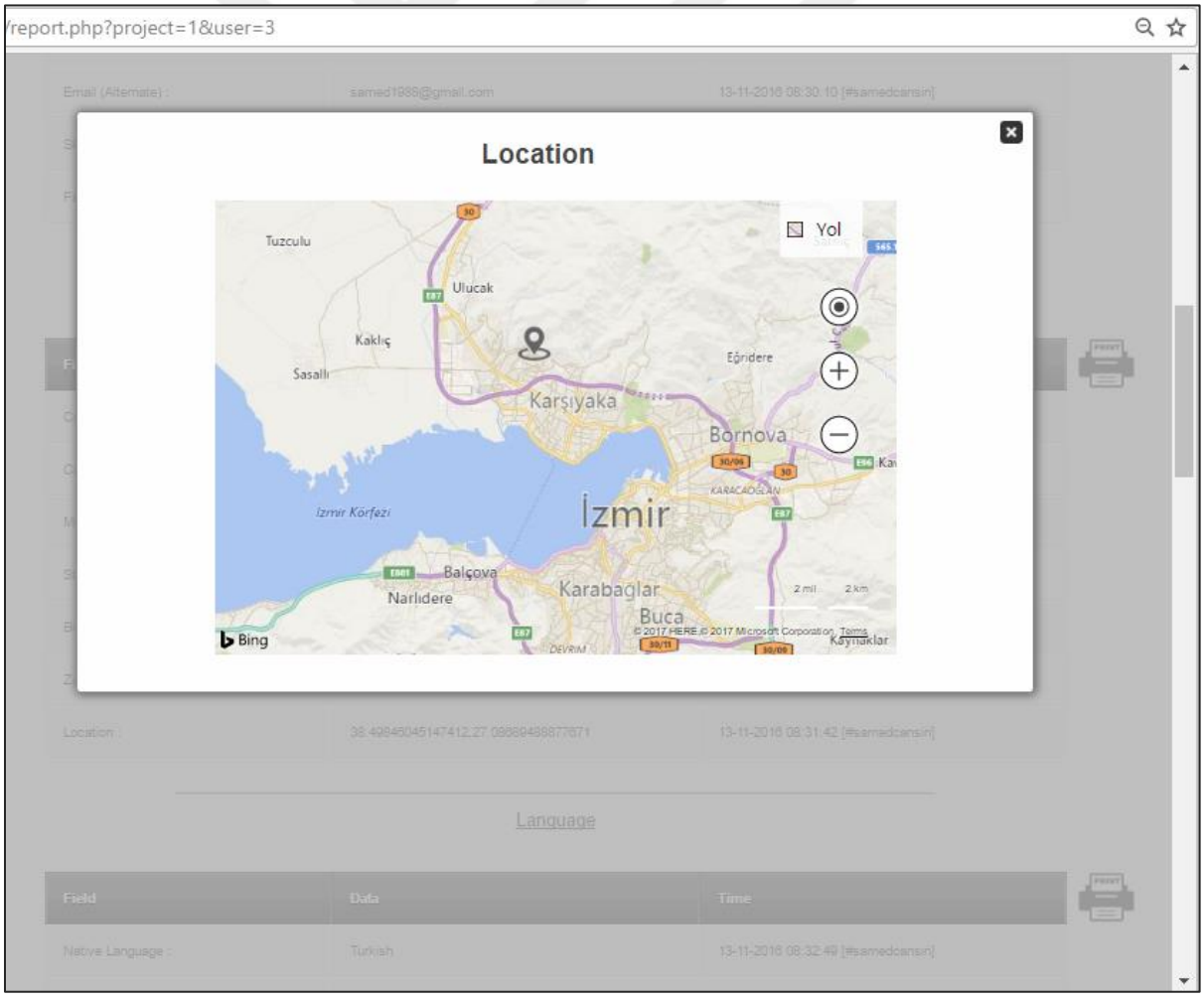
Fakat ele alınması gereken diğer bir mesele, bu raporlama kısmına giren kullanıcının da yetkileridir. Örneğin, 11 iş akışından ve dolayısıyla bunlara bağlı 11 form sayfasından oluşan ‘Öğrenci Kayıt Sistemi’ projesinin tüm sayfalarındaki formları dolduran ‘kullanici01’ tüm bilgilerini eksiksiz ve doğru şekilde girmiştir. Raporları okuması beklenen ‘kullanici02’ rumuzlu kullanıcının da 11 iş akışına da rapor okuma yetkisi olması gerekmektedir. Bu durum, proje ve iş akışları yönetim paneli üzerinden tasarlanırken, her bir iş akışının ayarlarından ‘Rep. Recipient’ yani rapor alıcı bilgisinin ‘Everyone’, ‘Members Only’ ya da ‘kullanici02’ olarak ayarlanması ile söz konusu olur. Eğer bir rapor okuyucusu kullanıcı, iş

akışlarından birinin raporlarını okuma yetkisine sahip değil ise, rapor dökümünde o iş akışına bağlı sayfadan girilen form girdi verilerini göremez.

Rapor dökümüne giren bir kullanıcı, başka (listeden seçtiği) bir kullanıcının o projedeki tüm form girdilerini tek bir sayfa üzerinden görüntüler. Ancak bu formal girdi bilgileri sayfada kısımlar halinde gösterilir yani her sayfa üzerinden girilen bilgiler ayrı ayrı gruplanmıştır. Her bir grubun yanındaki yazıcı ikonuna tıklayarak sadece o kısmın yazıcıdan bastırılarak çıktı olarak alınması da sağlanabilir.

Rapor dökümündeki satırlarından birine fare işaretçisi ile odaklanıldığı zaman o gruptaki diğer satırlar bulanıklaşır, bu sayede o satırın okunması kullanıcı tarafından daha kolay bir hale gelir. İlgili satır görsel açıdan odaklanılabilir ve diğer satırlardan ayırt edilebilir bir hal alır ve bilgiler göz ile kontrol edilirken satır atlama problemi yaşanmaz. Bu durum kullanıcı etkileşimi üzerine yapılan ayrı bir iyileştirme çalışmasıdır.

Şekil 33: HARPA İYS Rapor Dökümünde Formal Veri Detayı Görüntüleme İşlemi



Kaynak: Yazar tarafından derlenmiştir.

Her satırda bilginin adı, içeriği ve hangi kullanıcı tarafından hangi tarih ve saatte girdi olarak sisteme eklendiği görülür. Ancak içerik, ham veri bazında ilgili hücreye sığmadığından kısaltıldığı durumlar olabilir. Ya da, içeriğin direk olarak bir anlam ifade etmediği ve bir bağlam (context) ile birlikte görüntülenmesi gerekebilir. Bu sebeple, her satırdaki içeriğe tıkladıktan sonra açılan diyalog tarzı açılır pencerede içeriğin detayları incelenebilir.

Bazı bilgiler, örneğin koordinat verisi gibi bilgiler harita üzerinden girdi yapılarak sağlandığı için yine harita üzerinde görüntülenmelidir. Form sayfasında yükleme alanı üzerinden yüklenen dosyalar, raporlama kısmında diyalog türü penceredeki indirme (download) bağlantısı üzerinden kullanıcıya indirtilebilir. Bu gibi içerik raporlama biçemleri, ilgili içeriğin ilişkili olduğu türün kontrolör dosyalarından olan 'report.htm' dosyası üzerinde yapılır. Bu dosya bir raporlama temasıdır ve eğer ilgili kontrolörde bu dosya yok ise ham veri rapor diyalog penceresinde direk olarak ham veri şeklinde ekrana basılır.

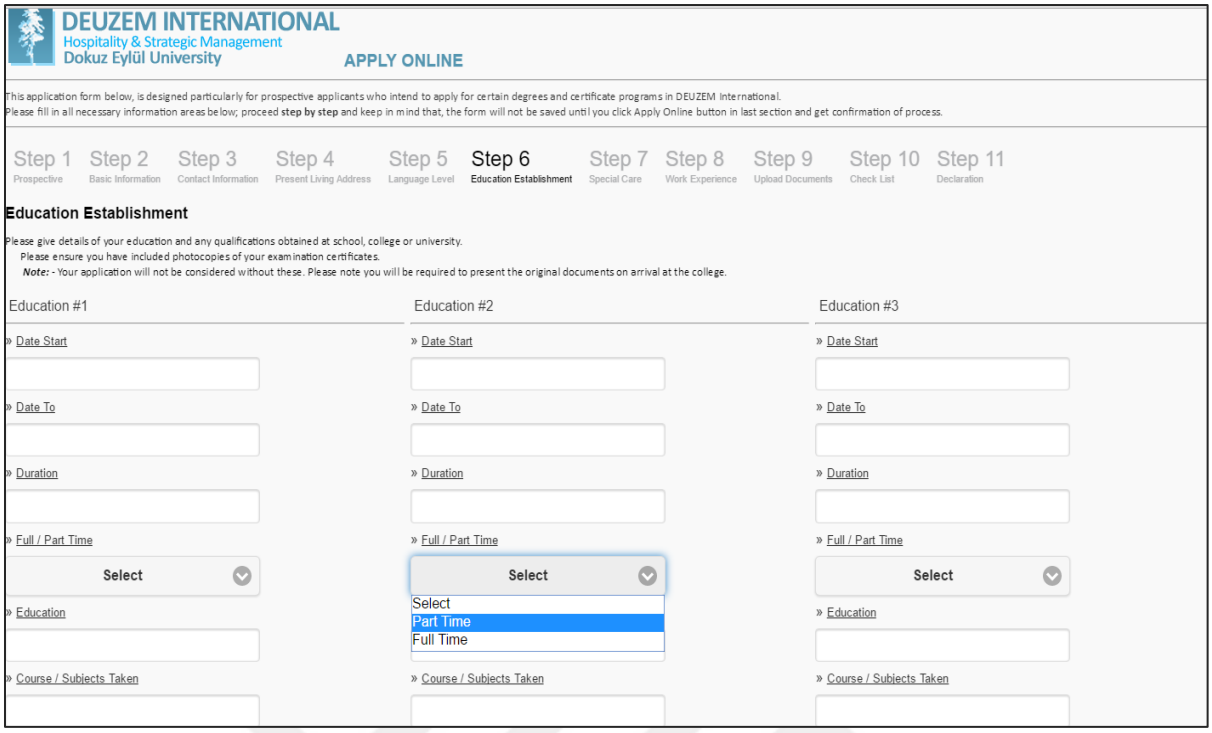
3.2. UYGULAMA ALANI (ULUSLARARASI ÖĞRENCİ KAYIT SİSTEMİ)

Harpa İçerik Yönetim Sistemi tabanı üzerinde kontrol paneli yardımıyla oluşturulan öğrenci kayıt sistemi, uluslararası yükseköğrenim programları için başvuruları kabul eden ve aday öğrencilerden talep edilen bilgileri girmesi sağlanan web tabanlı başvuru kanalıdır. Ön bilgilendirme amaçlı katalog projesi (PORTAL) ve form akışlarından oluşan başvuru projesinden (Application Form) oluşur. Bu projelerde barındırılan; bilgilendirme amaçlı arayüz katalog sayfaları dizayn sayfalarından oluşurken, başvuru sisteminin sayfaları dinamik form niteliğindedir.

Bunların dışında, kullanıcıların erişimde olan rapor sayfaları, yeni üyelik, üye giriş ve üye çıkış sayfaları, kontrol paneli sayfaları sistem sayfası niteliğindedir ve teknik anlamda dinamik yapılarda olsalar bile tasarım yapıları kullanıcı tarafından değiştirilemez.

Öğrenci Kayıt Sistemi Başvuru Projesi uluslararası kullanımda olduğundan sayfa içeriği İngilizcedir. Yapı olarak çok dilliliği desteklemez. Ancak, uygun eklentiler hazırlanırsa Google Translate, Microsoft Translate gibi dış kaynaklar üzerinden otomatik tercüme entegre edilebilir ya da kullanıcının diline göre hazırlanan farklı sayfalara otomatik yönlendirilebilir.

Şekil 34: HARPA İYS ile Tasarlanmış Mobil Destekli Öğrenci Kayıt Sistemi Çıktısı Kullanıcı Arayüzü



Kaynak: Yazar tarafından derlenmiştir.

Üye girişi yapmış kullanıcıdan talep edilen bilgiler aşağıda listelenmiştir:

- Kayıt Olunacak Eğitim Programı
- Kişisel Bilgiler
    - Başvuran
    - Unvan
    - Ad
    - İkinci Ad
    - Soy Ad
    - Meslek
    - İş Pozisyonu
    - İş Durumu
    - Uyruk
    - Doğum Tarihi
- İletişim Bilgileri
    - Telefon
        - Cep
        - Ev

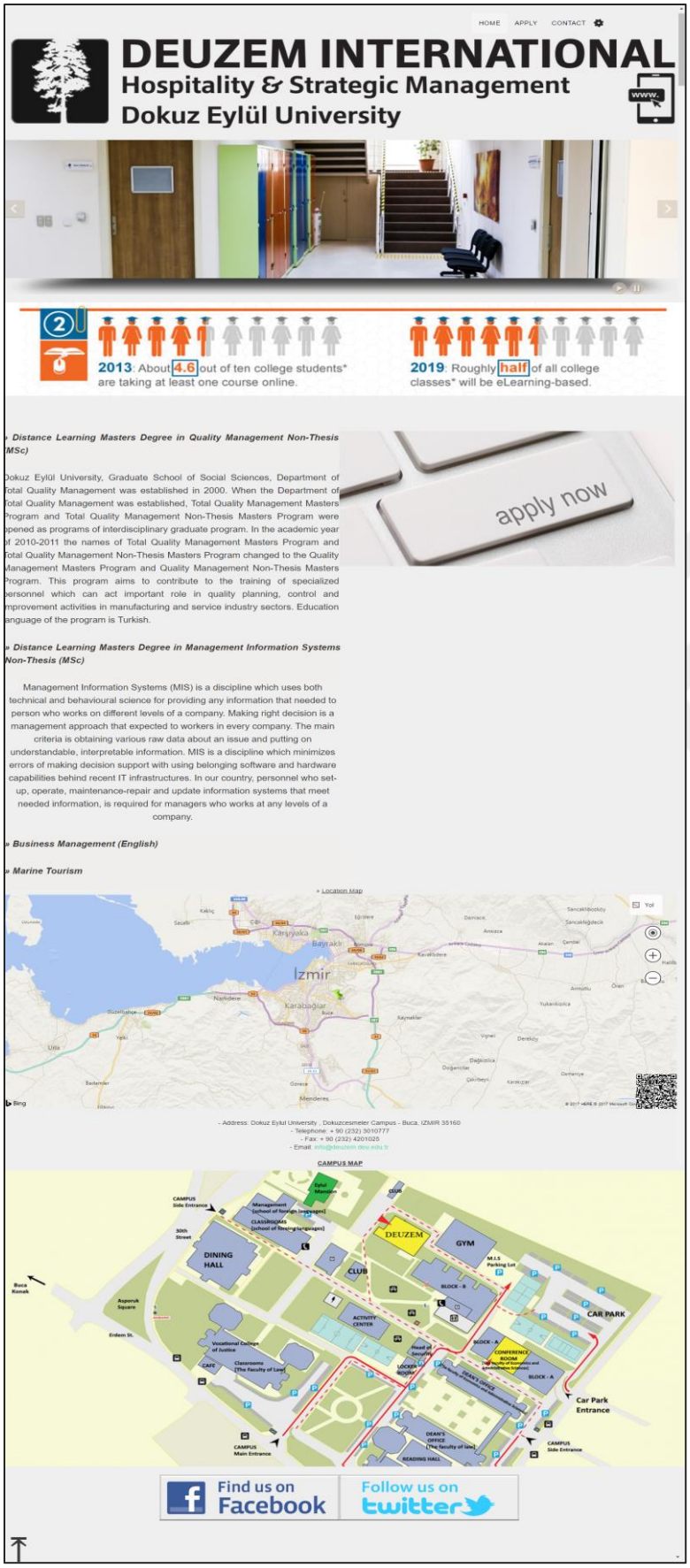
        - İş
        - Faks
    - İnternet
        - E-Posta
        - Alternatif E-Posta
        - Skype Hesabı Kullanıcı Adı
        - Facebook veya Twitter Hesabı Profil Adresi
- İkamet
    - Ülke
    - Eyalet
    - Şehir
    - Bölge
    - İlçe
    - Cadde/Sokak
    - Blok ve Daire No
    - Posta Kodu
    - Harita Konumu
- Dil
    - Ana Dil
    - İngilizce Seviyesi
    - 2. Yabancı Dil
    - 3. Yabancı Dil
    - TOEFL Sınav Puanı
    - IELTS Sınav Puanı
    - TOEIC Sınav Puanı
- Eğitim Geçmişi (Eğitim 1,2,3)
    - Program Adı
    - Başlangıç
    - Bitiş
    - Eğitim Süresi (Ay)
    - Kayıt Türü
    - Alınan Dersler
    - Not Ortalaması
    - Eğitim Kurumu

        - Ad
        - Mülkiyet
    - Denklik/Uzmanlık
    - Ülke
    - Şehir
- Sağlık Durumu
    - Okuma ve Öğrenme Bozukluğu
    - İşitme Bozukluğu
    - Görme Engeli
    - Fiziksel Engel
    - Diğer
- İş Geçmişi (İş 1,2,3)
    - Başlangıç
    - Bitiş
    - Süre
    - Çalışma Durumu
    - Sektör
    - Departman
    - Pozisyon
    - Detay
    - İşletme
        - Ad
        - Şirket/Kurum
        - Ülke
        - Şehir
- Evrak
    - Pasaport Fotoğrafı
    - Transkript
    - Diploma Örneği
    - Dil Testi Sonuçları
    - Diğer
- Teyit ve Dijital İmza

Öğrenci kayıt sistemi tanıtım projesi, hedef kitlesi uluslararası olduğu içeriği için İngilizce dilinde oluşturulmuştur. Örnek bazda üç alt dizayn sayfası barındırır. Sayfalar arası geçişleri kolaylaştırmak amacıyla pencerenin en üst kısmındaki menü sabit ve yerleşiktir. Sayfalar dizayn sayfasının kontrolör eklentisinin temeli olan kayar paralaks mekanizma (parallax) üzerine, iş akışı bazında art arda eklenmiştir. Sayfalar arası geçişler; fare tekerleği, pencere kaydırma çubuğu, klavyedeki üst ve alt ok tuşları ya da en üst kısımda bulunan mobil destekli menü sayesinde gerçekleştirilebilir. Bu menü sayfa kaydırılsa bile pencerenin en üst kısmında durmaya devam eder ve kullanıcının hangi sayfada olduğunu bilebilmesi için mevcut sayfanın menü elemanını yumuşak geçiş efekti ile açık renkteki bir kare içerisine alarak işaretler. Düşük ekran çözünürlüğüne sahip cihazlarda mobil erişimi kolaylaştırabilmek amacıyla, bu menü yapısı otomatik olarak dikey liste haline dönüşür.

Şekil 35'te görülen ana sayfada; başlık, animasyonlu fotoğraf galerisi ve sektöre ilişkin örnek istatistikler bulunur. Başvuru bilgilendirme sayfasında, dört eğitim programının başlığı ve ilk ikisinin açıklaması bulunurken, sağ kısımda başvuru sayfasına geçiş bağlantı görseli vardır. İletişim sayfasında, kampüsün şehirdeki harita konumunun işaretlendiği ve kare kod barındıran harita uygulaması, kampüs içi harita ve sosyal medya kanallarına bağlantı sağlayan logolar bulunur.

Şekil 35: HARPA İYS Tarafında Üretilen Uygulama Alanı Çıktısı (Öğrenci Kayıt Sistemi Tanıtım Sayfası)



Kaynak: Yazar tarafından derlenmiştir.

3.3. ÖN TANIMLI KONTROLÖRLER

HARPA İçerik Yönetim Sistemi, web sunucuya ilk yüklendiği zaman formal ve durağan nitelikte birçok mobil destekli kontrolör kullanıma hazır olarak gelir. Bu kontrolörler tema ve eklenti yapılarını kendi içinde barındırırlar ve yerleştirildikleri sayfa pozisyonunda tasarım unsurlarını oluştururlar, faal ve görsel nitelikte görevlerini yerine getirirler.

Yönetim panelindeki tasarım araçlarında, ön tanımlı kontrolörler aşağıdaki sırada kullanıma hazır halde bulunurlar:

- Data Form: Sayfalar arası iletişim elemanıdır, formal verilerin dinamik olarak kayıtlara geçebilmesi ya da başvuru sayfası gibi kullanıcıdan bilgi alan yapılarda kullanıcı tanımlı bilgilerin sistem tarafından okunabilmesi için, tüm form elemanlarının bu kontrolörün hiyerarşik olarak altına yerleştirilmesi gerekir.
- Plain Text: Düz metin elemanıdır. Yerleştirildiği yerdeki stil opsiyonlarını taşır.
- Text Box: Basit yazı alanıdır, tek satırlık metin yazılmasına olanak sağlar.
- Image/Picture: Sistemde yüklü fotoğraflardan yöneticinin seçtiğini sayfaya basar.
- Form Section/Step: Bir iş akışının içindeki sayfada bulunan bilgi aktarım formunun içerisinde, sayfa değişmeden farklı form adımları oluşturmak için kullanılır. Her bölüme ait kontrolörler bu kısmın altına hiyerarşik olarak eklenmelidir. Form sihirbazı olarak ta bilinir.
- Select List: Liste elemanıdır. Sistemde yüklü ve yöneticinin seçtiği veri setindeki liste içeriğini taşır ve kullanıcı tarafından seçilen opsiyon formun onaylanmasıyla işlem görür.
- HTML Content: Sayfaya ‘HTML’ içerik basar. Zengin metin olarak ta kullanılabilir.
- Multiple Check Box: Çoklu Seçim Kutusu elemanıdır. Sistemde yüklü ve yöneticinin seçtiği veri setindeki liste opsiyonlarını ekrana tek tek basar, kullanıcı birden fazlasını seçebilir ve formu onaylamasıyla tüm seçtikleri işlem görür.
- Expandable Select List: Liste elemanıdır, ancak kullanıcı listede olmayan bir opsiyon yazmak isterse içerisindeki yazı alanına kendi seçeneğini yazar, formu onayladığında seçtiği şık, seçmediyse yazdığı şık işlem görür. Varsayılan olarak sistemde yüklü ve yöneticinin seçtiği veri setindeki liste opsiyonlarını içerir.
- Large Text Area: Kullanıcıların çok satırdan oluşan metinler yazabilmesine olanak sağlayan formal yazı alanı kutucuğudur.

- Bing Map (Static): Harita bileşenidir. Microsoft BingMap altyapısını kullanır. Yöneticinin sayfayı oluştururken haritadan seçtiği koordinatı raptiyeleyerek kullanıcıya gösterir. Sağ alt köşesinde kare kod ile o koordinata link vererek, kullanıcıların ayrıca mobil cihazlarıyla kare kodu okutarak o koordinata ulaşabilmelerini de sağlar.
- Form Protection (CAPTCHA): Form sayfalarını otomatik girdi saldırılarına karşı koruyabilmek amacıyla geliştirilmiş, kullanımı kolay bir eklentidir. Data Form kontrolörünün hiyerarşik olarak altına konur, kendinden önceki tüm diğer form alanlarını saldırıya karşı koruduğu için sıralama olarak en sona konması gerekir.
- Bing Map (Dynamic): Harita kontrolörünün formal versiyonudur. Kullanıcılar kendi lokasyonlarını işaretleyebilirler ve form girdisi olarak işlenir. Kare kod desteği yoktur.
- QR Code: Yönetici tarafından atanan web link adresinin kare kodunu oluşturur ve sayfada konulduğu yere imaj olarak basar.
- Annual Chart: Yıllık çizelge bileşenidir. Sistemde yüklü ve yöneticinin seçtiği veri setindeki nümerik liste içeriğini yıllık çizelge olarak ekrana basar.
- Video Player: Video oynatıcıdır. Yöneticinin web adresini tanımladığı video dosyasını sayfada oynatır.
- Sound Player: Ses dosyası çalma bileşenidir. Yöneticinin web adresini tanımladığı ses dosyasını sayfada çalar.
- Donut Chart: Halka Grafiktir. Sistemde yüklü ve yöneticinin seçtiği veri setindeki nümerik liste içeriğini çember/halka grafik olarak ekrana basar. 3 boyutludur.
- Photo Slider: Fotoğraf galerisi bileşenidir. Yöneticinin seçtiği veri setindeki fotoğrafları efektler vasıtasıyla birer birer ekrana yansıtır.
- Password Area: Kullanıcının parola girebilmesi için örtülü parola alanı yerleştirir.
- URL Link: Yöneticinin tanımladığı web adresini link halinde ekrana basar.
- Embedded Flash Object: Gömülü Flash Objesidir. Yönetici tarafından adresi tanımlanan animasyon/oyun dosyasını ekrana basar.
- Annual Chart 3D: Yıllık Çizelge bileşeninin 3 boyutlu olanıdır.
- Single CheckBox: Tekli Seçim Kutusu ’dur. Yöneticinin tanımladığı içerik ve değeri yerleştirildiği yerde ekrana basar.
- Date Picker: Kullanıcının tarih işaretleyebilmesi amacıyla geliştirilen takvim bileşenidir.
- Multiple Option: Çoklu Radyo Buton bileşenidir. Yöneticinin seçtiği veri setindeki içeriklerin her birisini opsiyon olarak kullanır ve kullanıcı bir tanesini seçebilir.

- Multiple File Upload: Kullanıcı'nın birden fazla dosyayı sayfa üzerinde yükleyebilmesini sağlayan bileşendir.
- Formal Column: Form bileşeni altında dikey olarak kolon oluşturma amacıyla kullanılır. Bu sayede mobil bileşenler yan yana da yerleştirilebilir.
- Signature Box: İmza Bileşenidir. Kullanıcının mobil cihaz ya da fare yardımıyla kendi imzasını atabilmesini ve sisteme yükleyebilmesini sağlar.

Aşağıdaki tema bileşenleri diğer kontrolörler ile aynı yapıdadır ancak en üst seviyede kullanıldıkları için kontrol panelindeki araç kutucuğunda görünmezler. Bunlar, sayfa üretme kısımları tarafından otomatik olarak sisteme eklenirler.

- Form Supported Web-Page: Form sayfaları için kullanılan tema kontrolördür.
- Parallax Page: Durağan dizayn sayfaları için kullanılan kayar sayfa tema kontrolördür.

Sistemde varsayılan olarak bu iki tema mevcuttur ve uygulama tabanında örnek amaçlı geliştirilmiştir. Bu yapılara sadık kalarak farklı temalar geliştirilebilir ve sisteme yüklenebilir. Ancak geliştirilecek temalar mevcut iki sürükle-bırak destekli sayfa oluşturma editörüne yazılım katmanında tanımlanmalıdır. Bunun sebebi, sisteme yüklenen tema niteliğindeki kontrolör dosyalarının editörlere otomatik olarak tanımlanması için gerekli kısım olan tema ilişkilendirme aracının henüz geliştirilmemiş olmasıdır.

3.4. KISITLAR

Harpa İçerik Yönetim Sistemi, mimari olarak yapısal bir modeldir ve içerik yönetim alanında günümüz ihtiyaçlarını karşılayabilecek yapıdadır. Ancak kaynak yetersizliklerinden dolayı uygulama kısmında, modelin ispatlanması için yeterli olan temel kısımlar geliştirilmiştir. Bu sebeple tez uygulaması ancak simülatör düzeyinde çalışır. Sistemin doğru çıktılar verebilmesi için ancak doğru girdiler sağlanması gerekir. Hatalı bir içerik tanımlaması ya da veri seti, fotoğraf albümü gibi bir arşiv tanımlaması yapılması durumunda sistemin sıfırlanması gerekebilir. Ancak mevcut platform, içerik yönetim sistemi veri yaşam döngüsünü var olan sayfalar üzerinden asgari düzeyde tamamlayabildiği gibi, alan uygulaması olan 'Uluslararası Öğrenci Kayıt Sistemi' çıktısını doğru tanımlamalar ışığında başarıyla üretebilmiştir. Bu sayede pilot departmanın ihtiyacı karşılanırken, modelin sağlıklı çalışma yeteneği de teknik anlamda ispatlanmıştır.

Aşağıdaki kısımlar tez platform uygulamasında kapsam dışı bırakılmıştır:

- İçerik ve Sayfa Düzenleme
- İçerik ve Sayfa Silme
- Veri Seti ve Fotoğraf Albümü Düzenleme
- Veri, Seti ve Fotoğraf Albümü Silme
- Gömülü İçerik Çekme Eklentisi
- Proje Düzenleme ve Silme
- Rapor Düzenlene ve Silme
- Tema İlişkilendirme Aracı

SONUÇ

Yüksek Lisans Tezi, ‘Yeni Nesil İçerik Yönetim Sistemi Modeli’ konusu üzerine yürütülmüş olup, kuramsal çalışmanın yanı sıra çok kapsamlı bir yazılım geliştirme süreci ile uygulama tamamlanmış ve kontrol paneli üzerinden yapılan girdiler ile elde edilen çıktı uygulama alanına adapte edilmiştir.

Raporda, içerik kavramı ele alındıktan sonra İYS ve türevleri incelenmiş, güncel model analiz edilmiş, alternatif bir model önerisi yapılmış ve bu modelin taşıdığı isim olan ‘HARPA’ yapısal anlamda teknik altyapı ile ilişkileri ele alınmış, bu modelde geliştirilen uygulamanın ekran görüntüleri incelenmiş ve bu uygulamanın çıktısı olan ‘DEUZEM Öğrenci Kayıt Sistemi’ tanıtılmıştır.

Modüler ve yaygınlaştırılması kolay olan bu yapının, İçerik Yönetim Sistemleri ya da benzer web tabanlı sistemlerin geliştirme süreçlerini hızlandıracağı ve veri saklama veriminin artacağı öngörülmüştür. Ayrıca HARPA İYS geliştirilebilir bir platform niteliğinde olduğundan, kolay kullanımlı yapısının yanı sıra kendi içerisinde de farklı bileşenler üretildikçe alternatif sistemlerin önüne geçebilecek bir potansiyele sahiptir ve geniş kullanıcı kitlelerinin kolay internet sitesi tasarlama ihtiyacını doldurabilecek niteliktedir. Bu ise YBS alanında başlıca katkılardan biri olarak değerlendirilebilir.

Alan uygulaması bakımından, HARPA İYS ile üretilen örnek çıktı olan Uluslararası Öğrenci Kayıt Sistemi, bu sistemin orta ölçekli yapıların ihtiyaçlarına son derece esnek şekilde yanıt verebileceğinin ispatı niteliğindedir. Uygulamanın çıktıları, günümüzün mobil gereksinimlerine uygun olan kontrolör bileşenler üzerinden sağlandığından, kullanıcı arayüzü ve kullanıcı etkileşimi gibi faktörler açısından gelişmiş formlar ve durağan sayfalar üretilebildiği gözlemlenmiştir.

Diğer taraftan, yüksek lisans tezinde; içerik kavramının DIKW piramidi üzerinde gösterimi ve karşılaştırması yapılmıştır. Bununla birlikte, modelin standart İYSye göre işleyiş farklılıkları da ele alınırken, bu katkılar disiplinler arası bakış açısı ile çeşitli enformatik diyagramlar üzerinden incelenmiştir.

Sonuç olarak, İYSlere yeni bir konsept getirilerek teknolojik gelişmelere öncü niteliğinde bir model geliştirilmiş, bunun uygulaması örnek bir iş akışına adapte edilerek departmanın siber ihtiyaçları bu ürün kullanılarak asgari düzeyde sağlanmıştır.

Rapor edilen dokümantasyonda, gelecek akademik çalışmalara basamak teşkil edecek veriler ve yöntemler sonucu paylaşılmış, ticari ve teknolojik ürünlerin modeli kullanabilmesi amacıyla yazılım geliştirme sektörüne yönelik teknik detaylar aktarılmıştır.

Yönetim Bilişim Sistemleri Alanına Katkısı

Rapor edilen yüksek lisans tezinin, Yönetim Bilişim Sistemleri (YBS) ana bilim dalına olan katkıları farklı açılardan analiz edilmelidir.

Öncelikle basit proje yönetimi ve iş akışı destekli yeni bir modelin kurulması ve buna uygun bir platformun geliştirilerek çıktısı olan web sitesinin çalışan bir departmana entegre edilmesi en önemli katkılardan biri olarak kabul edilebilir. Zira proje yönetimi YBSnin en önemli araçlarından biridir ve içerik yönetimi gibi geniş spektrumlu bir alanın proje yönetimi desteği kazanırken dizayn çıktılarında geleneksel İYSlerin kalite çizgisi koruması önemli bir edinimdir. Diğer taraftan, YBS ana bilim dalı kapsamında ve İYS modellemesi üzerine yapılan ilk akademik tez çalışması niteliğinde olması destekleyici sebepler arasında kabul edilebilir.

Uygulamanın çıktıları, başvuru sistemleri ve dinamik iç hizmetler oluşturma faaliyetlerine daha uygun olduğundan, iş sektöründe kolaylıkla kullanılabilecek bir platform geliştirilmiştir. Kullanıcı etkileşimi üzerine yapılan çalışmalar YBSnin uygulama geliştirmelerde en hassas olduğu konulardan birisidir. Bu sebeple, uygulama içinde bulunan imza alanı gibi kontrolör eklentiler bu amaca hizmet eder niteliktedir. Özellikle, kare kod desteği ile birlikte gelen harita eklentisi, YBSnin Coğrafi Bilgi Sistemleri aracına katkıda bulunur. Bu gibi araçların kolay kullanılabileceği bir platform yaratmak ve yaygınlaştırmak YBS altında geliştirilecek başka çalışmalar için önemli bir kaynak niteliğinde kabul edilir.

KAYNAKÇA

Ackoff, R. L. (1989). "From data to wisdom". *Journal of Applied Systems Analysis.* 15(1): 3-9, http://faculty.ung.edu/kmelton/Documents/DataWisdom.pdf (3.05.2017).

Akçakaya, V. (2006). *Eğitimciler için yeni bir web aracı*. (Yüksek Lisans Tezi). Balıkesir: Balıkesir Üniversitesi, Bilgisayar ve Öğretim Teknolojileri Eğitimi, 88.

Altıntaş A. (2016). Microsoft Azure Servisleri ile Advanced Data Analytics Çözümleri, İstanbul, http://www.abdullahaltintas.com/wp-content/uploads/aa.pdf, (13.05.2017).

Avenoğlu, B. (2014). *A Context-Aware and Workflow-Based Framework for Pervasive Environments*. (Yayınlanmamış Doktora tezi). Ankara: Orta Doğu Teknik Üniversitesi Enformatik Enstitüsü.

CMS Review (2006). Seven Stages of the Content Lifecycle, http://www.cmsreview.com/Stages/, (19.04.2017).

Çatal, M. (2006). *Content Management System*. (Yayınlanmamış Yüksek Lisans Tezi). İstanbul: Haliç Üniversitesi, Fen Bilimleri Enstitüsü.

Çelik, S. (2011). İçerik Yönetim Sistemlerinin Eğitimsel Amaçlı Kullanımı. *5 th International Computer & Instructional Technologies Symposium Proceedings*, Düzenleyen Fırat Üniversitesi. 22-24 Eylül 2011.

Envato Market, Good Space - Responsive Minimal WP Theme by GoodLayers, https://themeforest.net/item/good-space-responsive-minimal-wp-theme/2278615, (11.02.2017).

Excolo. (2016). Between data, information and content, 2 Şubat 2016 http://excolo.com/information-v-content/, (23.05.2017).

Garrett, J. (2012). *The Elements of User Experience: User-Centered Design for the Web and Beyond:* Berkeley CA, Pearson Education.

Henriques G. (2014). Psychology Today A Wicked (WKID) Approach to Human Knowledge, https://www.psychologytoday.com/blog/theory-knowledge/201412/wicked-wkid-approach-human-knowledge/, (23.05.2017).

Hart, L. (2017). Web Content Management Comes of Age, http://www.cmswire.com/web-cms/web-content-management-comes-of-age/, (31.05.2017).

Hey, J. (2004). The Data, Information, Knowledge, Wisdom Chain: The Metaphorical link, Intergovernmental Oceanographic Commission - Oceanteacher: a Training System For Ocean Data and Information Management, http://www.citeulike.org/user/panda3000/article/7026249, (23.05.2017).

Innovative Architects, 7 Benefits of Using a Content Management System, https://www.innovativearchitects.com/KnowledgeCenter/basic-IT-systems/CMS-Benefits.aspx, (31.05.2017).

Jifa, G. (2013). "Data, Information, Knowledge, Wisdom and Meta-Synthesis of Wisdom-Comment on Wisdom Global and Wisdom Cities." Procedia Computer Science, 17(1): 713-719.

Kartal Karataş, E., Özen, Z., Üstünkaya, M.E., Zaim Gökbay, İ. ve Yarman, B.S. (2013). Organizasyonlarda İçerik Yönetim Sistemi Seçimi İçin Bir Karar Destek Sistemi Geliştirilmesi. *Marmara Üniversitesi Sosyal Bilimler Enstitüsü Öneri Dergisi*. 10(40): 155-162.

Kırbaş, İ. (2007). *PHP, MySQL ve XML tabanlı Türkçe Dinamik Web Sitesi İçerik Yönetim Sistemi: DyNA.* (Yayınlanmamış Yüksek Lisans Tezi). Kocaeli Üniversitesi Fen Bilimleri Enstitüsü.

Kızılboğa, İ. (2014). *Web İçerik Yönetim Sistemlerinin Sistem Yöneticiliği Üzerine Etkileri*. (Yüksek Lisans Tezi). İstanbul: Beykent Üniversitesi, Yönetim Bilişim Sistemleri Bilim Dalı, 4-25.

LeBlanc, S. (2011). When was the first web content management system (CMS) released?, https://www.quora.com/When-was-the-first-web-content-management-system-CMS-released/answer/Steve-LeBlanc/, (31.05.2017).

Lucid Charts (2017). What is a Data Flow Diagram, https://www.lucidchart.com/pages/data-flow-diagram/, (14.03.2017).

Marasco, J. (2005). *The Software Development Edge*: Pearson Education.

McIlwain, A. (2016). Custom website design or a CMS like WordPress? - The Garage. https://www.godaddy.com/garage/webpro/design/custom-website-design-or-a-cms-like-wordpress/, (01.03.2017).

McLean, S. (2012). Successful Writing, https://2012books.lardbucket.org/pdfs/successful-writing.pdf, (23.05.2017).

Mercier, O., Stevens, N. ve Toia, A. (2012). Mātauranga Māori and the data–information–knowledge–wisdom hierarchy: a conversation on interfacing knowledge systems. MAI Journal 2(1):103-116.

Mescan, S. (2004). Why Content Management Should Be Part of Every Organization's Global Strategy. *The Information Management Journal*. 54-57.

My2tech, CMS Nedir? Hangi CMS Türleri Vardır?, http://www.my2tech.com/cms-nedir-hangi-cms-turleri-vardir/, (13.05.2017).

North Patrol (2013). Future, present and history of CMSs as told by CMS expert Deane Barker, https://northpatrol.com/2013/11/11/future-present-and-history-of-cmss-as-told-by-cms-expert-deane-barker/, (31.05.2017).

Ozan, Ö. (2009). CMS, LMS, LCMS Kavramları. *Akademik Bilişim'09 - XI. Akademik Bilişim Konferansı Bildiri Kitabı*, Düzenleyen Harran Üniversitesi. 11-13 Şubat 2009.

Riley, J. (2017). Understanding Metadata: What is metadata, and what is it for?, National Information Standards Organization (NISO), Baltimore MD. http://www.niso.org/apps/group_public/download.php/17446/Understanding%20Metadata.pdf, (23.05.2017).

Samuels, J. (2013). Managing Information: Types of Content Management Systems Explained, TechWhirl, https://techwhirl.com/types-of-content-management-systems/, (23.05.2017).

Targowski, A. ve Tarn, J. (2007), *Enterprise Systems Education in the 21st Century:* Hershey PA, Information Science Publishing, 284.

Taşdelen, A. (2005). İleri ve Kurumsal Düzey .NET Uygulamaları, İstanbul: Pusula Yayıncılık.

TDK (1981). Türk Dil Kurumu, http://www.tdk.gov.tr/index.php?option=com_bts&view=bts&kategori1=veritbn&kelimesec=164024 (02.03.2017).

The University of Hong Kong (2013). What is a Portal?, http://www.its.hku.hk/news/ccnews100/portal.htm, (31.05.2017).

UX Magazine (2011). Content Strategy and UX: A Modern Love Story, http://uxmag.com/articles/content-strategy-and-ux-a-modern-love-story/, (27.04.2017).

WebHostingMedia (2016). Advantages & Disadvantages of Using a CMS for Building Your Website, http://webhostingmedia.net/advantages-disadvantages-using-cms/, (31.05.2017).

Wordpress Codex (2010). Database Description/2.9 « WordPress Codex, https://codex.wordpress.org/Database_Description/2.9/, (01.03.2017).

Yorulmaz, M., Yasemin, A. ve Şimşek, C. (2009). AkademiCMS: Akademisyenler için Web İçerik Yönetim Sistemi. *Akademik Bilişim'09 - XI. Akademik Bilişim Konferans Bildiri Kitabı* (ss. 1-6), Düzenleyen Harran Üniversitesi. 11-13 Şubat 2009.