

Kocaeli Üniversitesi
Bilgisayar Mühendisliği Bölümü
Yazılım Laboratuvarı I II.Proje
Web Uygulaması

Umut Kılıç-Göksel Kınalı

190201028@kocaeli.edu.tr-190201055@kocaeli.edu.tr

Projenin Özeti:

Yazılım laboratuvarı projesi için bizden kullanıcının sisteme dosya yükleyebileceği , bu dosyanın belli spesifik bilgilerinin alındığı , admin panelli bir web sitesi istendi.Admin ve kullanıcılar yüklenen dosyalara sorgu atarak ulaşması bekleniyordu.Kullanıcılar sadece kendi yükledikleri dosyalara ulaşırken , adminler herkesin dosyasına erişim sağlayabilecekti.

1.GİRİŞ:

Proje yaparken programlama dili olarak herhangi bir kısıtlama yoktu. Biz de python programlama dilini ve veritabanı için Django framework'ümü tercih ettik.Python dili nesneye yönelik bir programlama dili olması,dilin yazım kurallarının basit olması,yapılacak işlemlerin birkaç kod ile yazılabilmesi , istenilen isterleri sağlayabilecek kütüphaneler barındırması ve internette Python ile ilgili diğer dillere göre daha fazla kaynak olması bu programlama dilini seçme sebeplerimizdi.

İlk başta daha önce hiç pdf okutmadığımızdan pdf okuyan kütüphaneler ve yöntemleri araştırdık.pypdf2 gibi popüler pdf okuyan kütüphaneler, PDF okusa da Türkçe karakterler için olumlu çıktılar vermiyordu . Biz de daha çok araştırarak Pdfminer denen bir kütüphane bulduk ve projede pdfleri okurken bu kütüphaneyi kullandık.

Dosyaların içeriğine ulaşabildiğimizi gördükten sonra kullanıcın kayıt ve giriş işlemlerini yaparak projeye başladık.Kullanıcı giriş yaptıktan sonra yüklenen dosyaları tutmak için Django'nun veritabanındaki dosya alanlarını araştırdık ve projemize implement ettik.Daha sonra dosya , veritabanına yüklenirken içerdiği bilgileri almaya çalıştık.

Bunu temel olarak belli spesifik satırların yukarı veya aşağısına bakarak yaptık. Aldığımız bilgileri ilişkisel veritabanlarında tuttuk.

Yüklenen dosyaların içeriğini veritabanlarına attıktan sonra sorgulama işlemlerine başladık. Kullanıcı sadece kendi içeriğini görürken , admin tüm kullanıcıların yüklemiş olduğu dosyaları sorgulayabilecekti.Biz sorgularımızın daha fazla dinamik olması için sadece kullanıcın inputlara girdiği yerler , sorguları etkileyecek şekilde sistemi tasarladık.

Web sitesinin kullanıcı dostu olması gerekiyordu.Bunun için kullanıcının giriş yaptığındaki ana sayfaya bilgi güncellemesi ve çıkış yapması için butonlar koyduk.

2.YÖNTEM:

Projeyi başlatmak için ana dizinde

pip install -r requirements

komutunu çalıştırarak gerekli kütüphaneleri kurmalı ve manage.py olduğu dizinde

python manage.py runserver

komutun çalıştırarak sunucuyu başlatmalıyız.Sunucu başladıktan sonra <http://127.0.0.1:8000/> IP adresine giderek web sitesine ulaşabiliriz.Web sitesi ilk olarak Admin Sayfası ve Kullanıcı Sayfası ' na olmak üzere iki tane yönlendirme yapar. "Bu ekranlardan birinde adminin giriş ekranı bulunurken diğerinde ise Kullanıcının Giriş yapması veya kayıt olması için alt alta iki tane form bulunur.Eğer kullanıcı daha önceden kayıt olmuşsa kullanıcı adı ve şifresi veritabanında olduğundan login kısmından kullanıcı adı ve şifresini girerek sisteme giriş yapabilir.Eğer kullanıcının daha önce kaydı yoksa sisteme girerken

kullanıcının sisteme kayıtlı olmadığına dair bir uyarı yazısıyla karşılaşır.Bu durumda hemen alttaki forumdan kullanıcı adını ve şifresinin belirleyip kaydolurken,kaydolduktan hemen sonra giriş yap ekranından bilgilerini girerek sisteme giriş yapabilir.

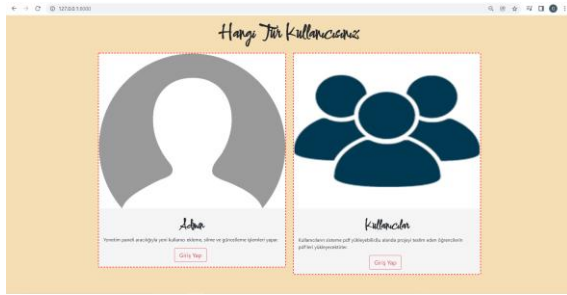
Giriş yaptıktan sonra kullanıcının sisteme dosya yükleyebileceği alan,bilgi güncelleme , sorgulama yapma alanına gitme ve çıkış yapma işlemi yapan butonlar bulunmaktadır.Kullanıcının sisteme dosya yükleyebilmesi için öncelikle 'Dosya Seçiniz' Butonuna basarak dosyanını seçmeli daha sonra 'Dosya Yükle' Butonuna basarak dosyasını yüklemelidir.

Kullanıcı bu ekranda isterse Bilgilerimi değiştir butonuna basarak şifre değiştirme işlemleri için diğer sayfa da gidebilir.Burada yeni bilgilerini belirleyerek bilgilerini güncelleyebilir.Ayrıca çıkış yapmak isterse de çıkış yap butonuna basarak giriş sayfasına gidebilir.

Son olarak da sorgulama yapmak için 'Sorgulama Yap' butonuna basabilir.Bu istediği şekilde birbiriyle ilişkili sorgular atabilir.

Deneysel Sonuçlar:

Uygulamanının Açılış:



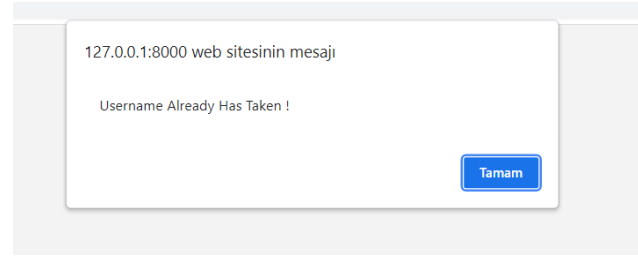
Uygulama başladığında iki ekran yan yana açılır.Soldaki ekranda admin'e yönlendirme bulunurken sağda kullanıcı giriş ekranına yönlendiren bir panel açılır.

Kullanıcı Giriş Ekranı :



Kullanıcı bu ekranda kullanıcı adını ve şifresini girerek oturum açabilir veya kullanıcı oluşturmak için gerekli bilgilerini vererek sistemde yeni bir kullanıcı oluşturabilir.

Kayıt Olurken Verilen Mesajlar:



Kayıt olunurken kullanıcı adının başka bir kullanıcıda olması veya şifrelerin eşlenmemesi halinde kullanıcıya alert basılır ve tekrar kayıt olma ekranına yönlendirilir.

Kullanıcın Doğru Giriş Yapma Durumu :



Kullanıcı doğru giriş yapınca 'Dosya Seç ' butonuna basarak dosyasını seçebilir ve 'Dosya Yükle ' butonuna basarak dosyayı sisteme yükleyebilir. 'Sorgulama Yap ' butonuna basarak sorgulama ekranına gidebilir. 'Bilgilerimi Değiştir ' butonuna basarak bilgi güncelleme ekranına gidebilir veya Çıkış Yap diyerek oturumunu sonlandırabilir.

Sorgulama Ekranı :



Sorgulama ekranında kullanıcın yüklemiş olduğu dosyaları yazara, derse , proje özelliklerine , anahtar kelimelere ve döneme göre filtreleyerek sorgulayabilmektedir. Burada her input alanı birbirini etkiler .



Ayrıca kullanıcı bir alanı doldurup sonra pencereyi kapatarak diğer alanları da sorgusuna dahil edebilir.

Sorgu Sonuç Ekranı :

Sorgulama yapıldığında ekrana sorgulanan dosyaların tek tek özellikleri basılır. Sayfanın altında bulunan 'Kullanıcı Ekranına Dön' butonu ile kullanıcı tekrar dosya yükleme ekranına dönebilir.

Admin Giriş Ekranı:

Giriş ekranından admin seçilmesi ile bu ekran açılır. Admin kullanıcı adını ve şifresini girerek admin içerik sayfasına gidebilir. Admin kullanıcısı ana dizinde:

python manage.py createsuperuser

ile oluşturulabilir.

Admin İçerik Ekranı:

Bu ekranda admin isterse 'Sorgu Ekranına Git' butonuna basarak sorgulama ekranına gidebilir veya manipüle etmek istediği tablonun ismine basarak, o tablodaki objeleri değiştirebilir, silebilir veya yenisini ekleyebilir.

Admin Sorgu Ekranı:

Sorgulama ekranında tüm kullanıcıların yüklemiş olduğu dosyaları kullanıcıya, yazara, derse, proje özelliklerine, anahtar kelimelere ve döneme göre filtreleyerek sorgulayabilmektedir. Burada her input alanı birbirini etkiler.

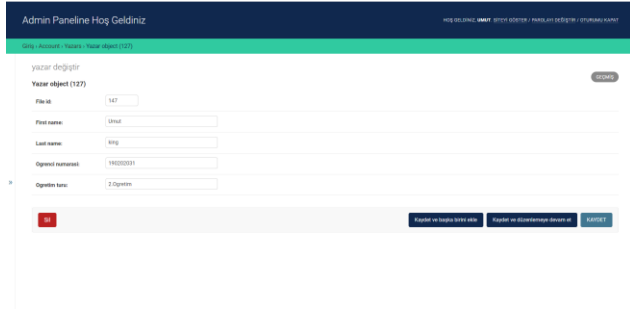
Admin Sorgu Sonuç Ekranı:

Sorgulama yapıldığında ekrana sorgulanan dosyaların tek tek özellikleri basılır. Sayfanın altında bulunan 'Admin Paneline Dön' butonu ile kullanıcı tekrar dosya yükleme ekranına dönebilir.

Admin Manipüle Ekranı :

Bu ekrana admin panelinden değiştirecek tablonun ismine basarak gidilir.

Burada sağ üstteki ekle butonu ile yeni obje eklenebilir veya değiştirilmek istenen obje seçilir.Obje seçilirse aşağıdaki ekran gelir.



Akış Diyagramı ektedir :

Akış diyagramı YAZLABdrawio.html'in içerisindeydir.

Sonuç:

Bu projede takım çalışmasının önemini çok iyi anladık. Projeyi işlere bölüp aramızda o işleri halletmeyi daha iyi kavradık. Proje esnasında gene karşılaştığımız zor durumlar olsa da yılmayıp üzerine gitmeyi ve bu zorlukların üstesinden geldik. Bu projeye birlikte Django dilinde kodlama yapmayı , HTML CSS Javascript ve Bootstrap ile Arayüz tasarımını, takım çalışmasını , zamanı verimli kullanmayı öğrendik.

Kaynakça:

<https://stackoverflow.com/questions/38157157/how-to-set-a-django-filter-to-select-all>

<https://stackoverflow.com/questions/6583877/how-to-override-and-extend-basic-django-admin-templates>

<https://www.youtube.com/watch?v=PtQiiknWUcl>

<https://www.youtube.com/watch?v=OTmQOjsl0eg>

<https://www.djangoproject.com/start/>

<https://www.youtube.com/watch?v=Yh8DMFPjuA0&list=PLGrTHqyRDvx5ZUs7h8mfGACFpnVipTNkA>

<https://www.youtube.com/watch?v=aHC3uTkT9r8>

<https://www.youtube.com/watch?v=uwVmWS1yJ1k&list=PLPrHLaaYVkhny4WRNp05C1qRl1Aq3WswH>

<https://getbootstrap.com/docs/4.0/components/buttons/>