Yazılım laboratuvarı 1.proje

Kütüphane Web Sitesi

Fazlı Sercan Yılmaz

190202113

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

**Özet**

Bu projede bizden istenilen öncelikle browser üzerinde çalışabilcek bir kütüphane sistemidir.Bu kütüphane sisteminde kullanıcılar kitap alabiliyor ve teslim süresi yaklaşmış ya da geçmiş kitaplarını teslim ediyorlar. Yöneticiler ise yeni kitap ekleyebiliyor,kullanıcılar üzerindeki kitapları görebiliyorlar ve tarihi ileriye öteleyebiliyorlar.

Kitap ekleme ve teslim etme kısımlarında her iki kullanıcı tipinde de opencv ile fotoğraf işlenerek isbn numarası ayıklanıyor.

Kitap alma kısımında ise kullanıcının üstündeki kitap sayısı ile teslim tarihleri kontrol ediliyor eğer kitap sayısı 3 den az ve teslim tarihleri geçmediyse yeni kitap almasına izin veriliyor.

**Giriş**

Bu projenin amacı,Web arayüzü olan bir kütüphane sistemi yapmaktır.

Projede frontend kısmı vanilla javascript ile yapılmış olup backend tarafı ise nodejs dili ile yapılmıştır.

Database ise mysql olarak belirlenmiştir eğer istenilirse connector dosyasından değişikliğe uygun yapıda yapılmıştır.

Ide olarak visual studio code kullanılmıştır.

**Kullanılan Teknolojiler**

Node.js,javascript,opencv.js,tesseract.js,express,cookie-parser,crypto-js,date-and-time,ejs,mysql,node-isbn,tmp,fs

**Tasarım Ve Algoritmalar**

**A)Front END**

**1-)Kullanıcı Front End**

**Ana sayfa (Kitaplar) (books.ejs)**

Kitaplar backend’den liste halinde alınarak ejs içinde for dönülerek bastırılır.

Kitaplar ısbn numarasına göre aranarak ekrana bastırılabilir.Eğer kitap bulunamaz ise uyarı mesajı gösterilir(“kitap bulunamadı...”,”Kitap bulundu...”).

Kitap almak için ise kitabın üstüne mouse ile gelip kitap al butonunu kullanarak alabilir.

Kitap alma durumunda 4 tane hata duruma vardır;

1-) Kullanıcının 3 kitabı varsa

2-)Kullanıcıda teslim süresi geçmiş kitap var ise

3-)Kitap önceden alınmış ise

hata durumlarına göre ekrana uyarı mesajı bastırılır.

Kullanıcı ISBN numarası kullanarak arama yapabilir.

Eğer kitap bulunursa kitap bulundu mesajı ekrana bastırılır ve sadece kitabın olduğu liste gösterilir

Kitap bulunamaz ise ekrana uyarı mesajı bastırılır

**Üyelik Sayfası (cart.ejs)**

Üyelik sayfasında tarih ve kulanıcının aldığı kitaplar ile alınma tarihi ve teslim tarihi listelenmektedir.

Eğer tarihi geçmiş kitap var ise uyarı mesajı gösterilmektedir.

**Kitap Teslim Sayfası(bookrefund.ejs)**

İade edilmek istenilen kitabın isbn numarası gözüken fotoğrafı sisteme yüklenir fotoğraf ön tarafda opencv ile işlenerek işlenmiş fotoğraf backend’e gönderilir.

4 tane hata durumu vardır;

1-) kitap bulunamaz ise

2-)kitap farklı bir kullanıcıda ise

3-)kitaba kimse sahip değil ise

4-)ISBN okunamaz ise

hata durumlarına göre ekrana uyarı mesajı bastırılır

**Kitap Teslim Sayfası(opencv)(refund.js)**

Yüklenen fotoğraf frontend de işlenerek iş yükü client tarafına bırakılmıştır.

Fotoğraf opencv kütüphanesi kullanılarak işlenmiştir.

Fotoğraf öncelikle gri tona çevili ve iki farklı foto işleme tekniği uygulanır birincisi fotoğraftaki gölge detaylarını çıkarır diğeri ise yazıları ikisini or metoduna soktuğumuzda sadece yazıların olduğu ve gölgelerin silindiği fotoğraf ortaya çıkar.

**1-)Yönetici Front End**

**Ana sayfa (Kitaplar) (books.ejs)**

Ana sayfa kullanıcılardakinden bir farkı yoktur tek farkı yönetici kitap alamaz.

**Yönetici Sayfası(admin.ejs)**

yönetici sayfasında tarih ile Kullanıcıların bilgileri bulunur.

Her kullanıcının aldığı kitaplar seçilen kullanıcıya göre listelenir.

Yönetici isterse tarihi öteleyebilir ötelemek için sadece gün değeri girmesi yeterlidir.

**Kitap ekle Sayfası(upload.ejs)**

Yönetici eklemek istediği kitabın bilgilerini,isbn numarası gözüken fotoğraf ve kapak fotoğrafı yükleyerek ekleyebilir.

Yükleme sırasında oluşabilcek 3 tane hata durumu vardır;

1-) Var olan kitap yüklenmeye çalışılıyor ise

2-)bilgiler eksik ise

3-)ISBN okunamaz ise

hata durumlarına göre ekrana uyarı mesajı bastırılır.

**Kitapyükleme Sayfası(opencv)(upload.js)**

Yüklenen fotoğraf frontend de işlenerek iş yükü client tarafına bırakılmıştır.

Fotoğraf opencv kütüphanesi kullanılarak işlenmiştir.

Fotoğraf öncelikle gri tona çevili ve iki farklı foto işleme tekniği uygulanır birincisi fotoğraftaki gölge detaylarını çıkarır diğeri ise yazıları ikisini or metoduna soktuğumuzda sadece yazıların olduğu ve gölgelerin silindiği fotoğraf ortaya çıkar.

**B)Back End**

**Index.js**

index.js backend yapısının maini olarak kabul edebiliriz içinde express modülü kullanılarak oluşturulmuş.

**Get /books**

Databaseden kimsenin almadığı kitaplar çekilerek ana sayfaya yapısı oluşturularak render edilir.

**Post /books**

Kullanıcının kitap almaya çalıştığında çalışan fonksiyondur. Öncelikle aranan kitap isbn numarası alınır ve database de aranır database de bulunursa kullanıcının kitap sayısı ve kitapların teslim tarihi kontrol edilerek anasayfaya yönlendilir.

**Post/Search**

Aratılan kitap databaseden bulunup anasayfada render edilir.

**Post /Auth**

Kullanıcı bilgileri frontend’den alınarak database de aranır ve kontrol edilir. Eğer mevcut ise kullanıcı tipine göre online kullanıcılara cookiesi eşlenir.

**Get/account**

Kullanıcı üzerindeki kitaplar database’den çekilerek kullanıcı sayfasında render edilir.

**Get/admin**

Kullanıcılarin listesi ve kitapları yönetici sayfasına render edilir.

**Post /Time**

Yönetici sayfasından çekilen gün server zamanına eklenir.

**GET /refund**

Kullanıcı kitap teslim sayfasını render eder.

**POST /refund**

Kullanıcının gönderdiği fotoğrafın isbn numarasını ocr ile çıkararak silme işlemi gerçekleştirir

**GET /Upload**

Yönetici kitap ekle sayfası render edilir.

**POST /Upload**

Eklenmek istenen kitabın bilgileri önce aratılır eğer databasede yok ise yeni kitap oluşturulur.

**GET /Login**

Login sayfasını render eder.

**Get /logout**

çıkış yapar.

**DATA.JS**

**booksprint()**

Alınmamış kitapları bastırır.

**KitapKontrol()**

Kullanıcı üstündeki kitapları kontrol ediyor.

**KitapAl()**

Kitapkontrol methodunuda kullanarak kullanıcı üstündeki kitapların tarihlerini ve sayısını kontrol eder eğer sıkıntı yok ise kitabı kullanıcının üzerine alır.

**KullanıcıKitapPrint()**

Kullanıcı üzerindeki kitapları döndürür.

**YoneticiKullanıcılar()**

Kullanıcılar ve kitaplarını döndürür.

**BookRefund()**

Kullanıcı üzerinden o kitabı siler.

**BookSearch()**

Aratılan kitabı döndürür.

**BookAuth()**

Kitap kontrolu yapar.

**OCR.JS**

**ISBNREADER()**

Base64 gelen fotoğraf bir temp dosyasına eklenir. Sonrasında ise dosya yolu ocr islemini yapan OCR’a yollanır sonrasında ise gelen string ISBNPARSER’a yollanır

**ISBNPARSER()**

String Satırlara göre ayrılır ve sonrasında her satırda ısbn regex’i aranarak isbn numarası çekilir.

**OCR()**

Tesseract ile ocr işlemi yapar.

**ZAMAN.JS**

**ZamanGuncelle()**

Server zamanını günceller.

**ServertimeOtele()**

Server zamanını öteler

**GetServerTime()**

Server zamanını döndürür.

**C)DATABASE**

3 Tane tablo bulunmaktadır

bunlar;

1-)Kullanıcılar

2-)Kitaplar

3-)Hangi kullanıcının hangi kitabı aldığı ve kitap tarihi

**Kaynakça**

[**https://www.w3schools.com/nodejs/**](https://www.w3schools.com/nodejs/)

[**https://www.tutorialspoint.com/nodejs/nodejs\_express\_framework.htm**](https://www.tutorialspoint.com/nodejs/nodejs_express_framework.htm)

[**https://docs.opencv.org/3.4/d5/d10/tutorial\_js\_root.html**](https://docs.opencv.org/3.4/d5/d10/tutorial_js_root.html)

[**https://www.w3schools.com/nodejs/nodejs\_mysql.asp**](https://www.w3schools.com/nodejs/nodejs_mysql.asp)

**[https://github.com/naptha/tesseract.js#tesseractjs](https://github.com/naptha/tesseract.js" \l "tesseractjs)**

[**https://www.npmjs.com/package/cookie-parser**](https://www.npmjs.com/package/cookie-parser)

[**https://www.npmjs.com/package/date-and-time**](https://www.npmjs.com/package/date-and-time)

[**https://www.npmjs.com/package/ejs**](https://www.npmjs.com/package/ejs)

[**https://www.npmjs.com/package/tmp**](https://www.npmjs.com/package/tmp)