**OCR TABANLI FİŞ TANIMA**

*MEHMET ALİ İNAL*

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

KOCAELİ ÜNİVERSİTESİ

mehmetaliinal10@gmail.com

15020263

**ÖZET:** Projenin amacı; kullanıcı GUI’si kullanarak , OCR tabanlı fiş tanıma uygulaması tasarlamak.JPEG veya PNG uzantılı fiş formunda bir görüntüden veri çekip veri tabanına aktarmak , veri listelemek ve filtrelemek.

Projenin hazırlık aşamasında önceliğim verilen verileri istenilen sonuçlara göre düzenleyip buna bağlı olarak kendime bir rota belirlemek oldu. Belirlediğim bu rotaya göre projeyi oluşturmaya başladım.

Projeyi Java yazılım diliyle yazmaya karar verdim. Bu projeyle ilgili olarak önceliğim Eclipse uygulamasına gerekli kütüphaneleri eklemek oldu. Sonrasında buna bağlı olarak gerekli kütüphaneleri daha rahat import edebilmek için projeyi Maven eklentisine dönüştürdüm. Kullanıcı GUI’si oluşturabilmek için ayrı bir class açıp Java’nın swing özeilligini projeme import ettim. Jframe kütüphanesini kullandım.

Kullanıcı ara yüzü veri çekme, listeleme ve filtreleme özelliklerine sahip. Projenin ana sınıfında veri tabanı bağlantıları ve görüntü işleme kodları bulunuyor. Veri tabanı işlemleri için Mysql veri tabanını kullandım. Görüntü işleme işlemi için OCR OpenCV ve Tesseract kütüphanelerini kullandım. İşlediğim görüntü içindeki verileri firma ismini baz alarak veri tabanına aktardım. Ara yüzü kullanarak veri tabanı içindeki verileri listeledim ve firma ismi ve tarih bazlı filtreledim.

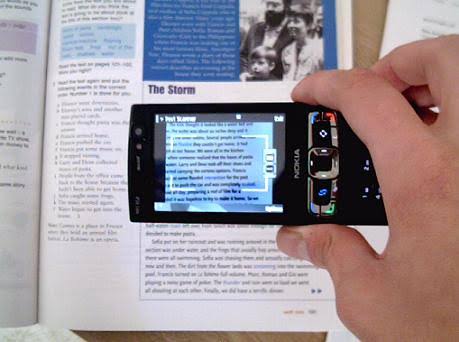
**GİRİŞ :** Optik Karakter Tanıma’ veya ‘Optik Karakter Okuyucu’ olarak adlandırılan bu yazılımlar, taranan bir belgede, bir fotoğrafta, el yazısıyla yazılmış bir metinde yer alan yazıların bilgisayar ortamına aktarılmasını sağlamak için geliştirilmiş yazılımlardır. Normal şartlarda bilgisayara tarayıcı vasıtasıyla aktarılan grafik dosyaları, bilgisayar tarafından resim olarak algılanır. Bu grafik dosyadaki yazılar ‘OCR’ yazılımları tarafından çözümlenip metin dosyasına çevrilebilir. Böylelikle klavye kullanmaya gerek kalmadan metin bilgisayara aktarılacaktır. ‘OCR’ ile metinleştirilen veriler, üzerinde düzenleme yapılabilecek verilerdir. Metin dosyaları, bilgisayarda görüntü dosyalarından daha az yer kaplar. ‘OCR’ yazılımlarının hata oranı sıfır olmasa da, günümüzde kullanımı oldukça yaygınlaşmıştır. Fotoğrafı metne dönüştürme programı denildiğinde kastedilen şey ‘OCR’ yazılımlarıdır.

‘Optik Karakter Tanıma’, en başta görme engelli kullanıcılar için metinlerin okunmasını sağlayacak cihazlar üretilmesi için ortaya çıkmıştır. Emanuel Golberg, 1914 yılında karakter okuması yapan ve onları telgraf kodlarına dönüştüren bir cihaz geliştirmiştir. Bu cihazı takip eden ‘Optophone’ ise, avuç içi büyüklüğünde bir tarayıcı olup, özel harfleri ve karakterleri de okuyabilen bir cihaz olarak ortaya çıkmıştır.

Optik Karakter Okuyucular’, akıllı telefon ve gözlüklerin ortaya çıkması ile yaygınlaşmış ve kullanılabilmesi için çeşitli uygulamalar geliştirilmiştir. Cihazların üzerindeki fotoğraf makineleri ile çekilen fotoğraflar veya belgeler, çok kısa bir süre içerisinde metin formatına dönüştürülebilmektedir.

İki temel ‘OCR’ algoritması bulunmaktadır. Matris eşleştirme, desen tanıma olarak da tanımlanabilecek olan bu algoritma, görüntüyü piksel-piksel tarayarak benzer bir dosya ile karşılaştırır ve metne döker. Bu algoritma, eski daktilo metinlerinde doğru bir biçimde çalışmaktadır. Ancak günümüzdeki yazı tiplerinde iyi sonuç vermemektedir.



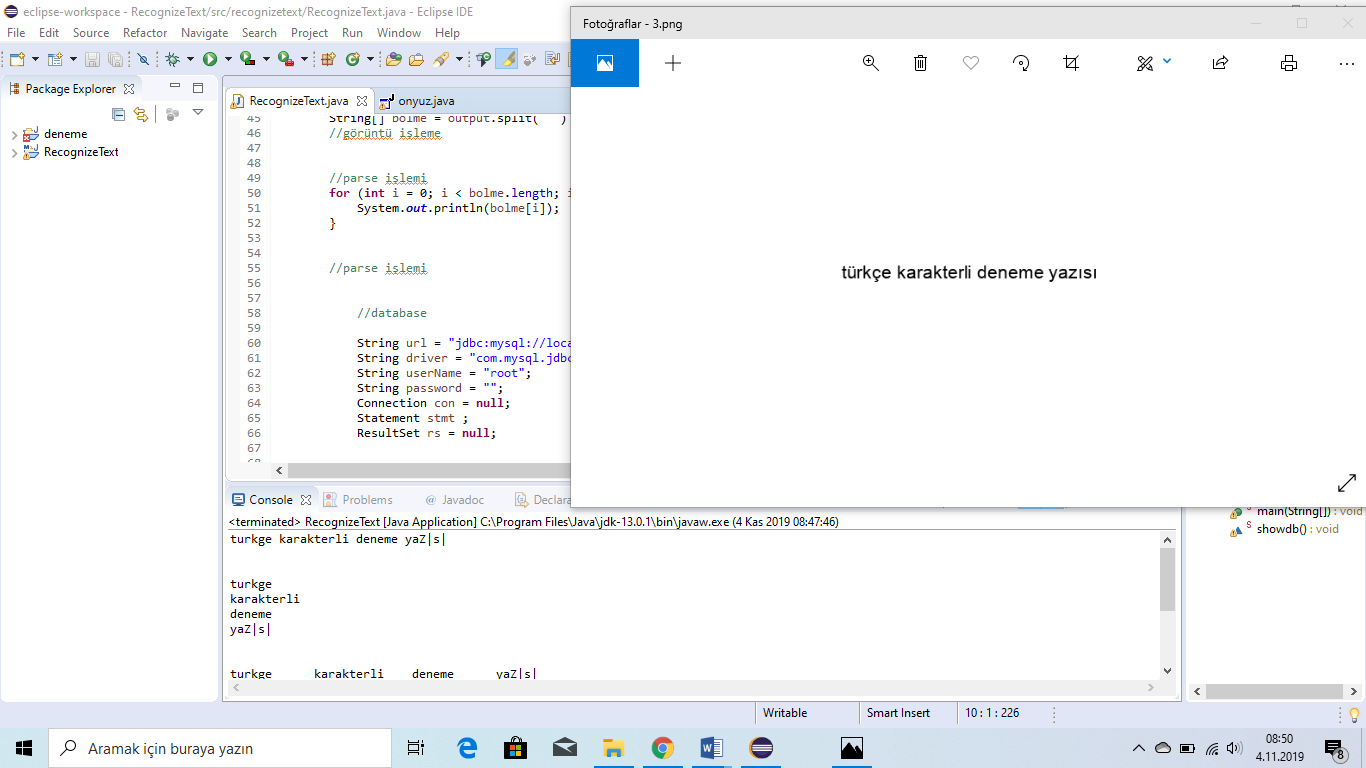


**TEMEL BİLGİLER :**

Yararlandığım teknolojiler sırasıyla;

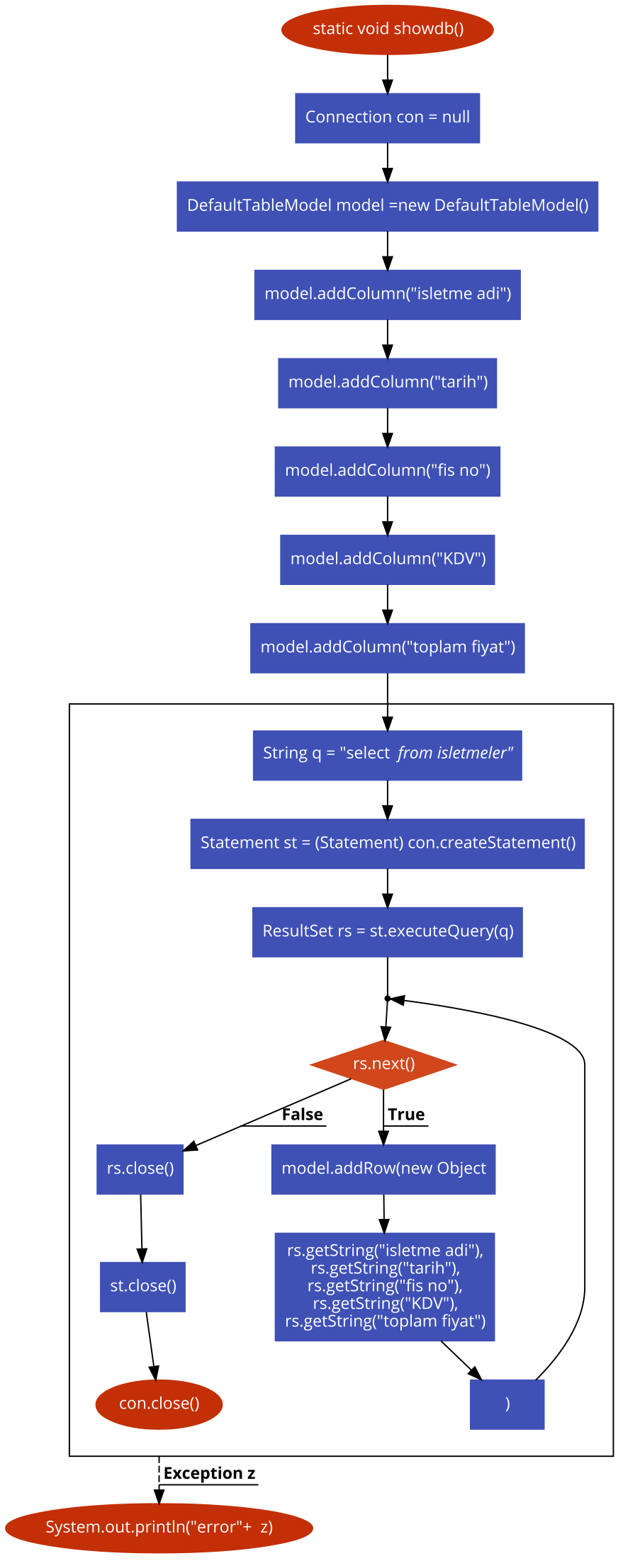
* Eclipse
* NetBeans
* Google
* GMail
* Youtube
* Microsoft Office

**Ekran Görüntüleri:**

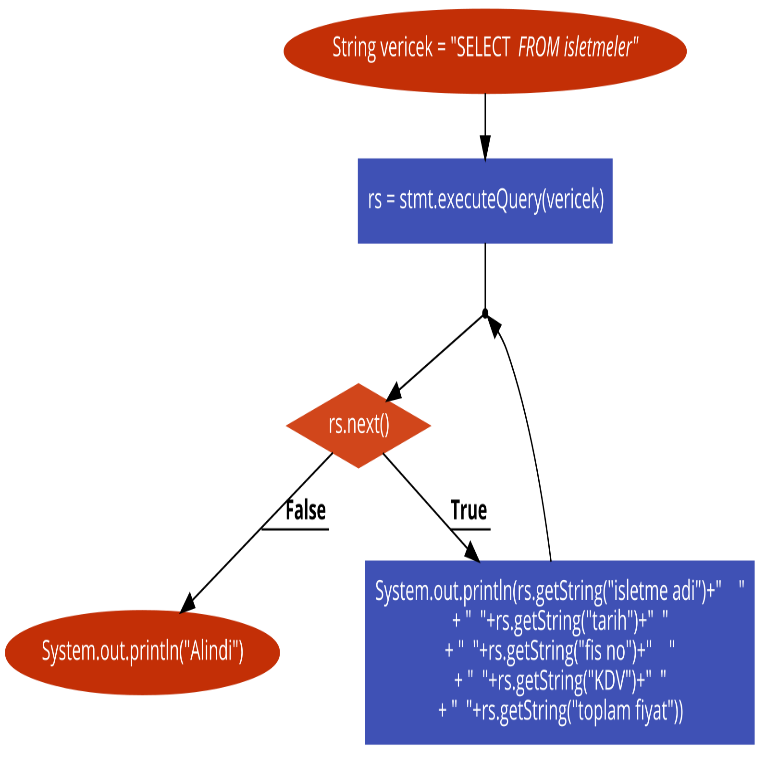
****

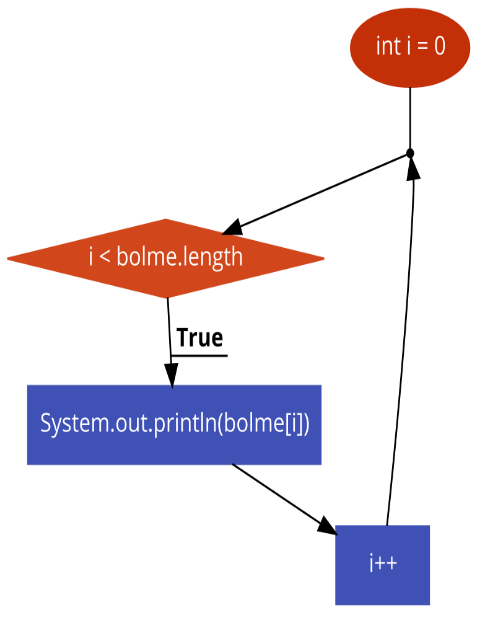
**AKIŞ DİYAGRAMLARI** :

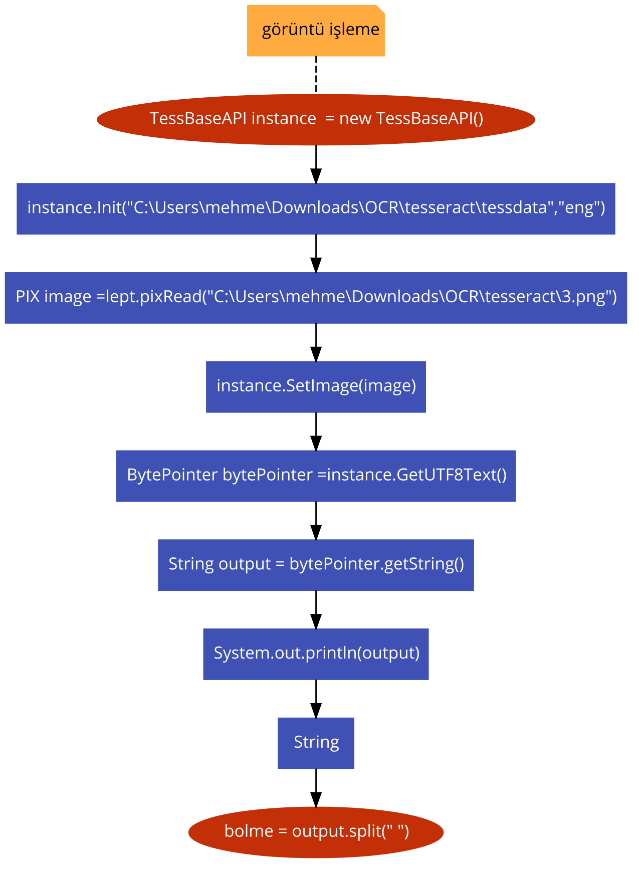
**Showdb Fonksiyonu:**



**Veri Çekme:**

**

**Parse İşlemi :**

**Görüntü İşleme :**

**KAYNAKÇA :**

**[1]** https://github.com/tesseract-ocr/tesseract/wiki/Downloads

**[2]** https://github.com/tesseract-ocr/tesseract/wiki/4.0-with-LSTM#400-alpha-for-windows

**[3]**https://www.youtube.com/watch?v=6sdhCwfP-lg

**[4]**https://stackoverflow.com/questions/40236472/how-do-i-separates-text-region-from-image-in-java

**[5]** https://www.geeksforgeeks.org/tesseract-ocr-with-java-with-examples/

**[6]** https://stackabuse.com/tesseract-simple-java-optical-character-recognition/

.