

FERDİ SÖNMEZ

BİLGİSAYAR MÜHENDİSİ

PROJE İÇERİKLERİ

PROJE KAYNAK KODLARI: <https://github.com/ferdisonmez>

Sayfa Numarası	Programlama Dili
2-3	C Programlama
4	C++ Programlama
4	Verilog
5	Docker,Python,Mule(DevOps)
5	Common Lisp
6	Java
7	C#--MsSql
8	Prolog
8-9	Splunk
11	Derin Öğrenme & Python
12-13	Makine Öğrenmesi & Python
14	ASP.NET

İletişim: ferdi.17810@gmail.com

0544 378 55 74

Mayıs 2022

C Programlama Dili

Proje1:Kullanıcıdan birden çok sayı (veri) alarak bu sayıların hepsinin ortalamasını bulup bir sayı doğrusu ile bu noktaları ayırır. Daha sonra ayrılan bu noktaların hepsinin orta noktasını bulabiliriz.

Proje2:Dosyadan okunan karışık bir mantıksal ifadenin sadeleştirilip kullanıcıdan alınan verilerin bu mantıksal ifade kapsamında değerlendirilip sonucun kullanıcıya bildirilmesi sağlandı.

Proje3:Kullanıcı tarafından 2 farklı dosya hazırlanır.Bu dosyaların birisinde karışık kelimeler diğesinde ise bu karışık kelimeler arasında bulunması istenen kelimeler bulunmaktadır.Aranan kelimeyi hem yatay hem de dikey olarak aramaktadır.Kelime bulunursa koordinatları kullanıcıya söylenmektedir.

Proje4: Program 2 dosya sahip olması gerekmektedir. Dosyaların ilkinde belirli bir polinom ifadesi bulunur. Diğer dosyada bu polinomda hesaplanması istenen değerler bulunmaktadır.Bu polinomda hesaplanan sonuç diğer dosyaya yazılır.

Proje5: Kullanıcı tarafından verilen metni dosya sistemi şeklinde sonuç dosyasında gösterir.

```
(Ali,(c,Beee,ec ee),K,Dayi,e,(f,(d,Ali)))
```

- This tree can be visualized as follows:

```
-Ali
--c
--Beee
--ec ee
-K
-Dayi
-e
--f
---d
---Ali
```

Proje6:Kullanıcı program ilk çalıştığında istediği bir nesnenin ismini ve o nesnenin hangi parçalardan oluştuğunu ve bu parçaların fiyatlarını girer. Daha sonra bu nesnenin toplam fiyatını hesaplar.

	Price
Bicycle x 1	
Wheel x 2	
Rim x 1	60.0
Spoke x 28	12.0
Hub x 1	
Gear x 6	25.0
Axle x 1	
Nut x 7	0.15
Bolt x 5	0.1
Frame x 1	
Rearframe x 1	170.0
Frontframe x 1	
Fork x 2	21.5
Handle x 1	12.0

Proje7:Kullanıcı 2 farklı dosya hazırlanır. İlk dosyada mantıksal bir devre bulunur.Diğer dosyada bu devre için input değerleri verilir.Bu devre ve input değerlerini hesaplayıp sonucu farklı bir dosyaya yazılır.

```
INPUT a b c d
AND and1 a b
OR or1 and1 c
NOT n1 d
FLIPFLOP f1 n1
AND a2 or1 f1
```

- `input.txt` has the following content:

```
1 1 0 1
1 0 1 0
1 1 1 0
```

Proje8:512x512 verilen bir matris içerisinde belirli bir sayı değeri verilir. Bu sayı değeri bu matris için su değeri kabul edilir.Bu sayı değerinin üzerinde kalanlar ada olarak değerlendirilir.

`output.txt`

- The first line is an integer (number of islands)
- Each line after the first line holds coordinate information of a point on an island. (single point for each island)
- Example: Here there are 5 islands. (`#...` are comments, which are not printed.)

```
5
x_coordinate1 y_coordinate1      #coordinate of a point which is on island1
x_coordinate2 y_coordinate2      #coordinate of a point which is on island2
x_coordinate3 y_coordinate3      #coordinate of a point which is on island3
x_coordinate4 y_coordinate4      #coordinate of a point which is on island4
x_coordinate5 y_coordinate5      #coordinate of a point which is on island5
```

C++ Programlama Dili

Proje1:Kullanıcıdan bir sayı değeri istenir. Bu alınan değer sonucunda bir fibonacci pramidi oluşturulur.Daha sonra bu pramidin en yüksek değeri ve bu yüksek değerın koordinatları bulunur.

```
• User enters the following number:
9
• Your program prints the following:
1
1 1
2 1 2
3 2 2 3
5 3 4 3 5
8 5 6 6 5 8
13 8 10 9 10 8 13
21 13 16 15 15 16 13 21
34 21 26 24 25 24 26 21 34
55 34 42 39 40 40 39 42 34 55
Biggest number: 55
Heart of the pyramid: 6 6
```

Proje2:Öğrenci otomasyon sistemi yapıldı.Bu sistemde hem öğrenciler hem de yöneticiler için farklı modüller bulunmaktadır.Öğrenciler derslerin saatlerini derslerin sayısını görebilir,istediği bir derse kaydını yapabilmektedir.Ayrıca öğrenci sisteme kayıt olacaktır.Dersten kaydını silecektir.

Yönetici modülünde dersin hocasının sisteme ders kaydını yapabilir.Bu dersin bütün bilgileri sisteme kaydederler.Bu derslerin kaydını yaparken sistem aynı saatte dersin olup olmadığı kontrol edilir.

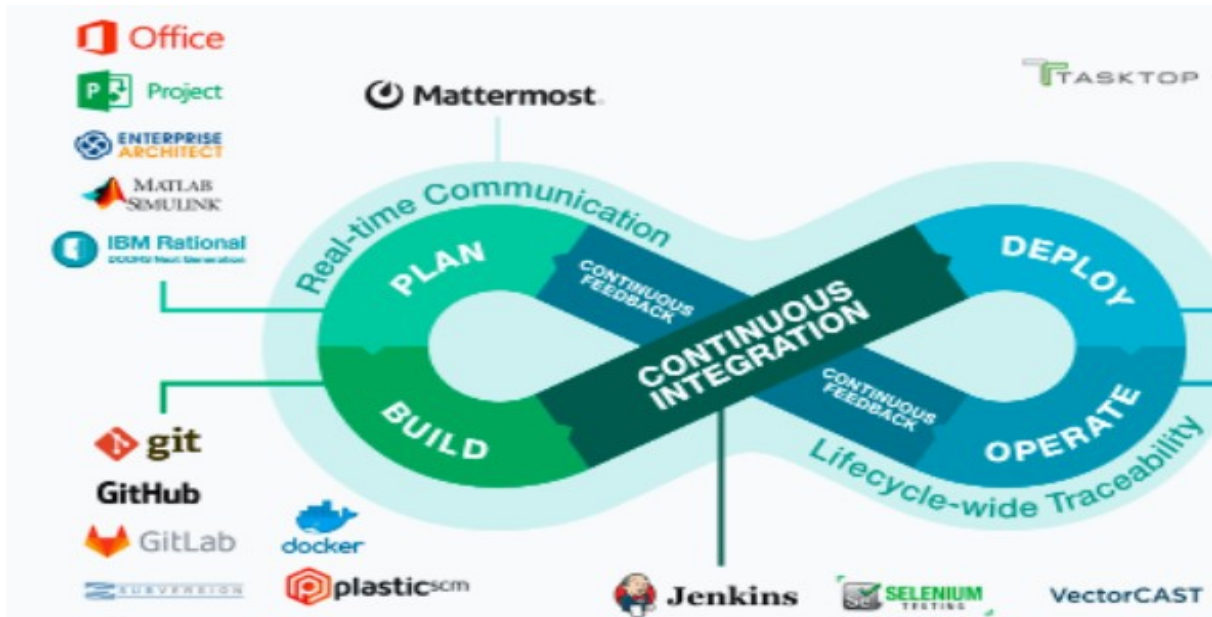
Proje3:Üniversitede çalışan tüm personelin yaptığı işler ve bu işler sonucunda üniversiteye yapılan katkıları ve kişinin değerlendirilmesi yapılır.Diğer modülde bu bunların hepsi farklı bir üniversite sistemine uygulandı.

Verilog:

32 bit single-cycle bir işlemci yapıldı.

Docker, Python Ve Mule(DevOps)

(Continuous integration) sürecinde yazılan java kodunun (mule esb) ile iletişimi sağlanmıştır.Yazılan kod mule ile yazmış olduğumuz script(python kodu) geldiğinde localde çalışmakta olan docker ile ayağa kaldırılmıştır.



Common Lisp

Gpp adında bir dil tasarlayıp bu dile ait özellikleri belirlenen özelliklere değerlendirme yapılır.Aslında bir nevi compiler diyebiliriz.

```
[1]> ;; helloworld.g++
[2]> (+ 10 10)
20
[3]> (list 1 2 123)
(1 2 123)
[4]> (set x (** (- 6 4) (/ 32 (* 4 2))))
16
[5]> (set y (concat (list 6 7 3 6) (list 1 2 5 7)))
(6 7 3 6 1 2 5 7)
[6]> (set z (append 4 (list 1 2 5 7)))
(4 1 2 5 7)
[7]> (if (and true false) (list 5 6) (list 6 78 9))
(6 78 9)
[8]> (if (or true false) (list 5 6) (list 6 78 9))
(5 6 0)
[9]> (not (and true true))
FALSE
[10]> (not (or false false))
TRUE
```

Java Programlama Dili

JAVA CORE

Proje1:Kargo sistemi tasarlanmıştır. Bu sistemde kargo şirketinde çalışan yönetici ,kargo görevlisi ,şube çalışanına ait ayrı modüller bulunmaktadır.Her modüle ait farklı özellikler bulunmaktadır.Ayrıca müşteri sistemde kargosunun nerede olduğunu görebilmektedir.Teslim edildiğinde müşteri bilgilendirilir.

Proje2:SimpleTextEditör yazıldı. String üzerinde istenilen işlemleri yapabilmektedir.

Proje3:Deque yapısı ders kapsamında yeniden tasarlandı.

Proje4:Online satış mağaza sistemi uygulanmıştır.Sistem içerisinde satıcılar ve müşteriler bulunmaktadır.Excel dosyası olarak satıcı ve ürünleri sistemde kategorilere ayrılıp gerekli veri yapıları ile depolanmaktadır.Daha sonra customerlar istedikleri kategorilere göre hızlıca istediği ürünlere ulaşabilmektedir.

Proje5:Dosya sistemi oluşturuldu.Verilen string üzerinde işlemlerle bilgisayarda directory ve file oluşturuldu.

Proje6:Farklı sıralama algoritmaları verimlilikleri değerlendirilip grafikleştirildi.

Proje7:Yazılım satış ve alış sistemi geliştirildi(google play benzeri).Sistemde iki farklı kullanıcı bulunmaktadır. Müşteri sistemde yazılım arayıp satın alabilir.Yönetici yazılımı ekler,siler ve güncelleyebilmektedir.

Proje8:Online beyaz eşya satış sistemi tasarlanmıştır. Sistem içerisinde çok fazla kullanıcı modülü bulunmaktadır. Mağaza şubeleri, şube çalışanları,müşteriler,online satış ve yönetici modülleri bulunmaktadır.Her modülün kendi içerisinde yapabileceği özellikler bulunmaktadır.

SPRING

Proje1:Dünyada ki tüm ülkelerin ve şehirlerinin özelliklerinin bulunduğu karmaşık bir json formatlı dosya java tarafında parse edilip spring yapısı ile web portunda sunucu olarak isteklere cevap vermektedir. Frontend tarafında react ile bu ülke ve şehir özellikleri aranabilmektedir. İstenilen ülke özellikleri isim,başkent ve bayrağı gösterilmektedir.

C# ve MsSql

Proje1:Tüm ticari işletmelerin kullanabileceği bir ticari otomasyon sistemi tasarlanmıştır. İçerisinde 9 farklı modül bulunmaktadır. C# programlama dili ve Mssql kullanılmıştır.

Ürünler→Veritabanında bulunan firmanın tüm ürünleri vardır ve bu ürünler tablosunu gerekli düzenlemeleri yapmaya olanak tanır.1-Ürün kaydet,2-Ürün silme,3-Ürünleri düzenleme işlemleri yapılabilmektedir.

Müşteriler→Firma ve bu firmanın çalışanlarının kaydedip, silinip ve bunların gerekli düzenlemelerinin yapılacağı modüldür. Müşterilerinin kaydedilip, silindiği ve bilgilerinin değişmesi gerekenlerin değiştirilebildiği işlemleri içerir.

Firmalar→Bizim kendi firmamız diğer firmalara da tedarik yapmaktadır. Bu tabloda bu tedarik yapılan firmalar bulunmaktadır. Uygulama arayüzün de firma ekleyip, silip ve düzenleyebilmekteyiz.

Personeller→Firmada çalışan personellerin listesi tutulmaktadır. Firma yetkilisi personel ekleyebilir, silebilir ve düzenleyebilmektedir.

Giderler→Firmada yapılan ekstra giderler tutulmaktadır. Elektrik, su , doğalgaz gibi kalemleri içerir.

Kasa→Firmanın ürün satışı ve diğer giderler için harcanan para miktarını gösterir.

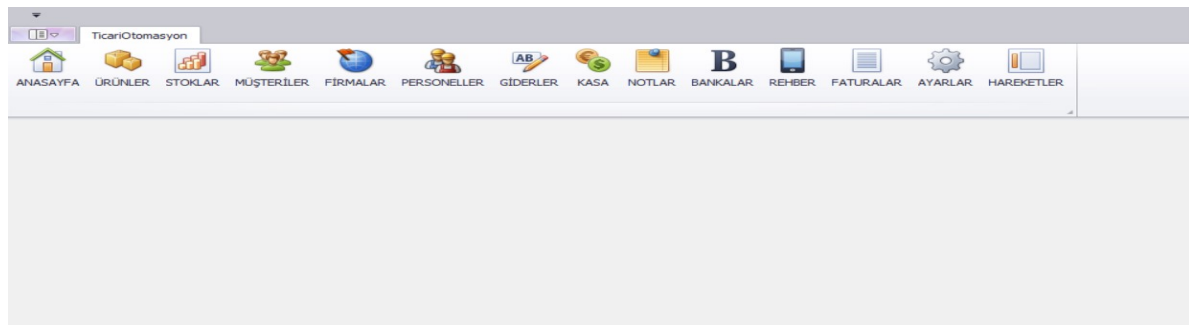
Bankalar→Firmanın ve işbirliği içinde bulunan firmaların banka bilgileri tutulmaktadır. Uygulama arayüzünde banka ekleyip çıkarılabilir, banka bilgileri silinebilir ve banka bilgileri düzenlenebilmektedir.

Rehber→Veri tabanında bir tablo değildir ama arayüzde firmaya ait bilgileri görüntülemek için kullanılır.

FirmaHareketler→Firmanın yapmış olduğu tüm işlemlere ait tablodur. Firmanın ürünlerine, adet miktarına, personellerine ve faturalarına ulaşılabilir.

MüşteriHareketler→Müşterinin yapmış olduğu işlemlere ait tablodur. Müşterinin almış olduğu ürün, adet, satış yapan personel, faturaID ulaşılabilir.

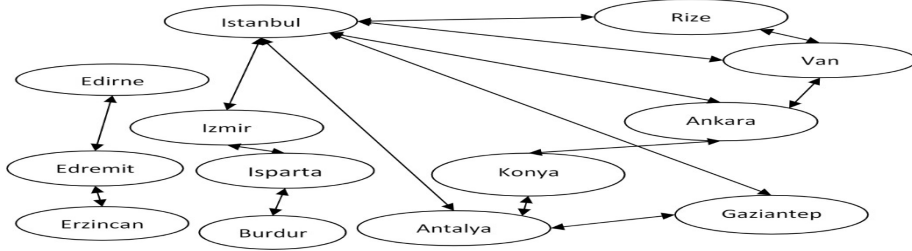
ARAYÜZ:



Prolog Programlama Dili

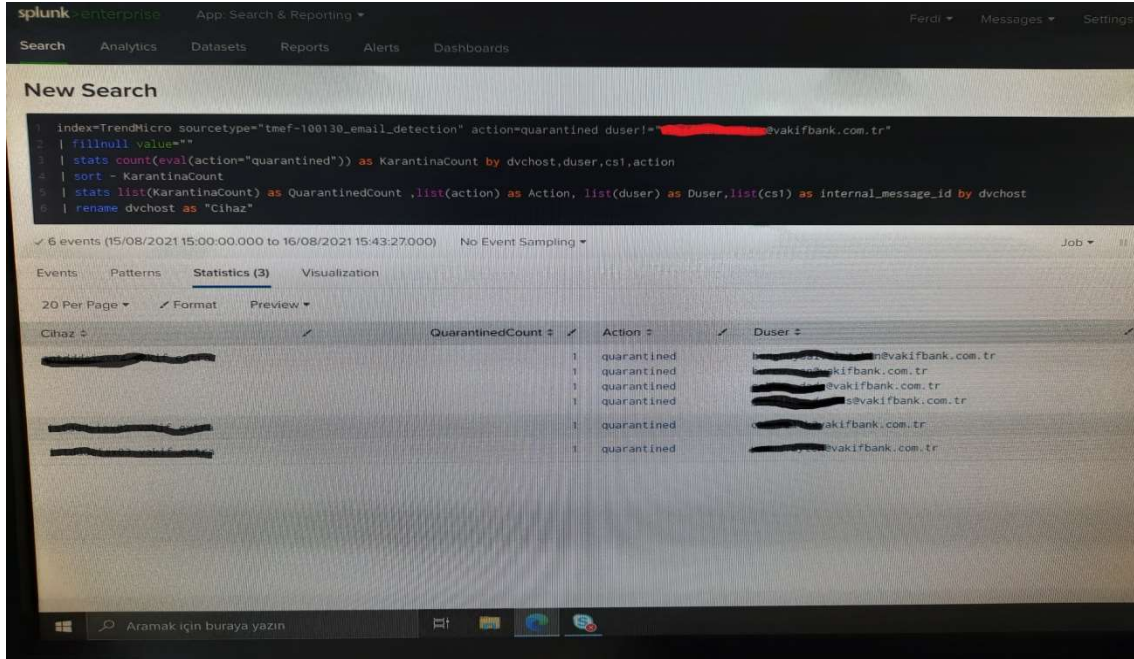
Proje1:Otobüsler için mesafe hesaplayan ve rotayı bulan bir sistem yapılmıştır.

Proje2:Öğrenci otomasyon sistemi yapılmıştır. Çakışan dersleri ve sınıflar bulunmaktadır.



SPLUNK

Proje1: E-mail cihazlarına gelen atak ve hangi kullanıcıya gönderildiğinin loglar üzerinden tespit edilip ve bu mail hakkında bilgi toplanmıştır. E-mail detection cihazlarında kullanıcılara gelen mailler kontrol edilip gerekli kontroller yapıldıktan sonra karantinaya alınmaktadır.



Proje2: DDOS Atak Tespiti:

Loglar üzerinde yapılan istek miktarı 100.000 kb dan fazla olan atakları listeleyen ve gerektiğinde alarm oluşturan bir projedir.

New Search

```
index=main service=PING | stats sum(rcvdbyte) AS download by dstip | where ( download > 100000 )
```

Proje3:Atak Öncesi Açık Port Arama Alarmı:

Belirli bir ip adresinden sürekli olarak aynı destination ip adresine yapılan istekler 20 den fazla olursa alarm oluşturup listeleyen splunk komutu aşağıda verilmiştir

New Search

```
index=main | dedup srcip dstip service | stats count by srcip dstip | where ( count > 20 )
```

Proje4: Bute Force Atak Tespiti:

Kullanıcının sürekli “root” olarak “login” olma isteği “failed” hatası alırsa ve bu miktar belirlenenden fazla olanları listeleyen ve alarm oluşturan bir projedir.

New Search

```
index=main action=login status=failed | stats count by user | where count > 1 AND (user="root")
```

Proje5: Kullanıcı Veri Kullanımını Kısıtlama:

Eğer kullanıcı belirlenen miktarda(300Mbyte) veri indirirse uyarı oluşturacak ve kullanımını kısıtlayacak kod parçası aşağıda verilmiştir.

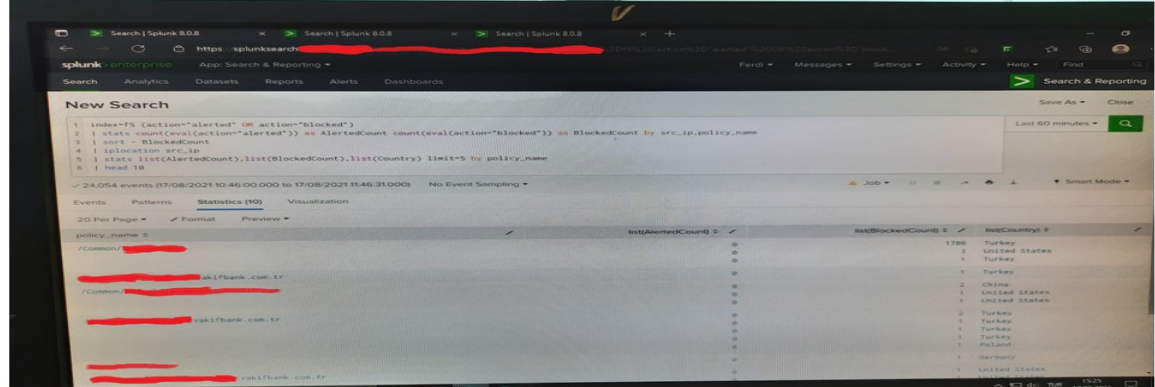
New Search

```
index=main | stats max(rcvdbyte) by srcip dstip sessionid  
| rename max(rcvdbyte) AS Download_Byte  
| eval Download_MByte=Download_Byte/1024/1024  
| eval Download_MByte2=round(Download_MByte,2)  
| fields - Download_Byte - Download_MByte  
| sort - Download_MByte2  
| where (Download_MByte2>300)
```

Proje6: WAF(Web Application Firewall) cihazına istek yapan ip adreslerinin bu cihazda actionunun("blocked" veya "alerted") nasıl engellendiğini, miktarının ne kadar olduğunu ve hangi ülkeden geldiğini sorgulamamızı ve tablo şeklinde gösterilmiştir.

Kod:

```
{
  index=f5 (action="blocked" OR action="alerted")
  | stats count(eval(action="alerted")) as AlertedCount
  | stats count(eval(action="blocked")) as BlockedCount by src_ip,policy_name
  | sort - BlockedCount
  | iplocation src_ip
  | stats list(AlertedCount),list(BlockedCount) limit=5 by policy_name
  | head 10
}
```



policy_name	listAlertedCount	listBlockedCount	listCountry
/Common/	0	1786	Turkey
/Common/	0	3	United States
/Common/	0	1	Turkey
/Common/	0	1	Turkey
/Common/	0	2	China
/Common/	0	1	United States
/Common/	0	1	United States
/Common/	0	2	Turkey
/Common/	0	1	Turkey
/Common/	0	1	Turkey
/Common/	0	1	Poland
/Common/	0	1	Germany
/Common/	0	1	United States

Derin Öğrenme & Python

Gerçekleştirilen bu çalışmada derin öğrenme ve görüntü işleme teknolojisi ile şehir içi trafiğe girmesi yasak olan araçların tespiti ve bu bilginin gerekli yerlere iletilmesi amaçlanmıştır.

Derin öğrenme yöntemlerinin görüntü işleme alanındaki uygulanabilirliği ve uygunluğu göz önüne alınarak çalışmada temel olarak derin öğrenme yöntemleri ve görüntü işleme algoritmaları kullanılmıştır. Bu çalışmada görüntü işlemede başarılı performans gösteren CNN algoritmasını kullanan YOLOV3 modeli kullanılmıştır. YOLO (You Only Look Once) görüntüdeki nesneleri konumlarıyla beraber tek seferde tespit edebilen ve CNN kullanan bir algoritmadır. Görüntüyü hücrelere bölerek her bir hücrede nesneler için confidence değerlerini ve bounding boxları tahmin etmeye çalışan çıktı vektörleri üretir.

Veri Seti:



Program Çıktıları:



Makine Öğrenmesi & Python

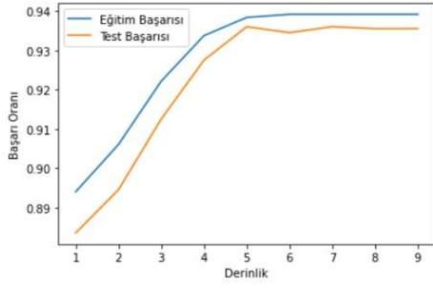
Günümüzde internetin kullanımı ve internete bağlı cihaz sayısı sürekli artmaktadır. İnternet ortamında veriler yoğun bir şekilde paylaşılabilir, depolanabilir ve etkileşime girilebilir bir duruma gelmiştir. İnternete olan ihtiyaç arttıkça kişisel verinin korunması son derece önemli bir hal almıştır. Bu çalışmanın amacı kimlik avı yapmak amacıyla hazırlanmış olan sitelerin makine öğrenmesi algoritmalarıyla kullanıcı siteyi ziyaret etmeden veya URL adresine tıklamadan önce kullanıcıyı uyararak kimlik avı saldırısının önüne geçmektir.

Çalışmada websitelerine istek atıldıktan sonra dönen cevap üzerinden python kütüphaneleri ile içerik analizi yapıldıktan sonra bu analizleri gradient boosting, random forest, lojistik regression gibi makine öğrenmesi modelleri ile öğrenmeler gerçekleştirilip, hangi algoritmanın daha optimize bir sonuç ürettiği karşılaştırılarak kullanılacak algoritmaya karar verilmiştir. Bu analiz sonucunda sistem için arayüz tasarlanmıştır. Bu arayüz aracılığı ile istenilen websitesi kontrol edilebilmektedir. Aşağıda belirtilen özelliklerin hepsi kontrol edilmiştir.

Özellik Numarası	Özellik İsmi	Özellik Numarası	Özellik İsmi
1	URL uzunluğu	12	Redirect kontrolü
2	Kısa link kontrolü	13	Mouse-over kontrolü
3	@ sembolü kontrolü	14	Sağ tık kontrolü
4	URL'de subdomain kontrolü	15	Pop-up kontrolü
5	SSL varlığı	16	Iframe kontrolü
6	Domain tarih kontrolü	17	Domain yaşı kontrolü
7	Port yönlendirme kontrolü	18	DNS kaydı
8	https kontrolü	19	Pagerank kontrolü
9	Site içerik kontrolü	20	Google index kaydı
10	SFH kontrolü	21	İstatistiksel rapor
11	Email gönderim kontrolü	-	-

Gradient Boosting Modeli

*****Gradient Boosting Classifier Test Ve Egitim Sonuclari**:



*****Gradient Boosting Classifier*****

Gradient Boosting Classifier Egitim Accuracy => 0.934
Gradient Boosting Classifier Test Accuracy => 0.927

Gradient Boosting Classifier Egitim f1_score => 0.941
Gradient Boosting Classifier Test f1_score => 0.936

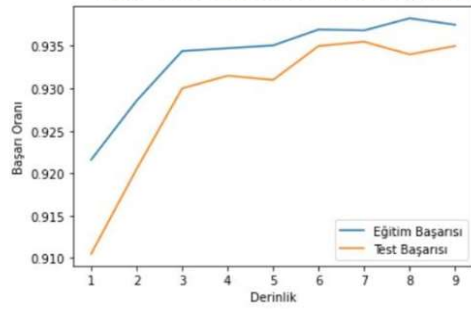
Gradient Boosting Classifier Egitim Recall => 0.944
Gradient Boosting Classifier Test Recall => 0.940

Gradient Boosting Classifier Egitim Precision => 0.937
Gradient Boosting Classifier Test Precision => 0.932

Şekil 3.9 Modellerin eğitim işlemi sonrası eğitim ve test verisiyle test edilen başarımlar

Random Forest Modeli

*****Random Forest Classifier Test Ve Egitim Sonuclari**:



*****RANDOM FOREST*****

Random Forest Egitim Accuracy => 0.938
Random Forest Test Accuracy => 0.936

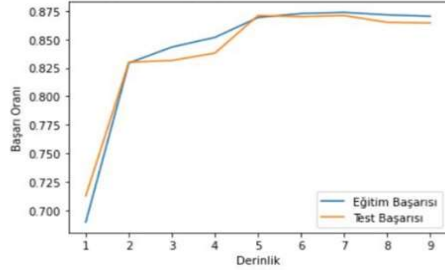
Random Forest Egitim f1_score => 0.945
Random Forest Test f1_score => 0.943

Random Forest Egitim Recall => 0.946
Random Forest Test Recall => 0.940

Random Forest Egitim Precision => 0.943
Random Forest Test Precision => 0.947

Lojistik Regresyon Modeli

*****Lojistik Regresyon Classifier Test Ve Egitim Sonuclari**:



*****Logistic Regression*****

Logistic Regression Egitim Accuracy => 0.870
Logistic Regression Test Accuracy => 0.864

Logistic Regression Egitim f1_score => 0.886
Logistic Regression Test f1_score => 0.883

Logistic Regression Egitim Recall => 0.910
Logistic Regression Test Recall => 0.912

Logistic Regression Egitim Precision => 0.863
Logistic Regression Test Precision => 0.855

Sistem Arayüzü:

PHISHING URL TESPİTİ

Kontrol Et

©2022

Website %79 oranında güvensizdir.

Hala Devam Edecek Misin?

<https://milliyet.com.tr>

- Long URL |----> 1
- Short URL |----> -1
- @ |----> 1
- subdomain |----> 0
- SSL varlığı |----> 1
- Domain Kayıt süresi |----> -1
- Port Yönlendirme |----> 1
- HTTPS Kontrolü |----> 1
- İçerik Oranı |----> 1
- Blank Kontrolü |----> 1
- Info mail |----> 1
- Redirect |----> -1
- MouseOver |----> 1
- Right Click Kontrolü |----> -1
- Pop-up |----> -1
- Iframe Kontrolü |----> 1
- Domain yaşı |----> 1
- DNS Kaydı |----> -1
- Rank değeri |----> 1
- Google Index Değeri |----> 1
- Zararlı Uzantı Kontrolü |----> 1

ASP.NET

Not:Projelerin kaynak kodlarını <https://github.com/ferdisonmez> adresinden inceleyebilirsiniz.

Ferdi SÖNMEZ