

**T.C.**  
**SAKARYA ÜNİVERSİTESİ**  
**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

**BSM 401 BİLGİSAYAR MÜHENDİSLİĞİ TASARIMI**

**Blockchain Altyapısı ile İletişim Bilgilerini  
Domainleştirme Sistemi Tasarımı**

**G171210021 - Onur Osman GÜLE**  
**G171210375 - Fatih Enis KAYA**

**Bölüm** : **BİLGİSAYAR MÜHENDİSLİĞİ**  
**Danışman** : **Doç. Dr. Ahmet ZENGİN**

**2020-2021 Güz Dönemi**

## ÖNSÖZ

Günümüzde internet sitelerini eskisi gibi IP adreslerini yazarak değil, alan adlarını yazarak ziyaret ediyoruz. Elbette alan adlarını bilmek, öğrenmek IP adreslerinden daha kolay bir yol. Ancak özellikle şirketlerin telefon numaralarını veya kişisel telefon numaralarının hepsinin hatırlanması bir hayli zor. Telefonlarda rehber özelliği olmasına rağmen her zaman yeterince kullanışlı olmayabiliyor. Bir şirketin müşteri temsilcilerine bağlanmak için telefon numaralarını internete erişebiliyorsak arama motorlarını kullanarak aramak durumunda kalabiliyoruz ve bu bize zaman kaybettiriyor.

## İÇİNDEKİLER

<a href="#">ÖNSÖZ</a>	I
<a href="#">İÇİNDEKİLER</a>	II
<a href="#">SİMGELER VE KISALTMALAR LİSTESİ</a>	V
<a href="#">ŞEKİLLER LİSTESİ</a>	VI
<a href="#">ÖZET</a>	IX
BÖLÜM 1.	
<a href="#">GİRİŞ</a>	1
<a href="#">1.1. Telefon Rehberleri</a>	1
<a href="#">1.2. Bilinmeyen Numaralar Servisi</a>	2
<a href="#">1.3. İletişim Bilgisi Paylaşımı</a>	2
<a href="#">1.3.1. Email</a>	2
<a href="#">1.3.2. Telefon</a>	3
BÖLÜM 2.	
<a href="#">VERİ TABANI</a>	3
<a href="#">2.1. MySQL</a>	3
<a href="#">2.2 Veri Tabanı Yapısı</a>	4
<a href="#">2.3. Tablo Yapıları</a>	5
<a href="#">2.3.1. Users (Kullanıcılar) Tablosu</a>	5
<a href="#">2.3.2. Domains (Linkler) Tablosu</a>	5
<a href="#">2.3.3. UserDomains Tablosu</a>	6
<a href="#">2.3.4. Types Tablosu</a>	6
<a href="#">2.3.5. Infos Tablosu</a>	7
<a href="#">2.3.6. Contacts Tablosu</a>	8
<a href="#">2.3.7. Privacies Tablosu</a>	8
<a href="#">2.3.8.DomainInfos Tablosu</a>	9
<a href="#">2.3.9. PublicPhones Tablosu</a>	9
<a href="#">2.3.10. CKeys Tablosu</a>	10

<a href="#"><u>2.4 Stored Procedure (Saklı Yordam)</u></a>	10
<a href="#"><u>2.4.1. DoLogin Yordamı</u></a>	11
<a href="#"><u>2.4.2. getDomainInfos Yordamı</u></a>	12
<a href="#"><u>2.4.3. getInfos Yordamı</u></a>	13
<a href="#"><u>2.4.4. DoSearch Yordamı</u></a>	13
<b>BÖLÜM 3.</b>	
<a href="#"><u>API ve GÜVENLİK</u></a>	14
<a href="#"><u>3.1 Veritabanı Bağlantısı</u></a>	14
<a href="#"><u>3.2 Linke Giriş</u></a>	14
<a href="#"><u>3.3 Linklerde Arama</u></a>	15
<a href="#"><u>3.4 Kayıt ve Giriş Güvenliği</u></a>	16
<a href="#"><u>3.5 Bilgi Güvenliği</u></a>	17
<b>BÖLÜM 4.</b>	
<a href="#"><u>EKRANLAR VE YÖNETİM PANELİ</u></a>	18
<a href="#"><u>4.1 Subdomain Yönetimi</u></a>	18
<a href="#"><u>4.2 Subdomain Ekranı</u></a>	19
<a href="#"><u>4.2.1 Subdomain Ekran Özellikleri</u></a>	20
<a href="#"><u>4.3 Yönetim Paneli</u></a>	21
<a href="#"><u>4.3.1 Bilgilerim Sayfası</u></a>	23
<a href="#"><u>4.3.2 Bilgi Ekle Sayfası</u></a>	23
<a href="#"><u>4.3.3 Rehber Sayfası</u></a>	24
<a href="#"><u>4.3.4 Domainler Sayfası</u></a>	24
<a href="#"><u>4.3.5 Yeni Domain Oluşturma Sayfası</u></a>	25
<a href="#"><u>4.3.6 Bilgileri Bağla Sayfası</u></a>	25
<a href="#"><u>4.3.7 Çıkış</u></a>	26
<a href="#"><u>4.3.8 Panel Ayarları</u></a>	26

**BÖLÜM 5.**

<b><u>MOBİL UYGULAMA</u></b>	27
<u>5.1 Uygulama Oluşturma</u>	27
<u>5.2 Uygulamada Arama</u>	27
<u>5.3 Uygulamada Yönetim Paneli</u>	28
<u>5.4 Uygulamaya Giriş</u>	28
<u>5.5 Uygulamada Linkler</u>	29

**BÖLÜM 6.**

<b><u>BLOCKCHAIN</u></b>	30
<u>6.1 Cloud Sanal Makine Oluşturma</u>	31
<u>6.2 VPC Ağ Konfigürasyonu</u>	31
<u>6.3 Sanal Makineye SSH Bağlantısı</u>	33
<u>6.4 NodeJS - CryptoJS Kütüphanesi</u>	34
<u>6.5 NodeJS - Express Kütüphanesi</u>	34
<u>6.6 Linux Forever Kütüphanesi</u>	35
<u>6.7 Blok Kazma İşlemi</u>	35
<u>6.8 Blockchain'i Diğer Düğümlere Eşleme</u>	35

**BÖLÜM 7.**

<b><u>SONUÇLAR VE ÖNERİLER</u></b>	37
<b><u>KAYNAKLAR</u></b>	38
<b><u>EK A</u></b>	39
<b><u>ÖZGEÇMİŞ</u></b>	39

## SİMGELER VE KISALTMALAR LİSTESİ

AES	: Advanced Encryption Standard
API	: Application Programming Interface
GCP	: Google Cloud Platform
AWS	: Amazon Web Services
SP	: Stored Procedure
IP	: Internet Protokol
RBDMS	: Relational Database Management System
SQL	: Structured Query Language
VPC	: Virtual Private Cloud

## ŞEKİLLER LİSTESİ

Şekil 2.1	MySQL logosu	3
Şekil 2.2	Veri Tabanı Yapısı	4
Şekil 2.3	Tablo Yapıları	5
Şekil 2.4	Users Tablosu	5
Şekil 2.5	Domains Tablosu	6
Şekil 2.6	UserDomains Tablosu	6
Şekil 2.7	Types Tablosu	6
Şekil 2.8	Types Mevcut Tipler	7
Şekil 2.9	Infos Tablosu	7
Şekil 2.10	Contacts Tablosu	8
Şekil 2.11	Privacies Tablosu	8
Şekil 2.12	Mevcut Gizlilik Türleri	8
Şekil 2.13	DomainInfos Tablosu	9
Şekil 2.14	PublicPhones Tablosu	9
Şekil 2.15	CKeys Tablosu	10
Şekil 2.16	Stored Procedure	11
Şekil 2.17	DoLogin Yordamı	12
Şekil 2.18	getDomainInfos Yordamı	12
Şekil 2.19	getInfos Yordamı	13
Şekil 2.20	DoSearch Yordamı	13

Şekil 3.1	Veritabanı Bağlantısı	14
Şekil 3.2	Linke Giriş	15
Şekil 3.3	Link Yordamı	15
Şekil 3.4	Linkte Arama	16
Şekil 3.5	Arama Yordam	16
Şekil 3.6	Kayıt ve Giriş Güvenliği	17
Şekil 3.7	Bilgi Güvenliği	17
Şekil 3.8	GCP örnek	17
Şekil 4.1	Subdomain Yönetimi	18
Şekil 4.2	Htaccess kod	18
Şekil 4.3	Subdomain Ekranı	18
Şekil 4.4	Subdomain Ekran Özellikleri	20
Şekil 4.5	Yönetim Ekran	21
Şekil 4.6	Yönetim 2.Ekran	21
Şekil 4.7	Yönetim İstatistik	21
Şekil 4.8	Yönetim Genel Ekran	22
Şekil 4.9	Bilgilerim Sayfası	23
Şekil 4.10	Bilgi Ekle Sayfası	23
Şekil 4.11	Rehber Sayfası	24
Şekil 4.12	Domainler Sayfası	24
Şekil 4.13	Yeni Domain Sayfası	25
Şekil 4.14	Bilgi Bağla Sayfası	25
Şekil 4.15	Panel Ayarları	26
Şekil 5.1	Uygulamada Arama	27
Şekil 5.2	Appte Yönetim Paneli	28



Şekil 5.3	Uygulamada Linkler	29
Şekil 6.1	Blockchain Json	30
Şekil 6.2	GCP Ekran	31
Şekil 6.3	VPC Ekran	32
Şekil 6.4	VPC Güvenlik Ekran	33
Şekil 6.5	Sanal Makine Ekran	33
Şekil 6.6	SSH Ekran	33
Şekil 6.7	CryptoJS Kütüphanesi	34
Şekil 6.8	Express Kütüphanesi	34
Şekil 6.9	Forever Örnek	35
Şekil 6.10	NPM forever	35
Şekil 6.11	Kazma İşlemi	35
Şekil 6.12	WebSocket Ekran	36

## ÖZET

Anahtar kelimeler: Blockchain, Domain, Mobil Uygulama, Veri Güvenliği, Bulut Bilişim

Veriler eskiden yalnızca bir merkez sunucuda saklanır ve kullanıcılar o sunuculardan verileri talep ederek erişirdi. Blockchain sisteminin doğuşuyla dağıtık sistemler ve peer to peer sistemler gündemleştirdi. Artık veriler yalnızca merkez sunucularda saklanmayıp, kullanıcılarda dahi saklanabilir hale geldi. Mobil uygulamada isteğe bağlı olarak indirilebilen blockchain ile domainlere ulaşmak için internete ihtiyaç duyulmayacak. Elbette veriler şifreli olarak saklanacağından herkese açık olmayan verilere erişim mümkün olmayacaktır.

Bu tasarım çalışması ile Google Cloud Platform ve Amazon Web Services kullanarak oluşturduğumuz Blockchain’de domainlerimiz saklanacak, kullanıcı bilgilerimiz ise MySQL veri tabanlarında mevcut olacaktır. Mobil uygulama ve websitesi için kullandığımız API’lar php ile yazıldı ve tamamen güvenli şekilde kullanıldı. Kullanıcı şifreleri Argon2 algoritması ile, kullanıcı bilgileri ise NaCl(libsodium) kütüphanesi kullanarak AES ve Açık Anahtarlı Şifreleme ile şifrelenmiştir.

Sonuç olarak yaptığımız proje sayesinde kişiler arası bilgi paylaşımı hem kolaylaştırıldı hem de verilmesi istenen ve istenmeyen bilgiler olmak üzere daha verimli şekilde iletmeye başlandı.

## BÖLÜM 1. GİRİŞ

İnternetin ilk yıllarında sunuculara bağlanmak için IP adresleri kullanılırdı. İnsanlar websitelerini ziyaret etmek için 193.140.253.240 gibi IP adresleriyle giriş yapıyorlardı. İlk domain kaydı 15 Mart 1985[1] tarihinde yapıldıktan sonra IP adresleri yerine domainler, alan adları kullanılmaya başladı. Artık 193.140.253.240 yerine sakarya.edu.tr yazarak sitelere giriş sağlayabiliyoruz. Ancak telefon numaralarında ilk çıktıklarından beri 11 haneli sayılar kullanılmakta. Elbette kolaylaştırmak için 444, 0850 gibi alan kodları çıksa da yeterince kolay değil.

### 1.1. Telefon Rehberleri

Telefonları unutmamak için eskiden her evde telefon rehberleri olurdu. A'dan Z'ye tüm tanıdıklarımız, ihtiyacımız olabilecek telefon numaralarını isimleriyle birlikte deftere yazardık.

Sonrasında dijitalleştirmeye gidilerek telefonlara rehberler eklendi ve bu sayede bir defter yerine direkt olarak telefonda aratılabilir olarak aranacak kişinin bulunması kolaylaştı.

Akıllı telefonlara geçiş yapıldığında artık Google ve Apple rehberleri Cloud'da tutarak telefonların kaybolması, sıfırlanması gibi durumlarda kaydedilen kişilerin kaybedilmemesini sağlayarak büyük kolaylık sağladı. Bu sayede artık yeni bir telefon alındığında rehberi aktarmak yerine Google veya Apple hesabını bağlamak yeterli oluyor.

## **1.2. Bilinmeyen Numaralar Servisi**

2000’li yıllarda ortaya çıkan bilinmeyen numaralar servisi[2] 118 ile kişiler devletin kendi oluşturduğu rehberden telefon numarası sorgulayabiliyordu. İnternetin yaygın olmadığı zamanlarda şirketlerin numaralarını öğrenmek için oldukça kullanılıyordu.

Şu anda 118 servisi özelleştirildi ve telekomünikasyon şirketlerinin kendi 118 numaralarından, kendi rehberlerinden numara sorgulama yapılabilir. Ancak elbette internetin yaygınlaşmasıyla bir şirketin numarasını öğrenmek için bilinmeyen numaralar servisini kullanan kişi sayısı epey azaldı.

## **1.3. İletişim Bilgisi Paylaşımı**

E-posta, cep telefonu numarası gibi iletişim bilgilerinin paylaşımı internetin olmadığı zamanlarda kişisel olarak paylaşıldığından bir zorluk oluşturmaktadır. Zaman içinde çeşitli yöntemler oluşturulmaya çalışılsa da henüz sağlıklı ve verimli bir çözüm ortaya çıkmış değildir. Çoğunlukla sözlü olarak 11 hane sayı veya noktalama işaretleri ile email bilgilerinin paylaşımı zorluk ortaya çıkarmaktadır.

### **1.3.1. Email**

Elektronik ortamda bu adres, e-mail (e-posta) adresi olarak adlandırılır. Bu bilginin normal şartlarda paylaşımı ve işlenmesi KVKK’ya tabi olduğundan aydınlatma metni ve kullanıcı onayı alındıktan sonra paylaşımına açık hale gelecektir[3]. 3. Kişilerle paylaşılması durumunda kullanıcılar sözleşme dahilinde bilgilendirilecektir.

### **1.3.2. Telefon**

Telefon numarası, bir telefon hattına bağlı sabit hatlı bir telefon abone istasyonuna veya telsiz telefon veya cep telefonu gibi bir kablosuz elektronik telefon cihazına ya da genel veri aktarımı için diğer cihazlara atanan bir dizi rakamdır. Bu bilgi de 1.3.1’deki KVKK’ya aynı şekilde uyularak paylaşılacaktır.

## BÖLÜM 2. VERİ TABANI

API'larımızda php kullanmayı tercih ettiğimizden php ile birlikte iyi çalışan MySQL MariaDB veritabanını kullanmayı tercih ettik.

### 2.1. MySQL

İstemci-sunucu modeli MySQL, açık kaynaklı bir ilişkisel veritabanı yönetim sistemidir (RDBMS). RDBMS ilişkisel bir modele dayalı veritabanı yaratmak ve yönetmek için kullanılan bir yazılım veya hizmettir[4].

Linux, macOS, Windows ve Ubuntu gibi birçok önemli platform ve programlama dilleriyle uyumlu olarak çalışmaktadır.



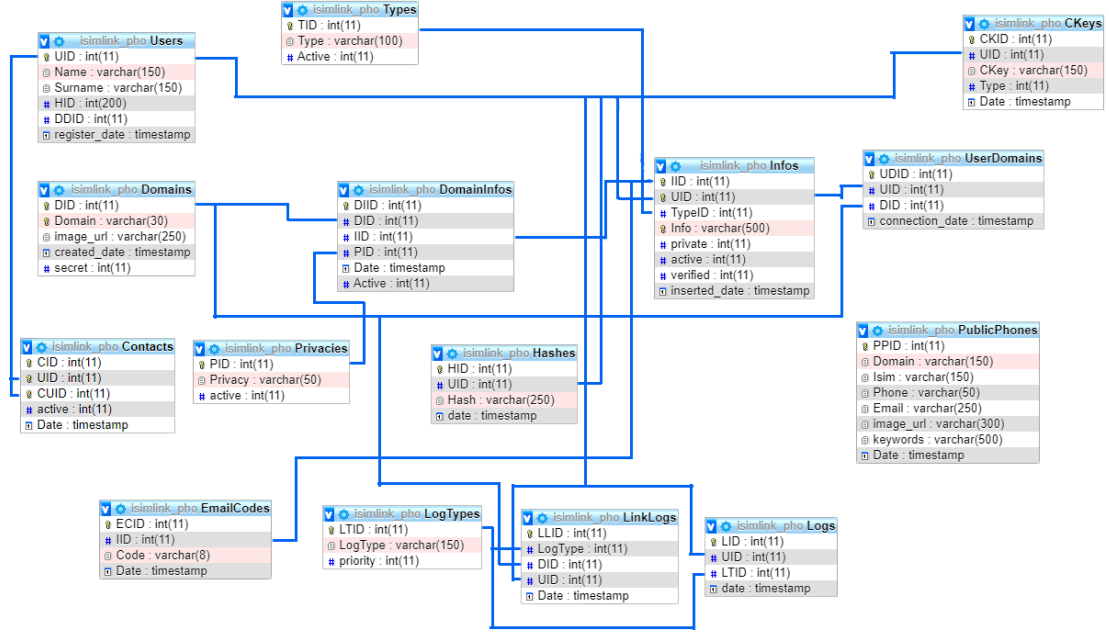
MySQL Logosu (Şekil 2.1)

Facebook, Twitter, Youtube, Google ve Yahoo gibi popüler uygulamalarının hemen hepsi veri depolamak için MySQL kullanmaktadır.

MySQL, SQL dilini kullanmaktadır ve biz de projemizde SQL dilini kullanarak sorgular oluşturduk.

## 2.2. Veri tabanı Yapısı

Veri tabanı ilişkisel yapımız aşağıdaki gibidir:



Veri Tabanı Yapısı Şekil 2.2

## 2.3. Tablo Yapıları

Veri tabanlarında, verileri tutmak için belli tablolar hazırlanmaktadır. Projemizin veri tabanında işlemekte olan toplam 15 tablomuz bulunmaktadır.

Tablo	Eylem	Satır	Türü	Karşılaştırma	Boyut	Ek Yük
<input type="checkbox"/> CKeys	★ Gözet Yapı Ara Ekle Boşalt Kaldır	18	MyISAM	latin1_swedish_ci	3.5 KiB	-
<input type="checkbox"/> Contacts	★ Gözet Yapı Ara Ekle Boşalt Kaldır	1	MyISAM	latin1_swedish_ci	3.0 KiB	-
<input type="checkbox"/> DomainInfos	★ Gözet Yapı Ara Ekle Boşalt Kaldır	13	MyISAM	latin1_swedish_ci	2.3 KiB	-
<input type="checkbox"/> Domains	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	MyISAM	latin1_swedish_ci	4.2 KiB	-
<input type="checkbox"/> EmailCodes	★ Gözet Yapı Ara Ekle Boşalt Kaldır	12	MyISAM	latin1_swedish_ci	2.3 KiB	-
<input type="checkbox"/> Hashes	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	MyISAM	latin1_swedish_ci	3.1 KiB	-
<input type="checkbox"/> Infos	★ Gözet Yapı Ara Ekle Boşalt Kaldır	8	MyISAM	latin1_swedish_ci	9.3 KiB	1.0 KiB
<input type="checkbox"/> LinkLogs	★ Gözet Yapı Ara Ekle Boşalt Kaldır	267	MyISAM	latin1_swedish_ci	10.5 KiB	-
<input type="checkbox"/> Logs	★ Gözet Yapı Ara Ekle Boşalt Kaldır	158	MyISAM	latin1_swedish_ci	6.6 KiB	-
<input type="checkbox"/> LogTypes	★ Gözet Yapı Ara Ekle Boşalt Kaldır	6	MyISAM	latin1_swedish_ci	2.2 KiB	-
<input type="checkbox"/> old_Users	★ Gözet Yapı Ara Ekle Boşalt Kaldır	20	MyISAM	latin1_swedish_ci	3.1 KiB	68 B
<input type="checkbox"/> Privacies	★ Gözet Yapı Ara Ekle Boşalt Kaldır	4	MyISAM	latin1_swedish_ci	2.1 KiB	-
<input type="checkbox"/> PublicPhones	★ Gözet Yapı Ara Ekle Boşalt Kaldır	1	MyISAM	latin1_swedish_ci	3.2 KiB	-
<input type="checkbox"/> Types	★ Gözet Yapı Ara Ekle Boşalt Kaldır	6	MyISAM	latin1_swedish_ci	2.1 KiB	-
<input type="checkbox"/> UserDomains	★ Gözet Yapı Ara Ekle Boşalt Kaldır	9	MyISAM	latin1_swedish_ci	2.1 KiB	-
<input type="checkbox"/> Users	★ Gözet Yapı Ara Ekle Boşalt Kaldır	9	MyISAM	latin1_swedish_ci	2.5 KiB	-
<b>16 tabloları</b>	<b>Toplam</b>	<b>552</b>	<b>MyISAM</b>	<b>latin1_swedish_ci</b>	<b>62.1 KiB</b>	<b>1.1 KiB</b>

Tablo Yapıları Şekil 2.3

### 2.3.1 Users (Kullanıcılar) Tablosu

Users tablosu, kullanıcılar üye olduklarında verilerinin eklendiği tablodur.

1. UID: Kullanıcı Numarası
2. Name: Kullanıcının Adı
3. Surname: Kullanıcının Soyadı
4. HID: Hash Numarası (Argon2 ile şifrelenen kullanıcı şifresini doğrulamak için)
5. DDID: Kullanıcının varsayılan linkinin numarası
6. register\_date: Kullanıcı kayıt tarihi

#	Adı	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Açıklamalar	Ekstra	Eylem
<input type="checkbox"/>	1	UID		int(11)	Hayır	Yok		AUTO_INCREMENT	Değiştir  Kaldır  Daha fazla
<input type="checkbox"/>	2	Name		varchar(150) utf8_turkish_ci	Hayır	Yok			Değiştir  Kaldır  Daha fazla
<input type="checkbox"/>	3	Surname		varchar(150) utf8_turkish_ci	Hayır	Yok			Değiştir  Kaldır  Daha fazla
<input type="checkbox"/>	4	HID		int(200)	Hayır	Yok	HashID		Değiştir  Kaldır  Daha fazla
<input type="checkbox"/>	5	DDID		int(11)	Evet	NULL	Default Domain ID		Değiştir  Kaldır  Daha fazla
<input type="checkbox"/>	6	register_date		timestamp	Hayır	current_timestamp()			Değiştir  Kaldır  Daha fazla

Users Tablosu Şekil 2.4

### 2.3.2. Domains (Linkler) Tablosu

Domains tablosu, kullanıcıların oluşturduğu linklerin eklendiği tablodur.

1. DID: Link Numarası
2. Domain: Link İsmi
3. image\_url: Linkin Resim Bağlantısı
4. created\_date: Linki oluşturma tarihi
5. secret: Linkin gizlilik türü

#	Adı	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Açıklamalar	Ekstra	Eylem
<input type="checkbox"/>	1	DID		int(11)	Hayır	Yok		AUTO_INCREMENT	Değiştir  Kaldır  Daha fazla
<input type="checkbox"/>	2	Domain		varchar(30) latin1_swedish_ci	Hayır	Yok			Değiştir  Kaldır  Daha fazla
<input type="checkbox"/>	3	image_url		varchar(250) latin1_swedish_ci	Evet	NULL			Değiştir  Kaldır  Daha fazla
<input type="checkbox"/>	4	created_date		timestamp	Hayır	current_timestamp()			Değiştir  Kaldır  Daha fazla
<input type="checkbox"/>	5	secret		int(11)	Hayır	0			Değiştir  Kaldır  Daha fazla

Domains Tablosu Şekil 2.5

### 2.3.3 UserDomains Tablosu

UserDomains tablosu, kullanıcı ve linklerin birleştirildiği tablodur. Birden fazla linki olan kullanıcıların yönetilmesi için zorunluluktur.

1. UDID: UserDomain Numarası
2. UID: Kullanıcı Numarası
3. DID: Link Numarası
4. connection\_date: Kullanıcı ve Linkin bağlantı tarihi

#	Adı	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Açıklamalar	Ekstra
<input type="checkbox"/>	1 UDID	int(11)			Hayır	Yok		AUTO_INCREMENT
<input type="checkbox"/>	2 UID	int(11)			Hayır	Yok		
<input type="checkbox"/>	3 DID	int(11)			Hayır	Yok		
<input type="checkbox"/>	4 connection_date	timestamp			Hayır	current_timestamp()		

UserDomains Tablosu Şekil 2.6

### 2.3.4. Types Tablosu



















Types tablosu, linklerin verilerinin hangi türlerden olduğunu içeren bir tablodur.

1. TID: Tip Numarası
2. Type: Tip Adı
3. Active: Tipin aktif olup olmadığı.

#	Adı	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan
1	TID	int(11)			Hayır	Yok
2	Type	varchar(100)	utf8_turkish_ci		Hayır	Yok
3	Active	int(11)			Hayır	1

Types Tablosu Şekil 2.7

Mevcut Tipler:

				TID	Type	Active
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	1	Email	1
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	2	Telefon	1
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	3	Adres	0
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	-1	İsim	0
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	-2	Soyisim	0
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	-3	Fotoğraf	0

Types Mevcut Tipler Şekil 2.8

### 2.3.5. Infos Tablosu

Infos tablosu, linklerin verilerini içeren tablodur.

1. IID: Veri Numarası
2. UID: Kullanıcı Numarası
3. TypeID: Tip Numarası
4. Info: Veri
5. private: Gizlilik
6. active: Aktiflik
7. verified: Onaylanmış Veri
8. inserted\_date: Verinin Eklendiği Tarih



#	Adı	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Açıklamalar	Ekstra
1	<b>IID</b> 🔑	int(11)			Hayır	Yok		AUTO_INCREMENT
2	<b>UID</b> 🔑	int(11)			Hayır	Yok		
3	<b>TypeID</b>	int(11)			Hayır	Yok		
4	<b>Info</b> 🗃️	varchar(500)	latin1_swedish_ci		Hayır	Yok		
5	<b>private</b>	int(11)			Hayır	0	hide_everywhere	
6	<b>active</b>	int(11)			Hayır	0		
7	<b>verified</b>	int(11)			Hayır	0		
8	<b>inserted_date</b>	timestamp			Hayır	current_timestamp()		

Infos Tablosu Şekil 2.9

### 2.3.6. Contacts Tablosu

Contacts tablosu, kişilerin kendi panellerinden ekledikleri rehber tablosudur. Bazı veriler yalnızca rehber ile paylaşılabilir.

1. CID: Rehber Numarası
2. UID: Ekleyen Kullanıcı Numarası
3. CUID: Eklenen Kullanıcı Numarası
4. active: Aktiflik
5. Date: Rehber ekleme tarihi

#	Adı	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Açıklamalar	Ekstra
1	<b>CID</b> 🔑	int(11)			Hayır	Yok		AUTO_INCREMENT
2	<b>UID</b> 🔑	int(11)			Hayır	Yok		
3	<b>CUID</b> 🔑	int(11)			Hayır	Yok	contactUID, eklenen	
4	<b>active</b>	int(11)			Hayır	1		
5	<b>Date</b>	timestamp			Hayır	current_timestamp()		

Contacts Tablosu Şekil 2.10

### 2.3.7. Privacies Tablosu

Privacies tablosu, verilerin gizliliklerinin türlerinin tutulduğu tablodur.

1. PID: Gizlilik Türü Numarası
2. Privacy: Gizlilik Türü
3. active: Aktiflik

#	Adı	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Açıklamalar	Ekstra
1	<b>PID</b> 🔑	int(11)			Hayır	Yok		AUTO_INCREMENT
2	<b>Privacy</b>	varchar(50)	utf8_turkish_ci		Hayır	Yok		
3	<b>active</b>	int(11)			Hayır	1		

Privacies Tablosu Şekil 2.11

Mevcut Gizlilik Türleri:

PID	Privacy	active
1	Herkese Açık	1
2	Üyeler	1
3	Rehberim	1
4	Yetkilendirdiklerim	0

Mevcut Gizlilik Türleri Şekil 2.12

### 2.3.8. DomainInfos Tablosu

DomainInfos tablosu, linkler ile verilerin eşleştiği tablodur. Veri tabanımızın en karmaşık yapıli tablosudur.

1. DIID: Link Veri Numarası
2. DID: Link Numarası
3. IID: Veri Numarası
4. PID: Gizlilik Türü Numarası
5. Date: Veri ile Linkin bağlandığı tarih
6. Active: Aktiflik

#	Adı	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Açıklamalar	Ekstra
1	DIID	int(11)			Hayır	Yok		AUTO_INCREMENT
2	DID	int(11)			Hayır	Yok		
3	IID	int(11)			Hayır	Yok		
4	PID	int(11)			Hayır	1	PrivacyID	
5	Date	timestamp			Hayır	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()
6	Active	int(11)			Hayır	1		

DomainInfos Tablosu Şekil 2.13

### 2.3.9. PublicPhones Tablosu

Kullanıcılarımıza kolaylık sağlaması amacıyla oluşturulan PublicPhones tablosu, kayıt olmadan, bizim tarafımızdan eklenen ve genellikle en çok aranan firmaların iletişim bilgilerinin bulunduğu bir tablodur.

1. PPID: Herkese Açık Bilgi Numarası
2. Domain: Link Adı
3. Isim: Linkin Bağlı Olduğı Yasal Ad
4. Phone: Telefon
5. Email: Mail Adresi
6. image\_url: Linkin Resim Bağlantısı

7. keywords: Linkin arama ihtimallerinin listelendiği anahtar kelimeleri
8. Date: Linkin eklendiği tarih

#	Adı	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Açıklamalar	Ekstra
1	PPID 	int(11)			Hayır	Yok		AUTO_INCREMENT
2	Domain	varchar(150)	latin1_swedish_ci		Hayır	Yok		
3	Isim	varchar(150)	latin1_swedish_ci		Hayır	Yok		
4	Phone	varchar(50)	latin1_swedish_ci		Hayır	Yok		
5	Email	varchar(250)	latin1_swedish_ci		Evet	NULL		
6	image_url	varchar(300)	latin1_swedish_ci		Evet	NULL		
7	keywords 	varchar(500)	latin1_swedish_ci		Evet	NULL		
8	Date	timestamp			Hayır	current_timestamp()		

PublicPhones Tablosu Şekil 2.14

### 2.3.10. CKeys Tablosu

CKeys tablosu, kullanıcıların verilerinin şifrlenmesini sağlayan, şifreleri barındıran tablodur.

1. CKID: Kriptolu Şifre Numarası
2. UID: Kullanıcı Numarası
3. CKey: Kriptolu Şifre
4. Type: Şifre Tipi (Kişisel ve Herkese Açık)
5. Date: Şifrenin Güncellenme Tarihi

#	Adı	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Açıklamalar	Ekstra
1	CKID 	int(11)			Hayır	Yok		AUTO_INCREMENT
2	UID	int(11)			Hayır	Yok		
3	CKey	varchar(150)	latin1_swedish_ci		Hayır	Yok		
4	Type	int(11)			Hayır	Yok	0:private, 1:public	
5	Date	timestamp			Hayır	current_timestamp()		

Ckeys Tablosu Şekil 2.15

Projemiz yukarıda yer alan tablolar ile sağlanmaktadır. Bunların yanında kayıt tutma tablolarımız da mevcuttur.

## 2.4. Stored Procedure (Saklı Yordam)

Veri tabanlarında, programlama dillerinde fonksiyon olarak tanımlanan sorgular saklı yordam ve fonksiyonlar olarak tanımlanabilir. Projemizde genellikle API'lara sonuç döndürmek için saklı yordamlar kullandık. Kullandığımız saklı yordamlar aşağıdadır:

Yordamlar					
Adı	Eylem		Turu	Dönüşler	
<input type="checkbox"/> DoLogin	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> DoLogin2	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> DoSearch	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> LogToDom	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> Register	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> activeInfo	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> addContact	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> addEmail	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> addInfo	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> connectDomainInfo	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> emailVerif	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getContacts	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getDomainInfos	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getDomainInfos_new	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getDomainInfos_old	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getDomainLinksOfUser	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getDomainsOfUser	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getHash	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getIndexStats	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getInfos	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getPrivacies	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
<input type="checkbox"/> getViewStats	Düzenle	Çalıştır	Dışa aktar	Kaldır	PROCEDURE
Tümünü işaretle					Seçilileri: Dışa aktar Kaldır

Stored Procedure Şekil 2.16

Projede en sık kullanılan birkaç saklı yordamı inceleyelim.

### 2.4.1. DoLogin Yordamı

DoLogin yordamı, kullanıcıların sisteme giriş yapmak için kullandığı bir yordamdır. fdomain parametresi kullanıcıdan giriş yaparken alınan link ismi, fsuccess parametresi ise kullanıcıdan alınan şifrenin veritabanındaki hash ile karşılaştırıldığında dönen sonuçtur.

Ayrıntılar					
Yordam adı		DoLogin			
Türü		PROCEDURE			
Parametreler		Yön	Adı	Türü	Uzunluk/Değerler
		IN	fdomain	VARCHAR	150
		IN	fsuccess	INT	
Parametre ekle					
Tanım		<pre> 1 BEGIN 2 SET @did = (SELECT DID FROM Domains d WHERE d.Domain = fdomain); 3 IF ISNULL(@did) THEN 4 SELECT 'Domain bulunamadı.' AS 'return'; 5 ELSE 6 SET @realuid = (SELECT UID FROM UserDomains ud WHERE ud.DID = @did); 7 IF ISNULL(@realuid) THEN 8 SELECT 'Kullanıcı bulunamadı.' AS 'return'; 9 ELSE 10 IF fsuccess = 0 THEN 11 INSERT INTO Logs(UID,LTID) VALUES(@realuid, 6); 12 SELECT 'Şifre hatalı.' AS 'return'; 13 ELSE 14 (SELECT UID, Name, Surname INTO @uid,@name,@surname FROM Users u WHERE u.UID = @realuid); 15 INSERT INTO Logs(UID,LTID) VALUES(@uid, 1); 16 SELECT @uid AS UID, @name AS Name, @surname AS Surname, 'Başarılı' AS 'return'; 17 END IF; 18 END IF; 19 END IF; 20 END </pre>			

DoLogin Yordamı Şekil 2.4.1

### 2.4.2. getDomainInfos Yordamı

getDomainInfos yordamı, linklere girildiğinde dönmeleri gereken verileri listeleyen bir yordamdır.

fdomain parametresi girilen linkin adını, fuid parametresi ise giriş yapan kişinin kullanıcı numarasını gösterir.

The screenshot shows the 'Ayrıntılar' (Details) tab for the 'getDomainInfos' stored procedure. The procedure is of type 'PROCEDURE'. It has two input parameters: 'fdomain' (VARCHAR, length 150) and 'fuid' (INT). The 'Tanım' (Definition) section contains the following SQL code:

```

1 BEGIN
2 SET @ppid = (SELECT pp.PPID FROM PublicPhones pp WHERE pp.Domain = fdomain);
3 IF @ppid IS NOT NULL
4 THEN
5 (SELECT pp.Phone, pp.Isim, pp.Email, pp.image_url INTO @telefon, @name, @email, @image FROM
6 PublicPhones pp WHERE pp.PPID = @ppid);
7 SELECT @name AS Name, @surname AS Surname, @telefon AS Telefon, @email AS Email, @image AS Image,
8 'public' AS LinkType;
9 ELSE
10 SET @did = (SELECT d.DID FROM Domains d WHERE d.Domain = fdomain);
11 SET @uid = (SELECT ud.UID FROM UserDomains ud WHERE ud.DID = @did);
12 (SELECT u.Name, u.Surname INTO @name, @surname FROM Users u WHERE u.UID = @uid);
13 IF ISNULL(fuid) THEN
14 (SELECT i.Info INTO @telefon FROM Infos i INNER JOIN DomainInfos di ON i.IID = di.IID WHERE di.DID
15 = @did AND i.TypeID = 2);
16 #son telefon geliyor, ona göre. Sonradan satır satır alırsız farklı bir proc ile.
17 ELSE
18 #burada uidye verilen bilgiler de gelsin. sonra...
19 (SELECT i.Info INTO @telefon FROM Infos i INNER JOIN DomainInfos di ON i.IID = di.IID WHERE di.DID
20 = @did AND i.TypeID = 2 LIMIT 0,1);
21 END IF;
22 IF ISNULL(@did) THEN
23 SET @res = '';
24 ELSE
25 SELECT @name AS Name, @surname AS Surname, @telefon AS Telefon, 'personal' AS LinkType;
26 END IF;
27 END IF;
28 END

```

getDomainInfos Yordamı Şekil 2.4.2

### 2.4.3. getInfos Yordamı

getInfos yordamı, kullanıcının panelinde bilgilerini görmesini, listelemesini sağlayan yordamdır.

fuid parametresi kullanıcının numarası olarak tanımlanır.

**Ayrıntılar**

**Yordam adı**

**Türü** PROCEDURE

Parametreler	Yön	Adı	Türü	Uzunluk/Değerler	Seçenekler
↑	I	fuid	INT		Kaldır

Parametre ekle

**Tanım**

```

1 BEGIN
2 SELECT DISTINCT
  i.IID,i.TypeID,i.Info,i.private,i.active,i.verified,i.inserted_date,di.DIID,di.
  DID, (SELECT d.Domain FROM Domains d WHERE d.DID = di.DID) AS 'Domain' FROM
  Infos i LEFT JOIN DomainInfos di ON di.IID = i.IID WHERE i.UIID = fuid AND
  di.DID IN (SELECT ud.DID FROM UserDomains ud WHERE ud.UIID = fuid) GROUP BY
  i.IID;
3 END

```

getInfos Yordamı Şekil 2.4.3

#### 2.4.4. DoSearch Yordamı

DoSearch yordamı, kullanıcıların arama yapmasına olanak sağlayan yordamdır. fquery parametresi kullanıcının arama alanına girdiği yazıdır.

**Yordamı düzenle**

**Yordam adı**

**Türü** PROCEDURE

Parametreler	Yön	Adı	Türü	Uzunluk/Değerler	Seçenekler
↑	I	fquery	VARCHAR	50	Karakter Kaldır

Parametre ekle

**Tanım**

```

1 BEGIN
2 SELECT pp.Isim AS `Isim`, pp.Domain AS `Domain`, 'c' AS Type FROM PublicPhones
  pp
3 WHERE pp.keywords LIKE CONCAT('%',fquery,'%') OR pp.Domain LIKE
  CONCAT('%',fquery,'%') OR pp.Isim LIKE CONCAT('%',fquery,'%')
4 UNION
5 SELECT CONCAT(u.Name, ' ',u.Surname) AS `Isim`, d.Domain AS `Domain`, 'p' AS
  Type FROM Domains d INNER JOIN UserDomains ud ON d.DID = ud.DID INNER JOIN
  Users u ON u.UIID = ud.UIID
6 WHERE d.Domain LIKE CONCAT('%',fquery,'%') OR  CONCAT(u.Name, ' ',u.Surname)
  LIKE CONCAT('%',fquery,'%');
7 END

```

DoSearch Yordamı Şekil 2.20

## BÖLÜM 3. API ve GÜVENLİK

Veri tabanını, tabloları, saklı yordamları oluşturduktan sonra bunları kullanıcı ile buluşturmak için API kullanmamız gerekiyor. Arka tarafta API için PHP programlama dili kullanmayı tercih ettik. PHP ile MySQL veri tabanını birbirine bağlamak için en popüler 3 yöntem mevcuttur.[5] PHP MySQL Eklentisi, MySQLi Eklentisi ve PHP Veri Objeleri(PDO) kullanabiliriz. Proje güvenliğini ele aldığımızda SQL Injection saldırılarını engellemek amacıyla PDO kullanmayı tercih ettik.

### 3.1. Veritabanı Bağlantısı

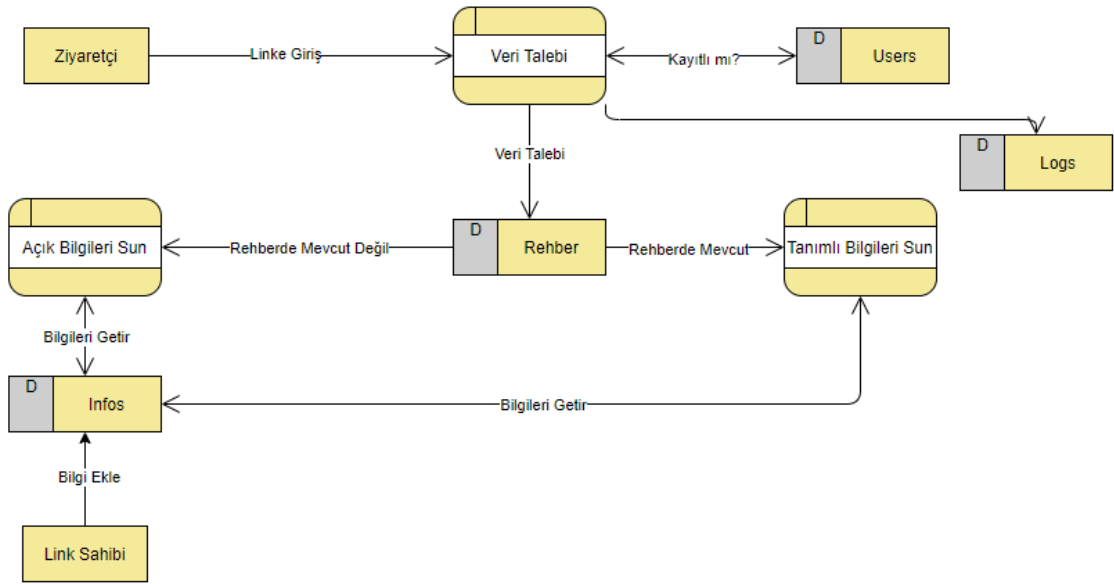
Veritabanı bağlantımızı yaparken PHP üzerinde PDO kullanıyoruz.

```
<?php
header('Content-Type: text/html; charset=utf-8');
try {
    $db = new PDO("mysql:host=localhost;dbname=isimlink_pho", "isimlink_upho",
"{PASSWORD}");
    $db->exec("SET NAMES 'utf8'; SET CHARSET 'utf8'");
} catch ( PDOException $e ){
    print $e->getMessage();
}
?>
```

Veritabanı Bağlantısı Şekil 3.1

### 3.2 Linke Giriş

Herhangi bir ziyaretçi, daha önce kayıtlı bir linke giriş yaptığında gösterilecek veriler link sahibi tarafından tanımlanır.



Linke Giriş Şekil 3.2

Linkteki bilgiler istenirken getDomainInfos saklı yordamı kullanılır.

```

<?php
$query = $_GET["query"];
$uid = $_GET["uid"];
include_once "../conf/db.php";
$q = $db->prepare("CALL getDomainInfos_new(:query,:uid)");
$q->execute(array("uid" => $uid,"query" => $query));
$f = $q->fetchAll(PDO::FETCH_ASSOC);
echo json_encode($f);
?>
    
```

Link Yordamı Şekil 3.3

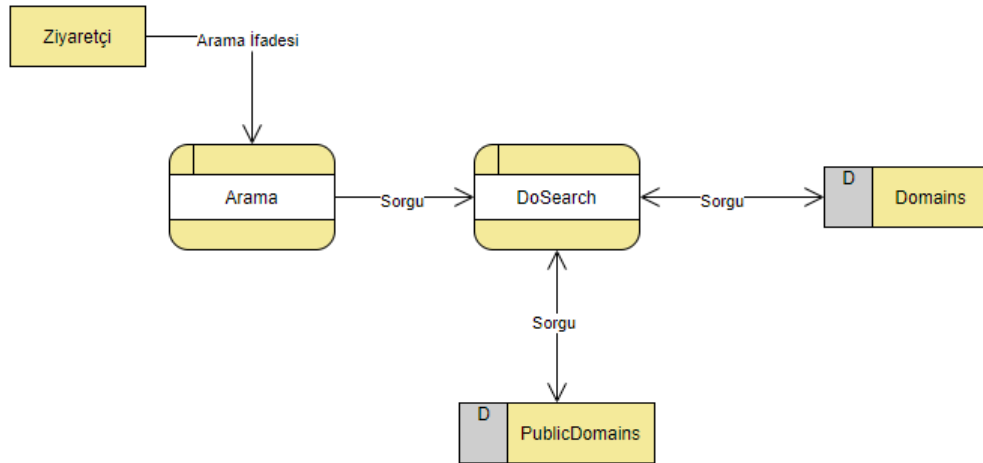
### 3.3 Linklerde Arama

Websitesi veya mobil uygulama üzerinden arama yapılırken 2 kategoride arama yapılır.

1. Kişisel Linkler
2. Kurumsal Linkler



Kişisel linklerde link sahibinin ismi veya linki ile arama yapılabilir. Kurumsal linklerde ise kişisel linklerdeki özelliklerin yanı sıra anahtar kelime araması yapılabilir. Sakarya Üniversitesi linkini aramak için sakarya, sau gibi kelimeler kullanılabilir.



Linkte Arama Şekil 3.4

Tek bir saklı yordam ile kişisel ve kurumsal linklerde arama sağlanır. Bu saklı yordama API yardımıyla erişilebilir.

```

<?php
$query = $_GET["query"];
include_once "../conf/db.php";
$q = $db->prepare("CALL doSearch(:query)");
$q->execute(array("query" => $query));
$f = $q->fetchAll(PDO::FETCH_ASSOC);
echo json_encode($f);
?>

```

Arama Yordam Şekil 3.5

### 3.4 Kayıt ve Giriş Güvenliği

Kayıt olurken yalnızca domain ismi ve yeni oluşturulacak bir şifre talep ediyoruz. Şifreleri Argon2 şifreleme algoritmasıyla şifreleyip hash değerini veri tabanımızda saklıyoruz.

```
<?php
function sifrele($sifre){
    return password_hash($sifre, PASSWORD_ARGON2I, ['memory_cost' => 2048, 'time_cost'
=> 4, 'threads' => 3]);
}

function dogrula($sifre,$hash){
if(password_verify($sifre,$hash) == 1) return 1;
else return 0;
}
```

Kayıt ve Giriş Güvenliği Şekil 3.6

### 3.5 Bilgi Güvenliği

IsimLink'e kaydolurken LibSodium kütüphanesi kullanılarak Diffie-Hellman algoritması ile size ait herkese açık ve gizli anahtar oluşturulup kaydedilir[6].

```
$skeys = file_get_contents("http://crypt.isim.link/pairkeygen.php");
$private_key = (json_decode($skeys))[0];
$public_key = (json_decode($skeys))[1];
```

Bilgi Güvenliği Şekil 3.7

Herhangi biri herkese açık olmayan bir verinize erişmeye çalışırken bu veri sizin gizli ve karşıdaki kişinin herkese açık anahtarı ile şifrelenir.

Verinize erişmeye çalışan kişi bu şifreyi yalnızca kendi gizli anahtarı ve sizin herkese açık anahtarınız ile açabilir.

Bu sayede herkese açık olmayan verileriniz güvende olur.

```
isimlink@isimlink-crypto: ~
GNU nano 3.2 pairkeygen.php

<?php
$keypair = sodium_crypto_box_keypair(); //user
$keypair_public = sodium_crypto_box_publickey($keypair); //upk
$keypair_secret = sodium_crypto_box_secretkey($keypair); //usk
echo json_encode(array(bin2hex($keypair_secret),bin2hex($keypair_public)));
```

GCP örnek Şekil 3.8

Şifreleme işlemlerini GCP sanal makinemizde yapmaktayız

## BÖLÜM 4. EKРАНLAR VE YÖNETİM PANELİ

IsimLink üzerinden herhangi bir linke girildiğinde subdomain olarak algılanır. Her kullanıcıya varsayılan olarak bir subdomain atanır ve bu subdomaini istediği yerde paylaşabilir.

### 4.1. Subdomain Yönetimi



Öncelikle her kullanıcı için subdomain oluşturmamız gerekmekte. Onun yerine genel bir tanımlayıcı oluşturduk. cPanel üzerinden DNS Zone Editor altında bir kayıt ekledik:

Name ^	TTL	Class	Type	Record
*.isim.link.	14400	IN	A	78.142.210.12

Subdomain Yönetimi Şekil 4.1

Bu kayıtla birlikte artık her subdomaini kontrol edebileceğiz.

Ayrıca cPanel üzerinden yine genel tanımlayıcı ile bir subdomain oluşturduk:

Subdomains	Document Root	Redirection
*.isim.link	 /public_html/subs 	not redirected

Htaccess kod Şekil 4.2

Artık her subdomain FTP üzerindeki subs klasörüne yönlendiriliyor.

Bu subdomainleri yönetmek için .htaccess kullanıyoruz:

```
RewriteEngine On
RewriteCond %{HTTP_HOST} ^(.*)\.isim\.link [NC]
RewriteRule ^$ index.php?user=%1 [L]
```

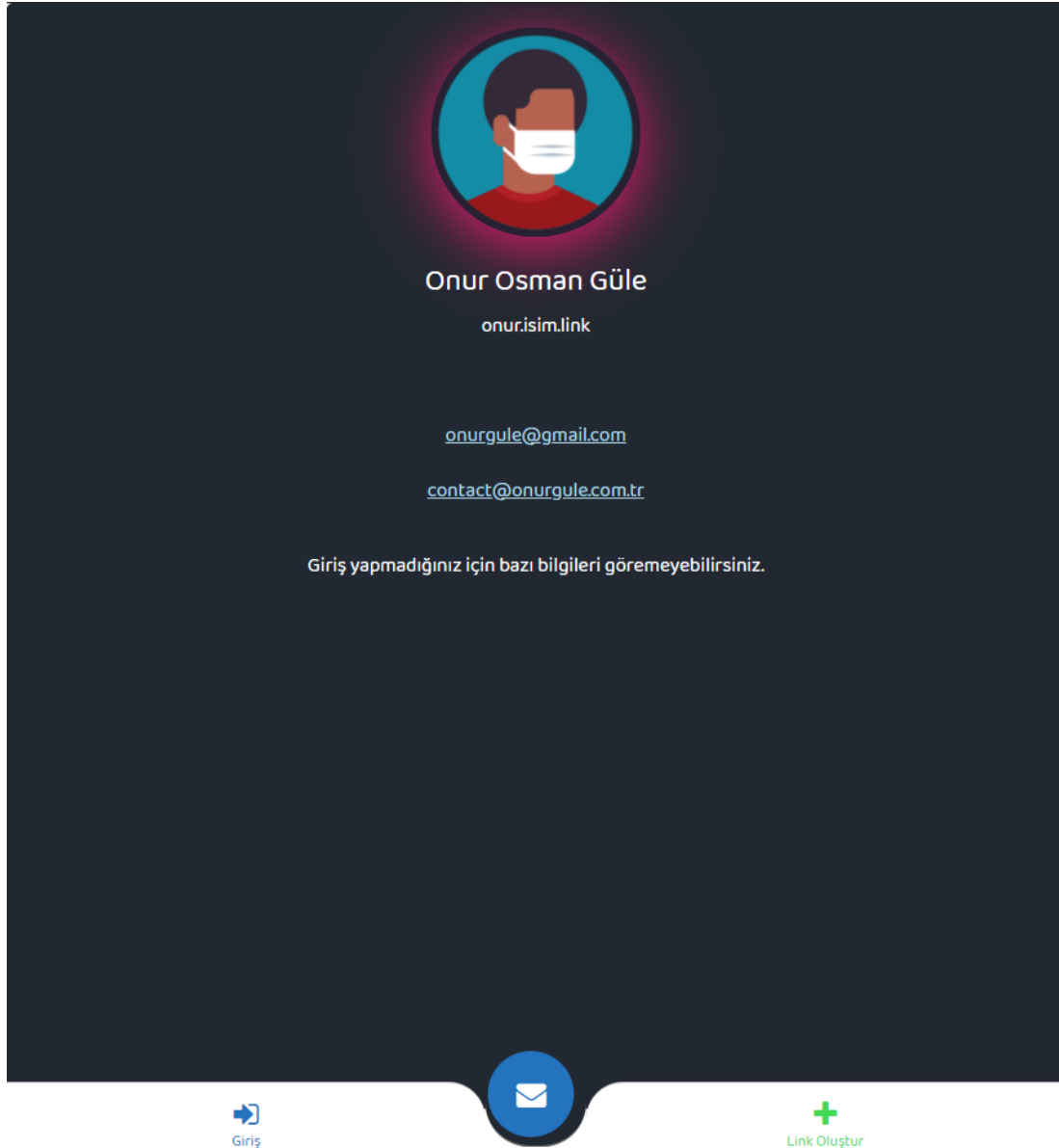
Subdomain Ekranı 4.3

Rewrite Rule ile gelen subdomain adresini index.php adresine parametre olarak gönderiyoruz.

## 4.2. Subdomain Ekranı

Artık subdomain girişlerinde gelen subdomain adını bildiğimiz için API ve Stored Procedurelerden faydalanabiliriz.

Gelen subdomain adına göre UID'yi bulup getDomainInfos yordamını kullanarak link bilgilerini gösteriyoruz.

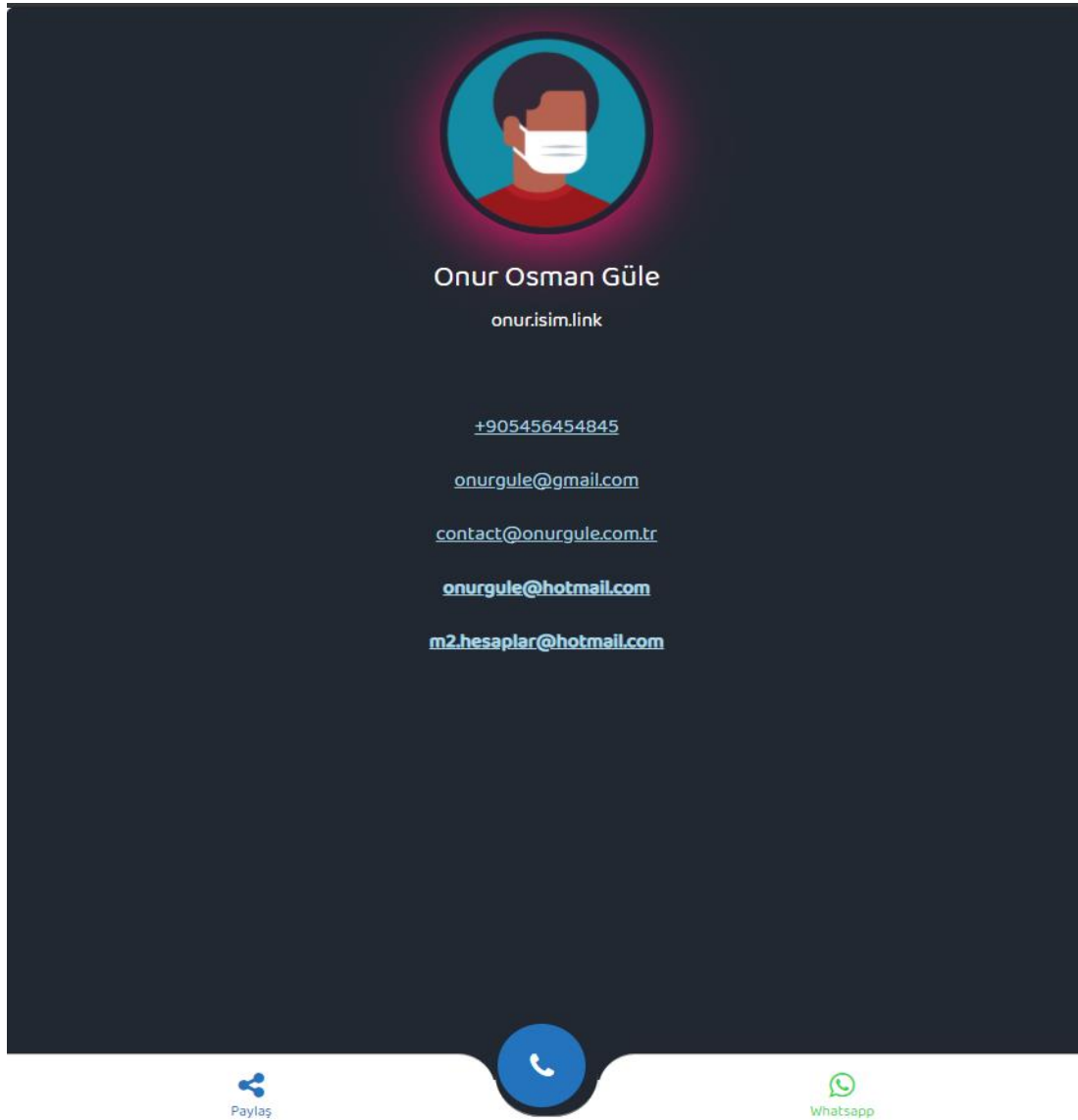


Subdomain Ekran Özellikleri Şekil 4.4

HTML, CSS, JS ve jQuery kullanarak şık bir sayfa tasarlayıp verileri aktardık.

#### 4.2.1. Subdomain Ekran Özellikleri

Subdomain ekranına giren kişiye göre gösterilen veriler değişkenlik gösterebilir. Herkese açık, yalnızca üyeler, yalnızca rehber gibi gizlilik türlerine sahip veriler o gizlilik türü kümesi içerisindeki kullanıcılara gösterilecektir. Yukarıdaki resimde giriş yapılmadığı halde görüldüğü gibi yalnızca herkese açık bilgiler gözükmemektedir. Giriş yapıldığı takdirde gözüken bilgilerde değişiklik gözlemlenebilir:



Subdomain Ekran Özellikleri Şekil 4.4

Aşağıda yer alan butonlar gösterilen verilere göre değişiklik gösterebilir. Telefon mevcut ise arama ve whatsapp butonları, mail mevcut ise mail gönderme butonu, hiçbiri mevcut değilse paylaş, giriş, oluşturma butonları eklenir.

### 4.3. Yönetim Paneli

Kullanıcılar IsimLink'te bir link sahibi olabilmek için üye olmaları gerekmektedir.

IsimLink anasayfasında Giriş veya Oluştur butonlarından birine tıklandığında Login sayfasına yönlendirilmektedir. Login sayfasından giriş ya da kayıt olunabilir.

Kayıt ya da giriş yapıldığında kullanıcı, yönetim paneline yönlendirilir.

Yönetim Ekranı Şekil 4.5

Yönetim 2.Ekran Şekil 4.6

Yönetim panelinin giriş sayfasında bazı istatistikler yer almakta:



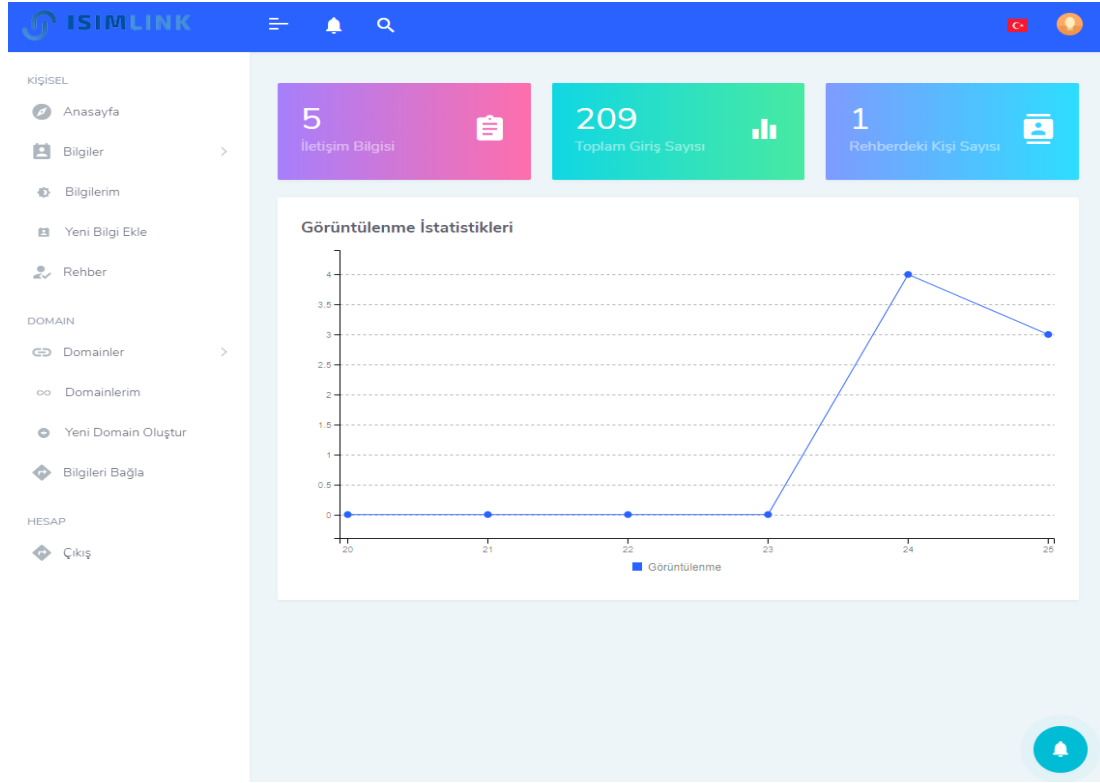
Yönetim İstatistik Şekil 4.7

1. İletişim Bilgisi Sayısı: Kaç adet iletişim bilginiz olduğunu gösterir.

2. Toplam Giriş Sayısı: Linkinize toplam giriş sayısını gösterir.
3. Rehberdeki Kişi Sayısı: Rehberinize eklediğiniz kişi sayısını gösterir.

Ayrıca son 5 günün görüntülenme istatistikleri de chart şeklinde verilir.

Yönetim paneline giriş sayfası aşağıdaki gibidir:



Yönetim Genel Ekran Şekil 4.8

Sol tarafta navigasyon menüsü mevcuttur, buradan yönetim panelinin belli sayfalarına giriş yapılabilir.

1. Anasayfa: İstatistiklerin görülebildiği giriş sayfası.
2. Bilgiler
  - 2.1. Bilgilerim: Var olan bilgilerin listelendiği sayfa.
  - 2.2. Yeni Bilgi Ekle: Panele yeni bir bilgi ekleme sayfası.
3. Rehber: Rehberdekileri görüp yeni birini ekleme sayfası.
4. Domainler
  - 4.1. Domainlerim: Mevcut linklerin gösterildiği sayfa.
  - 4.2. Yeni Domain Oluştur: Yeni link oluşturma sayfası.
5. Bilgileri Bağla: Var olan bir bilgiyi bir domaine bağlama sayfası.
6. Çıkış: IsimLink bağlantısını sonlandırma bağlantısı.

### 4.3.1. Bilgilerim Sayfası

IsimLink üzerinde daha önce eklenmiş, mevcut olan bilgilerin listelendiği sayfadır.

Bilgilerim			
IsimLink'te mevcut olan bilgileriniz			
Yeni Bilgi Ekle		Filtrele	
Bilgi Türü	Bilgi	Ekleme Tarihi	Düzenle
Telefon	+905456454845	2020-11-06 02:53:48	
Email	onurgule@gmail.com	2020-11-13 23:50:52	
Email	onurgule@hotmail.com	2020-11-19 09:45:25	
Email	contact@onurgule.com.tr	2020-11-19 10:39:30	
Email	m2.hesaplar@hotmail.com	2020-11-19 10:42:41	
« 1 »			

Bilgilerim Sayfası Şekil 4.9

### 4.3.2. Bilgi Ekle Sayfası

IsimLink üzerine yeni bir bilgi ekleme sayfası.

### Bilginizi Ekleyin

Linkinizde gösterilmesini istediğiniz bilgiyi ekleyin!

1

2

3

4

Bilgi Türü

Bilgi Girişi

Doğrulama

Link

Bilgi Türü : \*

Telefon Numarası

Ekleyeceğiniz bilgi türünü seçiniz...

Previous

Next

Bilgi Ekle Sayfası Şekil 4.10



### 4.3.3. Rehber Sayfası

IsimLink üzerinde bazı bilgilerinizi özel olarak paylaşmak istediklerinizi ekleyebileceğiniz rehber sayfası.

Rehber Sayfası Şekil 4.11

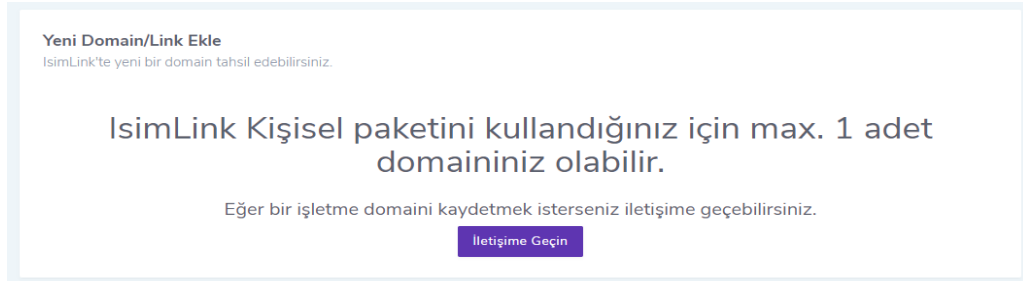
### 4.3.4. Domainler Sayfası

IsimLink üzerinde mevcut olan domainler, linklerinizi listeleyen sayfa.

Domainler Sayfası Şekil 4.12

### 4.3.5. Yeni Domain Oluşturma Sayfası

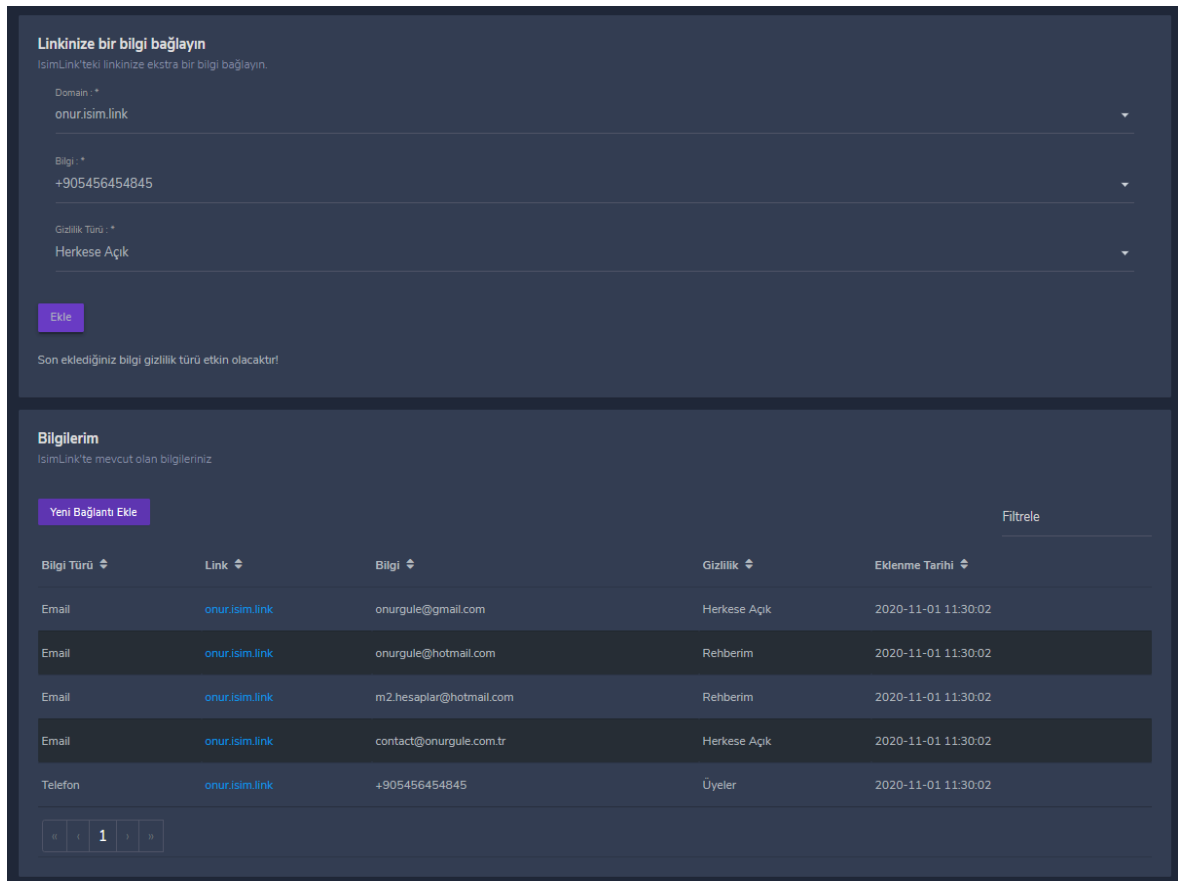
IsimLink'te birden fazla domain oluşturmak için kullanılan sayfa. Ancak şimdilik yalnızca 1 adet ile sınırlandırılmıştır.



Yeni Domain Sayfası Şekil 4.13

#### 4.3.6. Bilgileri Bağla Sayfası

Bilgileri Bağla sayfası, domain, bilgi ve gizlilik türlerini bağlayabileceğiniz, en karmaşık sayfadır.



Bilgileri Bağla Sayfası Şekil 4.14

#### 4.3.7. Çıkış

Çıkış, bir sayfa değildir, API'ya bağlı bir hyperlink'tir. Basıldığında IsimLink üyelikinden çıkış yapar, panele girebilmek için tekrar giriş yapılması gerekir.

### 4.3.8. Panel Ayarları

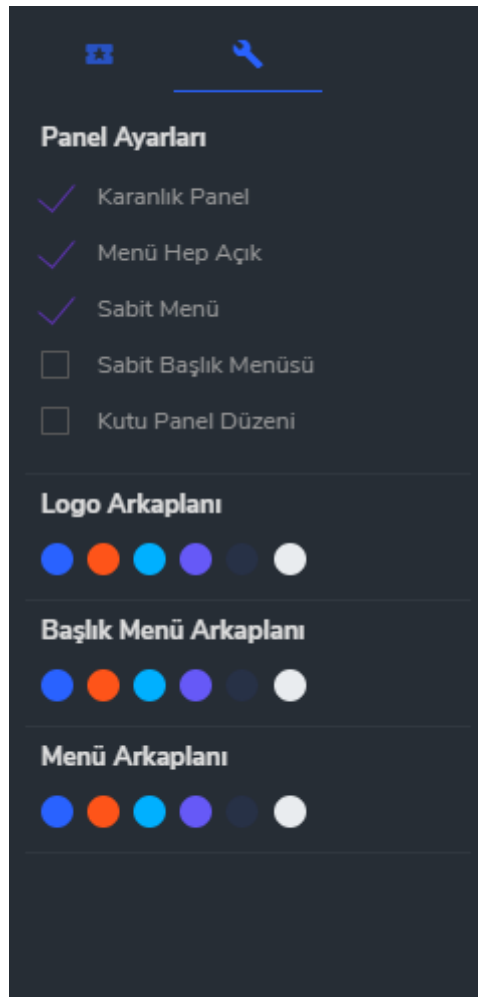
Panelin görünümünü değiştirebildiğiniz, her sayfadan ulaşılabilen ayarlardır. Panelde kullanıcının isteğine göre;

1. Karanlık Panel
2. Sol Menü Hep Açık
3. Sabit Menü (Kaymayan)
4. Sabit Başlık Menüsü
5. Kutu Panel Düzeni
6. Logo Arkaplan Rengi
7. Başlık Menüsü Arkaplan Rengi
8. Menü Arkaplan Rengi

şeklinde seçim yapılabilir.

Bu özellikler javascript ile window.localStorage üzerinde tutulup, sessionStorage içerisinde yer almaktadır.

Eğer session biter veya çıkış yapılırsa tanımlı ayarlara geri dönecektir.



Panel Ayarları Şekil 4.15

## BÖLÜM 5. MOBİL UYGULAMA

IsimLink, web tarafında mobil uyumlu olarak çalışsa da mobil uygulama olarak da yayınlamayı önemli olarak düşündük.

React Native ile daha işlevli bir uygulama yazılabileceğini düşündük ancak boyutunu daha çok önemsedik ve Flutter ile çalıştık[7].

7MB boyutunda bir Flutter uygulaması elde ettik. React Native ile muhtemelen 45MB büyüklüğünde bir paket elde edecektik.

Google Play: <https://play.google.com/store/apps/details?id=onurgule.isimlink>

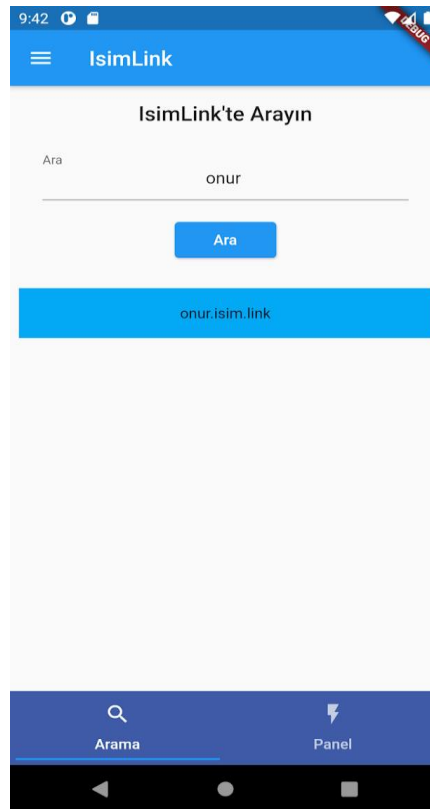
### 5.1 Uygulama Oluşturma

Uygulamamızı Flutter üzerinden oluşturduk, web paketleri yardımıyla güçlendirdik.

Uygulamamızı minimal tutmak için projemizin en temel özelliklerini ekledik, karmaşık özelliklerini webview ile sunduk.

### 5.2 Uygulamada Arama

Uygulama üzerinden arama yapmak için anasayfada yer alan arama çubuğuna talep edilen anahtar kelimeyi yazarak arama yapılabilir.

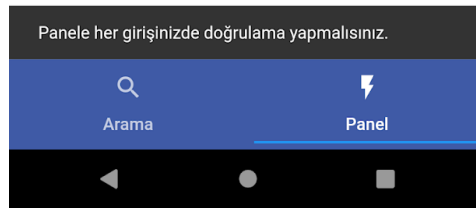
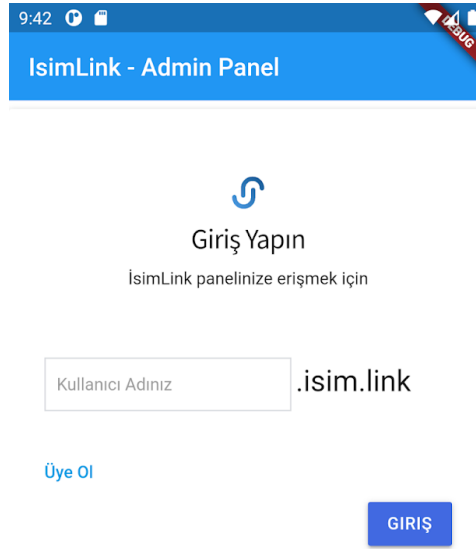


Uygulamada Arama Şekil 5.1

### 5.3 Uygulamada Yönetim Paneli

Uygulama üzerinden yönetim paneline erişmek için aşağıdaki sekmelerden seçim yapılabilir.

Panel üzerinden websitede yapılan her işlem, uygulama üzerinden de yapılabilir.



Appte Yönetim Paneli Şekil 5.2

Ancak güvenlik nedeniyle panele her girişinizde doğrulama yapma gereksinimi duyuluyor.

### 5.4 Uygulamaya Giriş

Uygulamada arama yaptıktan sonra herhangi bir linke giriş yapıldığında ziyaretçilere veya kişiye özel bilgilere erişmek için giriş yapılmalıdır.

Giriş işlemi sol taraftaki açılır menüden yapılabilir.

Uygulamaya giriş yapıldıktan sonra çıkış yapılana dek oturum kaydedilir, uygulama tekrar açıldığında tekrar giriş yapma zorunluluğu olmaz.

### 5.5 Uygulamada Linkler

Uygulamada herhangi bir arama sonucu gireceğiniz bir linkte, erişebileceğiniz tüm veriler listelenir.

Eğer telefon numarasına erişebilme yetkiniz mevcutsa arama ve whatsapp butonları açılır.

Eğer telefon numarası değil de emaile erişme yetkiniz mevcutsa yalnızca email gönderme butonu açılır.



Uygulamada Linkler Şekil 5.3

Sağ aşağıdaki paylaş butonu ile linki başka platformlarda(Whatsapp, Messenger, Facebook, Twitter, Instagram vs.) paylaşabilirsiniz.

## BÖLÜM 6. BLOCKCHAIN

IsimLink'te bir domain oluşturulduğunda bu hem MySQL veritabanına, hem de daha stabil, daha güvenilir olması amacıyla kendi oluşturduğumuz NodeJS Blockchain altyapımıza eklenir[8].

```

1 // 20201225183750
2 // http://crypt.isim.link:3001/blocks
3
4 ▾ [
5 ▾ {
6     "index": 0,
7     "previousHash": "0",
8     "timestamp": 1607625279,
9     "data": "isim.link",
10    "hash": "816534932c2b7154836da6afc367695e6337db8a921823784c14378abed4f7d7"
11  },
12 ▾ {
13    "index": 1,
14    "previousHash": "816534932c2b7154836da6afc367695e6337db8a921823784c14378abed4f7d7",
15    "timestamp": 1607627214.113,
16    "data": "onur.isim.link",
17    "hash": "4080ac07db994defec2daf979bc9841ff8a8319bee813b84224e36f5b30228fc"
18  },
19 ▾ {
20    "index": 2,
21    "previousHash": "4080ac07db994defec2daf979bc9841ff8a8319bee813b84224e36f5b30228fc",
22    "timestamp": 1607627305.961,
23    "data": "fek.isim.link",
24    "hash": "8ac7049b782c5cf597e2bf02cf892eefd9e266e474d78d267d91bbbd77824dd"
25  },
26 ▾ {
27    "index": 3,
28    "previousHash": "8ac7049b782c5cf597e2bf02cf892eefd9e266e474d78d267d91bbbd77824dd",

```

Blockchain Json Şekil 6.1

<http://crypt.isim.link:3001/blocks> endpointi ile blockchain blokları görüntülenebilir.

Blockchain teknolojisi, oluşturulan domain bilgisinin değiştirilmemesi veya silinmemesi için kullanılmıştır.

### 6.1 Cloud Sanal Makine Oluşturma

Blockchain ve verileri barındırmak için bir sanal makine oluşturmamız gerekiyor[9].

Sanal makinemizi GCP üzerinde barındırıyoruz.

**Ad** ?

Ad değiştirilemez

isimlink-blockchain

**Etiketler** ? (isteğe bağlı)

+ Etiket ekle

**Bölge** ?

Bölge değiştirilemez

europe-west3 (Frankfurt)

**Alt bölge** ?

Alt bölge kalıcıdır

europe-west3-b

**Makine yapılandırması****Makine ailesi**

Genel amaçlı

İşlem açısından optimize edilmiş

Bellek açısından optimize edilmiş

Maliyet ve esneklik için optimize edilmiş, yaygın iş yüklerine yönelik makine türleri

**Diziler**

E2

Müsaitlik durumuna göre CPU platformu seçimi

**Makine türü**

e2-medium (2 vCPU, 4 GB bellek)



vCPU

1 paylaşılan çekirdek

Bellek

4 GB

GPU sayısı

-

GCP Ekran Şekil 6.2

**6.2 VPC Ağ Konfigürasyonu**

Sanal makinemizi internete açmak için VPC ağ konfigürasyonu ile port açmamız gerekiyor.



←

Güvenlik duvarı kuralı oluşturun

Ad \*

http-3001

?

Küçük harf, sayı ve kısa çizgilere izin verilir

Açıklama

Blockchain Port

Trafik yönü ?

☒ Giriş
 ☐ Çıkış

Hedefler

Belirtilen hedef etiketler

▼

?

Hedef etiketler \*

http-3001 ✕

Kaynak filtresi

IP aralıkları

▼

?

Kaynak IP aralıkları \*

0.0.0.0/0 ✕ örneğin, 0.0.0.0/0, 192.168.2.0/24

?

İkinci kaynak filtresi

Yok

▼

?

Protokoller ve bağlantı noktaları ?

☐ Tümüne izin ver
 ☒ Belirtilen protokoller ve bağlantı noktaları

☒ tcp : 3001

☐ udp : tümü

☐ Diğer protokoller

protokoller, virgülle ayrılmış, ör: ah, sctp

VPC Ekran Şekil 6.3

NodeJS'i erişime açmak için 3001 portunu kullandık.

Oluşturduğumuz makinenin ayarlarından Ağ İletişimi sekmesinden oluşturduğumuz VPC güvenlik duvarı kuralımızın etiketini ağ etiketi olarak ekliyoruz.

**Güvenlik Duvarı** ?  
İnternet'ten belirli ağ trafiğine izin vermek için etiketler ve güvenlik duvarı kuralları ekleyin

☒ HTTP trafiğine izin ver  
☐ HTTPS trafiğine izin ver

Yönetim    Güvenlik    Diskler    **Ağ İletişimi**    Tek Kiralamalı


**Ağ etiketleri** ? (İsteğe bağlı)

http-3001 x


**Ana makine adı** ?  
Bu örnek için özel ana makine adı ayarlayın veya varsayılan adı kullanın. Seçim kalıcıdır

isimlink-blockchain.europe-west3-b.c.fluid-fiber-293913.internal

**Ağ arayüzleri** ?  
Ağ arayüzü kalıcıdır

default default (10.156.0.0/20) 

+ ağ arayüzü ekle



 Başka bir ağ arayüzü oluşturabilmek için öncelikle yeni bir ağınız olmalıdır.

VPC Güvenlik Ekran Şekil 6.4

### 6.3 Sanal Makineye SSH Bağlantısı

Sanal makinemize bağlanıp kodları aktarmak için FTP veya SSH bağlantısı yapmamız gerekiyor.

GCP, panel üzerinden SSH bağlantısı yapma kolaylığı sunuyor.

<input type="checkbox"/> Ad ^	Alt Bölge	Öneri	Kullanıcı	Dahili IP	Harici IP	Bağlantı oluştur
<input checked="" type="checkbox"/> isimlink-crypto	europe-west3-b			10.156.0.4 (nic0)	35.246.254.183 	SSH 

Sanal Makine Ekran Şekil 6.3

SSH butonuna bastığımızda web üzerinde bir terminal penceresi çıkıyor karşımıza.

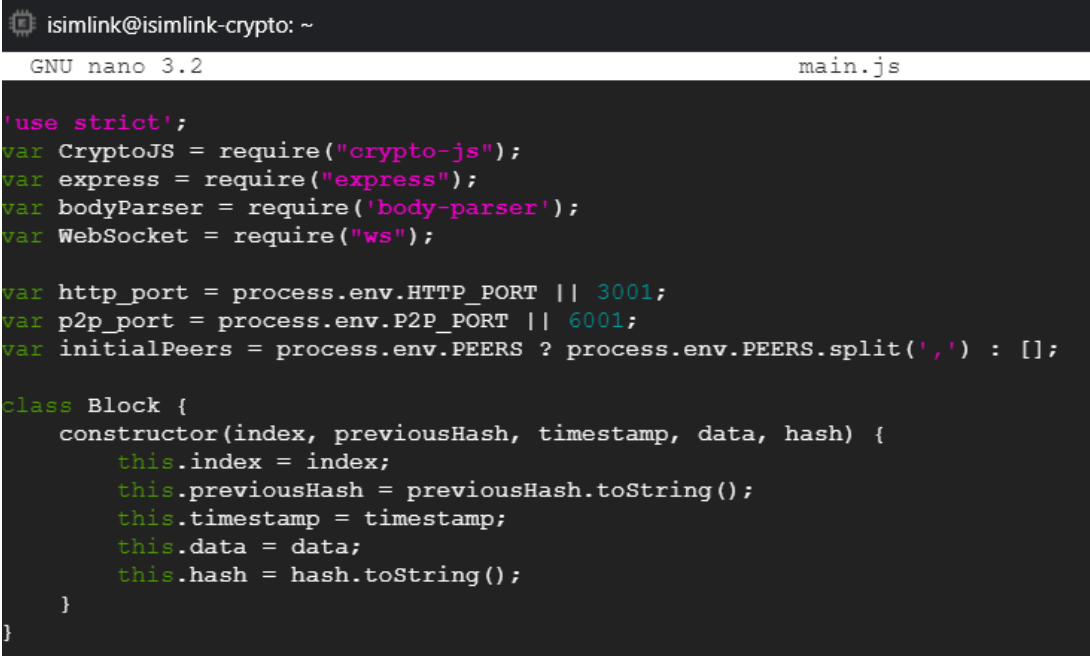
```
isimlink@isimlink-crypto: ~
root@isimlink-crypto: /home/isimlink#
```

SSH Ekran Şekil 6.5

## 6.4 NodeJS - CryptoJS Kütüphanesi

NodeJS üzerinden Blockchainimizi yazarken CryptoJS[10] kütüphanesini tercih ettik.

Block özelliklerini ihtiyaca göre belirledik.



```
isimlink@isimlink-crypto: ~
GNU nano 3.2 main.js

'use strict';
var CryptoJS = require("crypto-js");
var express = require("express");
var bodyParser = require('body-parser');
var WebSocket = require("ws");

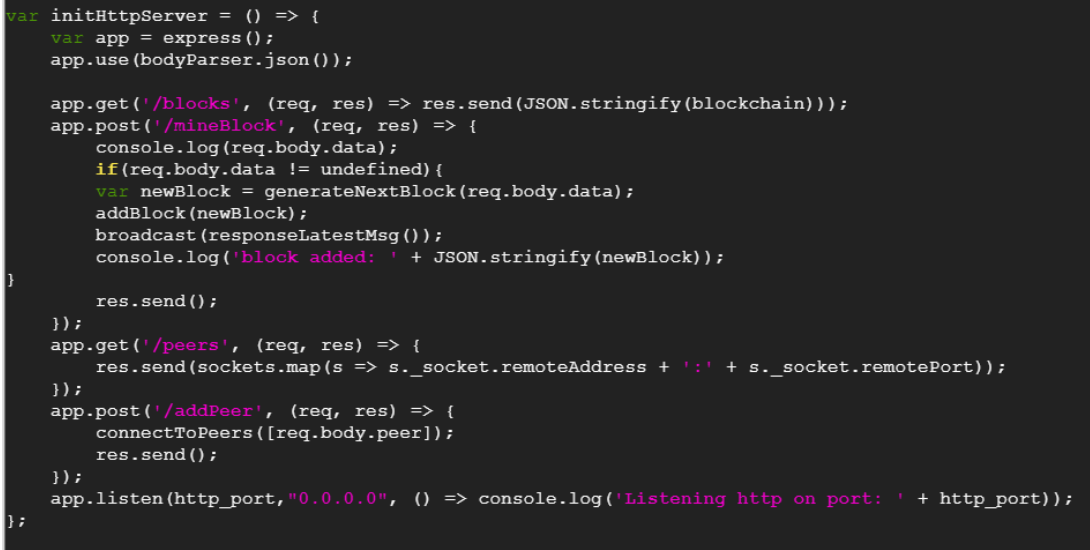
var http_port = process.env.HTTP_PORT || 3001;
var p2p_port = process.env.P2P_PORT || 6001;
var initialPeers = process.env.PEERS ? process.env.PEERS.split(',') : [];

class Block {
  constructor(index, previousHash, timestamp, data, hash) {
    this.index = index;
    this.previousHash = previousHash.toString();
    this.timestamp = timestamp;
    this.data = data;
    this.hash = hash.toString();
  }
}
```

CryptoJS Kütüphanesi Şekil 6.6

## 6.5 NodeJS - Express Kütüphanesi

NodeJS üzerinden Blockchainimizi internete sunmak için Express[11] kütüphanesini kullandık.



```
var initHttpServer = () => {
  var app = express();
  app.use(bodyParser.json());

  app.get('/blocks', (req, res) => res.send(JSON.stringify(blockchain)));
  app.post('/mineBlock', (req, res) => {
    console.log(req.body.data);
    if(req.body.data != undefined){
      var newBlock = generateNextBlock(req.body.data);
      addBlock(newBlock);
      broadcast(responseLatestMsg());
      console.log('block added: ' + JSON.stringify(newBlock));
    }
    res.send();
  });
  app.get('/peers', (req, res) => {
    res.send(sockets.map(s => s._socket.remoteAddress + ':' + s._socket.remotePort));
  });
  app.post('/addPeer', (req, res) => {
    connectToPeers([req.body.peer]);
    res.send();
  });
  app.listen(http_port, "0.0.0.0", () => console.log('Listening http on port: ' + http_port));
};
```

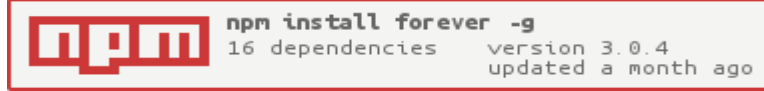
Express Kütüphanesi Şekil 6.5

## 6.6 Linux Forever Kütüphanesi

Blockchain'i sürekli çalıştırmak için Forever[12] kütüphanesi kullandık.

```
root@isimlink-crypto:/home/isimlink# forever list
info: Forever processes running
data:   uid  command  script  forever pid  id logfile  uptime
data: [0] T736 /usr/bin/node main.js 26944 26951 /root/.forever/T736.log 19:21:53:8.784999999916181
```

Forever Örnek Şekil 6.9



NPM Forever Şekil 6.10

## 6.7 Blok Kazma İşlemi

Herhangi biri IsimLink'te kayıt olduğunda ve bir domain oluşturduğunda otomatik olarak Blockchain'de bir block kazılır ve değeri domain olur. Bu block hiçbir zaman silinemez veya güncellenemez.

```
{
  "index": 1,
  "previousHash": "816534932c2b7154836da6afc367695e6337db8a921823784c14378abed4f7d7",
  "timestamp": 1607627214.113,
  "data": "onur.isim.link",
  "hash": "4080ac07db994defec2daf979bc9841ff8a8319bee813b84224e36f5b30228fc"
},
```

Kazma İşlemi Şekil 6.11

Her bloklar bir önceki block hash'ını tuttuğundan dolayı herhangi bir block silinir veya güncellenirse ondan sonraki blocklarda problem meydana geleceğinden böyle bir ihtimal mümkün değildir.

Böylelikle oluşturulan tüm domainler bir Blockchain'de saklanarak IsimLink, güvenilir bir proje haline gelecektir.

## 6.8 Blockchain'i Diğer Düğümlere Eşleme

Blockchain, eğer tek makinede, herhangi başka bir düğümlerle eşleşmeden çalışırsa makine istemli ya da istemsiz bir şekilde kapandığı anda tüm veriler silinecektir. Bunun çözümü ise başka makineleri düğüm olarak eşleştiriyoruz ve ilk makinemiz kapansa bile ikinci makinede blockchain devam edecektir. İlk makine tekrar açıldığında diğer düğümlerle eşleşecektir ve gerçek zamanlı devam edecektir.

```
listening websocket p2p port on: 6002
Listening http on port: 3002

listening websocket p2p port on: 6001
Listening http on port: 3001
Received message{"type":0}
```

WebSocket Ekran Şekil 6.12

Bir makine oluşturup addPeer fonksiyonumuzla bir websocket ile blockchainimize bir düğüm ekledik.

## BÖLÜM 7. SONUÇLAR VE ÖNERİLER

Projemiz şu anda canlı ortamda çalışmaktadır ve kaynak kodları tamamen açık şekilde Github'da paylaşılmıştır.

Github Repository: <https://github.com/onurgule/isimlink>

Bu proje, aşağıdaki özelliklerle geliştirilebilir:

- **Bilgi İsteği Gönderme**  
Herhangi kayıtlı bir ziyaretçi, herhangi bir domain sahibine verilerine erişmek için istek gönderebilir. Domain sahibi bu isteği kabul ettiğinde ziyaretçi artık istenen bilgilere hesabıyla erişebilir.
- **0850 Geçici Numara Servisi**  
Herkese açık olarak telefon numarası paylaşımı riskli olduğundan geçici numara servisi ile geçici olarak 0850 uzantılı bir numara tahsis edilir ve geçici süreyle ziyaret edilen domain sahibi ile iletişim kurulabilir.

Projenin geleceğinde ise:

- İletişim bilgisi paylaşılan e-ticaret siteleri ile anlaşılabilir ve iletişim bilgilerinin gizliliği sağlanabilir. Geçici numara servisi ile kullanıcıların telefon numaralarını paylaşma gereksinimi ortadan kalkar ve kötü amaçlı kullanımlar engellenir.
- Sosyal medya platformlarında iletişim bilgi paylaşımı IsimLink ile kısıtlanabilir, yalnızca tanıdıklara özel iletişim bilgileri verilebilir.
- Şirketlerin özel linklerine dahili linkler oluşturup şirket içi iletişim sağlanabilir.

## KAYNAKLAR

- [1] AtakDomain.com, Dünyada İlk Kaydedilen 100 Domain, 2018  
<https://www.atakdomain.com/blog/dunyada-ilk-kaydedilen-100-domain>
- [2] Wikipedia.org, Türkiye Ulusal Numaralandırma Planı, 2014  
[https://tr.wikipedia.org/wiki/T%C3%BCrkiye\\_Ulusal\\_Numaraland%C4%B1rma\\_Plan%C4%B1](https://tr.wikipedia.org/wiki/T%C3%BCrkiye_Ulusal_Numaraland%C4%B1rma_Plan%C4%B1)
- [3] KVKK, Kişisel Verilerin Korunumu Kanunu ve Uygulaması  
<https://www.kvkk.gov.tr/yayinlar/K%C4%B0%C5%9E%C4%B0SEL%20VER%C4%B0LER%C4%B0N%20KORUNMASI%20KANUNU%20VE%20UYGULAMASI.pdf>
- [4] Hostinger, MySQL Nedir?, 2020  
<https://www.hostinger.web.tr/rehberler/mysql-nedir/>
- [5] Code-Boxx, 3 Ways to Connect MySQL in PHP, 2020  
<https://code-boxx.com/connect-mysql-php/>
- [6] Libsodium Documentation  
<https://libsodium.gitbook.io/doc/quickstart>
- [7] Hackr.io, Youssef Nader, React Native vs Flutter, 2020  
<https://hackr.io/blog/react-native-vs-flutter>
- [8] Tenomad, A Blockchain in 200 Lines of Code, 2018  
<https://tenomad.com/a-blockchain-in-200-lines-of-code/>
- [9] Google Cloud, Compute Engine  
<https://cloud.google.com/compute?hl=tr>
- [10] CryptoJS Documentation,  
<https://cryptojs.gitbook.io/docs/>
- [11] ExpressJS Documentation,  
<https://expressjs.com/>
- [12] Forever Github Repository,  
<https://github.com/foreversd/forever>

## **EKLER**

### **ÖZGEÇMİŞ**

Onur Osman Güle 25.08.1999'da İstanbul'da doğdu. İlk ve orta eğitimini Bakırköy'de, lise eğitimini Avcılar'da tamamladı. 2017 yılında Haydar Akın Anadolu Teknik Lisesi, Bilişim Teknolojileri Bölümü'nden mezun oldu. 2017 yılında Sakarya Üniversitesi Bilgisayar Mühendisliği Bölümü'nü kazandı. 2020 yılında VBT Bilişim Teknolojileri A.Ş şirketinde yazılım stajını ve yine 2020 yılında Hepsiburada şirketinde donanım stajını yapmıştır.

Fatih Enis Kaya 21.03.1996'da Sakarya'da doğdu. İlk ve orta öğretimini Bursa Yıldırım'da, Liseyi Nilüfer Bahçeşehir Anadolu Lisesi'nde tamamladı. 2014 yılında Milli Piyango Anadolu lisesinden mezun oldu. 2 Yıl Viyana Teknik Üniversitesi'nde Business Informatics okudu. Sakarya Üniversitesinde 4. sınıf öğrencisidir. Stajlarını Uludağ Üniversitesi Teknokent'inde bulunan Serim Yazılım Teknolojileri'nde yapmıştır.

# BSM 401 BİLGİSAYAR MÜHENDİSLİĞİ TASARIMI

## DEĞERLENDİRME VE SÖZLÜ SINAV TUTANAĞI

KONU: Blockchain Altyapısı ile İletişim Bilgilerini Domainleştirme Sistemi

ÖĞRENCİLER (Öğrenci No/AD/SOYAD):

G171210021 / Onur Osman / GÜLE

G171210375 / Fatih Enis / KAYA

Değerlendirme Konusu	İstenenler	Not Aralığı	Not
<b>Yazılı Çalışma</b>			
Çalışma klavuza uygun olarak hazırlanmış mı?	x	0-5	
<b>Teknik Yönden</b>			
Problemin tanımı yapılmış mı?	x	0-5	
Geliştirilecek yazılımın/donanımın mimarisini içeren blok şeması (yazılımlar için veri akış şeması (dfd) da olabilir) çizilerek açıklanmış mı?			
Blok şemadaki birimler arasındaki bilgi akışına ait model/gösterim var mı?			
Yazılımın gereksinim listesi oluşturulmuş mu?			
Kullanılan/kullanılması düşünülen araçlar/teknolojiler anlatılmış mı?			
Donanımların programlanması/konfigürasyonu için yazılım gereksinimleri belirtilmiş mi?			
UML ile modelleme yapılmış mı?			
Veritabanları kullanılmış ise kavramsal model çıkarılmış mı? (Varlık ilişki modeli, noSQL kavramsal modelleri v.b.)			
Projeye yönelik iş-zaman çizelgesi çıkarılarak maliyet analizi yapılmış mı?			
Donanım bileşenlerinin maliyet analizi (prototip-adetli seri üretim vb.) çıkarılmış mı?			
Donanım için gerekli enerji analizi (minimum-uyku-aktif-maksimum) yapılmış mı?			
Grup çalışmalarında grup üyelerinin görev tanımları verilmiş mi (iş-zaman çizelgesinde belirtilebilir)?			
Sürüm denetim sistemi (Version Control System; Git, Subversion v.s.) kullanılmış mı?			
Sistemin genel testi için uygulanan metotlar ve iyileştirme süreçlerinin dökümü verilmiş mi?			
Yazılımın sızma testi yapılmış mı?			
Performans testi yapılmış mı?			
Tasarımın uygulamasında ortaya çıkan uyumsuzluklar ve aksaklıklar belirtilerek çözüm yöntemleri tartışılmış mı?			
Yapılan işlerin zorluk derecesi?	x	0-25	
<b>Sözlü Sınav</b>			
Yapılan sunum başarılı mı?	x	0-5	
Soruları yanıtlama yetkinliği?	x	0-20	
<b>Devam Durumu</b>			
Öğrenci dönem içerisindeki raporlarını düzenli olarak hazırladı mı?	x	0-5	
<b>Diğer Maddeler</b>			
<b>Toplam</b>			

DANIŞMAN: DOÇ. DR. AHMET ZENGİN

DANIŞMAN IMZASI: