

TÜBİTAK BİLGEM BLOKZİNCİR ARAŞTIRMA LABORATUVARI



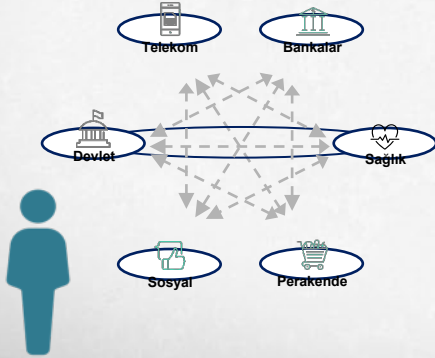
<http://blockchain.tubitak.gov.tr/>

- Literatür takibi ve bilgi havuzu oluşturma
- Blokzincir ölçeklenebilirlik, güvenlik ve mahremiyet sorunları detaylı incelenmektedir
- Platformlar, teorik ve deneysel olarak araştırılmaktadır.
 - Hyperledger Fabric, Indy, Corda, Ethereum, Bitcoin, IOTA
 - Örnek uygulamalar geliştirilmektedir
 - Test ağları kurulmaktadır

Faaliyetler

Araştırma ve Geliştirme

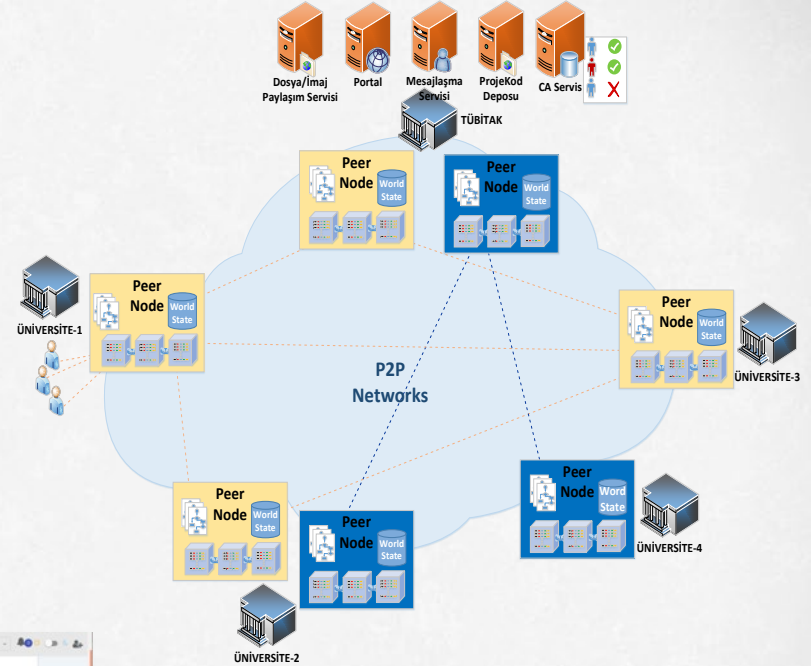
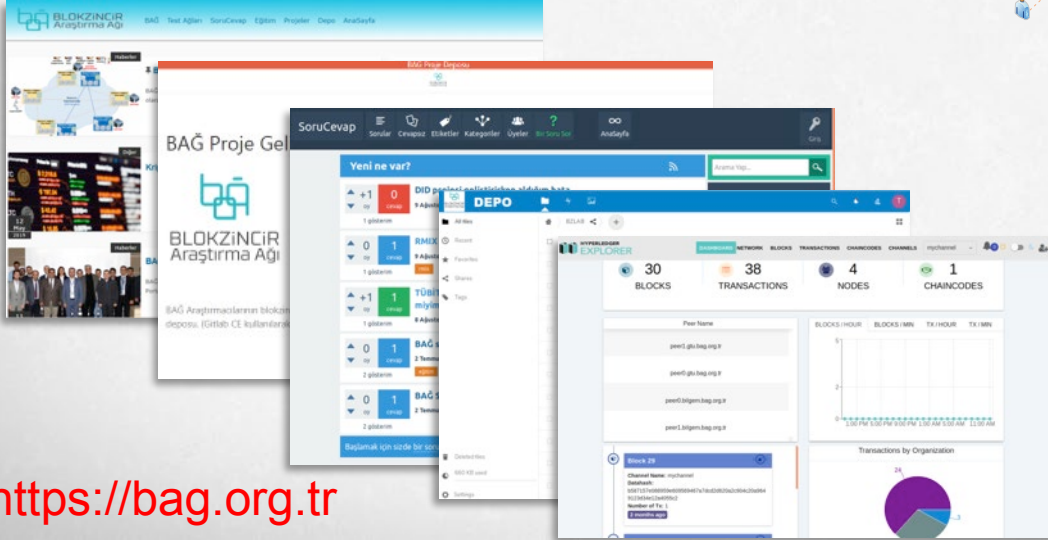
- Dijital Kimlik ve DID (Decentralized Digital ID) çalışmaları
 - DID İç Projesi
 - BAĞ-DID Kalkınma Bakanlığı Proje Önerisi (2020-2022)
- Blokzincir tabanlı Dijital kimlik yönetim sistemleri araştırılmaktadır
- Yeni nesil Blokzincir Tabanlı Dijital Kimlik Yönetim Sistemi geliştirilmesi
- TCKK e-Kimlik ile entegrasyon çalışmaları



- Akademik amaçlı başladı, firmaların da katılma isteği var
- Kendi bağımsız yönetim modeline sahip
- Araştırma amaçlı çalıştırılan blokzincir platformları
- Üye kuruluşların sağladığı kaynak havuzu ile genişleyen
- Blokzincir projeleri
 - Proje fikirleri olgunlaştırma
 - Proje ortağı bulmak
 - Geliştirme ve test aşında devreye alma
- Araştırma ve geliştirme odaklı tartışmalar
- Araştırmacıların ve kurumların birbirini tanınması



- Bilgi ve Haber Portalı
- Proje geliştirme ortamı
- Forumlar
- Soru Bankası
- Dosya Paylaşım Servisi
- Test Ağları (Ethereum, Indy, Fabric,...)



BAĞ platformunun önemi

- Blokzincir teknolojileri konusunda ülkemizin **rekabet gücünü** yüksek tutmak
- Ülkemizdeki blokzincir alanındaki **araştırmacı envanterini daha etkin kullanmak** ve hızla artırmak
- Ülkemizin ihtiyaçlarını ve süreçlerini dikkate alarak ülkemiz için **blokzincir dönüşümüne** katkı sağlamak
- Ulusal blokzincir **bilgi havuzunun** daha hızlı genişlemesi sağlamak
- Farklı üniversitelerin farklı bileşenler üzerinde uzmanlaşması sonucunda **büyük projeler için işbirlikleri** oluşturulabilmesine katkı sağlamak



26 Eylül 2019

*Ethereum Üzerinde
Akıllı Kontrat Geliştirme
Eđitimi*

TANSEL KAYA

Geleneksel Sözleşmeler



Otomatlar

A canonical real-life example, which we might consider to be the primitive ancestor of smart contracts, is the humble vending machine.

1997, Formalizing and Securing Relationships on Public Networks,
Nick Szabo



Nick Szabo

Bitcointalk

20.07.2010, Satoshi, Founder, Sr. Member:

Bitcoin is an implementation of Wei Dai's b-money proposal on Cypherpunks in 1998 and Nick Szabo's Bitgold proposal

27.12.2008, Nick Szabo: *Bit Gold would greatly benefit from a demonstration, an experimental market (with e.g. a trusted third party substituted for the complex security that would be needed for a real system). Anybody want to help me code one up?*

03.01.2009, Bitcoin Genesis Block: The Times 03/Jan/2009

Chancellor on brink of second bailout for banks.



Otonom
Programlar

Şeffaf

İz Bırakır
Değiştirilemez

Para Gönderir,
Alır ve Depolar

Geri
Döndürülemez

İnternete
Diğer
Sözleşmelere
Bağlı

SMART CONTRACT PLATFORMS

PUBLIC (PERMISSIONLESS)



ETHEREUM



EOS



LISK



RADIX



UBIQ



ETHEREUM
CLASSIC



NEO



TEZOS



CALYSTO



QTUM



BITSHARES



NXT



WAVES



DFINITY



NEBLIO



EXPANSE



OMNI LAYER



BYTEBALL



POA NETWORK



COUNTERPARTY



URBIT



ROOTSTOCK

PRIVATE (PERMISSIONED)



UNIVERSA



KOMODO

c.rda

CORDA



HYPERLEDGER



NEM



CIPHER

MONAX

MONAX



QUORUM



MIJIN



EXONUM



WANCHAIN



AETERNITY

EITHER PRIVATE OR PUBLIC



SEELE



MATRIX



WANCHAIN



APLA



CARDANO



ethereum

Imogen Heap



UJO MUSIC

Imogen Heap

Junge Deutsche Philharmonie



Tiny Human



download
\$0.60

Mastering Engineer

Credits Stems Lyrics Inspiration Licensing ▾

Tiny Human

Tiny Human Transactions

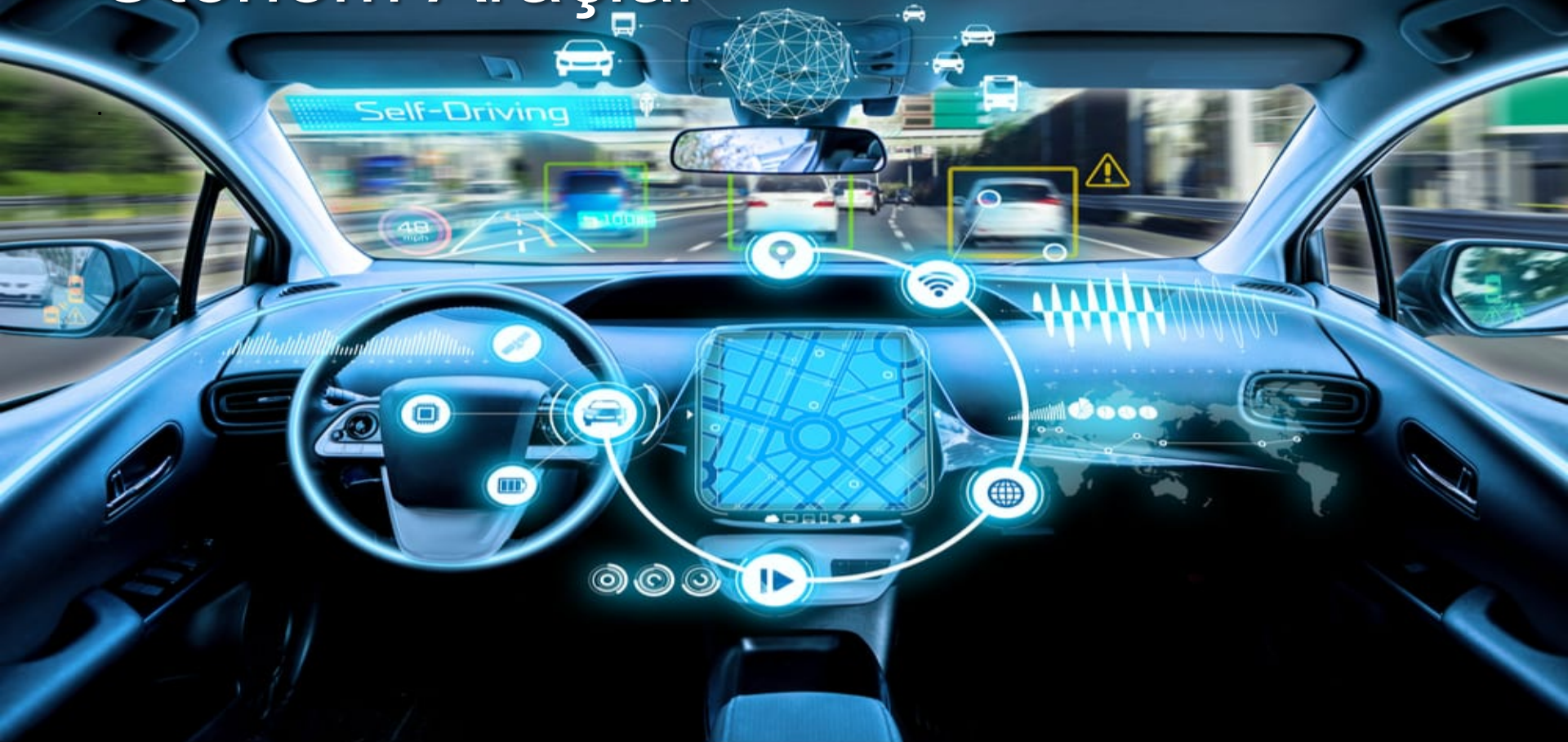
Transactions

| Payee Id | License Type | Block Number | Amount (ETH) |
|-------------------------------|--------------|--------------|---------------|
| 0x773bfc27b1b0a76ccbd486c... | DOWNLOAD | 1541107 | 0.04507888805 |
| 0x8f0455e03228ab62d16b47... | DOWNLOAD | 1443161 | 0.06204756980 |
| 0x9b8ce09f1c0673afaf6a611e... | DOWNLOAD | 1413506 | 0.07672634271 |
| 0xb438792a639eafcb9ff40d9... | DOWNLOAD | 1412121 | 0.08130081300 |
| 0xe2b78f7c509a23321f5ba08... | DOWNLOAD | 1354909 | 0.06349206349 |
| 0xbb1e11a9db28804c0e8b70... | DOWNLOAD | 1344368 | 0.07308160779 |
| 0x60f1701d67c7017183e3cfd... | DOWNLOAD | 1327485 | 0.08174386920 |
| 0x5ec245ca2b7bb420a2130f5... | DOWNLOAD | 1286224 | 0.05509641873 |
| 0x7b53c10b0572a659e2b8f1a... | DOWNLOAD | 1249477 | 0.05046257359 |

Tiny Human Distribution

| | | |
|---------------------------------------|-------|-----------------|
| Across all Licenses | 100% | \$110.44 |
| Performer: Imogen Heap | 91.2% | \$100.74 |
| Performer: Stephanie Appelhans | 1.3% | \$1.48 |
| Performer: Diego Romano | 1.3% | \$1.48 |
| Performer: Yasin Gundisch | 1.3% | \$1.48 |
| Performer: Hoang Nguyen | 1.3% | \$1.48 |
| Performer: Simon Minshall | 1.3% | \$1.48 |
| Performer: David Horwich | 1.3% | \$1.48 |

Otonom Araçlar



KİRA SÖZLEŞMESİ

(Meskenler/Konutlar İçin)

| | |
|--|---|
| DAİRESİ | (BURAYA TAŞINMAZIN BAĞLI OLDUĞU TAPU DAİRESİ YAZILACAK) |
| Mahallesi | (Burayı tapudaki mahallesi hanesinde yazan yazılacak) |
| Cadde/Sokağı | (Cadde veya sokak yazılacak) |
| Numarası | (Apartman kapı numarası ve Daire kapı numarası yazılacak) |
| Kiralananın Cinsi | Apartman dairesi |
| Kiralayan | (Kıraya verenin tam adı ve soyadı yazılacak) |
| Kiralayanın T.C. Kimlik No | (Kıraya verenin T.C. Kimlik no yazılacak) |
| Kiralayanın Adresi | (Kıraya verenin tam adresi yazılacak) |
| Kiracı | (Kıracının tam adı ve soyadı yazılacak) |
| Kıracının T.C. Kimlik No | (Kıracının T.C. Kimlik no yazılacak) |
| Kıracının Adresi | (Kıracının tam adresi yazılacak) |
| Akdin Başlangıç Tarihi | (Kıra sözleşmesinin başlangıç tarihi yazılacak) |
| Akdin Süresi | (Kıra sözleşmesinin süresi yazılacak) |
| Yıllık Kıra Bedeli | (Kıra bedelinin yıllık toplamı yazılacak) |
| Aylık Kıra Bedeli | (Her ay ödenecek kıra bedeli yazılacak) |
| Kıra Bedelinin Odeme Şekli | Her ayın beşinci günü akşamına kadar peşin olarak |
| Kiralananın Kullanım Şekli | Yalnızca mesken (konut) olarak |
| Kiralananın Durumu | Sağlam, tam ve kullanılmaya elverişlidir |
| Kiralananla Birlikte Teslim Edilen Demirbaşlar | marka kombi, elektrik (endeks.....), su (endeks.....) ve doğalgaz (endeks.....) sayaçları, su ve doğalgaz sayaçlarına ait kartlar ve kiralananın apartman kapısı ve daire kapısının ikişer adet anahtarı. |

(imza)

(imza)

(imza)

ÖZEL KOŞULLAR

- Kiralanan alt kiraya verilemez, ortak alınamaz; devir ve temlik edilemez.
 - Kiralanan, mesken dışında herhangi bir amaçla kullanılamaz.
 - Kiralanan, kiracı, eşi ve çocuklarının dışında kimse kalamaz.
 - Kıra bedelleri, her ayın beşinci günü akşamına kadar, kiralayanın numaralı hesabına yatırılacaktır. Şubesindeki Kira parasının başka bir şubeden havale edilmesi halinde, aynı süre içinde hesapta olacak şekilde işlem yaptırılacak olup, aksi durumda temerrüt hükümleri uygulanacaktır. Bir ayın kira parasının ödenmemesi halinde dönem sonuna kadar işleyecek kira paralarının tümü muacceliyet kazanacaktır.
 - Kiralananın kapıcı/kaloriferci, yaktı ve genel giderleri kiracı tarafından ödenecektir.
 - Kapılar, pencereler, sıhhi tesisat araçları sağlam, tam ve kullanılmaya elverişli olarak teslim edilmiştir.
 - Kiracı, kiralananı özenle kullanacak; kiralayan da gerekli onarımları, kiracının uyarısından itibaren on gün içinde -teknik olanaksızlar hariç- yaptıracaktır.
 - Kiracı, elektrik aboneliğini kendi adına yaptıracak, sözleşme sonunda hesabı kestirerek, buna ilişkin makbuz fotokopisini kiralayana verecektir.
 - Kiracı, üç gün içinde, aile beyannamesini mahalle muhtarlığına verecektir.
 - Sözleşmeden doğacak uyuşmazlıklardan dolayı, Eskişehir Mahkemeleri ve icra müdürlükleri yetkili olacaktır.
- İşbu, on sekiz genel ve on özel koşuldan oluşan sözleşmeyi, hür iradelerimizle iki nüsha olarak imzaladık. Tarih.

Kiracı:
(imza)
(isim)

Müteselsil Borçlu ve Kefil:
(imza)
(isim)

Kiralayan:
(imza)
(isim)

```

pragma solidity ^0.4.0;
contract RentalAgreement {
    /* This declares a new complex type which will hold the
rents*/
    struct PaidRent {
        uint id; /* The paid rent id*/
        uint value; /* The amount of rent that is paid*/
    }

    PaidRent[] public paidrents;

    uint public createdTimestamp;

    uint public rent;
    /* Combination of zip code and house number*/
    string public house;

    address public landlord;

    address public tenant;
    enum State {Created, Started, Terminated}
    State public state;

    function RentalAgreement(uint _rent, string _house) {
        rent = _rent;
        house = _house;
        landlord = msg.sender;
        createdTimestamp = block.timestamp;
    }
}

```

```

modifier require(bool _condition) {
    if (!_condition) throw;
    _;
}
modifier onlyLandlord() {
    if (msg.sender != landlord) throw;
    _;
}
modifier onlyTenant() {
    if (msg.sender != tenant) throw;
    _;
}
modifier inState(State _state) {
    if (state != _state) throw;
    _;
}

/* We also have some getters so that we can read the values
from the blockchain at any time */
function getPaidRents() internal returns (PaidRent[]) {
    return paidrents;
}

function getHouse() constant returns (string) {
    return house;
}

function getLandlord() constant returns (address) {
    return landlord;
}
}

```

```

function getTenant() constant returns (address) {
    return tenant;
}

function getRent() constant returns (uint) {
    return rent;
}

function getContractCreated() constant returns (uint) {
    return createdTimestamp;
}

function getContractAddress() constant returns (address) {
    return this;
}

function getState() returns (State) {
    return state;
}

/* Events for DApps to listen to */
event agreementConfirmed();

event paidRent();

event contractTerminated();

/* Confirm the lease agreement as tenant*/
function confirmAgreement()
inState(State.Created)

```

```

event contractTerminated();

/* Confirm the lease agreement as tenant*/
function confirmAgreement()
inState(State.Created)
require(msg.sender != landlord)
{
    agreementConfirmed();
    tenant = msg.sender;
    state = State.Started;
}

function payRent()
onlyTenant
inState(State.Started)
require(msg.value == rent)
{
    paidRent();
    landlord.send(msg.value);
    paidrents.push(PaidRent({
    id : paidrents.length + 1,
    value : msg.value
    }));
}

/* Terminate the contract so the tenant can't pay rent
and the contract is terminated */
function terminateContract()
onlyLandlord
{
    contractTerminated();
}

```

```
→ ~ ganache-cli -e 10000 -m 'sustain seek diet attend ready neutral damage join food extra monkey usage'
Ganache CLI v6.1.8 (ganache-core: 2.2.1)
```

Available Accounts

- ```
=====
(0) 0xee398666ca860dfb7390b5d73ee927e9fb41a60a (~10000 ETH)
(1) 0x0e626b761fef2092f26cf431b1f8840c19a85bb4 (~10000 ETH)
(2) 0x0d3cf8aedf0f84ae07a0a44fbfcabd1e04826b71 (~10000 ETH)
(3) 0x2a3dd934b167d1b7fe9d7c971b383bf5545ea4bf (~10000 ETH)
(4) 0xf26a2c21eafa74b9e549ef61b95f75b4ddc6b15f (~10000 ETH)
(5) 0x49903d54c2585fa80f512645492b3fce5448de00 (~10000 ETH)
(6) 0xeb91e372c064168d522b5015fde9c59b38e2567c (~10000 ETH)
(7) 0x4b06c7b64f3e293356f9419015f087312bf88219 (~10000 ETH)
(8) 0x91e5153577569fd9c8b7748ff15ce2894f535c7b (~10000 ETH)
(9) 0x533f22f3601e07392dcf220c047b476a32f60ce7 (~10000 ETH)
```

Private Keys

- ```
=====
(0) 0xb0f8338f9c68593144a93bd5cf4a397bd141d88636d9c50ac18e587848f98d7c
(1) 0xa885d56d1ce4cb666454ac296a3429735c3a887750211a4bc9545531d6a228d6
(2) 0xebf1e93d781e6ca5c8aca9ca7827f094f6d0079bd8aa8086405f25385f274eea8
(3) 0x789cf7ba9d92f9fac3b7b18f87488cc8d059a5130e34832c6e19bb20b4234eec
(4) 0xa11160dfbc930d6e1804ec23f3aa162e5f1ba4089781ab5b4cc1d7fa0666a1d8
(5) 0xaaacf0a910468246843fbbae288f918f1b42a2205fe82e3bf3b9923b74d5b869b
(6) 0x4ee5c56aaf4273721e042c04f9e35ecbaf9cfb2de9ce7347d02dc265b3fce26
(7) 0xe7fbef6e523d0de7180137cee767749b67feedd6b9b95b2f79d49cc584b94762
(8) 0xa52dc95a1e3635872d9cf86376d887fe5fd3243b3908bf80e8cf3e8fa746d83a
(9) 0xf08cb8398a7a6f2d12b8d77a5af66c82033ed04357c7d16a732f309d6c3e9c17
```

HD Wallet

```
=====
Mnemonic:      sustain seek diet attend ready neutral damage join food extra monkey usage
Base HD Path:  m/44'/60'/0'/0/{account_index}
```

Gas Price

```
=====
20000000000
```

Gas Limit

```
=====
6721975
```

Listening on 127.0.0.1:8545

You can now view **eth-ips** in the browser.

```
Local:      http://localhost:3000/
On Your Network: http://192.168.1.91:3000/
```

Note that the development build is not optimized.
To create a production build, use `npm run build`.

□

→ ~ truffle

Truffle v5.0.13 - a development framework for Ethereum

Usage: truffle <command> [options]

Commands:

- build Execute build pipeline (if configuration present)
- compile Compile contract source files
- config Set user-level configuration options
- console Run a console with contract abstractions and commands available
- create Helper to create new contracts, migrations and tests
- debug Interactively debug any transaction on the blockchain (experimental)
- deploy (alias for migrate)
- develop Open a console with a local development blockchain
- exec Execute a JS module within this Truffle environment
- help List all commands or provide information about a specific command
- init Initialize new and empty Ethereum project
- install Install a package from the Ethereum Package Registry
- migrate Run migrations to deploy contracts
- networks Show addresses for deployed contracts on each network
- obtain Fetch and cache a specified compiler
- opcode Print the compiled opcodes for a given contract
- publish Publish a package to the Ethereum Package Registry
- run Run a third-party command
- test Run JavaScript and Solidity tests
- unbox Download a Truffle Box, a pre-built Truffle project
- version Show version number and exit
- watch Watch filesystem for changes and rebuild the project automatically



INKUBE

EĞİTMENLERİMİZ



TANSEL KAYA, MSc



MÜH. ŞEN FERHAT



Av. MUTLUCAN SOLAK



Av. ELÇİN KARATAY



AFŞAR AKAL, MSc



INKUBE

EKİBİMİZ



BERKE ŞİPKA



MERT BAŞER



TUNA ÖZEN



SİNAN KOÇ



Av. BURÇAK ÜNAL



KEMAL APAYDIN