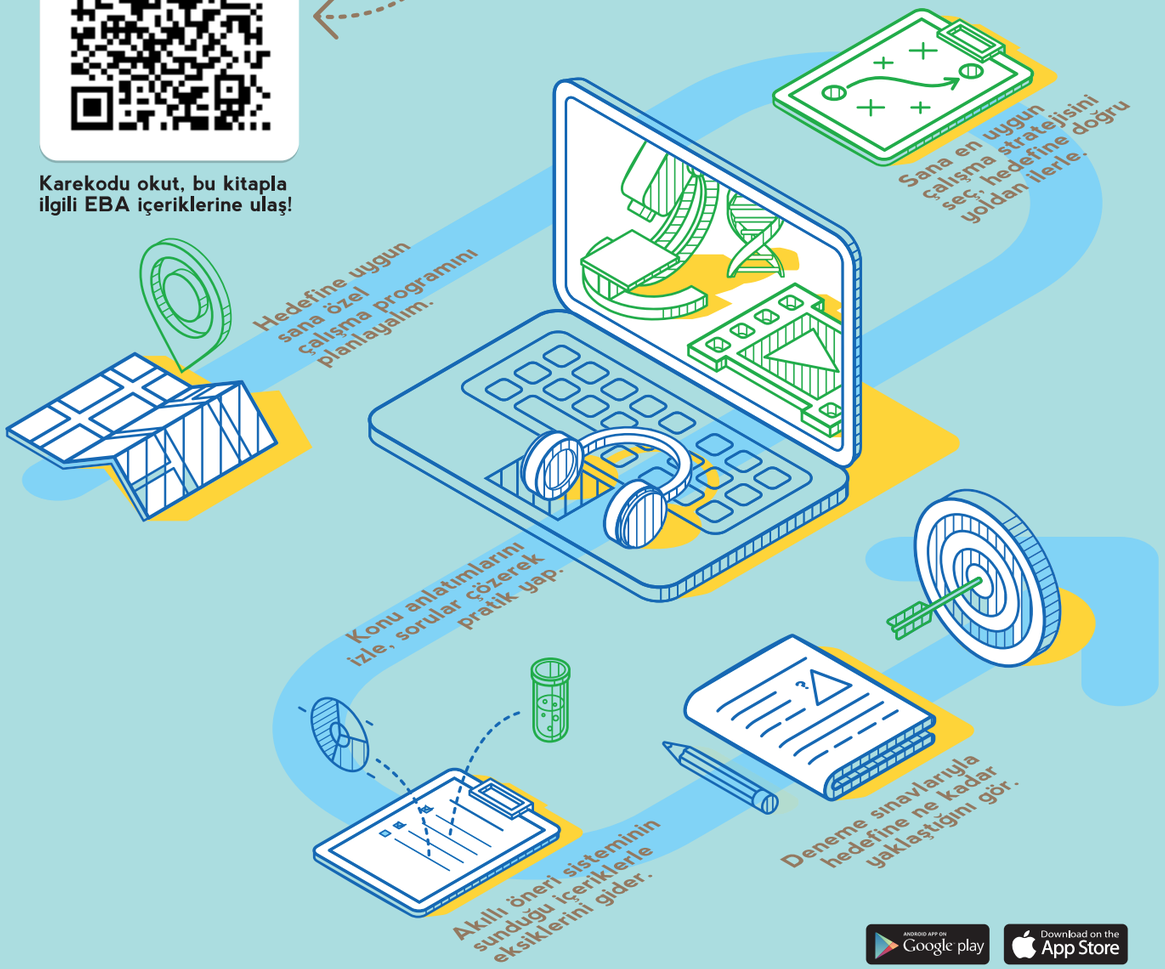
Bu kitaba sığmayan daha neler var!

Karekodu okut, bu kitapla ilgili EBA içeriklerine ulaş!



Google play | Download on the App Store

BİLİŞİM TEKNOLOJİLERİ ALANI

BLOK ZİNCİRİ

11-12

DERS KİTABI

BLOK ZİNCİRİ

MESLEKİ VE TEKNİK ANADOLU LİSESİ
BİLİŞİM TEKNOLOJİLERİ ALANI

T.C. MİLLÎ EĞİTİM BAKANLIĞI

MESLEKİ VE TEKNİK ANADOLU LİSESİ

BİLİŞİM TEKNOLOJİLERİ ALANI

BLOK ZİNCİRİ

DERS KİTABI

YAZARLAR

Dr. Arzu KİLİTCİ CALAYIR
Ahmet KARBUKAN
Ali GÖKDEMİR
Özgü ASKER

DEVLET KİTAPLARI

MİLLÎ EĞİTİM BAKANLIĞI YAYINLARI 0000
YARDIMCI VE KAYNAK KİTAPLAR DİZİSİ 0000

HAZIRLAYANLAR

Dil Uzmanı
Erman Erşan YORGANCILAR

Program Geliştirme Uzmanı
Emel DOLDUR

Ölçme ve Değerlendirme Uzmanı
Fatma YILMAZ

Rehberlik Uzmanı
Gülşen YALIN

Görsel Tasarım Uzmanı
Sermin FIRAT SOYDAN

ISBN:

Millî Eğitim Bakanlığının ..... gün ve ..... sayılı oluru ile Meslekî ve Teknik Eğitim Genel Müdürlüğünce ders materyali olarak hazırlanmıştır.

İSTİKLÂL MARŞI

Korkma, sönmez bu şafaklarda yüzen al sancak;
Sönmeden yurdumun üstünde tüten en son ocak.
O benim milletimin yıldızıdır, parlayacak;
O benimdir, o benim milletimindir ancak.

Çatma, kurban olayım, çehreni ey nazlı hilâl!
Kahraman ırkıma bir gül! Ne bu şiddet, bu celâl?
Sana olmaz dökülen kanlarımız sonra helâl.
Hakkıdır Hakk'a tapan milletimin istiklâl.

Ben ezelden beridir hür yaşadım, hür yaşarım.
Hangi çılgın bana zincir vuracakmış? Şaşarım!
Kükremiş sel gibiyim, bendimi çiğner, aşarım.
Yırtarım dağları, enginlere sığmam, taşarım.

Garbın âfâkını sarmışsa çelik zırhlı duvar,
Benim iman dolu göğsüm gibi serhaddim var.
Ulusun, korkma! Nasıl böyle bir imanı boğar,
Medeniyyet dediğin tek dişi kalmış canavar?

Arkadaş, yurduma alçakları uğratma sakın;
Siper et gövdeni, dursun bu hayâsızca akın.
Doğacaktır sana va'dettiği günler Hakk'ın;
Kim bilir, belki yarın, belki yarından da yakın.

Bastığın yerleri toprak diyerek geçme, tanı:
Düşün altındaki binlerce kefensiz yatanı.
Sen şehit oğlusun, incitme, yazıktır, atanı:
Verme, dünyaları alsan da bu cennet vatanı.

Kim bu cennet vatanın uğruna olmaz ki feda?
Şüheda fışkıracak toprağı sıksan, şüheda!
Cânı, cânânı, bütün varımı alsın da Huda,
Etmesin tek vatanımdan beni dünyada cüda.

Ruhumun senden İlâhî, şudur ancak emeli:
Değmesin mabedimin göğsüne nâmahrem eli.
Bu ezanlar -ki şehadetleri dinin temeli-
Ebedî yurdumun üstünde benim inlemeli.

O zaman vecd ile bin secde eder -varsa- taşım,
Her cerîhamdan İlâhî, boşanıp kanlı yaşım,
Fışkırır ruh-ı mücerret gibi yerden na'şım;
O zaman yükselerek arşa değer belki başım.

Dalgalan sen de şafaklar gibi ey şanlı hilâl!
Olsun artık dökülen kanlarımın hepsi helâl.
Ebediyyen sana yok, ırkıma yok izmihlâl;
Hakkıdır hür yaşamış bayrağımın hürriyyet;
Hakkıdır Hakk'a tapan milletimin istiklâl!

Mehmet Âkif Ersoy

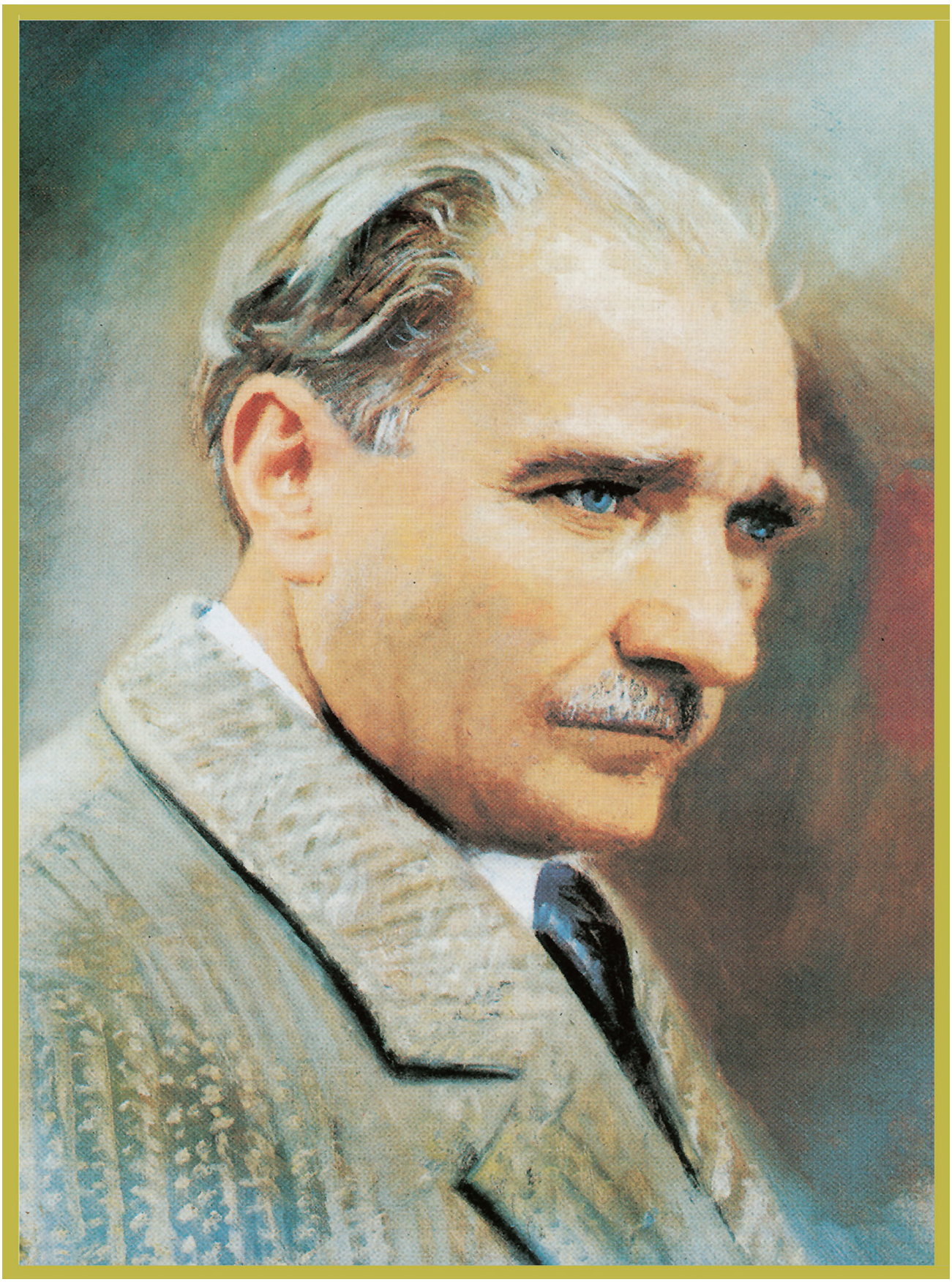
GENÇLİĞE HİTABE

Ey Türk gençliği! Birinci vazifen, Türk istiklâlini, Türk Cumhuriyetini, ilelebet muhafaza ve müdafaa etmektir.

Mevcudiyetinin ve istikbalinin yegâne temeli budur. Bu temel, senin en kıymetli hazinendir. İstikbalde dahi, seni bu hazineden mahrum etmek isteyecek dâhilî ve hâricî bedhahların olacaktır. Bir gün, istiklâl ve cumhuriyeti müdafaa mecburiyetine düşersen, vazifeye atılmak için, içinde bulunacağın vaziyetin imkân ve şeraitini düşünmeyeceksin! Bu imkân ve şerait, çok namüsait bir mahiyette tezahür edebilir. İstiklâl ve cumhuriyetine kastedecek düşmanlar, bütün dünyada emsali görülmemiş bir galibiyetin mümessili olabilirler. Cebren ve hile ile aziz vatanın bütün kaleleri zapt edilmiş, bütün tersanelerine girilmiş, bütün orduları dağıtılmış ve memleketin her köşesi bilfiil işgal edilmiş olabilir. Bütün bu şeraitten daha elîm ve daha vahim olmak üzere, memleketin dâhilinde iktidara sahip olanlar gaflet ve dalâlet ve hattâ hıyanet içinde bulunabilirler. Hattâ bu iktidar sahipleri şahsî menfaatlerini, müstevlîlerin siyasî emelleriyle tevhit edebilirler. Millet, fakr u zaruret içinde harap ve bîtap düşmüş olabilir.

Ey Türk istikbalinin evlâdı! İşte, bu ahval ve şerait içinde dahi vazifen, Türk istiklâl ve cumhuriyetini kurtarmaktır. Muhtaç olduğun kudret, damarlarındaki asil kanda mevcuttur.

Mustafa Kemal Atatürk



MUSTAFA KEMAL ATATÜRK

İÇİNDEKİLER

KİTABIN TANITIMI

Öğrenme biriminde neler öğrenileceğinin ön bilgilerini gösterir.

Öğrenme biriminde yer alan konuları gösterir.

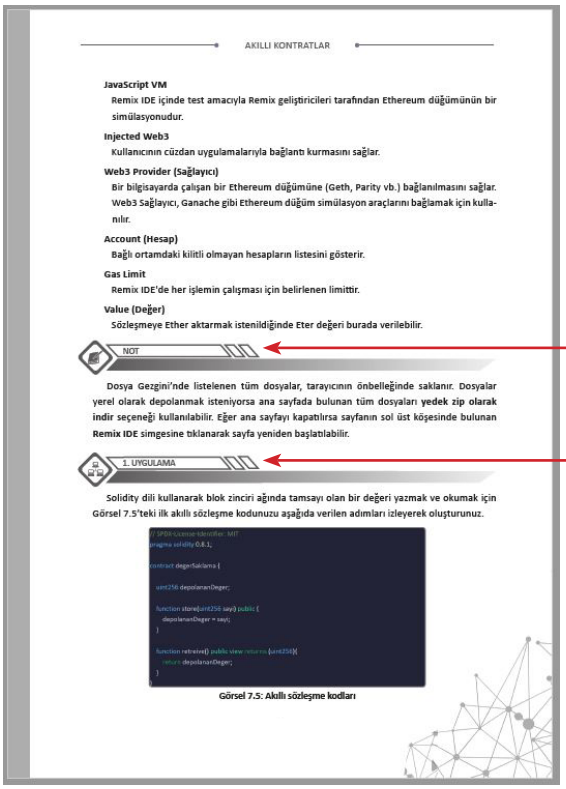
Öğrenme biriminin adını gösterir.

Karekod okuyucu ile taranarak resim, video, soru ve çözümleri gibi ilave kaynaklara ulaşılabilecek karekod bölümüdür.

Öğrenme biriminin numarasını gösterir.

Öğrenme birimindeki önemli kavramları gösterir.

Derse başlamadan yapılacak olan hazırlıkları gösterir.



AKILLI KONTRATLAR

JavaScript VM
Remix IDE içinde test amacıyla Remix geliştiricileri tarafından Ethereum düğümünün bir simülasyonudur.

Injected Web3
Kullanıcının cüzdan uygulamalarıyla bağlantı kurmasını sağlar.

Web3 Provider (Sağlayıcı)
Bir bilgisayarda çalışan bir Ethereum düğümüne (Geth, Parity vb.) bağlanılmasını sağlar. Web3 Sağlayıcı, Ganache gibi Ethereum düğüm simülasyon araçlarını bağlamak için kullanılır.

Account (Hesap)
Bağlı ortamdaki kilitli olmayan hesapların listesini gösterir.

Gas Limit
Remix IDE'de her işlemin çalışması için belirlenen limittir.

Value (Değer)
Sözleşmeye Ether aktarmak istenildiğinde Eter değeri burada verilebilir.

NOT

Dosya Gezgini'nde listelenen tüm dosyalar, tarayıcının önbelleğinde saklanır. Dosyalar yerel olarak depolanmak isteniyorsa ana sayfada bulunan tüm dosyaları **yedek zip olarak indir** seçeneği kullanılabilir. Eğer ana sayfayı kapatılırsa sayfanın sol üst köşesinde bulunan **Remix IDE** simgesine tıklanarak sayfa yeniden başlatılabilir.

1. UYGULAMA

Solidity dili kullanarak blok zinciri ağında tamsayı olan bir değeri yazmak ve okumak için Görsel 7.5'teki ilk akıllı sözleşme kodunuzu aşağıda verilen adımları izleyerek oluşturunuz.

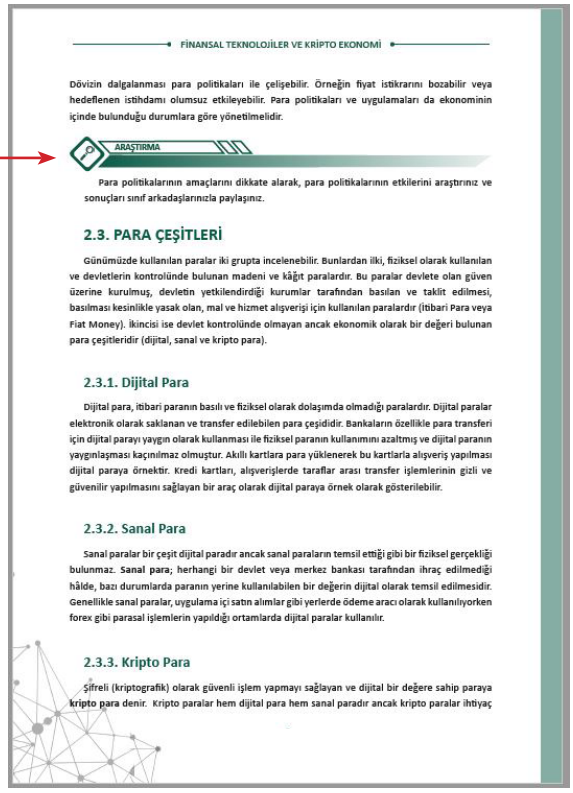
```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.1;

contract degerSaklama {

  uint256 depolananDeger;

  function store(uint256 say) public {
    depolananDeger = say;
  }

  function retrieve() public view returns (uint256){
    return depolananDeger;
  }
}
```

Görsel 7.5: Akıllı sözleşme kodları

Konuya ilişkin ek bilgi ve ipuçlarını gösterir.

Öğrencilerin edindiği bilgiyi kullanmasını sağlayacak çalışmaları gösterir.

Araştırılması gereken konuları gösterir.



FİNANSAL TEKNOLOJİLER VE KRİPTO EKONOMİ

Dövizin dalgalanması para politikaları ile çelişebilir. Örneğin fiyat istikrarını bozabilir veya hedeflenen istihdamı olumsuz etkileyebilir. Para politikaları ve uygulamaları da ekonominin içinde bulunduğu durumlara göre yönetilmelidir.

ARAŞTIRMA

Para politikalarının amaçlarını dikkate alarak, para politikalarının etkilerini araştırınız ve sonuçları sınıf arkadaşlarınızla paylaşınız.

2.3. PARA ÇEŞİTLERİ

Günümüzde kullanılan paralar iki grupta incelenebilir. Bunlardan ilki, fiziksel olarak kullanılan ve devletlerin kontrolünde bulunan madeni ve kâğıt paralardır. Bu paralar devlete olan güven üzerine kurulmuş, devletin yetkilendirdiği kurumlar tarafından basılan ve taklit edilmesi, basılması kesinlikle yasak olan, mal ve hizmet alışverişi için kullanılan paralardır (İtibari Para veya Fiat Money). İkincisi ise devlet kontrolünde olmayan ancak ekonomik olarak bir değeri bulunan para çeşitleridir (dijital, sanal ve kripto para).

2.3.1. Dijital Para

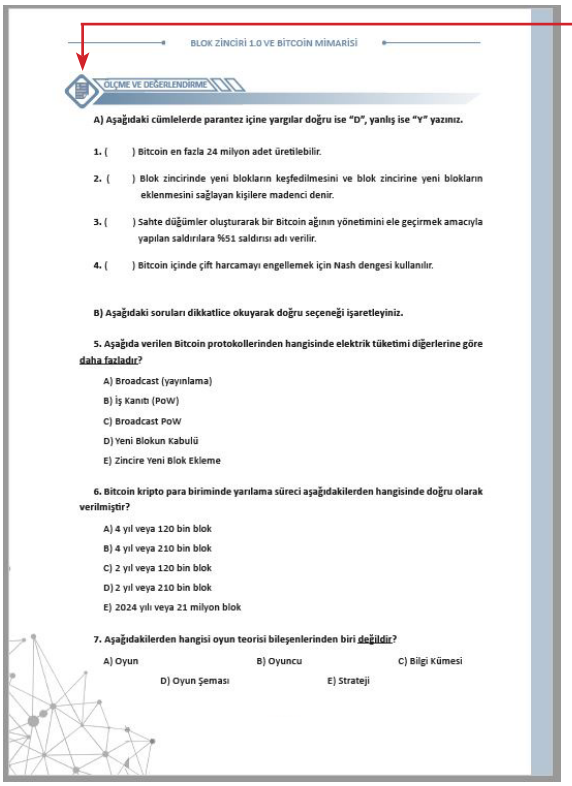
Dijital para, itibari paranın basılı ve fiziksel olarak dolaşımda olmadığı paralardır. Dijital paralar elektronik olarak saklanan ve transfer edilebilen para çeşididir. Bankaların özellikle para transferi için dijital parayı yaygın olarak kullanması ile fiziksel paranın kullanımını azaltmış ve dijital paranın yaygınlaşması kaçınılmaz olmuştur. Akıllı kartlara para yüklenerek bu kartlarla alışveriş yapılması dijital paraya örnektir. Kredi kartları, alışverişlerde taraflar arası transfer işlemlerinin gizli ve güvenilir yapılmasını sağlayan bir araç olarak dijital paraya örnek olarak gösterilebilir.

2.3.2. Sanal Para

Sanal paralar bir çeşit dijital paradır ancak sanal paraların temsil ettiği gibi bir fiziksel gerçekliği bulunmaz. Sanal para; herhangi bir devlet veya merkez bankası tarafından ihraç edilmediği hâlde, bazı durumlarda paranın yerine kullanılabilen bir değerin dijital olarak temsil edilmesidir. Genellikle sanal paralar, uygulama içi satın alımlar gibi yerlerde ödeme aracı olarak kullanılıyorken forex gibi parasal işlemlerin yapıldığı ortamlarda dijital paralar kullanılır.

2.3.3. Kripto Para

Şifreli (kriptografik) olarak güvenli işlem yapmayı sağlayan ve dijital bir değere sahip paraya **kripto para** denir. Kripto paralar hem dijital para hem sanal paradır ancak kripto paralar ihtiyaç



BLOK ZİNCİRİ 1.0 VE BİTCOİN MİMARİSİ

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerde parantez içine yargılar doğru ise "D", yanlış ise "Y" yazınız.

1. ( ) Bitcoin en fazla 24 milyon adet üretilebilir.
2. ( ) Blok zincirinde yeni blokların keşfedilmesini ve blok zincirine yeni blokların eklenmesini sağlayan kişilere madenci denir.
3. ( ) Sahte düğümler oluşturarak bir Bitcoin ağının yönetimini ele geçirmek amacıyla yapılan saldırılara %51 saldırısı adı verilir.
4. ( ) Bitcoin içinde çift harcamayı engellemek için Nash dengesi kullanılır.

B) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

5. Aşağıda verilen Bitcoin protokollerinden hangisinde elektrik tüketimi diğerlerine göre daha fazladır?

A) Broadcast (yayınlama)
B) İş Kanıtı (PoW)
C) Broadcast PoW
D) Yeni Blokun Kabulü
E) Zincire Yeni Blok Ekleme

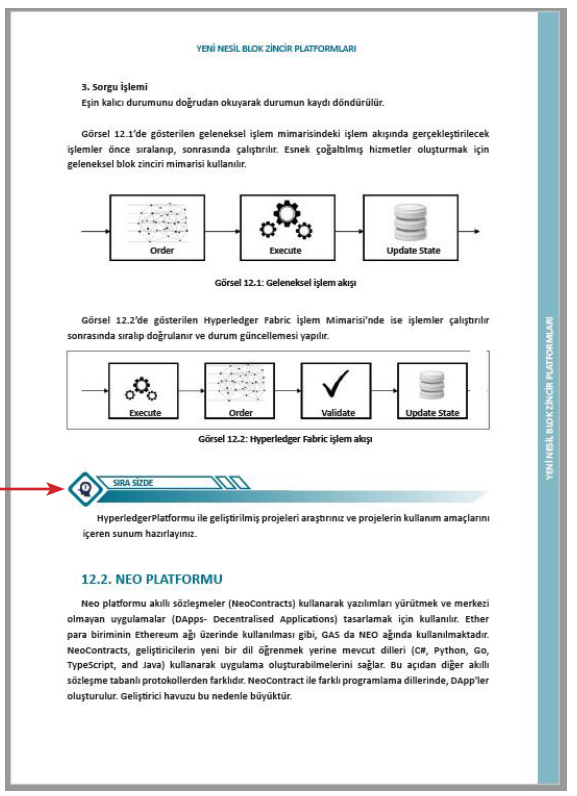
6. Bitcoin kripto para biriminde yarılama süreci aşağıdakilerden hangisinde doğru olarak verilmiştir?

A) 4 yıl veya 120 bin blok
B) 4 yıl veya 210 bin blok
C) 2 yıl veya 120 bin blok
D) 2 yıl veya 210 bin blok
E) 2024 yılı veya 21 milyon blok

7. Aşağıdakilerden hangisi oyun teorisi bileşenlerinden biri değildir?

A) Oyun B) Oyuncu C) Bilgi Kümesi
D) Oyun Şeması E) Strateji

Kazanılan bilgi ve becerilerin her öğrenme birimi sonunda ölçüldüğü çalışmaları gösterir.



YENİ NESİL BLOK ZİNCİR PLATFORMLARI

3. Sorgu İşlemi
Eşin kalıcı durumunu doğrudan okuyarak durumun kaydı döndürülür.

Görsel 12.1'de gösterilen geleneksel işlem mimarisindeki işlem akışında gerçekleştirilecek işlemler önce sıralanıp, sonrasında çalıştırılır. Esnek çoğaltılmış hizmetler oluşturmak için geleneksel blok zinciri mimarisi kullanılır.

Görsel 12.1: Geleneksel işlem akışı

Görsel 12.2'de gösterilen Hyperledger Fabric İşlem Mimarisi'nde ise işlemler çalıştırılır sonrasında sıralıp doğrulanır ve durum güncellemesi yapılır.

Görsel 12.2: Hyperledger Fabric işlem akışı

SIRA SİZDE

HyperledgerPlatformu ile geliştirilmiş projeleri araştırınız ve projelerin kullanım amaçlarını içeren sunum hazırlayınız.

12.2. NEO PLATFORMU

Neo platformu akıllı sözleşmeler (NeoContracts) kullanarak yazılımları yürütmek ve merkezi olmayan uygulamalar (DApps- Decentralised Applications) tasarlamak için kullanılır. Ether para biriminin Ethereum ağı üzerinde kullanılması gibi, GAS da NEO ağında kullanılmaktadır. NeoContracts, geliştiricilerin yeni bir dil öğrenmek yerine mevcut dilleri (C#, Python, Go, TypeScript, and Java) kullanarak uygulama oluşturabilmelerini sağlar. Bu açıdan diğer akıllı sözleşme tabanlı protokollerden farklıdır. NeoContract ile farklı programlama dillerinde, DApp'ler oluşturulur. Geliştirici havuzu bu nedenle büyüktür.

YENİ NESİL BLOK ZİNCİR PLATFORMLARI

Konuyu pekiştirmek için öğrencilerin yapması gereken etkinlikleri gösterir.

KONULAR

1.1. BLOK ZİNCİRİ TEKNOLOJİSİ KAVRAMLARI
1.2. BLOK ZİNCİRİ VE BLOK ZİNCİRİ KAVRAMLARI
1.3. BLOK ZİNCİRİ TEKNOLOJİSİNİN UYGULAMA ALANLARI VE KULLANIM SENARYOLARI
1.4. KRİPTOGRAFİ
1.5. HASH FONKSİYON ÖRNEKLERİ
1.6. DAĞITIK DEFTER TEKNOLOJİSİ KAVRAMI VE ÇEŞİTLERİ
1.7. MADENCİLİK KAVRAMI VE BLOK ZİNCİRİNDE KULLANIMI

NELER ÖĞRENECEKSİNİZ?

- Blok Zinciri Kavramları
- Blok Zinciri Teknolojisinin Uygulama Alanları
- Kriptografi Kavramı
- Hash Kavramı
- Dağıtık Defter Kavramı
- Madencilik Kavramı

ANAHTAR KELİMELER

Blok zinciri, dağıtık defter, hash, kriptografi, madencilik.

HAZIRLIK ÇALIŞMALARI

1. Günlük hayatta kullandığımız metal zincirlerin hazırlanma süreci ile ilgili düşüncelerinizi arkadaşlarınızla paylaşınız.

2. Gizli belgelerin şifrelenmesinin önemi neler olabilir? Arkadaşlarınızla değerlendiriniz.

BLOK ZİNCİRİ TEKNOLOJİSİ

1.
ÖĞRENME BİRİMİ

1.1. BLOK ZİNCİRİ SİSTEMİ

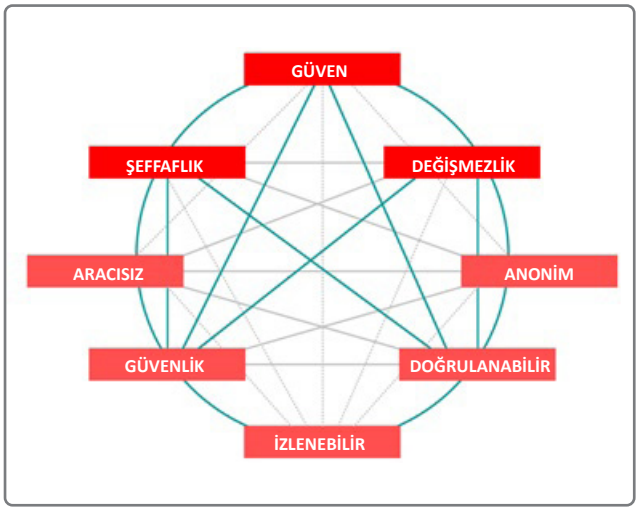
Görsel 1.1: Blok zinciri

Blok zinciri (blockchain), şifrelenmiş verilerin (kriptografi) birbirlerine zincir şeklinde bağlı, güvenli bloklardan oluşan ve sürekli büyüyen bir kayıt listesidir. Blok zinciri kavramında süreç boyunca yapılan tüm faaliyetlere ait veriler, şifreli bir şekilde kronolojik olarak kaydedilir. Blok zinciri, her bir blokun bir önceki blokun bilgilerini içerdiği, veri bloklarının art arda sıralanarak oluşturduğu zincir olarak da tanımlanabilir (Görsel 1.1). Blok zinciri aslında bir veri tabanıdır. Blok zinciri sisteminde veriler sıralı bir şekilde bloklara kaydedilir. Her bir kaydın bir zaman damgası bulunur. Bir blok dolunca bir sonraki blok oluşturulur ve bloklar birbirine zincir şeklinde bağlanır.

Ağustos 2014'te, ağ üzerinde gerçekleşen tüm işlemlerin kayıtlarını içeren kripto para blok zinciri dosya boyutu 20 GB'a ulaştı. Ocak 2016'dan Ocak 2017'ye kadar kripto para blok zinciri dosya boyutu 50 GB'tan 100 GB'a çıktı. 2021 yılı sonunda 380 GB'a ulaştı.

1.1.1. Blok Zinciri Teknolojisinin Özellikleri

Blok zinciri teknolojisindeki şeffaflık, güven ve değişmezliğin birleşimi blok zinciri teknolojisinin **özgünlüğüdür.** Blok zinciri teknolojisinin diğer önemli potansiyel özellikleri ise takma ad, doğrulanabilirlik, kontrol edilebilirlik, güvenlik ve aracısız bir yapıya sahip olmaktır. Potansiyel özellikler sadece blok zinciri teknolojisine özgü değildir fakat teknolojinin potansiyelini ve zorluklarını anlamak için önemlidir (Görsel 1.2).



Görsel 1.2: Blok zinciri teknolojisinin özellikleri (Lapointe ve Fishbane, 2019).

Şeffaflık (Transparency): Tüm işlem kaydının kopyaları her zaman katılımcılara açıktır. Dağıtık defter erişimi sayesinde herkes için işlemlerin şeffaflığı sağlanmaktadır.

Güven (Trust): Kriptografi ve işlemlerin değişmezliği ile güvenilir merkezî bir otorite olmadan da dağıtılmış bir ağ üzerindeki etkileşimde katılımcılara güçlü güvenlik sağlanır.

Değişmezlik (Immutability): Blok zinciri ağı üzerinde güncelleme işlemi yapılmak istenildiğinde ağa yeni bir işlem eklenir. Bu sayede defter üzerinde bulunan bir işlemin kaydı değiştirilemez çünkü defter üzerindeki işlemlerin silinmesinin herhangi bir yolu yoktur.

Aracısızlaştırma (Disintermediation): Doğrudan eşler arası iletişimi kullanan blok zinciri teknolojisi, aracıların yaptığı işlem adımlarına gereksinimi ortadan kaldırır. Bu durum, kişi ve kurumların işlem yoğunluğundan kaynaklanabilecek hatalarla karşılaşma riskini azaltır. Örneğin bu özellik sayesinde sigorta sistemindeki sigortalanan kişiler ile sigorta firmaları arasında broker, acente veya eksper gibi aracılara ihtiyaç duyulmadan birebir, kolay ve şeffaf bir iletişim sağlanır.

Güvenlik: Blok zinciri yapısında yapılan işlemlerin korunması ve doğru işlemlerin doğru kişilerce, doğru zamanda gerçekleştirilmesidir.

Anonim: Sistemin herkese açık olması, sistemden herkesin faydalanabilmesi ve onu ortaklaşa kullanabilmesidir.

Doğrulanabilir: Sistemdeki tüm işlemlerin herkes tarafından güvenli bir şekilde teyit edilmesidir.

İzlenebilir: Yapılan tüm işlemlerin herkes tarafından baştan sona kadar takip edebilmesidir.

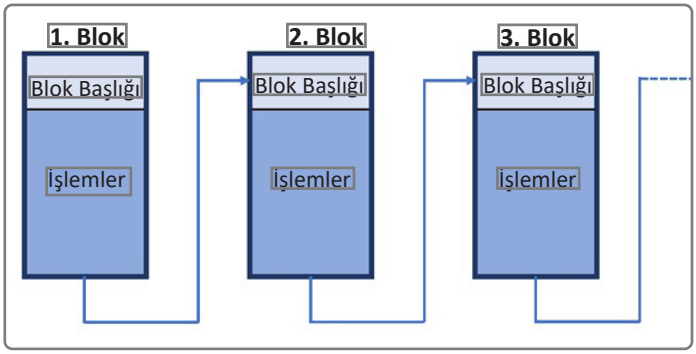
1.2. BLOK ZİNCİRİ KAVRAMLARI

Blok zinciri, bloklar ve bu blokları meydana getiren kayıtlardan oluşur.

Kayıtlar: İlgili blok zinciri yapısının üzerine oluşturulduğu her türlü içerik bilgisidir. Bu bilgiler blok zinciri tasarımına göre para transfer işlemi, demirbaş veya tedarik girdisi, müşteri kayıtları gibi değerler olabilir.Dijital para birimleri için bu kayıtlar, para transferi bilgileridir. Sistemde kayıtlı bir kullanıcının bir diğer kullanıcıya yaptığı para transferleri bu kayıtlarla tutulur. Yeni transfer istekleri de sıraya alınarak bir sonraki işlem ile kaydedilir ve zincirdeki yerini alır.

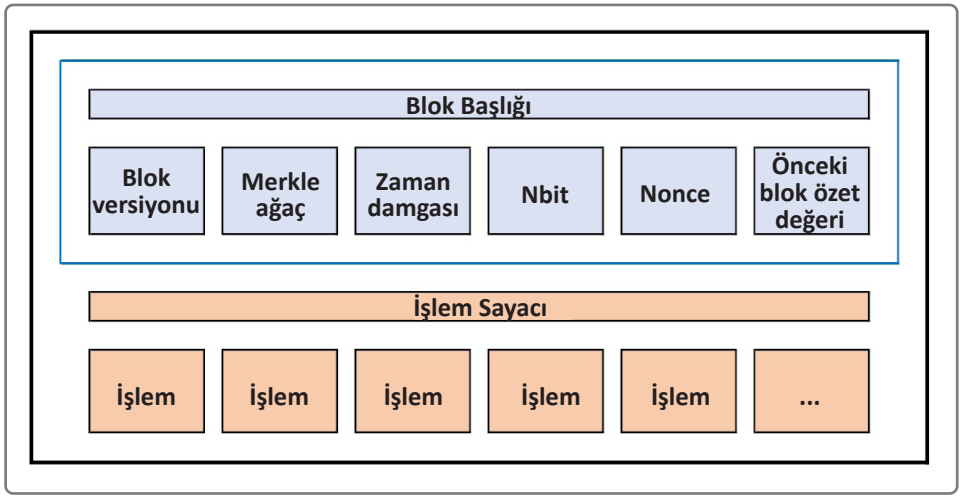
Bloklar: Kayıtlar belirli aralıklarla birleştirilip işlenerek blokların içine yazılır. Bloklar oluşturulurken içlerinde kaç adet kayıt bulunacağı, bu kayıtların hangi işlemlerden geçtikten sonra bir blok oluşturacağı gibi ölçütler belirlenir. Bu ölçütler blok zinciri tasarımına özgüdür. Bir blokun oluşturulması sırasında kriptografik özet algoritmaları ve dijital imza kullanılır.

Blok zinciri yapısında çok sayıda blok yer alır. Bu bloklar doğrusal ve kronolojik bir zincir oluşturacak şekilde birbirine bağlanır (Görsel 1.3), aynı zamanda kayıtların geçerliliğini doğrulayan dijital parmak izlerine sahiptir.



Görsel 1.3: Genel blok yapısı

Her bir blok içinde blok başlık özeti, Merkle ağaç kökü özeti, blokun oluştuğu zamana dair zaman damgası, Nbit, Nonce değeri ve işlemlere ait değerler bulunur (Görsel 1.4). Yapılan işlemlerin tamamı ağda bulunan tüm paydaşlar tarafından takip edilebilir.



Görsel 1.4: Blok başlığı yapısı

Blok başlığının yapısında bulunan bilgiler kısaca şu şekildedir:

- Blok versiyonu, blok doğrulama kurallarının hangisinin / hangilerinin uygulanacağını belirler.
- Merkle ağaç kökü özeti, bloktaki tüm işlemlerin kayıtlarının özet değerini tutar.
- Zaman damgası, 1 Ocak 1970 tarihinden beri evrensel zamanda saniye cinsinden geçerli zaman bilgisini içerir.
- Nbit, geçerli bir blok özet değeri için eşik değer bilgisi içerir.
- Nonce, genelde 0 değeriyle başlayan, her bir hesaplama için artan 4 byte boyutunda bir alandır.

- Önceki blok özet değeri, zincirde bir önceki bloka karşılık gelen 256 bit boyutunda tutulan özet değeridir.

1.3. BLOK ZİNCİRİ TEKNOLOJİSİNİN UYGULAMA ALANLARI VE KULLANIM SENARYOLARI

Blok zinciri teknolojisinin dağıtık ve şifrelenmiş yapısı, bloklardaki verilerin saldırıya uğramasını zorlaştırır. Ayrıca bilgilerin tüm katılımcılar tarafından görüntülenebilir ve değiştirilemez olması şeffaflığı sağlayarak sahtekârlığı azaltarak güven oluşturur. Blok zinciri teknolojisi dağıtık yapısıyla güveni sağlamak için aracı kişi veya kurum ihtiyacını ortadan kaldırarak maliyetleri azaltır ve işlemlerin daha hızlı gerçekleşmesine imkân sağlar. Sağladığı bu imkânlar nedeniyle birçok iş alanında kullanılabilir.

Blok zinciri teknolojisi başta finans sektörü olmak üzere birçok farklı alanda yeni ürün ve hizmetler oluşturmak için kullanılabilir. Özellikle tarafların birbirlerine karşı güven duymadığı durumlarda aracı kişi veya kurumlara ihtiyaç duyulmadan gerçekleştirilebilecek birçok farklı iş alanında tercih edilir.

Blok zincirinin uygulama alanları şunlardır:

- Para transferleri
- Takas işlemleri
- E-ticaret ve ödeme işlemleri
- Doğrulama ve yetkilendirme
- Değerli belgelerin yaratılması ve saklanması
- Borsalar ve hisse senetleri
- E-noter işlemleri
- Bağış ve mikro ödeme sistemleri
- Bulut bilişim ve güvenli bulut depolama
- Oylama sistemleri
- Tedarik zinciri işlemleri
- Dijital kimlik ve pasaport yönetimi
- Sosyal güvenlik sistemi
- Sigorta işlemleri

SIRA SİZDE

Blok zinciri ve kullanım senaryoları ile ilgili bir sunum hazırlayıp sınıfta arkadaşlarınızla paylaşınız.

1.4. KRİPTOGRAFİ

Blok zinciri yapısında tüm işlemler herkese açık gerçekleşir. Bu işlemlerdeki verilerin içinde özel ve kişisel bilgiler bulunduğundan bilgilerin şifrelenmesi yapılır. Yapılan bu işlemlerin "güven gerektirmeyen" bir yapıda fakat güvenli ve herkese açık olmasını sağlamak için **kriptografi** kullanılır. Böylece herhangi bir kişi hakkında hiçbir bilgi bulunmadan da onunla güvenli işlem yapabileceği anlamı ortaya çıkar.

Kriptografi, blok zinciri yapısındaki önemli unsurlardan biridir. **Kripto**, TDK'ye göre "saklı yazı" olarak tanımlanır yani bir yazı ya da verinin gizlenmesi amacıyla şifrelenmesidir. **Kriptografi**, bir verinin şifrelenmesi olarak kabul edilebilir. Gönderen, mesajı "şifreleyerek" üçüncü taraflardan gizler. Alıcı ise mesajın "şifresini çözerek" bunu tekrar okunur hâle getirir.

1.4.1. Kriptografinin Blok Zinciri Yapısındaki Kullanımı

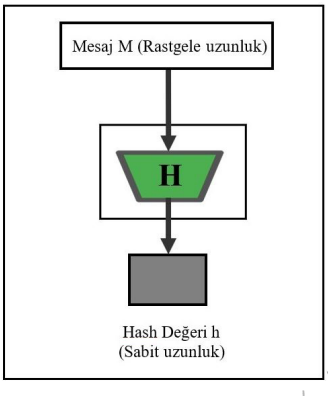
Kullanıcı kimliklerini ve yapılan işlemlerin bilgilerini korumak amacıyla blok zinciri teknolojisinde kriptografiden faydalanılır. Depolanan bilgilerin güvenliğinin yanı sıra yapılan işlemlerin de güvenli bir şekilde gerçekleşmesini sağlar. Blok zincirinde genellikle **açık anahtar şifrelemesi** kullanılır. Bu yöntemde bilgi, herkesle paylaşılabilecek bir açık anahtar aracılığıyla aktarılır. Şifreleme ve şifre çözme için tek bir anahtar yerine, **ortak anahtar** ve **gizli anahtar** olmak üzere iki anahtar kullanılır. Blok zinciri teknolojisinde yapılan işin ispat mekanizması ile sistemde bulunan kayıtlar tüm katılımcılara gönderilerek verinin bütünlüğü garanti edilir.

Açık anahtar şifreleme ile verilerin bütünlüğünü güvence altına alan bir elektronik imza üretilir. Elektronik imzanın üretimi, kullanıcıya ait özel anahtarın algoritma aracılığıyla imzalanmak istenilen verilerle birleştirilmesiyle elde edilir. Veriler böylece elektronik imzanın bir parçası olur. Veride oluşacak küçük bir değişiklik, elektronik imzayı da değiştireceği için blok zinciri ile verinin doğruluğu garanti edilir. Böylelikle verinin değişmezliği sağlanır.

Hash algoritmaları, blok zinciri işlem ayrıntılarını içeren blok verilerinin değiştirilmediğini ve verilerin bütünlüğünün korunduğunu garanti etmek için kullanılır. Blok zinciri tabanlı teknolojilerde özetleme algoritması olarak **SHA256** algoritması kullanılmaktadır.

1.5. HASH FONKSİYONU ÖRNEKLERİ

Hash fonksiyonu algoritmaları, girdi verinin boyutundan bağımsız sabit uzunlukta bir çıktı üretir. Hash, tek yönlü algoritmadır. Üretilen bir çıktıdan tekrar orijinal veriye dönülmez. Orijinal veri içerisinde yapılan küçük bir değişiklik bile hash algoritma çıktısını etkiler ve çıktının sonucu değişir (Görsel 1.5).



Görsel 1.5: Özet (hash) fonksiyonu

Hash, elektronik ortamda birçok doğrulama ve bütünlük kontrolü için kullanılan bir algoritmadır. Birçok hash algoritması geliştirilmiştir. Örneğin 128 bit MD5 algoritması, 160 bit SHA-1 algoritması, 256 bit SHA256 (SHA-2) algoritması bunlardan bazılarıdır. Bu algoritmalar, belirtilen bit büyüklüğünde bir çıktı üretir (Görsel 1.6).

Görsel 1.6: Hash fonksiyonu çıktısı

SIRA SİZDE

Hash fonksiyon örneklerini araştırarak bu fonksiyonların özelliklerini karşılaştıran bir sunum hazırlayınız.

Örneğin "Bilişim", "bilişim", "bilisim", "Bilişim Teknolojileri Alanı" kelimelerinin SHA-256 algoritmasıyla hash değerleri hesaplanırsa Tablo 1.1'deki sonuçlar elde edilir (https://andersbrownworth.com/blockchain/hash).

Tablo 1.1: Hash Değerleri

Veri	Hash Algoritması	Hash Değeri
Bilişim	**SHA-256**	595fa6e5eac144e5d02cbd06b21e50f774c647a4dac56c9fb60915b5cef6e36d
bilişim	**SHA-256**	24e1eecc339d53a8a48e401c47a4537b4ccf3e4a4ee6a6d950b2a450f62f29de
bilisim	**SHA-256**	deb718b22c5ed1369ba2657f67f73d9018a0feca1d085d1fed21cb14ff7ed98e
Bilişim Teknolojileri Alanı	**SHA-256**	5ab9c1905fd98f24e18f5676ddf1c1e99dc2cc153879f21ecc586d6367e4b66a

Hash algoritmalarında en önemli özelliklerden biri de çakışma olmamasıdır. İki farklı veri, hash algoritmasıyla işlem yapıldığında sonuç aynı çıkıyorsa çakışma oluşmuştur ve o algoritma artık işlevsizdir. SHA-1, MD5 gibi algoritmalarda çakışmalara rastlandığından günümüzde kullanımları tercih edilmez. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde ve Ethereum gibi kripto para sistemlerinde ise kullanılan hash algoritması **SHA-256**'dır.

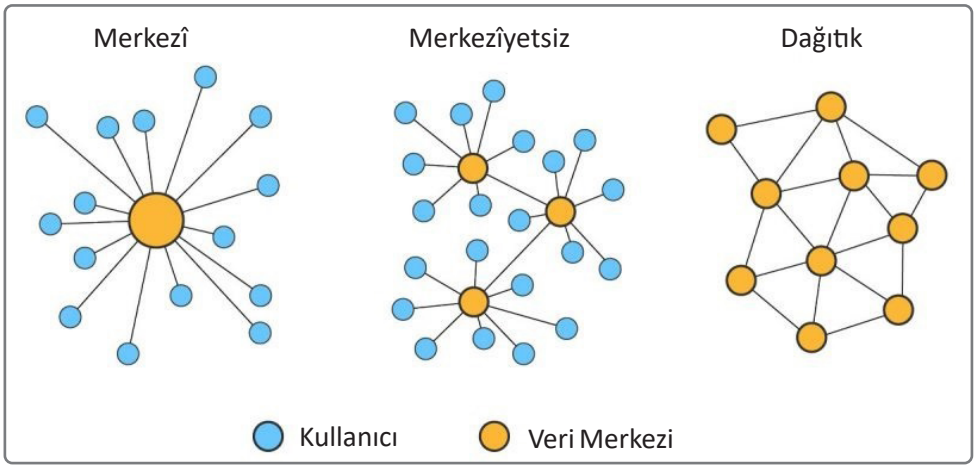
SIRA SİZDE

https://andersbrownworth.com/blockchain/hash sitesini kullanarak adınızın hash değerini bulunuz. Blok zinciri üzerinde, blok üzerine veri yazarak blok ve blok zinciri oluşturunuz.

1.6. DAĞITIK DEFTER TEKNOLOJİSİ KAVRAMI VE ÇEŞİTLERİ

Dağıtık defter teknolojisi, birçok düğüm ve katılımcı tarafından yönetilen merkezî olmayan veri tabanıdır. Bu teknolojide eşler arası ağ (P2P) olmalı, katılımcılar bulunmalı, sözleşmeler kullanılmalı ve verilerin değiştirilemez hâlde olması sağlanmalıdır. Eşler arası ağ ile ağda bulunan katılımcılar, herhangi bir merkezî otoriteye ihtiyaç duymadan aracısız işlem yapabilir. Böylece merkezî sunucuların maruz kaldığı çeşitli siber saldırıların önüne geçilmiş olur.

Verilerin saklanma biçimine göre üç çeşit ağ yapısı kullanılır (Görsel 1.7). **Merkezî** ağ yapısında her kullanıcı aynı bilgisayara bağlanır. **Merkezîyetsiz** ağ yapısında kullanıcılar, kendileri için en verimli sunucuya bağlanır. **Dağıtık** ağ yapısında ise kullanıcılar aynı zamanda veri sağlayıcısıdır.



Görsel 1.7: Ağ yapıları

Katılımcılar; bilgisayarlar aracılığıyla yapılan işlemlerin kaydedilmesi, paylaşılması, senkronizasyonu ve doğrulanmasını sağlarlar. Dağıtık defter teknolojisinde veri ekleme işlemleri için bir uzlaşma (konsensus) algoritması kullanılması gereklidir. Böylece ağa dâhil olmuş art niyetli kişilerin veya siber saldırıların ağı bozma çabaları engellenir. **Değiştirilemezlik**, yapılan işlem

sonrası dağıtık defter teknolojisine kaydedilen verilerin hiçbir katılımcı tarafından değiştirilememesidir. Kaydedilen dijital belgelere bir zaman damgası bilgisi girilir ve hash fonksiyonu ile dijital olarak imzalanır. Hash fonksiyonu, girilen veriden bağımsız, tek yönlü ve sabit uzunlukta bir şifreleme yapar. Kaydedilen veride yapılacak bir değişiklikte hash fonksiyon çıktısı farklı sonuç verdiği için sistemin güvenliği de sağlamış olur.

Dağıtık defter teknolojisinin genel yapısında **herkese açık** veya **özel izinli** olmak üzere iki yapı bulunur. **Herkese açık olan dağıtık defterler**, isteyen herhangi bir kullanıcının erişimine açık olarak tasarlanmıştır. Bu yapıda son kullanıcılar herhangi bir izin olmadan ağa katılabilir, doğrulayabilir veya istediği zaman ağdan ayrılabilir. **Özel izinli dağıtık defterler**, ağı oluştururken belirlenen şartları sağlayan belirli bir kullanıcı grubu için tasarlanır. Belirli bir katılımcı grubu, ağın yönetiminden sorumludur. Bu yapıda ağa, sadece izin verilen son kullanıcılar katılabilir, doğrulayabilir veya ağdan ayrılabilir.

2008 yılında **Satoshi Nakamoto** tarafından yayımlanan makalede, bir merkezî otoriteye ihtiyaç olmadan taraflar arasında düşük maliyetli ödeme yapılmasını sağlayan yapı açıklanmış ve buna Eşten-eşe Elektronik Nakit Ödeme Sistemi adı verilmiştir. Kullanıcılar tarafından bu finansal blok zinciri sisteminin ticari amaçla kullanmasıyla Eşten-eşe Elektronik Nakit Ödeme Sistemi, kripto para olarak kabul görmüştür. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde veriler, dağıtık defter teknolojisi kullanılarak bloklar hâlinde saklanıp katılımcılar tarafından doğrulanır. Ağa veri ekleme işlemi, **madenciler** adı verilen **doğrulayıcı düğümler** tarafından gerçekleştirilir. Madenciler, hash fonksiyonu bulma problemini bilgisayarları ile çözmeye çalışır. Problemi ilk çözen madenci tarafından blok ağa eklenir. Bu işlem, İş Kanıtı **(Proof of Work) uzlaşma mekanizması** olarak adlandırılır.

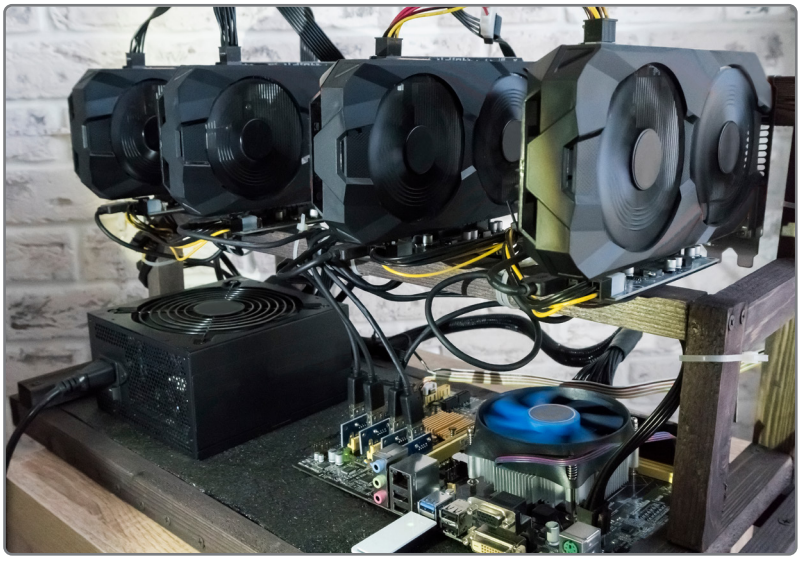
2013 yılında **Vitalik Buterin** tarafından yayımlanan makalede **Ethereum** blok zinciri altyapısı ve **ethereum kripto parası** sistemini tanımlanmıştır. Herhangi bir kullanıcı, Ethereum sayesinde merkezî olmayan ve blok zinciri teknolojisini kullanan uygulamalar oluşturabilir. **Ethereum**, herkesin kullanmasına izin veren açık kaynak blok zinciri platformudur. Ethereum blok zinciri altyapısı akıllı sözleşmeleri de destekler. **Akıllı sözleşmeler (smart contract)**, blok zinciri ağında çalışan farklı kuruluşlar arasında iş süreçlerinin yürütülmesine ve varlıkların tanımlamasına izin veren programlardır. Ethereum ile ortaya çıkan akıllı sözleşmeler, blok zinciri teknolojisinin kripto para dışında farklı alanlarda da kullanılabilmesine olanak sağlamıştır.

1.7. MADENCİLİK KAVRAMI VE BLOK ZİNCİRİNDE KULLANIMI

Madencilik, blok zincirine yeni bir blok ve bu blok içinde yeni işlem kaydının eklenmesidir. Madenciler, blok zincirine yeni bir blok eklemek için hash fonksiyonunu sonucunu belirlenen zorluk değerine göre bulan ilk kişi olmaya çalışır. Blok zincirin kripto para amacıyla kullanılmasına, yeni kripto paranın üretilmesine de **madencilik** denir. Madenciler sadece yeni blok üretmez aynı zamanda blok zincirine kaydedilecek işlemleri ve üretilen bloku doğrular, dağıtık defter yapısının sürekliliğini sağlarlar.

Madenciler tarafından üretilen son blok zincirine yeni eklenen blok, dağıtık ağda bulunan tüm düğümlere yani kullanıcılara iletilir. Böylece blok zinciri sisteminin kaydı ağda bulunan herkes tarafından tutulmuş olur. Her blok üretiminde yeni bloklar yayınlanır ve bu bloku ekleyene sistem tarafından belirlenmiş ödül verilir.

Madenciler, kendi bilgisayar kaynaklarını kullanarak işlemleri doğrular ve kaydeder. Madenciler, hash fonsiyonunu hesaplama gücü için CPU, GPU, FPGA ve ASIC gibi donanımları kullanabilir. Madencilik, bulut tabanlı hesaplama aracılığıyla da yapılabilir. Hesaplama gücü açısından CPU ve FPGA yavaş olduğundan yüksek hesaplama kapasitesine sahip, paralel işlem yapan GPU (Görsel 1.8) ya da hash fonksiyonu hesaplama için özel geliştirilmiş daha hızlı ASIC donanımları (Görsel 1.9) kullanılır.



Görsel 1.8: GPU donanımıyla madencilik ekipmanı

Görsel 1.9: ASIC sistemi ile madencilik ekipmanı

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerde parantez içine yargılar doğru ise "D", yanlış ise "Y" yazınız.

1. ( ) Veri blokları sıralanarak blok zinciri oluşturur.

2. ( ) Blok zincirinde kayıtlar rastgele sıralanır.

3. ( ) Blok zincirinin dağıtık yapısı, onun saldırılara açık olmasına neden olur.

4. ( ) Blok zincirinde işlemlerin uzun sürmesinin sebebi, dağıtık yapısıdır.

5. ( ) Tüm işlemler, blok zincirinde herkese açıktır.

6. ( ) Dağıtık ağ yapısı, kullanıcıları aynı zamanda veri sağlayıcı olduğu yapıdır.

B) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

7. I. Şifrelenmiş olması

II. Bloklardan oluşması

III. Hash

IV. Gizli kod kullanılması

V. Merkezî olması

Yukarıdakilerden hangileri blok zincirinin özelliklerinden <u>değildir</u>?

A) I-II

B) II-III-IV

C) I-V

D) I-IV-V

E) IV-V

8. Aşağıdakilerden hangisi hash fonksiyonu özelliklerden <u>değildir</u>?

A) Sabit uzunlukta çıktı vermesi

B) Tek yönlü algoritması olması

C) Çıktıdan girdiye ulaşılamaması

D) Girdi değişse bile çıktının değişmemesi

E) Her seferinde aynı sonucu vermesi

9. I. Hash fonksiyonu

II. Merkle ağaç kökü özeti

III. Nbit

IV. Sonraki blok özeti

V. Önceki blok özeti

VI. Zaman damgası

Yukarıdakilerden hangileri blok başlık yapısında bulunur?

A) I-II-III

B) II-III-IV-V

C) II-III-V-VI

D) I-IV-V-VI

E) I-II-IV-V

10. I. Veri ekleme

II. Veri silme

III. Doğrulama

IV. Hash hesaplama

Yukarıdakilerden hangileri madencilerin görevlerindendir?

A) I-III-IV

B) II-III-IV

C) I-II-III

D) I-II-IV

E) I-II-III-IV

11. Blok zincirinde ağlara eklenen verilerin doğrulanması aşağıdakilerden hangisi tarafından sağlanır?

A) Doğrulayıcı

B) Ekleyici

C) Kazıcı

D) Madenci

E) Uzlaştırıcı

12. I. CPU

II. GPU

III.EPROM

IV. ASIC

V. DVD

Yukarıdaki donanımlardan hangileri madencilik işlemleri için kullanılır?

A) I-II-III

B) I-II-IV

C) II-III-IV

D) I-II-V

E) I-III-IV

13. I. Şeffaflık

II. İzlenebilirlik

III. Değişmezlik

IV. Doğrulanabilirlik

V. Değişkenlik

Yukarıdakilerden hangileri blok zinciri teknolojisinin özellikleri arasında yer alır?

A) I-II

B) II-III-IV

C) I-III-V

D) I-II-III-IV

E) I-II-IV-V

KONULAR

2.1. PARANIN TARİHÇESİ
2.2. PARA POLİTİKASI
2.3. PARA ÇEŞİTLERİ
2.4. BLOK ZİNCİRİ 1.0
2.5. BLOK ZİNCİRİ 2.0
2.6. ETHEREUM

NELER ÖĞRENECEKSİNİZ?

- Paranın tarihçesi
- Para politikası ve amaçları
- Paranın özellikleri
- Paranın fonksiyonları
- Para çeşitleri
- Blok Zinciri 1.0 ve 2.0 teknolojisi
- Eşten-eşe Elektronik Nakit Ödeme Sistemi blok yapısı
- Soğuk ve sıcak cüzdan
- Etherum'un çalışma prensibi

ANAHTAR KELİMELER

Fiyat, kâğıt para, mal para, Okun Yasası, para, para politikası, takas.

HAZIRLIK ÇALIŞMALARI

1. Tasarruf yapma yöntemleri nelerdir? Siz hangisini tercih edersiniz? Düşüncelerinizi arkadaşlarınızla paylaşınız.

2. Sanal paraları satın aldıktan sonra nasıl saklayabilirsiniz? Arkadaşlarınızla değerlendiriniz.

FİNANSAL TEKNOLOJİLER VE KRİPTO EKONOMİ

2. ÖĞRENME BİRİMİ

2.1. PARANIN TARİHÇESİ

Güney Pasifik okyanusunda ada sakinleri değer değişim aracı olarak taşları kullanmıştır. Görsel 2.1'de Yap adasında değişim aracı olarak kullanılan taşlar görülmektedir.

Görsel 2.1: Yap adasındaki taş paralar

Tarih kitaplarına göre para ilk olarak Lidyalılar tarafından icat edilmiştir. Teknik olarak bu bilgi doğru olsa da pratik olarak paranın kökeni daha eskilere dayanır. Değişik coğrafyalardaki çeşitli tarihî kalıntılarda ticaret için faklı dönüşüm sistemlerine sahip, takas türünde para kullanımının olduğu ortaya çıkarılmıştır. İnsanlar, tarihin ilk döneminde ihtiyaç duyduğu mal ve hizmetleri kendi imkânları ile üretmekteydi ancak coğrafi nedenler, beceri, gerekli alet ve hayvanların eksikliği gibi nedenlerle üretemedikleri mal ve hizmetleri elde edebilmek için farklı yöntemler geliştirmiştir. Bu yöntemlerden biri de **takas** (Değişim veya trampa olarak da bilinir.) yöntemidir. İnsanlar, kendi üretemedikleri ve ihtiyaç duydukları mal ve hizmetleri de takas yöntemiyle başkalarından sağlamışlardır ancak birbirinin mal ve hizmetine ihtiyaç duyan kişi ya da grupların karşılaşması ve ortak bir değerde anlaşması oldukça zor bir durum olduğundan takas sistemi bazı sorunlar içermekteydi. Buna rağmen takas sistemi uzun yıllar boyunca kullanılmaya devam etmiştir. Örneğin buğday üreten kişi, kendi ihtiyacından fazlasını hayvancılık yapan kişiye vermiş ve karşılığında et almıştır.

İnsanlık tarihi ilerledikçe takas sistemindeki zorluklar iyice artmış, bu sistemin zayıflıkları daha fazla hissedilmeye başlanmıştır. Birinin ürettiği mal veya hizmete başkasının ihtiyaç duyması ve bu kişinin de karşı tarafın ürettiği mal ve hizmete ihtiyaç duyması takas işleminin işleyişini etkilemekteydi. Ayrıca takas yapacak kişilerin değişim oranı üzerinde anlaşması da gerekmekteydi. Bu sorunlar takas yapısının değişmesi veya yenilenmesini zorunlu kılmıştır. Günümüzde de kullanılan fayda, değer, üretim miktarı ve ihtiyaç sıklığı gibi ölçütler göz önüne alınarak belirli bir fiyata dayalı takas işleminin geliştirilmesi sağlanmıştır. Bu yeni takas sisteminde, ihtiyaçtan doğan mal veya hizmet ile karşılanan bu ihtiyacın sonunda elde edilen fayda, o mal veya hizmetin değerini belirlemiştir. Bu değere de **fiyat** denilmiş ve takas işlemleri bu fiyatlandırma üzerinden yapılmaya başlanmıştır. Her mal ve hizmete bir fiyat belirlenerek takas işlemlerine bir düzen

getirilmeye çalışılmıştır. Bu sistem takas işlemlerindeki zayıflıkları ortadan kaldırmaya yardımcı olmuştur.

Çeşitli dönemlerde bir malın başka bir mal ile takası şeklinde kullanılmasına para tarihinde **mal para** denilmektedir. Mal para, değerli madenlerin bulunması ve bu madenlerin değişim aracı olarak kullanılmaya başlanmasının ardından önemini yitirmiştir. Lidyalıların parayı değerli metal standardına dönüştürmesi takas yöntemini altın, gümüş ve bakır gibi değerli madeneler ile ödeme yapılmasına dönüşmüştür. Bu değerli madenlerden yapılan paraya **sikke** adı verilmiştir. Görsel 2.2'de eski döneme ait sikkelerden örnekler görülmektedir.

Görsel 2.2: Antik Yunan ve Roma sikkeleri

Kişisel ihtiyaçların karşılanması için kullanılan sikkeler, bir sorun teşkil etmezken büyük ticaretlerde fazla yer kaplayan ve ağır sikkelerin kullanımı oldukça zordu. Ticaret hacmi büyüdükçe sikkelerin kullanımı daha büyük sorun yaratmaya başladı. Ticaret ve ekonominin gelişmesi ile metal paranın taşınması ve güvenliğine dair sorunlar artmış, farklı çözüm arayışına gidilmiştir. Doğal madenlerden yapılmış sikkelerin hacim olarak yer kaplaması ve ağırlık anlamında problem yaratması neticesinde kâğıt para kullanılmaya başlanmıştır. Paranın kâğıt formunda basılması ve kullanılması bu soruna bir çözüm olarak geliştirilmiştir. Kâğıdın icadıyla birlikte insanlar, yüksek hacimli ticaretlerinde üzerinde yazılı değeri olan ve bugün de kullandığımız senede benzer kâğıt para kullanımına geçmiştir. Tarihte ilk kâğıt ve matbaa teknolojisini Çinliler icat etmiştir (https://www.tcmb.gov.tr) ancak 1200'lü yılların ortasından sonra bu icadı kullanarak ilk kez kâğıt para basan Moğol imparatorluğu olmuştur. Kâğıt paranın kullanımı ve yaygın şekilde kullanılması coğrafi olarak Çin'de başlamıştır.

Mal ve hizmet satın almada kullanılan kâğıt para sistemi günümüzde de kullanılan bir takas sistemidir. **Kâğıt para sistemi**; tedavüle çıkarılan ödeme araçlarının altın gibi değerli bir madene çevrilme zorunluluğunun bulunmadığı, para basma yetkisine sahip merkez bankaları tarafından çıkarılan ve altın gibi değerli madenlerden bağımsız, paranın miktarının tespit edildiği sistemdir. Bu sistemde ödeme aracı olarak kullanılan değerli madenler yerine yeni bir ödeme aracı kullanılmaktadır. Bu sistemde paranın karşılığı değerli bir maden olarak mevcut değildir.

Kâğıt para, kıymetini esas itibarıyla takasta kullanılması, sınırlı miktarda bulunması ve kendisine kanuni bir ödeme kabiliyeti tanınmasından alır. Paranın evrim süreci Görsel 2.3'te gösterilmiştir.

Görsel 2.3: Paranın evrimi

2.1.1. Paranın Özellikleri

Paranın başlıca özellikleri aşağıdaki başlıklardaki gibidir.

Taşınabilirlik: Paranın ağırlığı ve hacmi bakımından taşımaya uygun olmasıdır.

Dayanıklılık: Para, kullanım şartlarından doğan nem, ısı, ışık, aşınma, yıpranma gibi süreçlere dayanıklı malzemelerden üretilir.

Bölünebilirlik: Farklı miktarlardaki ödemelerin yapılabilmesi için paranın bölünebilir ve birbirine dönüşebilir olmasıdır (para üstü almak).

Genel Kabul Görme: Paranın halk ve diğer ülkeler tarafından tanınması ve kullanılmasıdır.

2.1.2. Paranın Fonksiyonları

Paranın başlıca fonksiyonları aşağıdaki başlıklardaki gibidir.

Değişim (Mübadele) Aracıdır: Para, takas sisteminin getirdiği güçlükleri ortadan kaldırmak için kullanılmaya başlanmıştır. Para ile birlikte mal ile malın değiştirilmesi yerine, mal ile paranın değiştirilmesi sağlanmıştır.

Ortak Değer Ölçüsüdür: Mal veya hizmetlerin alım satım değeri toplum tarafından kabul edilmelidir. Mal veya hizmetlerin değeri fiyatı, fiyat da parayı ifade eder.

Tasarruf ve Borçlanma Aracıdır: İhtiyaç fazlası paranın harcanmayarak elde tutulmasına **tasarruf** denilir. Benzer şekilde ihtiyaç duyulan bir mal veya hizmetin eldeki para ile alınamaması hâlinde ileriye yönelik ödeme işlemlerinin yapılmasında para borçlanma aracı olarak kullanılır.

Ekonomi Politikası Aracıdır: Ülkedeki para ile ilgili alınan karar ve yapılan uygulamalara **para politikası** denir. Ülkeler, mal ve hizmet alım gücü açısından para ile ilgili kararlar alabilir ve bu kararlarda ekonomi politikalarını uygular.

Mal ve hizmetlerin değişim aracı olarak kullanılan paranın tarihi, uygarlık tarihi kadar eskidir. Ülkelerin egemenlik ve özgürlüklerinin ifadesi anlamına gelen para, dilimize küçük parça anlamına gelen Farsça **pare** kelimesinden geçmiştir. **Lira** ise, Latince terazi anlamına gelen **libre** kelimesinden geçmiştir. Değerli madenlerin kullanımı ile başlayan para kullanımı daha sonra yerini kâğıt paraya bırakmıştır. Kâğıt para ile alınan hizmet ve mallara ait ödemeler kolaylaşmıştır ancak 21. yüzyıla gelindiğinde paranın yapısında önemli değişiklikler meydana gelmiştir. Bunların en önemlisi ise kripto paralardır. **Kripto paralar**, doğrudan bilişim teknolojisinin bir ürünü olarak ortaya çıkmış ve paranın yapısını değiştirebilecek bir etken olarak tanımlanmaya başlamıştır.

21. yüzyılda bilişim teknolojilerinde çok önemli yenilikler meydana gelmiş, finansal sistemler de bu yeniliklerden etkilenmiştir. Bu süreçte finansal piyasa çevrelerinde kullanılmak üzere farklı türde para birimleri ortaya çıkmıştır. Ortaya çıkartılması ve piyasada dolaşımı için herhangi bir devletin desteğine ve güvencesine ihtiyaç duymayan bu para birimleri, geleneksel para birimlerinin aksine sanal ortamda yönetilmektedir. Bu para birimlerine **kripto para** denir. Kripto paralar, taraflar arasında transfer edilebilecek olan değer veya para biriminin dijital gösterimine dayanır. Dijital olarak üretilen kripto paralar, depolanabilir ve işlem görebilir. Kripto paraların işlem görebilmesi için sanal bir cüzdandanın şifre ile alınması ve çıkartılması gerekmektedir. Görsel 2.4'te kripto para birimlerinin sembolik gösterimi bulunmaktadır.

Görsel 2.4: Kripto para birimlerinin sembol paraları

Kripto paralar, fiziki olarak kullanılmadığı için değerli madenler gibi gerçek bir değere sahip olmayan ancak talep, değer, güven ve kabul sebebiyle sanal olarak var olan dijital paralardır. Kripto paraların taşıdığı en büyük risk, henüz bilgisayar korsanlarına (hacker) karşı yeterince güvenilir olup olmadığının test edilememiş olmasıdır.

2.2. PARA POLİTİKASI

Para politikası, merkez bankalarının genel ekonomi politikası hedefleri doğrultusunda para arzı ve faiz oranı gibi değişkenleri yönlendirme çalışmalarına verilen bir isimdir. Para arzının artırılmasının ya da azaltılmasının, ekonominin büyüklüğü ve fiyat düzeyi üzerinde önemli etkileri olabilir. Ekonominin içinde bulunduğu durumlar, uygulanacak para politikasının niteliğini de belirlemektedir.

2.2.1. Para Politikasının Amaçları

Para politikası fiyat istikrarının sağlanması ve enflasyon gibi sorunların sebep olduğu durumların iyileştirilmesi amacıyla yapılır ancak genel itibari ile para politikasının amaçları aşağıdaki gibidir.

Fiyat İstikrarının Sağlanması: Para politikasının en temel amacıdır. Fiyat istikrarı, her türden iktisadi faaliyetleri yürütürken fiyatların değişimlerini ana değişken olarak almaz. Para politikası, fiyat seviyesinin artmasının ekonomiye olumsuz etkisini azaltmak ve gelecekle ilgili daha sağlıklı kararlar alınması için uygulanır.

İstihdam: Para politikasının bir amacı da istihdamdır. Gerçek işsizliğin önlenmesini ve ortadan kaldırılmasını sağlamak amacıyla para politikaları planlanır ve şekillendirilir. %3'lük işsizlik oranı normal olarak kabul edilirken gelişmekte olan ülkelerde bu oran %5 olarak kabul edilir. İstihdamın artması ekonominin büyümesiyle mümkün olacağı için işsizliğin 1 puan azalması ekonominin 2,2 puan artması ile sağlanabilir (**Okun Yasası** olarak bilinir.).

Ekonomik Büyüme: İstihdamın artması ekonomik büyümeyle gerçekleşebilir. Para politikası uygulamalarıyla ekonomik büyüme desteklenmelidir. Bu sayede genelde bir hasıla ve üretim artışı meydana gelir. Bu artışlar ekonomide yekûn bir büyümeyi getirir.

Faiz İstikrarı: Ekonomide istikrarın sağlanması için faiz oranlarında bir dengenin sağlanması gerekir. Faizdeki aşırı oynaklık, ekonomi üzerinde belirsizliklere neden olabilir. Bu sebeple para politikasının bir amacı da faizde belli bir istikrar düzeyinin korunması ve belirsizlik ortamına yol açılmamasıdır.

Piyasaların İstikrarı: Ekonomide en önemli görevlerin başında, finansal paniklerin ortaya çıkmasını engellemek veya buna sebep olabilecek sorunları ortadan kaldırmaya çalışmak gelir. Örneğin ekonomide nakit para taleplerinin artması bazı finansal krizleri tetikleyebilir veya krizi derinleştirebilir. Bu gibi zamanlarda, merkez bankalarının çözüm için devre girmeleri ve ellerindeki argümanları kullanmaları beklenir. Para politikası bu gibi durumları düzenleyen faaliyetleri kapsamalıdır.

Dış Ödemeler Dengesi: Para politikasının bir amacı da dış ödemeler dengesinin sağlanmasıdır. Dış ödemeler dengesinde ödemeler dövizle yapılır. Dövizin değeri sabit kur veya esnek kurla belirlenir. Esnek kurda yabancı paralar, piyasa üzerinden dalgalanmaya bırakılır.

Dövizin dalgalanması para politikaları ile çelişebilir. Örneğin fiyat istikrarını bozabilir veya hedeflenen istihdamı olumsuz etkileyebilir. Para politikaları ve uygulamaları da ekonominin içinde bulunduğu durumlara göre yönetilmelidir.

ARAŞTIRMA

Para politikalarının amaçlarını dikkate alarak, para politikalarının etkilerini araştırınız ve sonuçları sınıf arkadaşlarınızla paylaşınız.

2.3. PARA ÇEŞİTLERİ

Günümüzde kullanılan paralar iki grupta incelenebilir. Bunlardan ilki, fiziksel olarak kullanılan ve devletlerin kontrolünde bulunan madeni ve kâğıt paralardır. Bu paralar devlete olan güven üzerine kurulmuş, devletin yetkilendirdiği kurumlar tarafından basılan ve taklit edilmesi, basılması kesinlikle yasak olan, mal ve hizmet alışverişi için kullanılan paralardır (İtibari Para veya Fiat Money). İkincisi ise devlet kontrolünde olmayan ancak ekonomik olarak bir değeri bulunan para çeşitleridir (dijital, sanal ve kripto para).

2.3.1. Dijital Para

Dijital para, itibari paranın basılı ve fiziksel olarak dolaşımda olmadığı paralardır. Dijital paralar elektronik olarak saklanan ve transfer edilebilen para çeşididir. Bankaların özellikle para transferi için dijital parayı yaygın olarak kullanması ile fiziksel paranın kullanımını azaltmış ve dijital paranın yaygınlaşması kaçınılmaz olmuştur. Akıllı kartlara para yüklenerek bu kartlarla alışveriş yapılması dijital paraya örnektir. Kredi kartları, alışverişlerde taraflar arası transfer işlemlerinin gizli ve güvenilir yapılmasını sağlayan bir araç olarak dijital paraya örnek olarak gösterilebilir.

2.3.2. Sanal Para

Sanal paralar bir çeşit dijital paradır ancak sanal paraların temsil ettiği gibi bir fiziksel gerçekliği bulunmaz. **Sanal para**; herhangi bir devlet veya merkez bankası tarafından ihraç edilmediği hâlde, bazı durumlarda paranın yerine kullanılabilen bir değerin dijital olarak temsil edilmesidir. Genellikle sanal paralar, uygulama içi satın alımlar gibi yerlerde ödeme aracı olarak kullanılıyorken forex gibi parasal işlemlerin yapıldığı ortamlarda dijital paralar kullanılır.

2.3.3. Kripto Para

Şifreli (kriptografik) olarak güvenli işlem yapmayı sağlayan ve dijital bir değere sahip paraya **kripto para** denir. Kripto paralar hem dijital para hem sanal paradır ancak kripto paralar ihtiyaç

duyduğu güveni devlet veya merkez bankası gibi bir güven kaynağı yerine, kendi kriptolojik temelli algoritmalarındaki yapılardan sağlar. Kripto parayı sanal ve dijital paradan ayıran en temel özelliği de budur.

Kripto paralar önceden belirlenmiş kurallar çerçevesince üretilir. Kripto para, kendi içerinde belirlenmiş olan arz ve onay mekanizmaları aracılığıyla kullanıcılar tarafından kontrol edilir. İtibari para sisteminde olduğu gibi dışarıdan para girişi veya üretimine herhangi bir şekilde müdahale söz konusu değildir.

Kripto paraların geleneksel paralara göre bazı olumlu ve olumsuz özellikleri bulunmaktadır.

Olumlu Özellikleri

- Merkez bankalarından bağımsızdır, bu yüzden ülkelerin alacak oldukları parasal kararlardan ve enflasyonda etkilenmez.
- Kimlik doğrulama işlemleri yapılarak hem zamandan hem de paradan tasarruf sağlarlar.
- Para transferlerinde komisyon ücreti neredeyse yok denecek kadar azdır veya hiç yoktur.
- Para transferi istenilen zamanda yapılabilir. Zaman kısıtlaması yoktur.

Olumsuz Özellikleri

- Kripto paraları denetleyecek bir mekanizma henüz yoktur.
- Para arzı sınırlandırılmıştır (Ancak alt coinler bu sorunu aşmak için üretilmektedir.).
- Ani değer artışı veya azalışı olabilir. Bu durum yatırımcıyı etkiler.
- Kripto para transferi izlenemez. Bu nedenle vergi kaçakçılığı ihtimali bulunur.
- Verilerin sanal ortamda paylaşılması ve bu verilerin (şifrelenmiş olarak da olsa) her bir kripto para kullanıcısında bulunması risk oluşturur.
- Diğer para birimleri ile karşılaştırıldığında kripto paraların kullanım alanı daha sınırlıdır.
- Kripto paralar, bilgisayar veya akıllı telefonda hesap cüzdanlarında tutulur. Sanal ortamda bulunan bu cüzdanlardan kripto paranın çalınması veya kaybolması durumunda diğer para birimleri gibi izinin sürülme imkânı yoktur.

2.3.4. Dijital Para ve Kripto Para Arasındaki Farklar

- Dijital paralar bir otorite (devlet veya merkez bankası) tarafından çıkarılırken kripto paraların arkasında herhangi bir otorite bulunamaz.
- İşlemler, dijital paralarda gerçek kişilerce, kimlik bilgileri ile yapılırken (açık kimlik beyanı) kripto paralarda gizlidir.

- Dijital paralar için yasal düzenlemeler varken kripto paralar için yetersizdir.
- Dijital para işlemleri, kripto para ile yapılan işlemlere göre siber saldırılara karşı daha zayıftır. Blok zinciri tabanlı kripto para birimleri, siber saldırılara karşı çok daha güçlü ve güvenlidir.

SIRA SİZDE

Kripto para, dijital para ve sanal para konulu ve özellikle aralarındaki farkları anlatan sunum yapınız. Sunumunuzu infografik ve görsellerle destekleyiniz.

2.4. BLOK ZİNCİRİ 1.0

Dağıtılmış defter teknolojisi (DLT) uygulamasının finans işlemleri için kullanılmasıyla **Blok Zinciri 1.0** ortaya çıkmıştır. Blok zinciri 1.0 ile kripto para birimi üretiminin temelleri atılmıştır.

2.4.1. Finansal Blok zinciri

Kripto para birimlerinin geliştirilmesi için 2005 yılında Hal Finney tarafından bir konsept ortaya konmuştur. Bu yıllarda dünya ekonomilerinde bir mali kriz bulunmaktaydı. 2009 yılında mali krizin etkisinin zayıflamasının ardından Satoshi Nakamoto Eşten-eşe Elektronik Nakit Ödeme Sistemi hakkında bir teknik inceleme yazmış ve blok zinciri 1.0 çağının başlamasına sebep olmuştur.

Blok zinciri teknolojisinin ortaya çıkmasının en büyük nedeni, güvenilir bir kripto para birimini oluşturmadır. Devletlerden ve merkez bankalarından bağımsız, bir ağ tarafından yönetilen kripto para biriminin ortaya çıkartılması için gerekli uygulamayı içeren teknoloji Blok zinciri 1.0 olarak tanımlanır.

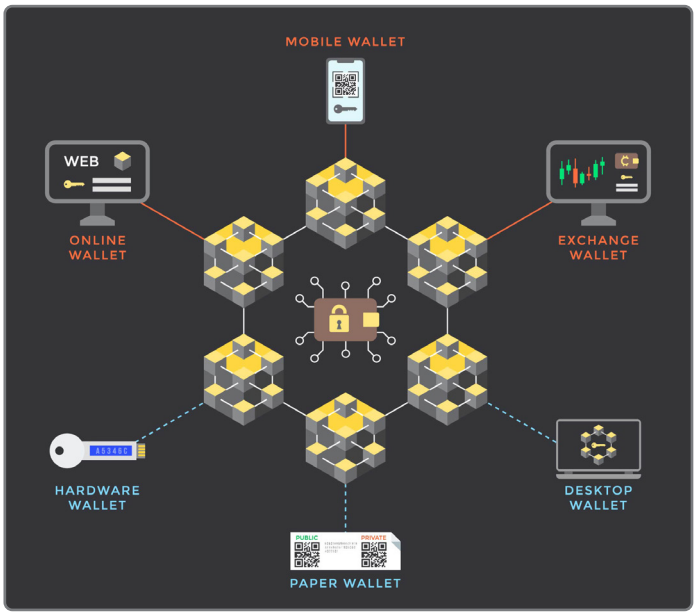
Blok Zinciri 1.0 teknolojisinin temel bileşenleri; kripto para birimi, bu parayı saklayacak cüzdan, kripto para madencilik teçhizatları ve kripto madencilik yazılımı için bir blok zinciri çekirdeğidir.

Kripto Para Birimi: Dijital veya sanal olarak var olan ve işlemleri güvence altına almak için kriptografi tekniğini kullanan para birimi biçimlerine verilen genel ifadedir. Kirpto para işlemlerinde yapılan işlemleri doğrulamak için bir bankaya gerek yoktur.

Kripto Cüzdan ve Yazılımı: Kripto paraları saklamak ve depolamak için kullanılan programa **kripto cüzdan** denilir. Kripto paralar bir tür dijital para oldukları için özel bir anahtardan oluşur. Bu özel anahtarı saklamak ve erişilebilir şekilde güvende tutmak için kripto cüzdan kullanılır. Ek olarak kripto paraları göndermek, almak içinde kullanılır. Kripto cüzdanlar ikiye ayrılır.

• Sıcak Cüzdan (Hot Wallet)

Kripto varlıkların özel anahtarlarını korumak için çevrimiçi yazılım kullanan kripto para depolayan cüzdandır. Görsel 2.5'te kritpo cüzdanların iki tipi de gösterilmiştir.



Görsel 2.5: Kripto sıcak ve soğuk cüzdanlar

• Soğuk Cüzdan (Cold Wallet)

Donanım cüzdanları olarak da bilinen soğuk cüzdanlar, özel anahtarları güvenli bir şekilde saklamak için çevrimdışı elektronik cihazları (flaş bellek gibi) kullanır. Görsel 2.6'da soğuk cüzdan olarak kullanılan Trezor ve Ledger gösterilmiştir.

Görsel 2.6: Trezor ve Ledger (kripto donanım cüzdanları)

Kripto Para Madencilik Teçhizatı: Kripto paralar madencilik adı verilen bir süreçle oluşturulur. Üretim aşamasında karmaşık matematiksel problemleri çözmek için bilgisayar gücünü kullanmayı içeren bu sürece madencilik adı verilir. Madencilik işlemleri için kullanılan ekipmanlara **Kripto Para Madencilik Teçhizatı** denilmektedir. Görsel 2.7'de kripto madencilikte kullanılan bir teçhizat görülmektedir.

Görsel 2.7: ASIC for kripto madencilik çiftliği

Kripto Madencilik Yazılımı: Coin üreten karmaşık matematiksel problemleri çözmek için geliştirilen yazılımlardır.

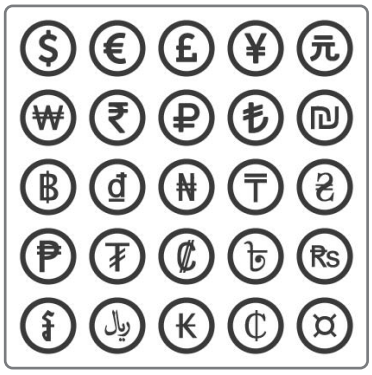
Blok Zinciri Çekirdeği: Blok zinciri ağına bağlanmak, bir düğüm çalıştırmak için kullanılan bir tür yazılımdır. Genellikle açık kaynaklı bir yazılımdır, yani herkes tarafından yükseltmeler görüntülenebilir ve yeni fikirler önerilebilir.

Kripto para birimleri, Blok Zinciri 1.0'ın ilk uygulamasıdır. Bu sebeple blok zinciri finansal işlemlere izin veren bir teknolojidir. Blok zinciri ve kripto para teknolojisinin amacı, her türlü döviz transferinde üçüncü taraf etkileşimi olan devlet otoritelerini ortadan kaldırmak olduğu için siyaseten bağımsız kalabilmektir. Kripto para, para oluşumunu kontrol etmek ve para transferinin doğrulanmasına izin vermek için şifreleme teknikleri kullanılarak blok zincirinde elektronik olarak oluşturulan ve saklanan bir değişim aracıdır.

2.4.2. Eşten-eşe Elektronik Nakit Ödeme Sistemi

Finansal blok zincirinin ilk ve en önemli uygulamalardan biri Eşten-eşe Elektronik Nakit Ödeme Sistemi'dir. Eşten-eşe Elektronik Nakit Ödeme Sistemi, açık zincir türünün temelini oluşturur. Eşten-eşe Elektronik Nakit Ödeme Sistemi esasen bir dijital veridir ve şifreleme sistemi sayesinde aracı bir kuruma ihtiyaç duymadan iki tarafın güvene dayalı bir ödeme sistemi oluşturmasını sağlar.

Görsel 2.8'de, kullanımda olan bazı para birimlerinin sembolleri bulunmaktadır.



Görsel 2.8: Para birimlerinin sembolleri

Kritpto para üretmeye madencilik adı verilir. Madencilik için özel donanıma sahip teçhizatlar gerekir. Bu teçhizata sahip bireyler veya işletmeler, verdikleri hizmet karşılığında sistem (network) tarafından coin denilen jetonlarla veya kripto paralarla ödeme alır. Merkezî olmayan bu sistemde ne kadar fazla madencilik yapılırsa kripto para elde etmek o kadar zorlaşır. Bunun sebebi, sisteme yeni eklenen blok arttıkça blok zincirinin blok uzunluğu da artacaktır (veri tabanında tutulan kayıtların artması gibi). Çıkartılabilecek maksimum kripto para sayısı belli olduğundan bu sayıya yaklaştıkça eklenen her bir blok başına verilen ödül kripto para da azalır. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin teknik tasarımından dolayı maksimum 21 milyon adet kripto para üretilebilir. Üretimi daha uzun bir süreye yaymak için üretim miktarı her dört yılda bir yarıya inmektedir.

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde veriler bir cüzdanda saklanır. Dijital ortamdaki bu cüzdanlar, genel (public) ve özel (private) anahtarlara sahiptir. Genel anahtar, blok zinciri ağı üzerinden cüzdanlar arası kripto para transferleri için kullanılırken özel anahtar cüzdana erişim sağlamak için kullanılır.

2.4.2.1. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin Temel Özellikleri

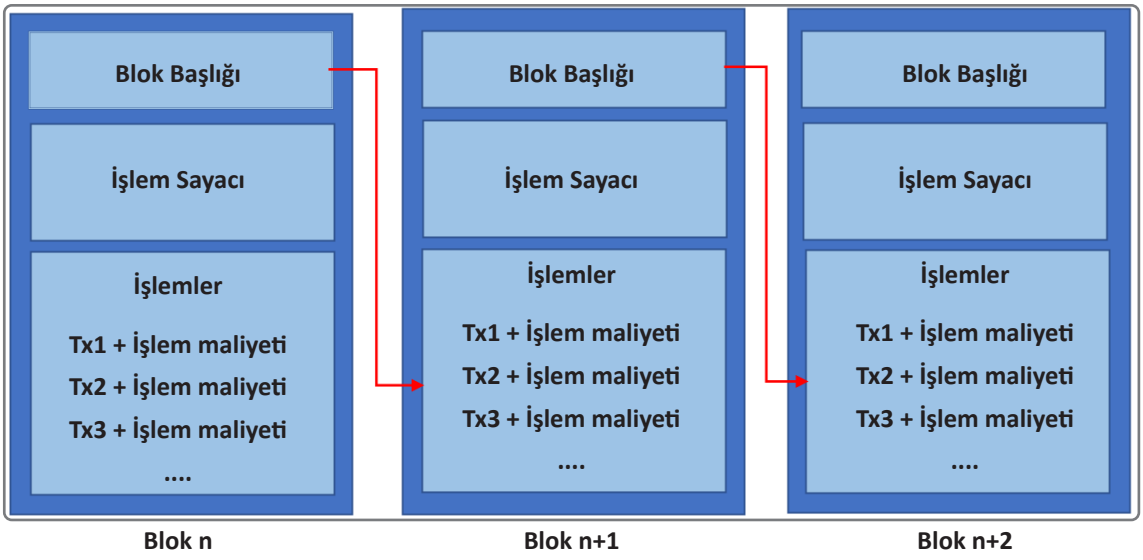
Diğer para birimleri ile karşılaştırıldığında Eşten-eşe Elektronik Nakit Ödeme Sistemi'ni diğer para birimlerinden ayıran özellikler şunlardır:

- Merkezî otoriteye bağlı olmaması
- Dijital bir teknoloji ile üretilmesi ve kullanılması
- Taklit edilememesi ve yeniden üretilememesi
- İşlem onaylama süresinin yaklaşık 10 dakika olması
- P2P [Peer to Peer (eşler arası)] teknolojisi ile işlemlerin yapılması

2.4.2.2. Eşten-eşe Elektronik Nakit Ödeme Sistemi Blok Yapısı

Eşten-eşe Elektronik Nakit Ödeme Sistemi bloklardan oluşur. Bloklar veri konteyneri olarak kabul edilir ve iletilen verilerin şifrelemesi için kriptografi teknolojisi kullanır. Bu yüzden Eşten-eşe Elektronik Nakit Ödeme Sistemi, kripto para birimi olarak adlandırılır. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde blok zincirinin bir bloku 500'e kadar işlem (Transactions) içerebilir. Bu durumda her bir blok boyutu 1 MB olur. Farkı durumlarda bir blokun boyutu 8 MB'a ulaşabilir ve bu durumda saniyede işlenen işlem sayısı artar.

Blok zinciri içindeki her blok, **SHA256 kriptografik hash** algoritması kullanılarak oluşturulan hash koduyla tanımlanan bir başlık bloku içerir. Bu başlık blokunun içinde kendisinden önceki blokun hash kodu bulunur. Her bloku ebeveynine bağlayan bu hash dizisi ilk bloktan itibaren son bloka kadar uzanan bir zincir oluşturur. Buna **Genesis Bloku** denir. Eşten-eşe Elektronik Nakit Ödeme Sistemi'ndeki blok yapısı Görsel 2.9'da, başlık blokunun yapısı ise Görsel 2.10'da gösterilmiştir.



Görsel 2.9: Eşten-eşe Elektronik Nakit Ödeme Sistemi blok yapısı

Görsel 2.10: Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde başlık bloku yapısı

2.4.2.3. Eşten-eşe Elektronik Nakit Ödeme Sistemi Çalışma Prensibi

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde her bir düğüm, açık bir kayıt defterine erişme hakkına sahiptir. Yapılacak tüm değişiklikler bu kayıt defterine kaydedilmek zorundadır (Bir veri tabanı olarak düşünülebilir.).Bu veriler, blok zincirin yapısından dolayı merkezî olmayan bir ağda -yani paylaşılmış bir veri tabanı mantığı ile- saklanır. Ağdaki her bir düğüm, yeni eklenen veri üzerinde onaylama sürecini gerçekleştirir. Hesaplamalar sonunda yeni eklenen veri, blok olarak onaylanmadan ve ağa katılmadan önce bellek havuzunda bekler. Bu süre daha önce de bahsedildiği gibi yaklaşık 10 dakikadır.

Bir cüzdandan, başka bir cüzdana kripto para göndermek için "XXX hesabına 2 kripto para gönderiyorum." şeklinde bir mesaj gönderici tarafından imzalanmalıdır. Bu mesaj bir veridir ve kayıt defterinde kaydedilir. Diğer bloklar bu işlemin doğruluğunu onaylar ve işlem bir blok olarak en sona eklenir. Bu işlemleri yapmak için bir kripto para cüzdanı gereklidir. Kripto para depolamak için bir cüzdan oluşturulduğunda bu cüzdana ait bir **genel (public)** bir de **özel (private)** anahtar tanımlanır. Genel anahtara kullanıcı adı ve şifresi de denilebilir ve harf-rakam kombinasyonundan oluşur.

Örnek olarak Ahmet, Berna'ya kripto para gönderecekse Ahmet, Berna'nın bu genel anahtarını ister. Genel anahtar, kişiye özgüdür ancak kişi hakkında herhangi bir bilgi içermediği için kişinin kimliği gizli kalmaya devam eder (E-mail adresi olsaydı kişinin adı ve soyadı gibi bilgilere hatta sorgulamalarla detaylı bilgilere ulaşılabilirdi.). Blok işleminin onaylanması sonrası kripto para, Berna'nın hesabına geçer. Berna, bu kripto parasına ulaşabilmek için özel anahtarına ihtiyaç duyar. Özel anahtar blok zinciri dünyasında kimlik bilgisidir. Kripto parasına ulaşabilmek, transfer etmek için özel anahtara ihtiyaç duyulmaktadır.

Blok zinciri içinde işlem yapılmakta olan kripto parayla yeniden işlem yapılmasını (çift işlem) önlemek için işlem verilerinin doğru ve gerçek zamanlı kaydının tutulması gerekir. Blok zinciri teknolojisinin yapısında bulunan merkezî olmayan yönetimi, bu sorunu **zaman damgası** ile çözmüştür. Blockcahin'de her bir blok, kronolojik olarak kayıt defterine kaydedilir ve aynı değerin tekrar kullanılmasını veya kopyalanmasını engellemek için her blokun kendi zaman damgası kullanılır.

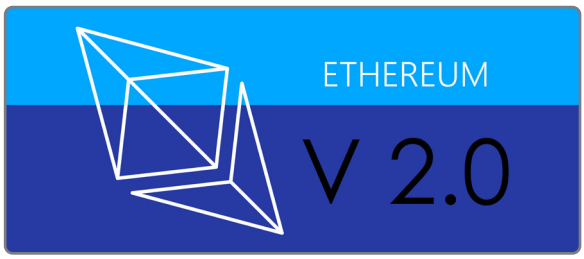
SIRA SİZDE

Finansal blok zincirinde kripto para üretiminin ne zaman dolacağını araştırarak sınıf arkadaşlarınızla paylaşınız.

2.5. BLOK ZİNCİRİ 2.0

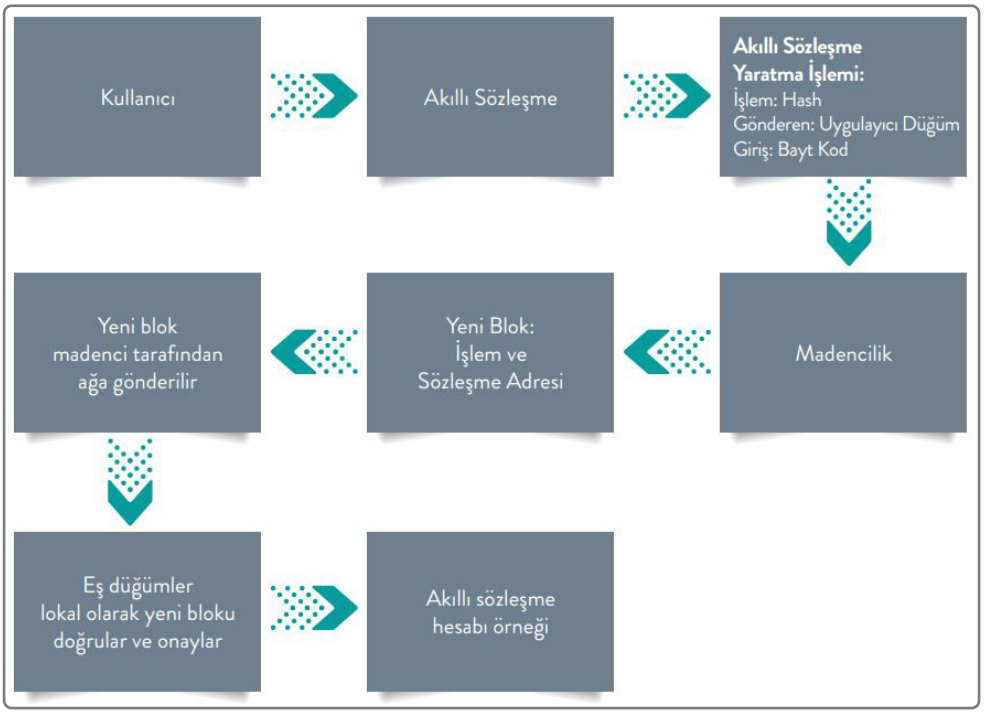
Blok Zinciri 1.0, sağlam bir zemine dayanır ancak düşük performans, buluta entegre etmede eksiklikler ve dağıtımların zorluğu gibi bazı sorunlar barındırır. Bu sorunlara çözüm

getirmek ve blok zinciri teknolojisini bir adım daha ileriye taşımak için merkezî olmayan ve eşler arası değer alışverişine dayanan bir sistem, **Blok Zinciri 2.0** olarak tanımlanmıştır. Aslında Blok Zinciri 1.0'dan Blok Zinciri 2.0'a geçişin sağlanması **Ethereum** ile gerçekleşmiştir. Görsel 2.11'de Blok Zinciri 2.0 Ethereum sembolü bulunmaktadır.



Görsel 2.11: Blok Zinciri 2.0 Ethereum

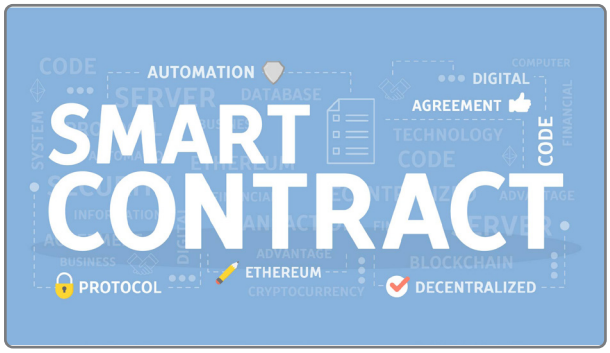
Blok Zinciri 2.0 teknolojisinin asıl amacı, kripto paraya odaklanmak yerine bu teknolojinin yeteneklerini piyasaların işini kolaylaştırmak için kullanmaktır. Buna, varlık değiş tokuşuna izin veren akıllı sertifikalar, akıllı sözleşmeler örnek olarak verilebilir. Görsel 2.12'de akıllı sözleşmelerin hazırlanması ve işlem akışı görülmektedir.



Görsel 2.12: Akıllı sözleşmenin hazırlanması ve işletilme akışı (Bankalararası Kart Merkezi, 2020)

Görsel 2.12'de görüldüğü üzere akıllı sözleşmeler, blok zinciri teknolojisi kullanılarak etkinleştirilen kod satır aralarına yerleştirilmiş anlaşmalardır. Bir sözleşmenin insan tarafından okunabilir terimleri, bir ağ üzerinde çalışabilen bilgisayar kodunda derlenir.

Bu yapı itibari ile akıllı sözleşmeler, kendi kendini yürüten bir bilgisayar programıdır. Bu bilgisayar programı, içerdiği anlaşma ile Ethereum gibi merkezî olmayan blok zinciri ağına dağıtılır. Belirli koşullar karşılandığında bir akıllı sözleşme devreye girer. Blok zincirine bağlanacak blokların doğrulama süreci ve buna bağlı olarak da sahip olunan güven nedeniyle sahte veya manipüle edilemeyen bir veri tabanı programı gibi çalışan sistem sayesinde daha önce hiç birbirini görmemiş ve tanışmamış kişiler arasında anlaşmalar ve sözleşmeler gerçekleştirilebilir. Görsel 2.13'te bir akıllı sözleşme konsepti gösterilmiştir.



Görsel 2.13: Akıllı sözleşme konsepti

Akıllı sözleşmeler herhangi bir adalet veya noterlik sistemi gibi üçüncü bir tarafa ihtiyaç duymadan iki veya daha fazla taraf arasında bir anlaşmanın yürütülmesini sağlar. Bu sözleşmelerle işletmeler, alacaklarını herhangi bir kayba karşı korurken müşteriler de kendilerini güvende hisseder. Bu yenilik Blok Zinciri 2.0 ile gelen en önemli özelliklerden biridir.

2.6. ETHEREUM

Ethereum, Eşten-eşe Elektronik Nakit Ödeme Sistemi üzerine inşa edilmiş ancak bazı büyük değişiklikleri içeren yeni ve farklı bir blok zinciri teknolojisidir. Her iki teknoloji de dijital paradır ve merkezî otoriteden bağımsızdır ancak Ethereum programlanabilirdir. Bu sebeple kripto para veya birçok farklı dijital varlık için kullanılabilir. Ethereum; finansal hizmetler, oyunlar, akıllı sözleşmeler ve uygulamalarda sıklıkla kullanılan bir kripto paradır. Ethereum, küçük bir ücret karşılığında kripto para göndermenize izin veren bir teknoloji olup aynı zamanda herkesin kullanabileceği ama kimsenin bireysel olarak kaldıramayacağı uygulamalara güç veren teknolojidir. Bu sebeple "Ethereum iki kısımda hizmet sağlayıcıdır." denilebilir.

1. Ethereum Ortamındaki Tüm Aktörler İçin Gerçek Bir Kaynak Sağlar

Ethereum blok zinciri, halka açık bir işlem veri tabanını koruyarak ekosistem için tek bir gerçek kaynağı oluşturur. Bu halka açık ve paylaşım hâlindeki veri tabanı, kullanıcılar ve uygulamalar arasında meydana gelen tüm işlemleri tutar. Benzersiz sanal adresler aktörleri tanımlar ve her işlem, katılan adresleri kaydeder. Bu adresler herhangi bir kişisel bilgiyi ifşa etmez ve kullanıcıların

kişisel bilgilerinin gizli kalmasını sağlar. İşlemler, halka açık olarak tutulan kayıt defterine kaydedilmeden önce, bloklar hâlinde gruplandırılır ve binlerce bilgisayar tarafından doğrulanır. Gönderilen işlemler, daha sonra hiç kimse tarafından kaldıramaz veya değiştiremez. Kayıtlar mühürlendiğinden kötü niyetli kişiler, bir işlemi geri alamaz veya ilgili kayıtları kurcalayamaz. Tüm işlemler gerçek zamanlı olarak kapatılır. Bu durum, blok zincirinin mevcut durumunun güvenilir olduğunu garanti eder.

2. Akıllı Sözleşmeler Olarak da Bilinen Uygulamalar İçin Bir Platform Sağlar

Her uygulama, bir sistem üzerinde çalışıyorsa bazı kaynaklara ihtiyaç duyar. Örneğin bir video gösterici uygulama, bilgisayarın işlemci, RAM, grafik belleği gibi kaynaklarına ihtiyaç duyar. Bilgisayar veya cihaz bu kaynakları sağlar. Tıpkı fiziksel bilgisayarlar gibi web hizmetleri de uygulamalar için altyapı sağlar. Bu web hizmetleri, bulut bilişim veya sanal makineler olarak da bilinir. Ethereum blok zinciri, web hizmetlerine benzer kritik uygulama altyapısı sağlar. Gerçek kaynağını koruyan binlerce düğüm veya bilgisayar sistemi, aynı zamanda; depolama, işlem gücü ve bant genişliği gibi kaynakları da sağlar. İnsanlar bu düğümleri veya bilgisayar sistemlerini, Dünya'nın herhangi bir yerinde çalıştırdığından bu birleşmiş toplu kaynaklar, tek bir bilgisayar sistemi gibi hizmet verir. Bu sebeple Ethereum **Dünya Bilgisayarı** olarak anılır. Ethereum, uygulamaların ve işlem verilerinin bir işletme tarafından kontrol edilen birkaç veri merkezi yerine, binlerce düğüme dağıtılmasıyla merkezî web hizmetlerinden farklı bir yapıdadır. Uygulamaları depolama ve güçlendirme özelliği, Ethereum blok zincirini Eşten-eşe Elektronik Nakit Ödeme Sistemi'nden ayıran temel özelliktir.

Ethereum'un sahip olduğu bazı özellikler şunlardır:

- **Herkes İçin Bankacılık**

Genel olarak sunulan finansal hizmetlere, herkesin erişimi yoktur. İnternet erişimi olan herkes, Ethereum ile finansal hizmetlere ulaşarak işlem yapabilir.

- **Daha Özel Bir İnternet**

Bir Ethereum uygulamasını kullanmak için tüm kişisel bilgilerin verilmesine gerek yoktur.

- **Eşler Arası Ağ**

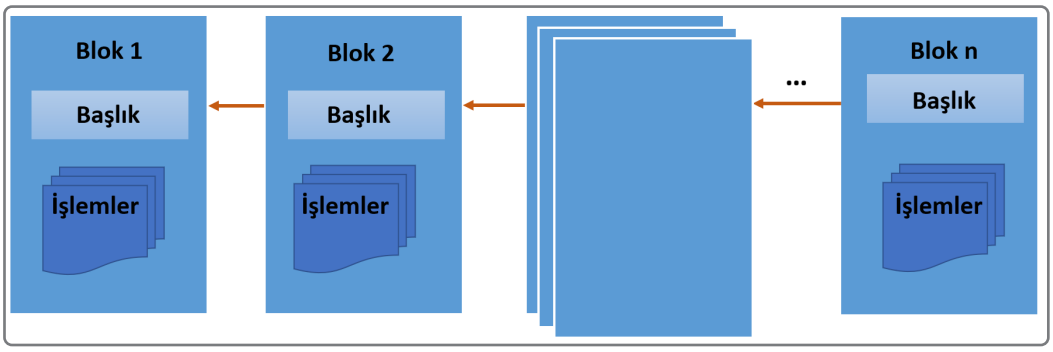
Ethereum, bir başka kişiye para göndermeye veya kişiler arası sözleşme yapmaya olanak verir. Bunlar için aracı kurumlara ihtiyaç duymaz.

- **Sansüre Karşı Dayanıklılık**

Hiçbir devlet veya şirketin Ethereum üzerinde kontrolü yoktur. Bu nedenle herhangi bir devlet, şirket ya da kişi, Ethereum'da para transferi veya diğer hizmetlerin kullanımını engelleyemez.

2.6.1. Ethereum'un Çalışma Prensibi

Ethereum'da aynı anda milyonlarca işlem yürütülür. İşlemler, blok olarak adlandırılan bir yapıda gruplandırılmıştır. Her blok, bir dizi işlem yürütür. Bloklar kendinden önceki bloklara zincirlenir. Görsel 2.14'te Ethereum blok zinciri yapısı görülmektedir.



Görsel 2.14: Ethereum blok zinciri yapısı

Ethereum blok zinciri, işleme dayalı bir **durum makinesini** ifade eder. Bu durum makinesi, girdiye bağlı bir dizi işlem yürütüp bu işlemlere göre yeni bir duruma geçer. Bir durumdan başka bir duruma geçmek için işlemlerin **geçerli** olması gerekir. Bir işlemin geçerli olabilmesi için ise **madencilik** adı verilen **doğrulama** sürecinin tamamlanması gerekir. **Ethereum madenciliği**, bir düğümün veya bilgisayar sisteminin geçerli bir işlem bloku oluşturmak için işlem kaynaklarını harcama sürecidir.

Ethereum blok zincirinde kendini madenci olarak tanıtan bir bilgisayar sistemi (düğümün), blok oluşturmaya veya doğrulama işlemi yapmaya çalışabilir. Madenciler, blok zincirine yeni bir blok gönderdiklerinde bir **kanıt** sunmalıdır. Matematiksel bir ifade şeklinde sunulan bu kanıt ile blok geçerli sayılır. Kısaca "Kanıt varsa blok geçerlidir." denir. Yeni bloku onaylayan ve geçerli sayan bir madenci, bu iş için ağ tarafından belirli bir değerle ödüllendirilir. Madenciler kanıtladıkları her blok için **Ether** adı verilen bu coinle ödüllendirilir.

2.6.2. Finansal Teknolojilerde Kullanılan Blok Zinciri Uygulamaları

Blok zinciri teknolojisi birçok alanda olduğu gibi finans alanında da büyük bir etkiye sahiptir. Görsel 2.15'te görüldüğü üzere ödeme sistemleri, finansal araştırmalar, sigorta, kredi ve borç işlemleri, yatırım yönetimi ve kitle fonlaması alanlarında kullanılmaktadır.

Görsel 2.15: Finansal Teknoloji (Fintech) örnek kullanım alanları

Finansal alanda kullanılan blok zinciri uygulamaları aşağıdaki gibi beş grupta ele alınabilir:

1. Ticari Finans Platformları

Ticari müşteriler arasında akıllı sözleşmeler yaparak verimliliği, güvenirliği ve şeffaflığı artırmak amacıyla finans platformlarda blok zinciri teknolojisi kullanılır.

2. Takas İşlemleri

Blok zinciri teknolojisinin ardında yatan doğrulama süreci ve bu sürecin verdiği güven nedeniyle takas işlemlerinde blok zinciri teknolojisinin kullanımını oldukça popüler hâle getirir. Takas işlemlerinde blok zinciri teknolojisinin kullanımının yaygınlaşmasıyla finansal maliyetlerin yakın zamanda daha da düşeceği öngörülmektedir.

3. Sınır Ötesi İşlemler

Günümüzde sınır ötesi para transferlerinde, paranın aracılar üzerinden aktarılması transfer gecikmesi ve maliyet artışı gibi olumsuzlukları beraberinde getirir. Blok zinciri, eşler arasında para transferini; aracısız, güvenli, hızlı ve çok düşük maliyetle yerine getirebilen bir teknolojidir.

4. Kredi Raporlama İşlemleri

Kişilerin veya işletmelerin finansal kredi raporları, veri ihlalleri veya manipülasyonlara açık olması nedeniyle blok zinciri teknolojisinin sunduğu imkânı daha cazip hâle getirir. Kaldı ki blok zinciri teknolojisi tabanlı kredi raporları, standart yöntemlerle elde edilen raporlara göre daha fazla güven sunar. Aynı zamanda blok zinciri teknolojisi, kredi raporlarını oluştururken daha fazla faktörü dikkate alabileceği için finansal alanda bu teknolojinin kullanımının artacağı ön görülmektedir.

5. Dijital Kimlik Doğrulama

Blok zinciri teknolojisi, bireylerin kimlik bilgilerini açık bir şekilde saklamadığından kimlik bilgileri güvenli bir şekilde tutulur. Bireylere ait tanımlayıcı bilgiler, blok zinciri teknoloji sayesinde güvence altına alındığından kişiler, kendilerini güvende hisseder ve finansal kurumlar ise kayıplara karşı korunur. Aynı zamanda kimlik doğrulama işlemi geleneksel yöntemlere göre oldukça hızlı olduğundan finansal işlemlerde tercih sebebidir.

SIRA SİZDE

Ethereum ortamını kullanınız ve edindiğiniz bilgileri sınıfta arkadaşlarınızla paylaşınız.

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerde parantez içine yargılar doğru ise "D", yanlış ise "Y" yazınız.

1. ( ) Sikke; altın, gümüş ve bakır gibi değerli madenlerden yapılmıştır.

2. ( ) Bir malın başka bir mal ile değiştirilmesi işlemine mal para denilir.

3. ( ) Kripto paralar merkez bankalarınca denetlenebilen para çeşididir.

4. ( ) Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde madencilik yıllar içinde azalsa da kripto para üretimi sınırsızdır.

B) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

5. Aşağıdakilerden hangisi paranın özelliklerinden biri değildir?

A) Bozdurulabilirlik
B) Bölünebilirlik
C) Dayanıklılık
D) Kabul görme
E) Taşınabilirlik

6. Aşağıdakilerden hangisi Ethereum için söylenemez?

A) Eşten-eşe Elektronik Nakit Ödeme Sistemi üzerine kurulmuştur.
B) Dijital para çeşididir.
C) "Dünya Bilgisayarı" olarak anılır.
D) Blok Zinciri 1.0 teknolojisi olarak bilinir.
E) Akıllı sözleşmelere olanak sağlar.

7. Aşağıdakilerden hangisi Ethereum'um özelliklerinden biri değildir?

A) İnternete sahip herkesin finansal işlemlere erişebilmesi
B) Merkezî bir yönetime sahip olması
C) Finansal işlemler için daha özel bir internet ortamı sağlaması
D) Kişiler arası sözleşme yapmaya imkân vermesi
E) Sağlanan hizmetlerin başkalarınca engellenemez olması

8. Genesis bloku kavramının açıklaması aşağıdaki seçeneklerin hangisinde doğru olarak verilmiştir?

A) Blok zincirindeki hatalı işlem blokudur.

B) Blok zincirindeki en son yapılan işleme ait bloktur.

C) Blok zinciri ağına kaydedilen ilk bloktur.

D) Arka arkaya gelen bloklar topluğudur.

E) Onaylanmamış işlem blokudur.

9. Çıkartılması ve dolaşımı için herhangi bir devletin desteğine ve güvencesine ihtiyaç duymayan para birimine ne ad verilir?

A) Kağıt para

B) Dijital para

C) Sikke

D) Mal para

E) Kripto para

10. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde her bir bloku kaç işlem içerebilir?

A) 100

B) 256

C) 500

D) 1 Milyon

E) 21 Milyon

KONULAR

3.1. BİZANS HATA TOLERANSI (BFT)

3.2. İŞ KANITI (PROOF OF WORK) MUTABAKAT MEKANİZMASI

3.3. HİSSE KANITI (PROOF OF STAKE) MUTABAKAT MEKANİZMASI

3.4. OTORİTE KANITI (PROOF OF AUTHORİTY) MUTABAKAT MEKANİZMASI

3.5. GECİKTİRİLMİŞ İSPATI (DELAYED PROOF OF WORK) MUTABAKAT MEKANİZMASI

3.6. HİBRİT MUTABAKAT ALGORİTMALARI

3.7. YENİ NESİL MUTABAKAT MEKANİZMALARI

NELER ÖĞRENECEKSİNİZ?

- Bizans hata toleransı
- İş kanıtı (Proof of Work) mutabakat mekanizması
- Hisse kanıtı (Proof of Stake) mutabakat mekanizması
- Otorite kanıtı (Proof of Authority) mutabakat mekanizması
- Geciktirilmiş ispat (Delayed Proof of Work) mutabakat mekanizması
- Hibrid mutabakat algoritmaları
- Yeni nesil mutabakat mekanizmaları

ANAHTAR KELİMELER

Eşten-eşe Elektronik Nakit Ödeme Sistemi, blok zinciri, DPoW, PBFT, PoA, PoS, PoW

HAZIRLIK ÇALIŞMALARI

1. Gündelik yaşamınızda size ulaşan bir mesaj ya da haberin doğruluğunu anlamak için hangi yöntemleri kullanırsınız? Sınıfta arkadaşlarınızla tartışınız.

2. Bizans generalleri problemini araştırarak araştırma sonuçlarınızı arkadaşlarınızla değerlendiriniz.

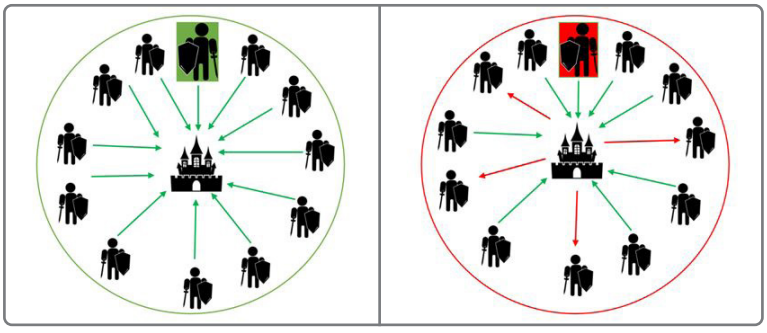
3. Fikir ayrılığına düştüğünüz bir arkadaşınızla nasıl anlaşırsınız? Düşüncelerinizi arkadaşlarınızla paylaşınız.

MUTABAKAT PROTOKOLLERİ

3.
ÖĞRENME BİRİMİ

3.1. BİZANS HATA TOLERANSI (BFT)

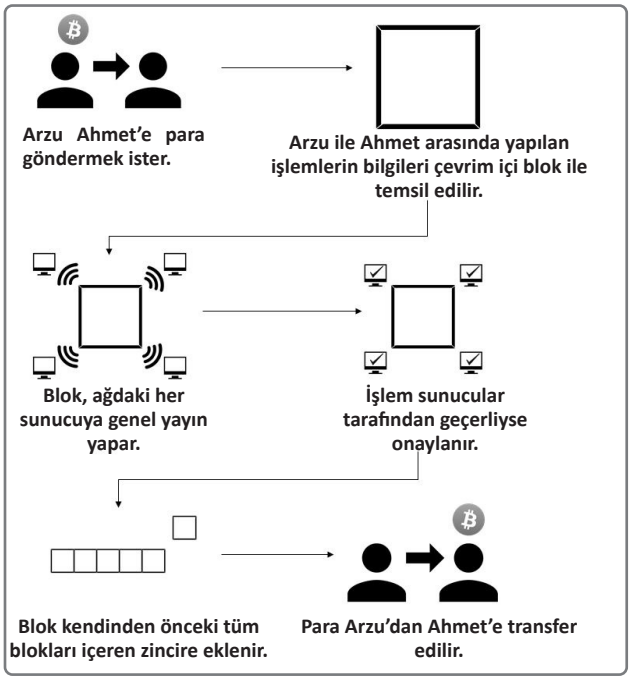
Bizans hata toleransının [Practical Byzantine Fault Tolerance (PBFT)] adı, Bizans generallerin kullandığı yöntemden gelir. Yöntemin kullanıldığı sistemde İmparator, emirlerini generallere birden fazla haberci göndererek ulaştırırdı. İmparatordan gelen emrin elçilerin çoğunluğu tarafından doğrulanmış olması durumunda ise generaller, emiri doğru olarak kabul ederdi (Görsel 3.1.a). Gelen emrin elçilerin çoğunluğu tarafından doğrulanmış olmaması durumunda ise imparatordan gelen emir kabul edilmezdi (Görsel 3.1.b). Generaller, bu yöntemi imparatordan gelen emrin gerçek olup olmadığını belirlemek için kullanırdı.



Görsel 3.1.a: Koordineli saldırı ile kazanma

Görsel 3.1.b: Koordinasyonsuz saldırı ile yenilgiye uğrama

Bizans hata toleransı blok zinciri teknolojisinde de kullanılmaktadır. Görsel 3.2'de blok zincirinde bir işlemin (transaction) gerçekleşme süreci gösterilmektedir.



Görsel 3.2: Blok zincirinde bir işlemin gerçekleşme süreci

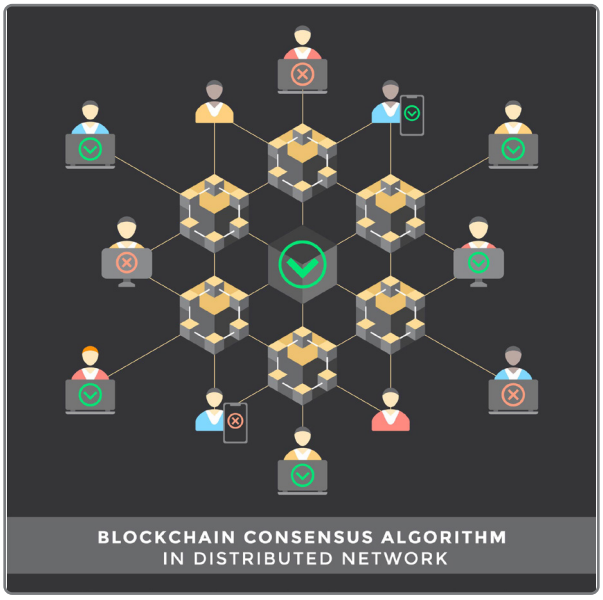
Blok zinciri ağına dâhil her cihaz, özel bir açık-gizli anahtar ikilisine ve diğer cihazların açık anahtar bilgisine sahiptir. Her cihaz kendisine ulaşan işlem (transaction) bilgisini kontrol eder ve onayladığında işlemi imzalayarak ağ ile paylaşır. Ağdaki bir işlemin belirli sayıda cihaz tarafından onaylanması durumunda mutabakat sağlanır ve işlem, ağ tarafından geçerli kabul edilir. Örneğin 2n cihazdan oluşan bir ağda mutabakat oluşması için n + 1 cihazın işlemi onaylaması gerekir.

SIRA SİZDE

Sınıfta arkadaşlarınızla oyunlaştırma yöntemini kullanarak blok zincirinde işlem gerçekleşme sürecini canlandırınız.

3.2. İŞ KANITI (PROOF OF WORK) MUTABAKAT MEKANİZMASI

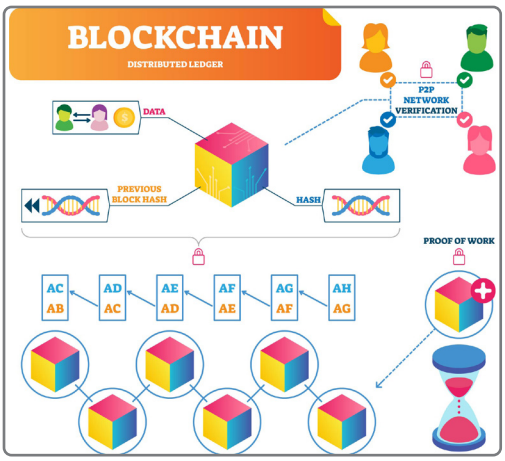
Mutabakat (Consensus) ya da **uzlaşma**, birbirlerine güven duymayan tarafların aralarında, verinin son haliyle ilgili anlaşmaya varma süreci olarak ifade edilir. Tarafların mutabakata varabilmeleri için birbirinden farklı algoritma türleri kullanılabilir. İki taraftan oluşan sistemlerde mutabakata varmak zor değildir. Görsel 3.3'te görüldüğü üzere birden fazla tarafın bulunduğu dağıtık sistemlerde ise tarafların tek bir değer üzerinde uzlaşması gerektiğinde mutabakata varmak zorlaşır.



Görsel 3.3: Blok zinciri mutabakat algoritması

Blok zinciri ağında kullanılan mutabakat mekanizmalarından iş kanıtında, blok zinciri ağına eklenecek yeni blok için çözümü zor fakat çözümün doğruluk kontrolünün kolay yapıldığı bir problemin çözülmesi gerekir.

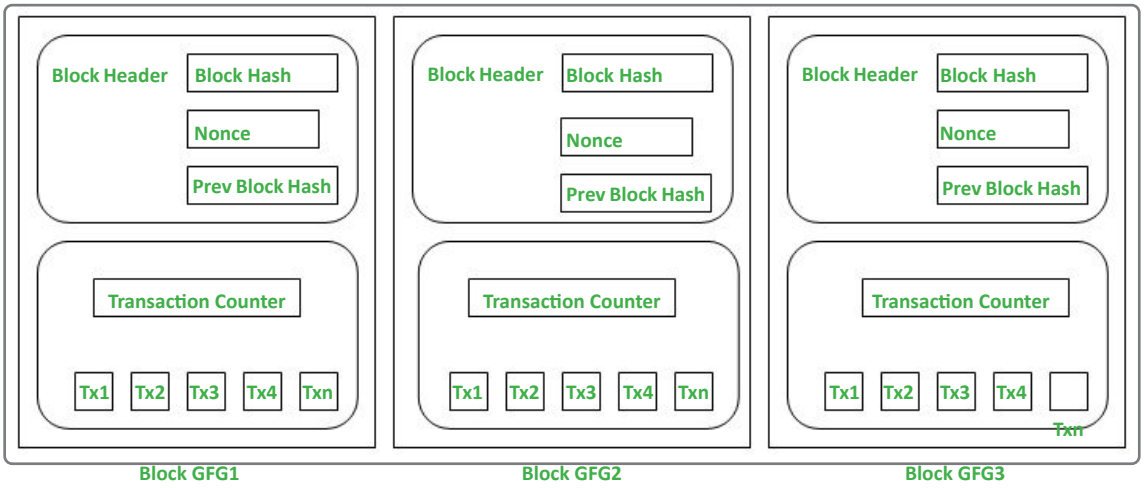
Görsel 3.4'te görüldüğü üzere iş kanıtı algoritmalarını kullanan blok zinciri sistemlerinde, katılımcıların blok zinciri ağına dâhil olabilmesi için bir çeşit iş (work) yapması beklenir. İş kanıtı algoritmasını kullanan Eşten-eşe Elektronik Nakit Ödeme Sistemi'ndeki madenci düğümler, sıradaki bloku çözmek için işlemci güçlerini sisteme verirken aynı zamanda işlemleri kayıt altında tutar ve geçerli kılar. Madencilik süreci yüksek enerjiyle birlikte işlemci gücü gerektirir. Madencilik için kullanılan elektrik enerjisinin büyüklüğü de çevreye ciddi zarar verir.



Görsel 3.4: İş kanıtı Algoritması

Görsel 3.5'te İş kanıtı algoritmasının çalışması gösterilmektedir. İş kanıtı, Eşten-eşe Elektronik Nakit Ödeme Sistemi için çift harcama sorununu çözer. Bu sayede finansal değerlerin aktarılabildiği güvenli bir ağ ortamı sunar. Bunun yanı sıra madenciler arasında gerçekleşen rekabet ve en uzun zincir kuralı, blok zincirinde eşitlik ortamı oluşturur. Blok zincirinin yok edilmesine karşı, %51 ve Genesis saldırısı için yüksek seviyede korunma sağlar.

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde herkes tam düğümünü çalıştırabilir ve madenciliğe başlayabilir. Ayrıca konu ile ilgili internet siteleri aracılığıyla kişiler blok zincirleri inceleyebilir veya denetleyebilir.



Görsel 3.5: İş kanıtı (Proof of Work) algoritmasının çalışma mantığı

Bir blok, o bloku tanımlayan bir blok başlığı (block header) ve blok özeti (block hash), Nonce değeri, önceki blok özeti (prev block hash), işlem sayacı (transaction counter) ve işlem kayıtlarından (transactions) oluşmaktadır. Yeni oluşan blokun girdilerinden biri kendisinden önce oluşturulmuş blokun özet değeridir. Önceki blok özetleri (Prev block hash) kendinden önceki

blok özetleri ile birbirlerine bağlanır. İş kanıtı algoritmasını kullanan Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde her blok 500'den fazla işlem (transaction) taşır. Ortalama işlem boyutu 250 byte olup blok başlığında 80 byte veri bulunur. Nonce değeri Pow tarafından kullanılan sayaç değeridir. Madenciler İş kanıtı problemlerini çözebilmek için de Nonce değerini kullanır. Blok bilgileri değişmeden yeni bir özetin oluşmasında kullanılır. Bu süreç blok zincirinin değiştirilemez olmasına imkan sağlar. Nonce değeri 0 ile 2^{32} arasında değer alabilir.

SIRA SİZDE

İş kanıtı (Proof of Work) mutabakat mekanizması'nın çalışması sürecinde ortaya çıkan ısı enerji verimliliği bakış açısı ile nasıl tekrar kullanabiliriz? Bu konuyu oluşturacağınız küçük gruplarda tartışınız ve bir sunum hazırlayarak sınıfta grup sözcünüz aracılığıyla arkadaşlarınızla paylaşınız.

3.3. HİSSE KANITI (PROOF OF STAKE) MUTABAKAT MEKANİZMASI

Blok zincirinde hisse kanıtı algoritmalarında madencilik süreci yoktur. Burada madencilerin yerine doğrulayıcılar (validator) bulunur. İş kanıtı algoritmasında blok ödülü yoktur. Doğrulayıcılar yaptıkları işlemler için ücret alır. Bu algoritmada, sistemde doğrulayıcı olmak için belirli miktarda kripto para hisse kanıtı için tutulur. Doğrulayıcılar, blok zincirinde blokları bulmak ve işlemleri blok zincirine eklemek için birlikte çalışır. Blok ekleme işleminden sonra toplanan işlem ücretleri hisse kanıtı için ayırdıkları kripto para miktarı oranında doğrulayıcılara paylaştırılır. Enerji kullanımı bakımından iş kanıtı algoritmasına göre çevreye zararı daha azdır. PoW ve PoS dışında birçok konsensüs (uzlaşma) algoritması da bulunur. Tablo 3.1'de İş kanıtı ve hisse kanıtı karşılaştırmalı olarak verilmiştir.

Tablo 3.1: İş Kanıtı (PoW) ve Hisse Kanıtı (PoS) Karşılaştırması

	Proof-of-work	**Proof-of-stake**
Güç tüketimi (Power Consumption)	Yüksek	Düşük
Güvenlik (Security)	Yüksek	Test edilmemiş
Ekipman Gereksinimi (Required Tools)	Madencilik ekipmanı gerekir	Madencilik ekipmanı gerekli değildir
Merkezileşme riski (Risk of centralisation)	Merkezi olma eğilimi yüksek	Kullanıcılar kendi varlıklarının kontrolünü sürdürebilir

SIRA SİZDE

İş kanıtı (Proof of stake) mutabakat mekanizması'nda neden güvenliğin yüksek olduğunu ve güç tükeminin de düşük olduğunun gerekçelerini araştırıp, arkadaşlarınızla tartışınız.

3.4. OTORİTE KANITI (PROOF OF AUTHORITY) MUTABAKAT MEKANİZMASI

Proof of Stake'in (PoS) düzenlenmiş biçimidir. Özel ve izin gerektiren blok zinciri ağlarında kullanılır. Blok zinciri ağında blokun üretilme, doğrulanma ve zincire eklenme adımlarında işlem gücü ya da hisse miktarı gibi süreçler bulunmamaktadır. Herhangi bir ekonomik varlık gerektirmeyen verimli bir mutabakat mekanizmasıdır. Ayrıca bu mutabakat algoritmasında madencilik sürecinin bulunmaması, algoritmanın performansına ve ölçeklenebilirliğine olumlu katkı sunmaktadır. Blok onayları özel ağlardaki güvenilir ve rastgele seçilen yetkili doğrulayıcılar tarafından yapılır. Verilen onaylar, ağın tamamı tarafından otomatik olarak kabul edilir. Onay süreçlerinde istenmeyen durumların yaşanmaması için yetkili makamlar iyi seçilmeli ve bunlar denetlenmelidir.

3.5. GECİKTİRİLMİŞ İSPAT (DELAYED PROOF OF WORK) MUTABAKAT MEKANİZMASI

Geciktirilmiş İspat (Delayed Proof of Work), İş Kanıtı (Proof of Work) mutabakat algoritmasının farklılaştırılmış versiyonudur. İş kanıtı algoritmasından ayrılan yönü mutabakat algoritması değil, iş kanıtı kurallarını destekleyen bir güvenlik mekanizması olmasıdır. Geciktirilmiş ispat, onaylanmış blokların tekrar düzenlenmesine izin vermez. Bu da blok zinciri ağlarını çok daha güvenli duruma getirir. Blok zinciri ağına gerçekleştirilebilecek %51 saldırılarına karşı, ağın daha dayanıklı olması imkânını sunar. Örneğin saldırganların, Eşten-eşe Elektronik Nakit Ödeme Sistemi'ni çökertebilmesi için blok zinciri ağındaki yedeklenmiş tüm anlık görüntüleri yok etmesi gerekir. Ayrıca geciktirilmiş ispat, sık yedekleme izinleriyle saldırı durumda ya da sistem hatası durumlarında tüm verinin hızlıca kurtarılmasını sağlar. En son yedeklemeden daha eski işlemlerle ilgili ortaya çıkabilecek sorunlarda, doğru kaydı bulmak için seçilen iş kanıtının blok zincirindeki yedeklerine bakılır.

3.6. HİBRİT MUTABAKAT ALGORİTMALARI

Hibrit mutabakat algoritmalarında; İş kanıtı (Proof of Work) ve Hisse kanıtı (Proof of Stake) mutabakat mekanizmaları birlikte kullanılarak, her ikisinin olumlu yönlerinden yararlanmak ve her birinin kendine özgü eksikliklerini dengelemek için diğerini kullanmak amaçlanır.

3.7. YENİ NESİL MUTABAKAT MEKANİZMALARI

Mutabakat mekanizmalarından yeni nesil mutabakat mekanizması kullanan bir blok zinciri, "Yönlü Düz Ağaç" (DAG) mutabakat protokolünü kullanarak işlemleri işlemek ve doğrulamak için bağlantılı tüm cihazları kullanır. DAG, ağdaki işlemlerin paralel şekilde işlenmesine olanak sağlar. Doğrulayıcılar ise yeni bir işlemin geçerli olup olmadığını belirlemek için rastgele seçilen diğer doğrulayıcıları sorgular. Alt örneklemeden rastgele yapılan doğrulama ile bir işlemin yanlış olma durumu istatistiksel olarak kanıtlanır.

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

1. Aşağıdakilerden hangisi Blok zinciri kavramının tanımına uygun değildir?

A) Herhangi bir merkeze bağlı kalmadan çalışır.

B) Tek bir merkezden çalışır.

C) Bir dijital güven protokolüdür.

D) Blok zincirine yazılan veriler kalıcıdır.

E) Blok zincirinde mutabakat algoritamaları kullanılır.

2. Aşağıdakilerden hangisi Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin kullandığı mutabakat algoritmasıdır?

A) Hisse Kanıtı(Proof of Stake) B) İş Kanıtı (Proof of Work)

C) Hibrit mutabakat algoritması D) Geçiktirilmiş İspat (Delayed Proof of Work)

E) Otorite Kanıtı (Proof of Authority)

3. Aşağıdakiler özelliklerden hangisi Hisse Kanıtı'na (Proof of Stake) ait değildir?

A) Güç tüketimi düşüktür. B)Madencilik ekipmanlarına ihtiyaç yoktur.

C) Güvenliği test edilmiştir. D) Kullanıcılar kendi hisselerini kontrol edebilir.

E) Merkezileşme riski düşüktür.

4. I.Hisse Kanıtı (Proof of Stake)

II. İş Kanıtı (Proof of Work)

III. Hibrit mutabakat algoritması

IV. Geciktirilmiş İspat (Delayed Proof of Work)

V. Otorite Kanıtı (Proof of Authority)

Yukarıdakilerden hangileri Ethereum'un kullandığı ve kullanacağı mutabakat algoritmalarıdır?

A) I-II B) I-III C) II-IV D) I-III-V E) II-IV-V

5. Aşağıdakilerden hangisi Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin blok başlığında bulunmaz?

A) Blok başlığı (Block header) B) Blok özeti (Block hash)

C) Nonce değeri D) Önceki blok özeti (Prev block hash)

E) Geri sayım sayacı(Countdown counter)

KONULAR

4.1. EŞTEN-EŞE ELEKTRONİK NAKİT ÖDEME SİSTEMİ PROTOKOLÜ

4.2. EŞTEN-EŞE ELEKTRONİK NAKİT ÖDEME SİSTEMİ OYUN TEORİSİ

4.3. EŞTEN-EŞE ELEKTRONİK NAKİT ÖDEME SİSTEMİ'NDE MADENCİLİK VE ZORLUK SEVİYELERİ

4.4. EŞTEN-EŞE ELEKTRONİK NAKİT ÖDEME SİSTEMİ SALDIRILARI

4.5. LIGHTNING AĞI

NELER ÖĞRENECEKSİNİZ?

- Eşten-Eşe Elektronik Nakit Ödeme Sistemi Protokolü
- Eşten-Eşe Elektronik Nakit Ödeme Sistemi Ödüllendirme Politikası
- Oyun teorisi
- Eşten-Eşe Elektronik Nakit Ödeme Sistemi ile Oyun Teorisi İlişkisi
- Oyun teorisi bileşenleri
- Eşten-Eşe Elektronik Nakit Ödeme Sistemi %51 Saldırısı
- Eşten-Eşe Elektronik Nakit Ödeme Sistemi Sybil Saldırısı
- Lightning ağı

ANAHTAR KELİMELER

Eşten-eşe Elektronik Nakit Ödeme Sistemi, çift harcama, kişilik doğrulama, madenci, madencilik, Nash dengesi, oyun teorisi, sybil saldırısı, yarılama süresi, %51 saldırısı

HAZIRLIK ÇALIŞMALARI

1. Üretimi yapılan bir ürün çeşitli nedenlerle zamanla üretimi zorlaşarak daha az üretilse nasıl sonuçlar doğabilir? Düşüncelerinizi arkadaşlarınızla paylaşınız.

2. Güven üzerine kurulmuş olan blok zinciri teknolojisine, bilgisayar korsanları nasıl saldırı gerçekleştirebilirler? Bu saldırıları kime nasıl zarar verebilir?

3. Bir savaş sırasında üst düzey komutanların bazılarının isteyerek yanlış bilgiler verip yanlış kararlar aldırması durumunda yeni bir strateji geliştirmek için nasıl bir yol izlenebilir? Düşüncelerinizi arkadaşlarınızla paylaşınız.

BLOK ZİNCİRİ 1.0 MİMARİSİ

4.
ÖĞRENME BİRİMİ

4.1. EŞTEN-EŞE ELEKTRONİK NAKİT ÖDEME SİSTEMİ PROTOKOLÜ

Protokol, bir verinin bilgisayar sistemleri arasında belirli sisteme göre gönderilmesi, alınması veya paylaşılmasını içeren kurallar dizisidir. **Kripto para için protokol**, dijital paranın internet ortamında güvenli bir şekilde takas edilmesini sağlayan ve blok zincirinin yapısını bilgilendiren bir dizi kuraldır. Bu zincir, daha önce de bahsedildiği gibi dağıtılmış veri tabanlarından oluşur.

Blok zinciri teknolojisi, dağıtılmış veri tabanları üzerinde çalışan bir dağıtık defter teknolojisidir. Blok zincirinde gerçekleştirilen her bir Bitcion işlemi, defterden izlenebilir ve doğruma işlemi yapılabilir. Bu yapı sayesinde sürekli bir kontrol işlemi sağlanır ve işlemler en üst seviyede güvenlik ile gerçekleştirilir. Sistemde yer alan madenciler, sahip oldukları bilgisayar sistemleri ile blok zincirinin çalışmasını sağlayarak karşılığında ödüllendirilir. Bahsedilen tüm bu işlemler, Eşten-eşe Elektronik Nakit Ödeme Sistemi protokolünü oluşturur.

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin protokolü sayesinde dijital paraların kişiler arasında takası, internet üzerinden güvenli bir şekilde ve herhangi bir merkezî otoriteye bağlı kalmadan mümkün hâle gelir. Eşten-eşe Elektronik Nakit Ödeme Sistem protokolünün ardından, kendi protokollerine sahip farklı kripto paralar geliştirilir. Kripto paralar, protokoller sayesinde blok zinciri aracılığı ile bilgisayar ağına erişir.

Eşten-eşe Elektronik Nakit Ödeme Sistemi protokolündeki en önemli özelliklerinden bir diğeri ise bir paranın iki defa harcanmasının önüne geçmesidir. Örneğin yemekhane kartı başkası tarafından kullanılan bir öğrencinin, okul yemekhanesine geldiğinde, kartın o gün kullanıldığını ve ikinci bir defa kullanılamayacağını öğrenmesi olayında olduğu gibi Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin sahip olduğu protokol sayesinde dijital paranın aynı anda ikinci kez kullanımı önlenir. Harcanmış veya takas edilmiş bir paranın tekrar kullanılması blok zincirindeki kayıtlar ve doğrulama süreci sayesinde mümkün değildir.

Tablo 4.1'de Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin ağ üzerindeki iletişim aşamaları ve bu aşamalar sırasında düğümlerin ne yaptığı gösterilir.

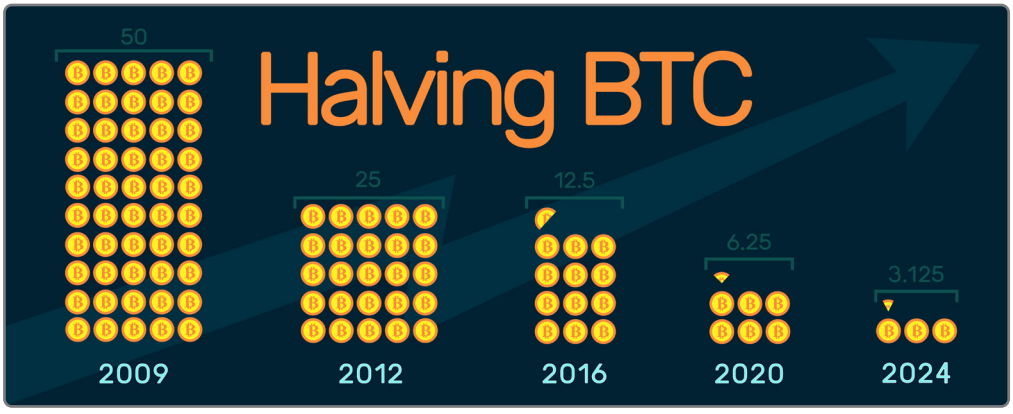
Tablo 4.1: Eşten-eşe Elektronik Nakit Ödeme Sistemi Protokol Aşamaları

	1. Adım	2. Adım	3. Adım	4. Adım	5. Adım	6. Adım
	Broadcast	**İşlem Bloku**	**İş Kanıtı (PoW)**	**Broadcast PoW**	**Yeni Blokun Kabulü**	**Zincire Yeni Blok Ekleme**
Aşama	Yeni işlemler tüm düğümlere yayınlanır.	Her düğüm, yeni işlemleri bir blokta toplar.	Her düğüm, bloku için zor bir PoW bulmaya çalışır.	Bir düğüm, PoW'unu bulduğunda bloku tüm düğümlere yayınlar	Düğümler, yalnızca içindeki tüm işlemler geçerliyse ve henüz harcanmamışsa bloku kabul eder.	Düğümler, önceki hash'i kullanarak zincirde, bir sonraki bloku oluşturmaya çalışarak bloku kabul ettiklerini ifade eder.
Elektrik tüketimi	Düşük	Yüksek	Yüksek	Düşük	Düşük	Düşük
Düğümler-CPU		X	X			
Ağ iletişimi	X			X		
Disk ve bellek kullanımı	X				X	

4.1.1. Eşten-eşe Elektronik Nakit Ödeme Sistemi Ödüllendirme Politikası

Blok zinciri protokollerinin ağı güvenceye almak ve sistemin çalışmaya devam etmesini sağlamak için bazı teşvikler sunması gerekir. Kritpo para birimlerinde blok ödülleri veya teşvikleri, kişilerin ağa katılmasını sağlamak için en önemli araçlardan biridir. Blok zincirine yeni blokların eklenmesi ve doğrulama gibi işlemlerden sonra, madencilik yapan kullanıcılarına bu teşvikler sunulur.

Verilen ödüller bazen sabit bir değerde olabilirken (her işlemden sonra aynı sayıda jeton ödülü) bazen de bu ödül sayısı kademeli olarak azalır. Örneğin Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde her dört yılda bir veya 210 bin blokta bir, yarılama süreci bulunur. Bu da kazanılan ödüllerin her dört yılda yarı yarıya azalacağı anlamına gelir. Blok ödülleri, dolaşımda 21 milyon adet paraya ulaşıncaya kadar bu yarılama süreci devam eder. Toplam 21 milyona ulaşan para arzı sonrasında ise Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde ödüllendirme sistemi sona erer. Yarılama süreçleri içinde ödül dağıtım grafiği Görsel 4.1'de gösterilmiştir.



Görsel 4.1: Eşten-eşe Elektronik Nakit Ödeme Sistemi yarılama süreci ve ödül miktarı

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde madenci, kendi ürettiği ve zincire dâhil ettiği bloklardan ücret kazabilirken aynı zamanda yapmış olduğu işlemlerden dolayı da ödüller kazanabilir. Zamanla blok ödülü daha az önemli hâle gelir ve düğümlerin madencilik yapma isteği azalır. Bunu telafi etmek için işlem ödülleri geliştirilmiştir. Arzdaki sınır, işlem ücretlerinin bir noktada blok ödülünün yerini alması gerektiğini açıkça ortaya koyar. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde üretim sürecinin tamamlanmasından sonra, sadece yapılan işlemlerden ödüller kazanılmaya devam edileceği varsayılır.

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde ödüllendirme politikası ile sağlanan avantajlar şunlardır:

Maliyet Verimliliği: Blok zinciri ağındaki işlem onaylamaya bağlı olarak yürütülen işlemlerden, ağdaki müşterilerinin yararlanması nedeniyle üçüncü şahıslara ödenen komisyon bulunur. Kendi içinde ödüllendirme politikası, yeni müşteri kazanımı açısından da etkili bir yöntemdir.

Dürüstlük: Ödüllendirme politikası, blok zincirine dâhil olan düğümlerin hangi işlemden ne kadar kazanacağını belirler. Kötü niyetli bir düğüm, güçlü teçhizatlar kullanarak en uzun zinciri oluştursa bile tüm blok zincirinin kontrolünü ele alamaz. Kötü niyetli düğüm, kontrolü ele geçirirse insanların blok zincirine olan güveni sarsılacağından Eşten-eşe Elektronik Nakit Ödeme Sistemi değer kaybedecek ve sonuç itibari ile daha fazla kaybedecektir. Bu durum, düğümler arasında güvensizlik duygusunu ortadan kaldırır.

4.2. EŞTEN-EŞE ELEKTRONİK NAKİT ÖDEME SİSTEMİ OYUN TEORİSİ

Oyun teorisi, iki veya daha fazla kişi arasında oynanan bir oyunda oyunun sonucunun, her bir oyuncunun eylemine bağlı olduğu stratejik bir etkileşim süreci olarak tanımlanabilir. Bu sürece örnek olarak satranç oyununda, bir kişinin yapacağı hamleye karşılık bir diğerinin yapacağı hamle ve oyunun bu hamleler sonucunda şekillendirilmesi ve sonlanması verilebilir.

4.2.1. Oyun Teorisi

Oyun teorisi, genel olarak iki net sonucu olmayan durumlarda kullanılır. Bir sistem içindeki bireyler, birbirleri ile etkileşime girdiklerinde sonuçlar tahmin edilemeyecek kadar karmaşık olabilir. Bir davranışın sonucunu tahmin etmede bireylerin aldıkları kararlar, stratejiler ve mevcut bilgiler önemli etkiye sahiptir.

Oyun teorisini anlatmak için aşağıda 2 farklı örnek senaryo verilmiştir:

1.Mahkûm Problemi

Bu senaryoya göre bir suça karışan A ve B kişileri, ayrı odalarda birbirlerini görme ve iletişim kurma imkânı olmadan sorgulanır. Her iki kişinin sorgusunu yapan savcı, bu kişilerle konuşarak ikna etmeye çalışır.

- A kişisi, B kişisi aleyhinde ifade verirse A kişisi serbest kalacak ve B kişisi 10 yıl ceza alacak ya da B kişisi, A kişi aleyhinde ifade verirse B kişisi serbest kalacak ve A kişisi 10 yıl ceza alacaktır.
- A kişisi, B kişisi aleyhinde ifade verir ve B kişisi de A kişisi aleyhinde ifade verirse (birbirleri aleyhinde iki ifade) her ikisi de beşer yıl ceza alacaktır.
- Her iki kişi de birbirleri aleyhinde ifade vermezlerse bu durumda her ikisi de delil yetersizliğinden birer yıl ceza alacaklardır.

A ve B kişilerinin bu durumda alacakları en yüksek ceza 10 yıl ve en az ceza ise 1 yıldır ancak en az ceza için bu iki kişinin sessiz kalması ve aleyhte bir ifade vermemesi gerekir. Birbiri ile iletişim kurmayan bu iki kişinin nasıl bir hamle yapacağı belli değildir. En iyi sonuç için bir güven gereklidir ama sonucun ne olacağı kişiler özelinde belli değildir. Her ikisi de birbirini suçlar ifade verirse her ikisi de beşer yıl ceza alacağından en iyi sonuçtan uzaktır.

2. Ortadaki Para

Bu senaryoda bir masa üzerinde ortada iki kutu bulunur. Bu kutuların birinin içinde 50 TL, diğerinin içinde ise 200 TL vardır. Birbirleri ile iletişim kurma imkânı olmayan A ve B adlı iki oyuncudan birer kutu seçmesi istenir ancak aynı kutuyu seçerlerse ikisi de bir şey kazanamayacaktır.

- Her ikisi 50 TL'lik kutuyu seçerse yine ikisi de bir para kazanamayacak ve kazanç 0 TL olacaktır.
- Her ikisi de 100 TL'lik kutuyu seçerse yine ikisi de bir para kazanamayacak ve kazanç 0 TL olacaktır.
- A kişi 50 TL'lik kutuyu ve B kişisi de 200 TL'lik kutuyu seçerse paraları kazanacaklar ve toplam kazanç 250 TL olacaktır. Tam tersi kutu seçimi için de durum aynıdır.

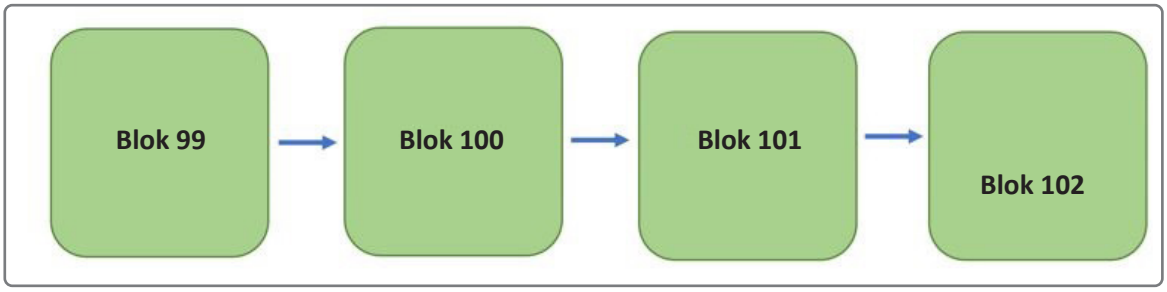
Kişilerin kararlarının sonuçlar üzerindeki bu etkisi ve yapmış oldukları eyleme bağlı olarak sonucun değişmesi bir oyun teorisi olarak ele alınır. Elde edilecek kazanımın bir başka kişinin eylemine bağlı olduğu durum, Nash dengesi ile açıklanmıştır. **Nash dengesi**, başlangıçta en iyi senaryo için strateji belirlendikten sonra oyun içinde kişinin kendisi için en uygun stratejiyi seçerek ilerlemesi ve sonuçta daha büyük bir kazanca erişemediği durumu formüle eder.

SIRA SİZDE

"Mahkûm Problemi" ve "Ortadaki Para" senaryolarını sınıfınızda canlandırınız.

4.2.2. Eşten-eşe Elektronik Nakit Ödeme Sistemi ile Oyun Teorisi İlişkisi

Eşten-eşe Elektronik Nakit Ödeme Sistemi üçüncü şahısların denetiminden bağımsız, merkezî olmayan bir ağ tarafından yönetilen blok zinciri içinde, birbiri ardına yerleştirilmiş sıralı bloklar bulunur ve her blok, bir önceki blokun hash değerini içerir. Bu şekilde bloklar birbirine bağlıdır ve zincir şeklini alır. Bu durumu ifade eden blok diyagramı, Görsel 4.2'de ifade edilmiştir.



Görsel 4.2: Eşten-eşe Elektronik Nakit Ödeme Sistemi blok zinciri blok diyagramı

Merkezî olmayan bir ağda yer alan tüm düğümlerin birbirini görmeden güvenmeleri ve doğru bir şekilde çalışmasını sağlamak oyun teorisi ile açıklanabilir. Oyun teorisi; sistemde yer alan kullanıcılar ve sistemi koruyan madenciler farkı yerlerde olsa, birbirleri ile iletişim kurmasa ve hatta birbirlerine güvenmese bile, ağın güven içinde doğru bir şekilde çalışmasını sağlar.

Kripto paraların ekonomisi, sisteme dâhil olan katılımcıların davranışları sonucunda ortaya çıkan potansiyel sonuçlar tarafından belirlenir. Sisteme dâhil olan her katılımcı iyi niyetli olmayabilir. Ağa dâhil olup sistemi içeriden bozmaya çalışabilecek katılıcıların davranışları da değerlendirilir. Eşten-eşe Elektronik Nakit Ödeme Sistemi, sistemde yer alan kullanıcıların ve madencilerin çıkarlarını bir dengeye getirerek onların belirli şekilde davranmalarını sağlamak için oyun teorisini ve ödüllendirme politikasını beraberce kullanır. Oyun teorisi ve kriptografiyi bir araya getiren Eşten-eşe Elektronik Nakit Ödeme Sistemi, işlem onaylama algoritması (proof of work) sayesinde saldırılara karşı korunaklı ve merkezî olmayan bir yapıya sahiptir. Bu sebeple Eşten-eşe Elektronik Nakit Ödeme Sistemi için Nash dengesine sahip, hileye veya sahtekârlığa karşı güvenli bir blok zinciri protokolü denilebilir.

4.2.3. Oyun Teorisinin Bileşenleri

Oyun teorisinin doğru bir şekilde işleyebilmesi için bazı bileşenlere sahip olması gerekir. Bu bileşenler şunlardır (Görsel 4.3):

Oyun: İki veya daha fazla kişinin davranışlarına bağlı bir sonuç meydana getiren sistemdir.

Bilgi Kümesi: Oyunun sürdürülebilmesi için gerekli ve mevcut olan bilgi bütünüdür.

Oyuncu: Oyunda yer alan ve karar veren kişidir.

Strateji: Oyun içindeki duruma göre bir oyuncunun alacağı karar veya davranış planıdır.

Ödeme: Bir oyuncunun oyun içindeki herhangi bir sonuca ulaşmasından dolayı hak ettiği ödüldür.

Denge: Oyuncuların verdikleri kararlar ve ulaştıkları sonuçların son durumudur.

Görsel 4.3: Oyun teorisi bileşenleri

Oyun teorisinin Eşten-eşe Elektronik Nakit Ödeme Sistemi'ne dâhil edilmesi, ağın doru ve düzgün bir şekilde çalışmasını ve sisteme dâhil olan kullanıcı ve madencilerin davranışlarını istenilen sonuca yönlendirmesini sağlar.

4.3. EŞTEN-EŞE ELEKTRONİK NAKİT ÖDEME SİSTEMİ'NDE MADENCİLİK VE ZORLUK SEVİYELERİ

Blok zinciri temelli çalışan Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde iki tür oyuncu bulunur. Bunlar, kullanıcılar ve madencilerdir.

Kullanıcılar: Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde kripto para göndermek ve almak için yer alan kişilerdir.

Madenciler: Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde gerçekleşen işlemlerin kimliğine doğrulama (onaylama) yapan ve yeni blokların keşfedilmesini, blok zincirine eklenmesini sağlayan kişilerdir. Bu işleme de **madencilik** adı verilir.

Kullanıcıların blok zinciri içinde para almak ve göndermek için yani herhangi bir kripto para ticareti yapabilmesi için bir genel (public) ve bir de özel (private) anahtara sahip olması gerekir. Madenciler de kullanıcıların yaptığı bu işlemleri doğrular ve onaylar.

Madenciler, birtakım hesaplamalar sonucunda yeni bir blok keşfedip bu bloku blok zincirinin sonuna ekler. Ekleme işlemi için de blokların doğrulanması gerekir. Eşten-eşe Elektronik Nakit Ödeme Sistemi merkezî bir yapıda olmadığından, dünya üzerinde yayılmış birçok düğümden (bilgisayar sistemi) oluştuğundan gerçekleştirilen tüm işlemlerin ve blok ekleme sürecinin doğrulanması ancak düğümlerin fikir birliğine varması sonrası gerçekleşebilir. Madenciler daha fazla ödüllendirilmek ve kazanmak için hile yapmayı tercih edebilir. Birbirini hiç görmeyen ve tanımayan tüm blok zinciri aktörlerinin birbirine güvenmesi oyun teorisi ile sağlanır. Eşten-eşe Elektronik Nakit Ödeme Sistemi'ndeki madenci için en doğru davranış, dürüst davranarak blok zinciri güvende tutmak olacaktır. Aksi takdirde sistemin aksaması hâlinde kendi kazancı da azalacaktır ancak Eşten-eşe Elektronik Nakit Ödeme Sistemi daha popüler hâle geldikçe blok zinciri ağına katılan bilgisayarların sayısı da artmaktadır. Artan bilgisayar sayısı ile daha fazla madenci, sınırlı blok ödülleri için birbirleri ile rekabet etmek durumunda kalır. Bu durum Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde madencilik zorluğu olarak adlandırılır. Bir madenci, Eşten-eşe Elektronik Nakit Ödeme Sistemi madencilik sistemi kurduğunda ve ağa dâhil olduğunda Eşten-eşe Elektronik Nakit Ödeme Sistemi'deki madencilik zorluğuna katkıda bulunur. Madencilerin bu donanımlarını Eşten-eşe Elektronik Nakit Ödeme Sistemi'ndeki işlemleri onaylamak ve imzalamak için çalıştırması gerekir.

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde madencilik zorluğu, ağda bulunan madenci sayına bağlı olarak değişebilir. Bir madencinin ödül elde edecek bir kod oluşturması veya blok zincirine yeni eklenecek bir bloku önerme hakkını kazanması yaklaşık olarak 10 dakika sürer. Bu sürenin korunması için Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde bir zorluk algoritması çalıştırılır. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde her 2.016 blok çıkarıldıktan sonra zorluk

seviyesi otomatik olarak yeniden ayarlanır. 2.016 blok çıkarımı yaklaşık olarak iki hafta sürer. Bu süreye zorluk dönemi denir. 2.016 blok çıkarımı tamamlandıktan sonra zorluk algoritması devreye girer. Yeni bir blok önerme ve ödül elde edilebilecek kod oluşturma süresine bakar. Bu süre 10 dakikadan az ise Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde çok madenci olduğu farz edilir ve madencilik zorluğu artırılır.

4.4. EŞTEN-EŞE ELEKTRONİK NAKİT ÖDEME SİSTEMİ SALDIRILARI

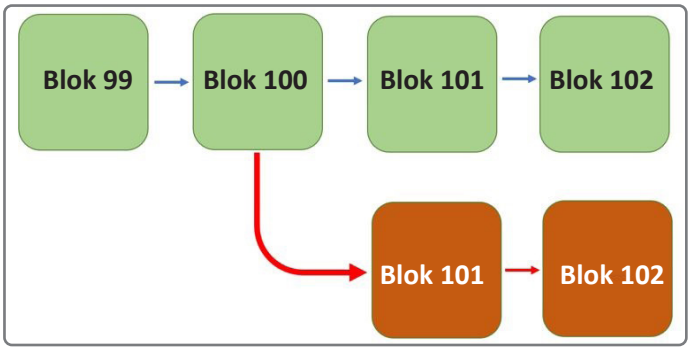
Eşten-eşe Elektronik Nakit Ödeme Sistemi'ndeki madenci için hile olarak adlandırılabilecek işlemler aşağıdaki gibi listelenebilir:

1. Geçersiz olan işlemleri onaylama
2. İş kanıtını yani doğrulama sürecini beklemeden zincire blok ekleme
3. Geçersiz bloklarla madenciliğe devam etme

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin kötü niyetli eylemlerden korunması için işlem kanıtı olarak adlandırılan ve kriptografik teknikler içeren İş Kanıtı (Proof of Work) algoritması kullanılır. Proof of Work algoritması kullanılarak Eşten-eşe Elektronik Nakit Ödeme Sistemi'ndeki kayıt defterinin doğru bir şekilde tutulduğu konusunda anlaşmaya varılır. Bu algoritma sayesinde dengeli ve rekabetçi bir kripto para madenciliği ortamı oluşturulur ancak bu işlem için harcanan süre ve teçhizatların kaynak tüketimi, madencilerin karşılarına bir dezavantaj olarak çıkar.

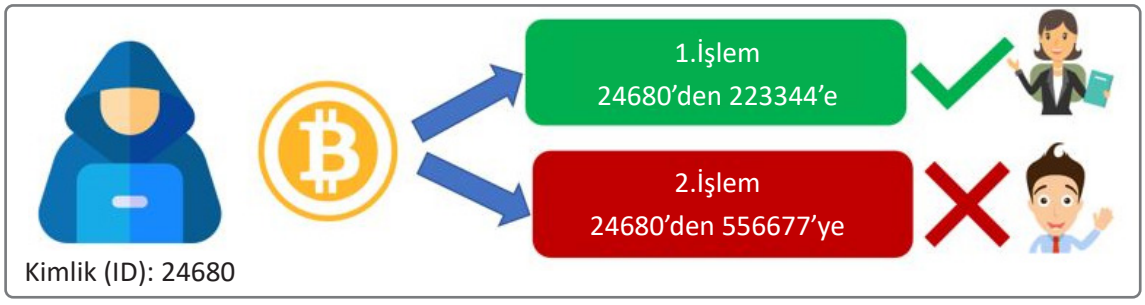
4.4.1. %51 Saldırısı

Madencilerin dürüst bir şekilde davranmalarını sağlamak için Eşten-eşe Elektronik Nakit Ödeme Sistemi, kendi içinde bir teşvik sistemine sahiptir. Örneğin yeni bir bloku keşfeden ilk madenciye belirli bir miktar ödül (kripto para) verilir. İşte bu noktada daha fazla ödül kazanmak isteyen madenciler, geçersiz blokları da keşfettiklerini bildirerek hile yapabilir. Bir blokta yürütülen geçersiz işlemi onaylayarak ödül kazanmak da hile olarak isimlendirilir. Buna bir örnek Görsel 4.4'te gösterilmiştir.



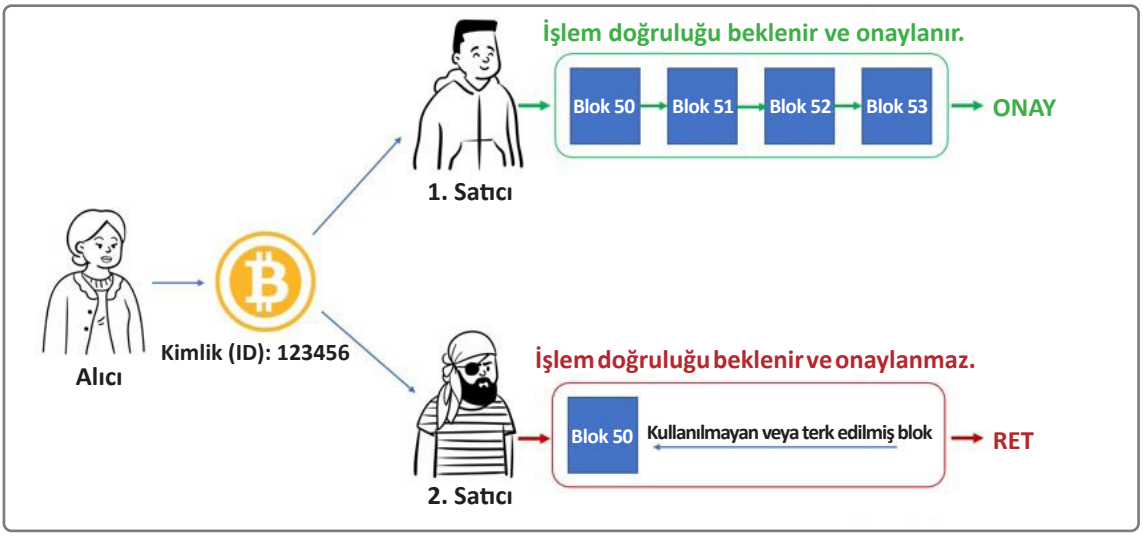
Görsel 4.4: Eşten-eşe Elektronik Nakit Ödeme Sistemi işlemlerinde hileli işlem blok diyagramı

Görsel 4.4'te yeşil blok ile gösterilen bloklar, Eşten-eşe Elektronik Nakit Ödeme Sistemi ana zincir bloklarıdır. Blok 101'de bir satma/harcama işlemi yapılmak istenmektedir.
Hileli işlem yapmak isteyen madenci, bir önceki (ana) blok olan Blok 100'den yeni bir zincirle ayrılır ve sahte yeni bir Blok 101 oluşturur. Yeşil ile gösterilen gerçek blok zincirinde satma/harcama işlemi tamamlar ve karşılığını alır ancak sahte olarak oluşturulan yeni zincirdeki Blok 101'de bu işlem yapılmaz ve para harcanmamış olarak kalmaya devam eder. Madenci hem yaptığı satışın parasını almış hem de sahte blok zincirinde hâlâ satış öncesindeki kripto parası bulunur. Bu işleme çift harcama adı verilir. Görsel 4.5'te kötü niyetli kişinin aynı kripto parayı iki defa harcamak veya satmak istemesi durumu ifade edilmiştir.



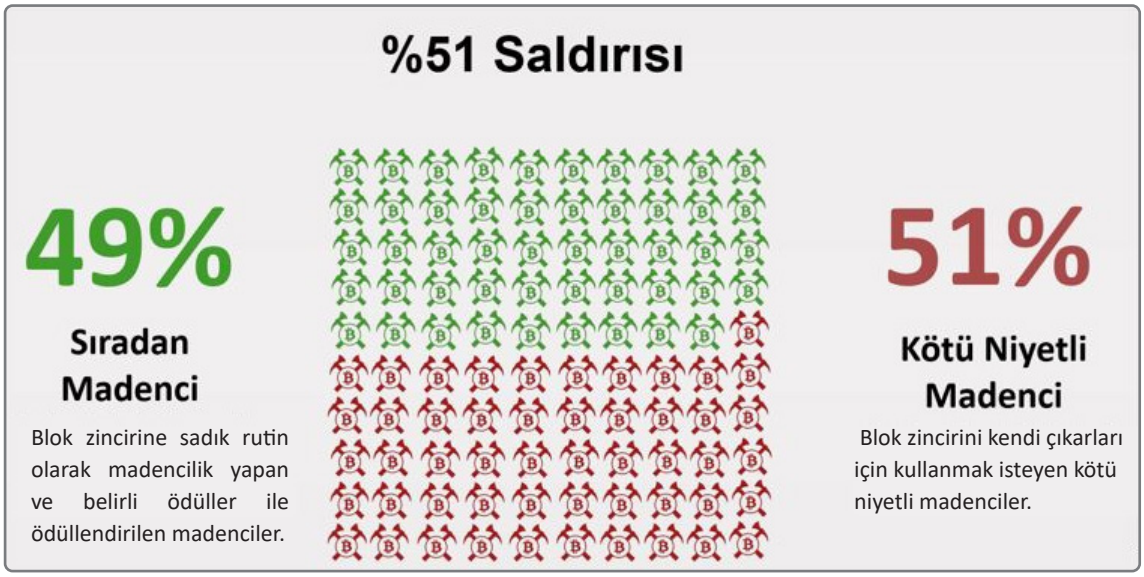
Görsel 4.5: Çift harcama işlemi

Bahsedilen bu çift harcama gibi blok zincirini tehlikeye sokacak durumların engellenmesi için Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin içerisinde Nash dengesini bulunduran ödül ve ceza sistemini barındırır. Eşten-eşe Elektronik Nakit Ödeme Sistemi'ni güçlendiren madenciler, bir blokta yürütülen işlemlerde ve zincire dâhil edilen bloklarda ödüllendirilir. Yapılacak herhangi bir çift harcamaya yönelik işlemler onaylanmaz ve ret ile sonuçlandırılır (Görsel 4.6).



Görsel 4.6: Çift harcama problemi ve blok zincirinin onaylama mekanizması

Sağlanan tüm teşvikler, aktörleri kötü niyetli olmaktan ve Eşten-eşe Elektronik Nakit Ödeme Sistemi'ne zarar vermekten korumak amaçlı geliştirilmiştir ancak yine de Eşten-eşe Elektronik Nakit Ödeme Sistemi'ni tehlikeye atmak isteyen olursa bu kişi veya kişi grubunun tek seçeneği toplam hash işlem miktarının yarısından fazlasını (%51) ele geçirmektir (Görsel 4.7).



Görsel 4.7: %51 saldırısı temsili gösterimi

Bu durumda blok zinciri zaten bozulmuş olarak sınıflandırılacağı için ekonomik ve güven endeksinden dolayı kripto para, anında değer kaybeder. Değerini kaybetmiş bir kripto paranın da önemli teçhizatlar ve enerji maliyeti olan madencilik üretimini etkileyeceğinden elde edilen kazanç değersiz hâle gelir.Bu sebeple blok zincirine saldıracak kötü niyetli madenciler, bu yolu tercih etmez ve kazançlarını azaltacak işlemler yapmak yerine kurallara bağlı kalarak madencilik faaliyetlerini yürütmeye devam eder.

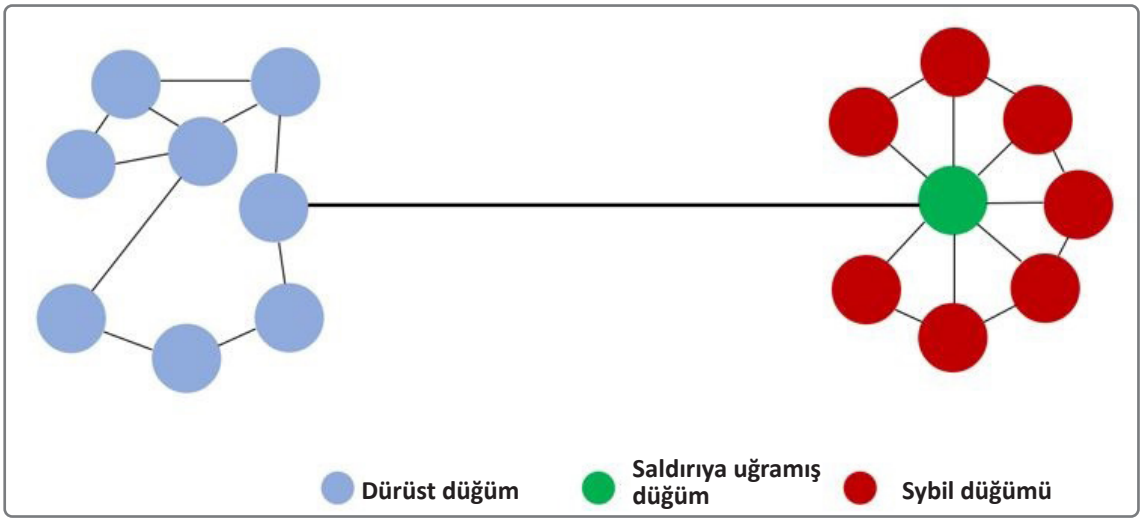
SIRA SİZDE

Saldırganların Eşten-eşe Elektronik Nakit Ödeme Sistemi'ne yönelik yapabileceği başka saldırı türleri ve olası sonuçlarını araştırınız. Bu konu hakkında bir sunum hazırlayarak arkadaşlarınızla paylaşınız.

4.4.2. Sybil Saldırısı

Blok zincirinde sybil saldırısı, madencilerin birden fazla hesap ve düğüm oluşturarak ağı ele geçirmeye çalıştığı bir saldırı türüdür. Ağa yönelik bir sybil saldırısının temel amacı, ağda alınan kararlar üzerinde daha fazla söz sahibi olarak güç elde etmektir. Bunun için ağ üzerinde farklı hesaplar oluşturur ve ağı etkilemeye çalışır. Bu durum, bir kişinin farklı e-mail adresi oluşturması veya farklı çok sayıda sahte sosyal medya hesabı açmasına benzetilebilir.

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin yönetiminde bazı kararlar madenciler tarafından karar verilir. Anket türünde bir oylama ile operasyon yönetimi için en uygun kararın alınması sağlanır. Birden fazla hesabı kontrol eden madenci, anket sonuçlarını kendi menfaati doğrultusunda etkileyebilir ayrıca sybil saldırıları ağdaki veri akışını kontrol etmek için de kullanılabilir. Ağdaki diğer kullanıcıların veya madencilerin IP adreslerini toplayarak, bu bilgileri kullanarak yeni sahte düğümler oluşturabilir. Gerçek kullanıcının veya düğümün ağı kullanmasını engelleyebilir. Bu durum, ağ güvenliğini ve gizliliğini de tehlikeye sokar. Görsel 4.8'de sybil saldırısına ait bir blok şema gösterilmiştir.



Görsel 4.8: Sybil saldırısı blok gösterimi

Görsel 4.8 incelendiğinde kötü niyetli bir düğümün veya madencinin (Kırmızı renk ile gösterilmiştir.), tüm işlemleri her iki yönde de kontrol etmek için hedef düğümü (Yeşil ile gösterilmiştir.) birden fazla düğümle çevrelediği görülür. Buradaki sybil düğümü, hedef düğümün ağ ile karşılıklı bağlantısını keserek işlemlerini engelleyebilir, komut dosyaları ve yazılım araçları üzerinden hedef düğümün yaptığı tüm işlemleri izleyebilir. Bu saldırılar genellikle çift harcama hilesi amacıyla gerçekleştirilir.

Sybil saldırıları Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde bazı sorunlara neden olabilir. Bu sorunlar kısaca aşağıdaki gibidir:

- Sybil saldırısını gerçekleştiren kişi, oluşturduğu sahte kimliklerle yeni blok eklenmesi gibi işlemleri reddederek ağın performansını düşürebilir.
- Sybil saldırısını gerçekleştiren kişi, eğer yeterince sahte kimlik oluşturmayı başarırsa ağın çalışması engellenebilir. Bu durum %51 saldırısına yol açar.

Sybil saldırılarını tamamen engellemek pek de olası değildir ancak gerçekleştirilecek sybil saldırı sayıları azaltılabilir. Bunun için alınacak önlemler ise Görsel 4.9'da gösterilmiştir.

Görsel 4.9: Sybil saldırılarını önlem için alınabilecek tedbirler

• Kimlik Doğrulama

Bu, saldıran kişi veya düğümün gerçek kimliğini ortaya çıkarmaya yönelik bir önlemdir. Genellikle telefon numarası, kredi kartı gibi doğrulama teknikleri ile kimlik bilgileri doğrulanır. Kimlik doğrulama iki yöntemle gerçekleştirilir.

Doğrudan Kimlik Doğrulama

Kimliğin doğrulanması için merkezî bir otoriteyi sorgulaması ile gerçekleştirilir.

Dolaylı Kimlik Doğrulama

Kimliğin doğrulanması için daha önce kimliği doğrulanan kişilerce onay verilmesi veya kefil olması ile gerçekleştirilir.

• Kişilik Doğrulaması

Bu, dijital ağları kimlik sahtekârlığından korumayı amaçlayan, yeni kimlik doğrulama mekanizmasıdır. Kimliğin değil kişiliğin, yani varlığın doğrulaması yapılır. Bu doğrulama için yer ve zamanı kişilerce belirlenen faaliyetler organize edilir. Bu faaliyetler sonunda madenci veya düğüm hem fiziksel hem de sanal kimliğini temsil eden bir kriptografik kimlik belirteci alır. Kişilik doğrulama yöntemiyle fiziksel ve sanal kimlikler birbirine bağlanır.

• Sosyal Güven Grafikleri

Bağlantı verilerinin analiz edilmesi sonunca sybil saldırıların azaltılması sağlanabilir. Advogato Trust Metric veya SybilGuard gibi teknikler kullanılarak sybil saldırıların önlenmesi sağlanabilir.

• Uzmanlaşmış Algoritma Kullanımı

Sybil saldırılarını azaltmak için iş kanıtı (proof of work), hisse kanıtı (proof of stake) gibi uzlaşma (consensus) algoritmaları kullanılır. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde yeni bir blok oluşturmak için iş kanıtına ihtiyaç bulunur. İş kanıtı sağlanması için harcanan işlem gücünün toplam iş gücüyle orantılı olması gerekir. Bu işlem gücü için madenci, gerçek ve güçlü bir bilgisayar sistemine ihtiyaç duyar.

Bahsedilen bilgisayar sisteminin maliyeti nedeniyle sybil saldırısı, her madencinin yönelebileceği bir saldırı olmaktan çıkar.

Sonuç olarak kripto para madenciliğinde saldırı gerçekleştirmek çok zor değildir ancak madenciler, saldırı gerçekleştirerek kripto paranın potansiyel değerini düşürmek yerine, madenciliği olması gerektiği gibi ve dürüst bir şekilde sürdürmeyi tercih eder. Ödül politikası ve madencilik teşvikleri de kullanıcıları bu saldırılardan vazgeçiren etkilerdendir.

ARAŞTIRMA

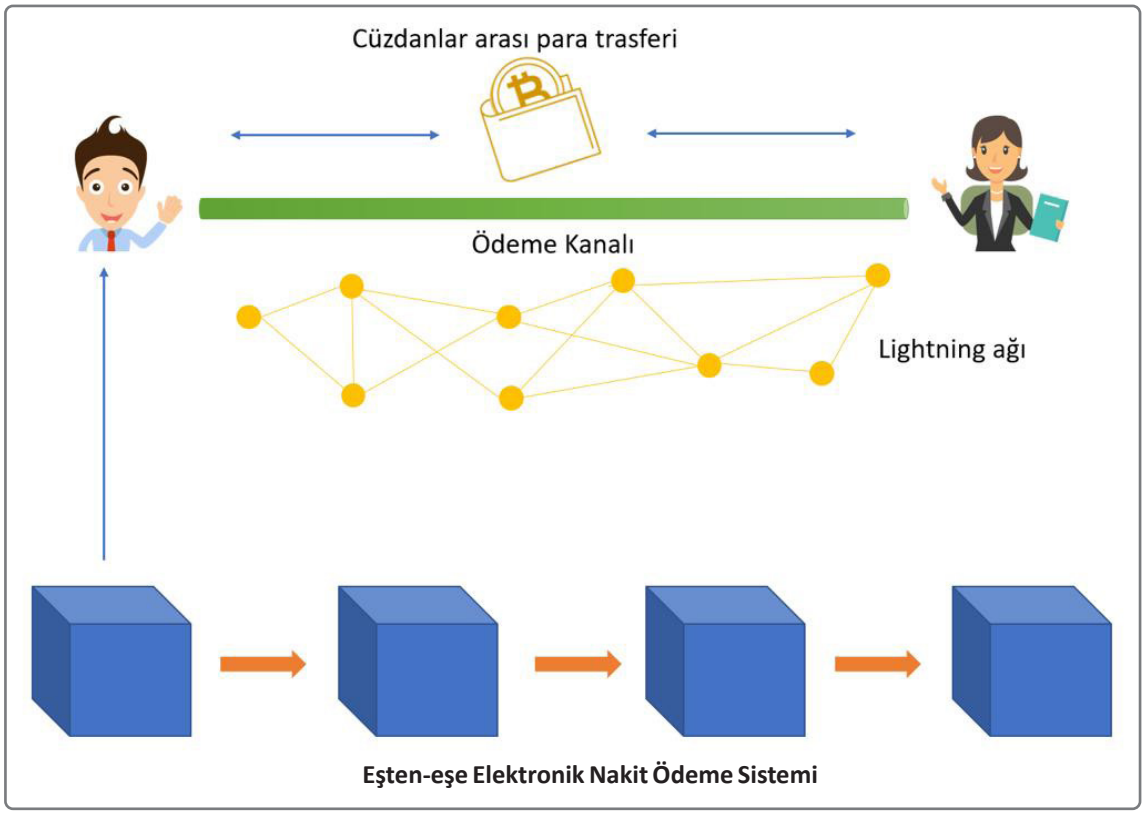
Sybil saldırısı, blok zinciri haricinde başka alanlarda da gerçekleştirilebilir mi? Araştırarak sonuçları sınıf arkadaşlarınız ile paylaşınız.

4.5. LIGHTNING AĞI

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin sahip olduğu, merkezî olmayan ve güvenli yapısının da bazı dezavantajları bulunur. Bunların başında ödeme işlemlerindeki işlem kapasitesinin yetersizliği gelir. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde blok zincirinin işlem kapasitesi, ödeme işlemlerinde kullanılmak için oldukça düşüktür. Ayrıca blok zincirinde ödeme işlemlerinde onay süreci, diğer ödeme işlemlerine göre uzun sürer. Örneğin kredi kartı kullanan bir kişi ödeme yapmak istediğinde bunu saniyeler içinde yapabilir. Kredi kartlarında ortalama işlem onaylama miktarı 2021 yılı için yaklaşık 5.200 iken Eşten-eşe Elektronik Nakit Ödeme Sistemi'ndeki blok zincirinde çok daha düşüktür. Bunun sebebi, her 10 dakikada bir blok meydana getirilir ve birkaç madencinin ağa dâhil olup bu bloku onaylaması gerekir. Bu da zaman alır.

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde işlemlerin bir bloka dâhil olması ve bu işlemleri madencilerin onaylama süreci, aslında bu sistemin ardında yatan sorunu ortaya koyar. Madenciler gelirlerini artırmak için en yüksek ücret veren işlemleri seçmek ister. Herhangi bir işlem talebinde ödeme düşük miktarda ise onaylama için ödenecek ücret, işlem talebindeki miktardan daha fazlaya mal olabilmektedir. Bu durum küçük ödemeler için yüksek işlem ücretlerini de beraberinde getirir. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin bu dezavantajını ortadan kaldırmak ve işlem ücretlerini rekabet edebilir hâle getirmek için ağ ile beraber çalışan başka bir ağ geliştirilmiştir. Bu ağ, işlemleri doğrulamak için blok zincirini kullanmadan gönderici ve alıcı arasında para transferine izin veren bir kanal sisteminden oluşmaktadır. Alıcı ve verici arasında ödeme kanalı oluşturulduktan sonra, ikili arasında anında para transferi yapılabilmektedir. Bu sisteme Lightning ağı adı verilmiştir. Lightning ağı, kullanıcı çiftleri arasında blok zinciri dışında ödeme kanalları oluşturmak için akıllı sözleşmeleri kullanır. Eşten-Eşe Elektronik Nakit Ödeme Sistemi para transfer işlemlerinin hem hızlı hem de düşük işlem ücretleri ile yapılabilmesi için tasarlanmıştır.

Lightning ağı, Eşten-eşe Elektronik Nakit Ödeme Sistemi'ne bir başka katman ekleyerek bu katman üzerinde alıcı ve gönderici arasında ödeme kanalı oluşturmasını sağlamaktadır (Görsel 4.10). Her bir kanal işlem tamamlanıncaya kadar açık kalır. İşlemler gerçek zamanlı olarak gerçekleştirilir. İkili arasında doğrudan bir alışveriş olduğundan başka bir madenci tarafından onaylanmasına gerek yoktur. Bu sebeple işlem onay ücreti çok düşük ve hatta çoğunlukla sıfır olmaktadır.



Görsel 4.10 Lightning ağında ödeme işlemi blok diyagramı

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerde parantez içine yargılar doğru ise "D", yanlış ise "Y" yazınız.

1. ( ) Kripto para en fazla 24 milyon adet üretilebilir.

2. ( ) Blok zincirinde yeni blokların keşfedilmesini ve blok zincirine yeni blokların eklenmesini sağlayan kişilere madenci denir.

3. ( ) Sahte düğümler oluşturarak bir ağının yönetimini ele geçirmek amacıyla yapılan saldırılara %50 saldırısı adı verilir.

4. ( ) Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde çift harcamayı engellemek için Nash dengesi kullanılır.

B) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

5. Aşağıda verilen işlemlerden hangisinde elektrik tüketimi diğerlerine göre daha fazladır?

A) Broadcast (yayınlama)

B) İş Kanıtı (PoW)

C) Broadcast PoW

D) Yeni Blokun Kabulü

E) Zincire Yeni Blok Ekleme

6. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde yarılama süreci aşağıdakilerden hangisinde doğru olarak verilmiştir?

A) 4 yıl veya 120 bin blok

B) 4 yıl veya 210 bin blok

C) 2 yıl veya 120 bin blok

D) 2 yıl veya 210 bin blok

E) 2024 yılı veya 21 milyon blok

7. Aşağıdakilerden hangisi oyun teorisi bileşenlerinden biri değildir?

A) Oyun B) Oyuncu C) Bilgi Kümesi

D) Oyun Şeması E) Strateji

KONULAR

5.1. ETHEREUM PROTOKOLÜ

5.2. ETHEREUM OYUN TEORİSİ

5.3. ETHEREUM MADENCİLİĞİ

5.4. MERKEZÎ OLMAYAN UYGULAMALAR (DApp)

5.5. ETHEREUM SANAL MAKİNESİ (EVM)

5.6. ETHEREUM ÇATALLANMA

NELER ÖĞRENECEKSİNİZ?

- Ethereum Protokolünü açıklar.
- Ethereum oyun teorisini ve örnek oyunları açıklar.
- Merkeziyetsiz uygulamaları (Dapps) bilir ve kullanır.
- Ethereum sanal makinasını bilir ve Ethereum Gas'ı kullanır.
- Çatallama işlemlerini tanır ve kullanır.

ANAHTAR KELİMELER

Akıllı sözleşme, ethereum, Gas, merkeziyetsiz uygulamalar

HAZIRLIK ÇALIŞMALARI

1. Sizce finansal blok zinciri ve genel amaçlı blok zinciri protokolleri arasında ne gibi benzerlik ve farklılıklar vardır? Düşüncelerinizi arkadaşlarınızla paylaşınız.

2. Kullanıcı olarak merkezi sistemi kullanmanın size sağlayacağı avantajlar ve getireceği sınırlılıklar neler olabilir?

BLOK ZİNCİRİ 2.0 MİMARİSİ

5.
ÖĞRENME BİRİMİ

5.1. ETHEREUM PROTOKOLÜ

Dünya, blok zinciri teknolojisi ile 2008 yılı sonunda Satoshi Nakomato'nun yayınladığı makale ile tanışmıştır. Marc Kenigsberg "Blok zinciri teknojisinin potansiyel ilk yaygın göstergesini Eşten-eşe Elektronik Nakit Ödeme Sistemi" olarak ifade eder. Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin sadece eşler arası para transferi için kullanılması, blok zinciri teknolojisinin farklı alanlarda kullanılmasının oluşturacağı değeri sınırlar. Bunun yanı sıra Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin sınırlı programlanabilirlik (limited programmability) yapısı ile merkeziyetsiz uygulamaların (Dapp) ağ üzerinde oluşturulmasına imkan tanımaz. Rus yazılımcı Vitalik Buterin ise programlanabilir bir blok zincirine olan ihtiyacı 19 yaşında fark etmiştir. 2013 yılı sonlarında Vitalik Buterin Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin üzerine Turing betik dili ile Ethereum'u oluşturmuştur. Görsel 5.1' de görüldüğü üzre Ethereum'un geliştirilmesindeki amaç akıllı sözleşmeler ve merkeziyetsiz uygulama platformu oluşturmaktır. **Ethereum**, farklı sektörler için değer oluşturabilecek merkezî olmayan uygulamalar için tasarlanmış yenilikçi ve açık kaynak blok zinciri platformdur. Ethereum üzerindeki uygulamaların işlemesi için herhangi bir merkezî varlığa ya da sunucuya gereksinim duyulmaz. Uygulamalar dağıtık olarak birden fazla bilgisayarda çalıştırılabilir. Bu sayede dağıtık uygulamaları kaldırmak mümkün değildir. Ethereum üzerinde dağıtık karakteristikte uygulama geliştirmek için Akıllı Sözleşme (Smart Contract) kullanmak gerekir. Akıllı sözleşmeler ise **Solidity** dili kullanılarak kodlanır. Eşten-eşe Elektronik Nakit Ödeme Sistemi 1. Nesil blok zinciri (Blok zinciri 1.0), Ethereum ise 2. Nesil blok zinciri (Blok zinciri 2.0) olarak tanımlanır.

Finansal ve Genel Amaçlı Blok zinciri	Finansal Blok zinciri	Genel Amaçlı Blok zinciri
Lansman Tarihi	3 Ocak 2009	30 Temmuz 2015
Kullanım Amacı	Dijital Para Birimi Eşten eşe Elektronik Nakit Ödeme Sistemi	Akıllı Sözleşmeler ve Merkezi Olmayan Uygulamalar Platformu
Mutabakat Mekanizması	İş Kanıtı	İş Kanıtı (Hisse Kanıtına Geçiş Planlanıyor)
Mucit	Satoshi Nakamoto	Vitalik Buterin ve Ekibi
Maksimum Arz	21 milyon	Sınırsız
İşlem Hızı	10 Dakika	Yaklaşık 20 Saniye

Görsel 5.1 : Finansal Blok zinciri ve Genel Amaçlı Blok zinciri Karşılaştırılması (https://kba.ai)

Ethereum ağına dâhil olan her bilgisayar Ethereum Sanal Makinesi (Ethereum Virtual Machine) adında sanal bir makineyi çalıştırır. Ethereum Sanal Makinesi; Solidity, Viper, Serpent gibi özel üst

düzey programlama dilleriyle kodlanmış uygulamaların Ethereum ağı üzerinde çalıştırılmasına imkân verir. Ethereum dil yapıları **Turing-complete** özelliğine sahiptir. Bu özellik sayesinde Ethereum içerisinde her şey bir program olarak kodlanabilir. Akıllı sözleşmeler elektronik ortamda oluşturulan programlanır yapılardır. Ayrıca sorgulanan bir şarta bağlı (if-else if- else yapısı) olarak içeriğe uygun şekilde otomatik olarak çalışır ve çalıştıktan sonrada çalıştırılan işlem geri alınamaz.

SIRA SİZDE

Ethereum protokolünün neden hisse kanıtı uzlaşma algoritmasını (Proof of Stake) kullanmaya geçmek istediğini araştırınız. Araştırma sonuçlarınızı sınıfta arkadaşlarınızla paylaşınız.

5.1.1. Ethereum Dünya Bilgisayarı

Ethereum'un vizyonu tüm dünyada **"Ethereum World Computer"** adında bir blok zinciri oluşturmaktır. **Ethereum World Computer**, bireysel bilgisayarların oluşturduğu tek bir bilgisayardır. Görsel 5.2'de görüldüğü üzere Ethereum ağındaki her bilgisayar, **istemci** adında bir programı çalıştıracaktır.

Görsel 5.2: Ethereum ağı

Ethereum istemcisi, diğer bilgisayar sistemleriyle iş birliği yapılmasına ve Ethereum Dünya Bilgisayarı'nın oluşturmasına olanak sağlayacaktır. Bu sistemin hayata geçebilmesi ve bütünlüğünü koruyabilmesi için bazı temel özellikleri taşıması gerekir. Bunlar; Deterministik (Deterministic), İzolasyon (Isolation) ve Sonlandırabilirlik'tir (Terminability).

Deterministik (Deterministic): Ethereum Dünya Bilgisayarı sisteminde çalışan her program ya da yapılan her işlem, deterministik nitelikte olmalıdır. Çünkü herhangi bir işlem ya da program, sistemlerden veya işlem hızından bağımsız olarak belirli girdi kümesi için her seferinde aynı çıktıyı üretiyorsa deterministiktir. Deterministiklik, sistemin bütünlüğünü korumayı sağlar.

İzolasyon (Isolation): Ethereum Dünya Bilgisayarı, dünyanın farklı yerlerinden bilgisayarların birbirine bağlanmasıyla oluşur. Akıllı sözleşmeler, bağlanan her bilgisayardan Ethereum Dünya Bilgisayarı'na yüklenebilir fakat herhangi bir sözleşmenin virüs ve siber zafiyet içermesi durumunda tüm ağ zarar görebilir. İzolasyon sayesinde bir bilgisayarda çalışan herhangi bir program, Ethereum Dünya Bilgisayarı'nın çalışmasını etkilememelidir. Çünkü Ethereum istemcileri izole olarak çalışır.

Sonlandırabilirlik (Terminability): Ethereum akıllı sözleşmeleri kendi kendini yürüten programlardır. Akıllı sözleşmelerin bir kez tetiklenmesi durumunda işleyişe müdahale etmenin ve işleyişi durdurmanın bir yolu yoktur. Bu durumda yürütülen programlar, ağdaki tüm kaynakları tüketebilecek sonsuz bir döngüye girebilir. Bunu önleyecek bir mekanizma olmalıdır.

5.1.2. Ethereum'un Temel İlkeleri

Ethereum, merkezî olmayan sözleşmeler ve uygulamalar oluşturmak için bilgisayar kodlarını depolayan ve yürüten programlanabilir bir blok zinciridir. Genel olarak temel ilkeleri aşağıdaki gibidir:

- **Basitlik**

Ortalama bir geliştirici, Ethereum ağının tüm özelliklerini kolayca kavramalı ve uygulamalıdır. Protokolü karmaşıklaştıran herhangi bir değişiklik, önemli fayda sağlamadıkça uygulanmamalıdır.

- **Evrensellik**

Geliştiriciler, Ethereum'da Akıllı Sözleşmeler yazabilir.

- **Modülerlik**

Ethereum protokolü, bağımsız yapıdaki modüllerden oluşur. Bir modelde küçük bir değişiklik yapılırsa, tüm protokol yığını herhangi bir değişiklik yapılmadan çalışabilir.

- **Çeviklik**

Ethereum ağındaki her değişiklik, uzun inceleme süre ve süreçleri olmaksızın kullanıma geçer.

5.2. ETHEREUM OYUN TEORİSİ VE ETHEREUM MADENCİLİĞİ

Bazı oyun teorisi fikirleri 18. yüzyıla kadar dayanabilir, fakat oyun teorisi fikrinin ana gelişimi 1920'li yıllarda matematikçi **Emile Borel** (1871-1956) ve bilge **John von Neumann'ın** (1903-1957) çalışmalarıyla başlamıştır. Teorinin gelişiminde John von Neumann ve **Oskar Morgenstern'in** 1944'te **Oyun ve Ekonomik Davranış Teorisi** kitabının yayınlaması etkili olmuştur. 1950'li yıllarda ise oyun teorisi modelleri iktisat teorisinde, siyaset biliminde ve psikolojide kullanılmış; psikologlar, deneysel oyunlarda insan deneklerin davranışlarını incelemek için bu teori modellerini kullanılmaya başlamıştır (Osborne, 2000).

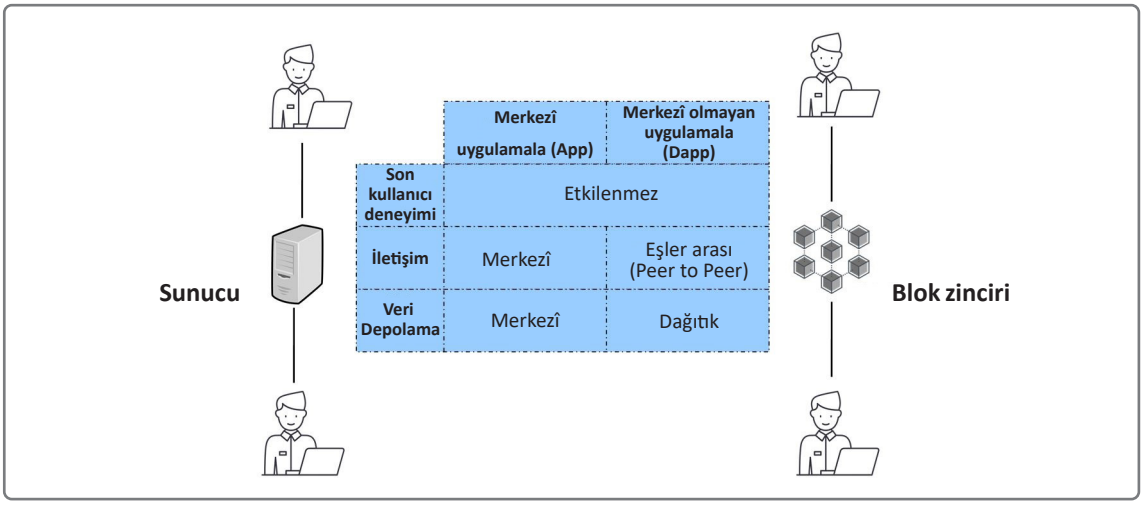
Oyun teorisi; stratejik düşünme, aktörleri etkileme, sıralı oyunlar ve zaman kavramından bahseder. Merkeziyetsiz sistemler tasarlanırken oyun teorisi modelleri dikkate alınmalıdır. Mahkûm İkilemi, oyun teorisinin bilinen örneklerinden biridir. Tutuklanan iki suçlu farklı odalarda sorgulanır ve mahkûmların birbirleriyle iletişim kurmalarına izin verilmez. Mahkûmlar, birbirleri aleyhine ifade vermeleri için ikna edilmeye çalışılır. Mahkûmların verecekleri cevaplara göre farklı sonuçlar gerçekleşir. Oyun teorisi modelleri kullanma fikrini örnekleyen farklı Mahkûm İkilemi çeşitleri bulunur ve bu örnekler, özünde insan davranışları ve rasyonel karar verme süreçlerinin sonuçlarını inceler.

Oyun teorisi ile kriptografinin birlikte kullanılması özellikle Ethereum ve Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin ve Etherum'un İş Kanıtı (Proof of Work) algoritmasını blok zincirini saldırılara karşı dayanıklı ve merkeziyetsiz bir sistem olmasına imkân sağlar.

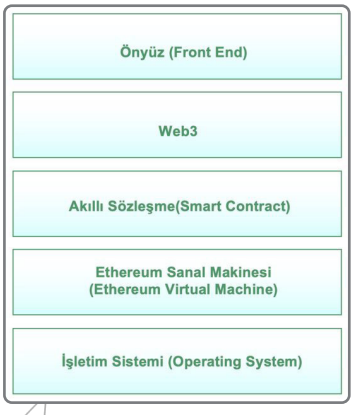
Ethereum madenciliği İş Kanıtı (Proof of Work) algoritmasını kullanır. Ethereum'un kullandığı İş Kanıtı algoritması ile Ethereum ve Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin kullandığı İş Kanıtı algoritması farklıdır. Ethereum'da çalışan İş Kanıtı, **Ethash** adında hafıza kullanılarak işlem yapar. Kullanılan hafıza ve işlemciyle düşük işlem maliyetleri ve verimlilik sağlanır.

5.3. MERKEZÎ OLMAYAN UYGULAMALAR (DApp)

DApp'lar, merkezî olarak oluşturulup kontrol edilmek yerine, merkezî olmayan teknoloji üzerine kurulmuş uygulamalardır. Örneğin telefon uygulamalarını indirmek için kullandığımız uygulama dükkanları içinde yer alan uygulamalar, tek bir şirket yani tarafından oluşturulup kontrol edilir. Ödeme taleplerinizi yürütmek ve işleme koymak için süreci ve koşulları tanımlayan tek yetkili şirkettir fakat DApp'lerdeki kontrol, ağ katılımcıları arasında paylaşılır. Görsel 5.3'te görüldüğü üzere DApp'larda kodlar, herkesin talimatları ile denetleyerek değiştirebileceği şekilde açık ve şeffaf tutulur.



Görsel 5.3: Merkezî ve merkezî olmayan uygulamaların karşılaştırılması



Görsel 5.4: Dağıtık uygulama mimarisi (https://kba.ai)

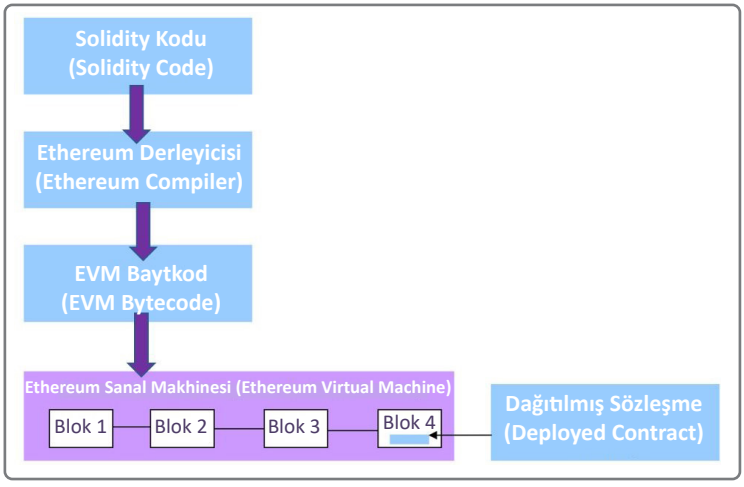
Görsel 5.4' te görüldüğü üzere dağıtık uygulama mimarisi beş farklı katmandan oluşur. Bunlar önyüz, web3, akıllı sözleşmeler, Ethereum sanal makinası ve işletim sistemidir.

Önyüz (Front End): Görsel 5.4'e göre ilk katman olan önyüz, blok zinciri teknolojisine özel bir okuryazarlık yetkinliği bulunmayan ve blok zinciri ile iletişim kurmak isteyen kullanıcıların blok zinciri ile sorunsuz bir şekilde iletişim kurmasını sağlar. Arayüz herhangi bir dilde geliştirilebilir. Kullanıcıyla etkileşim için metin kutuları ve düğmeler sunar.

Web3: İkinci katmanda bulunan Web3, kullanıcıların HTTP, IPC-WebSocket kullanarak Ethereum düğümleriyle iletişim kurmasını sağlar. Ayrıca Ethereum için belirli işlevleri içeren API listesi sunar.

Akıllı Sözleşme (Smart Contract): Üçüncü katman herhangi bir aracının yönetimine gereksinim duyulmadan; nakit, mülk, hisse veya değerli bir varlığın değişimini şeffaf ve çatışmasız bir şekilde yönetir. Akıllı sözleşmeler, Ethereum ağının omurgası olarak da ifade edilir.

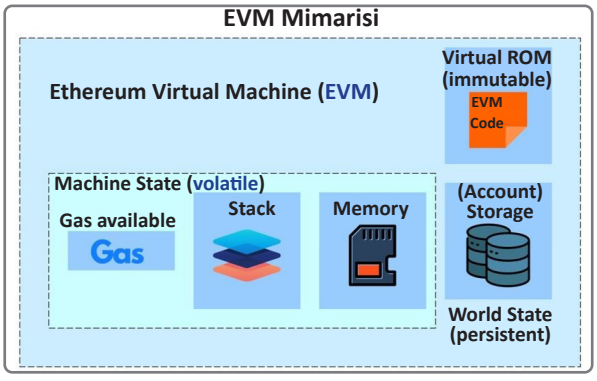
Ethereum Sanal Makinesi (EVM): Dördüncü katman Ethereum sanal makinesinde, tam bir düğümü çalıştırabilen Ethereum ağına giriş noktasıdır. Görsel 5.5'te gösterildiği üzere Ethereum sanal makinesi, çalıştırılan kod ile kodu çalıştıran makine arasında sanal bir katman oluşturur. Ethereum sanal makinesi, akıllı sözleşme kodunu yürütür.



Görsel 5.5: Ethereum sanal makinesi (https://kba.ai)

İşletim Sistemi (Operating System): Beşinci katman ise Ethereum sanal makinesinin kurulduğu işletim sistemini ifade eder.

5.4. ETHEREUM SANAL MAKİNESİ (EVM)



Görsel 5.6: Ethereum sanal makinesi [Virtual Machine (EVM)]

Görsel 5.6'da Ethereum sanal makinesi [Virtual Machine (EVM)] mimarisi gösterilmektedir. EVM Ethereum'daki akıllı sözleşmelerin çalıştırılabilmesine ortam sağlar. Akıllı sözleşmeler, EVM'de kendi kendini yürütür.

5.4.1. EVM'ye Neden İhtiyaç Duyulur?

Ethereum Dünya Bilgisayarı'nın varlığının devamı için üç özelliği yerine getirmesi gerekiyor. Ethereum Dünya Bilgisayarı'nda çalışan tüm programlar, tam olarak programlandığı gibi çalışmalıdır. Yani **Deterministik** olmalıdır. Ethereum Dünya Bilgisayarı'ndaki herhangi bir şey ya da her şey istendiğinde **sonlandırılabilir** olmalıdır. Ayrıca sistem, herhangi bir dış etkenden etkilenmemelidir. Yani **izole** olmalıdır.

EVM, akıllı sözleşmelerin (EVM Kodu) herhangi bir işletim sisteminde yürütülmesine olanak sağlar. Akıllı sözleşme ile işletim sistemi arasında orta katman görevi yapar. Ayrıca Ethereum'un kodu okumasında, işlem maliyetini hesaplamasında ve işlemleri yürütmesinde önemli bir rol oynar.

Ethereum sözleşmeleri, Ethereum Sanal Makinesi içinde yayındadır ve haricî bir hesaptan bir işlem tarafından tetiklendiğinde kodun belirli bir bölümünü yürütür. Görsel 5.6'da görüldüğü üzere EVM mimarisinde şu üç depolama alanı vardır:

1. Depolama (Storage): Sözleşme değişkenlerini depolayan bir alandır. Ayrıca sözleşme oluşturma süresi sırasında sözleşme değişkenleri için yer tahsis edilir.

2. Bellek (Memory): Bellek alanında geçici değişkenler tutulur. Belleğe sadece sözleşmenin yürütülmesi esnasında erişilebilir.

3. Yığın (Stack): Hesaplamalarda ara değerleri saklamak için kullanılır. Yığın, sözleşmenin yürütülmesinden sonra kullanılmaz.

Gas, Ethereum blok zincirinde bir işlemi başarıyla yürütmek için kullanılan, ödenmesi gereken ücret veya fiyatlandırma değeridir.

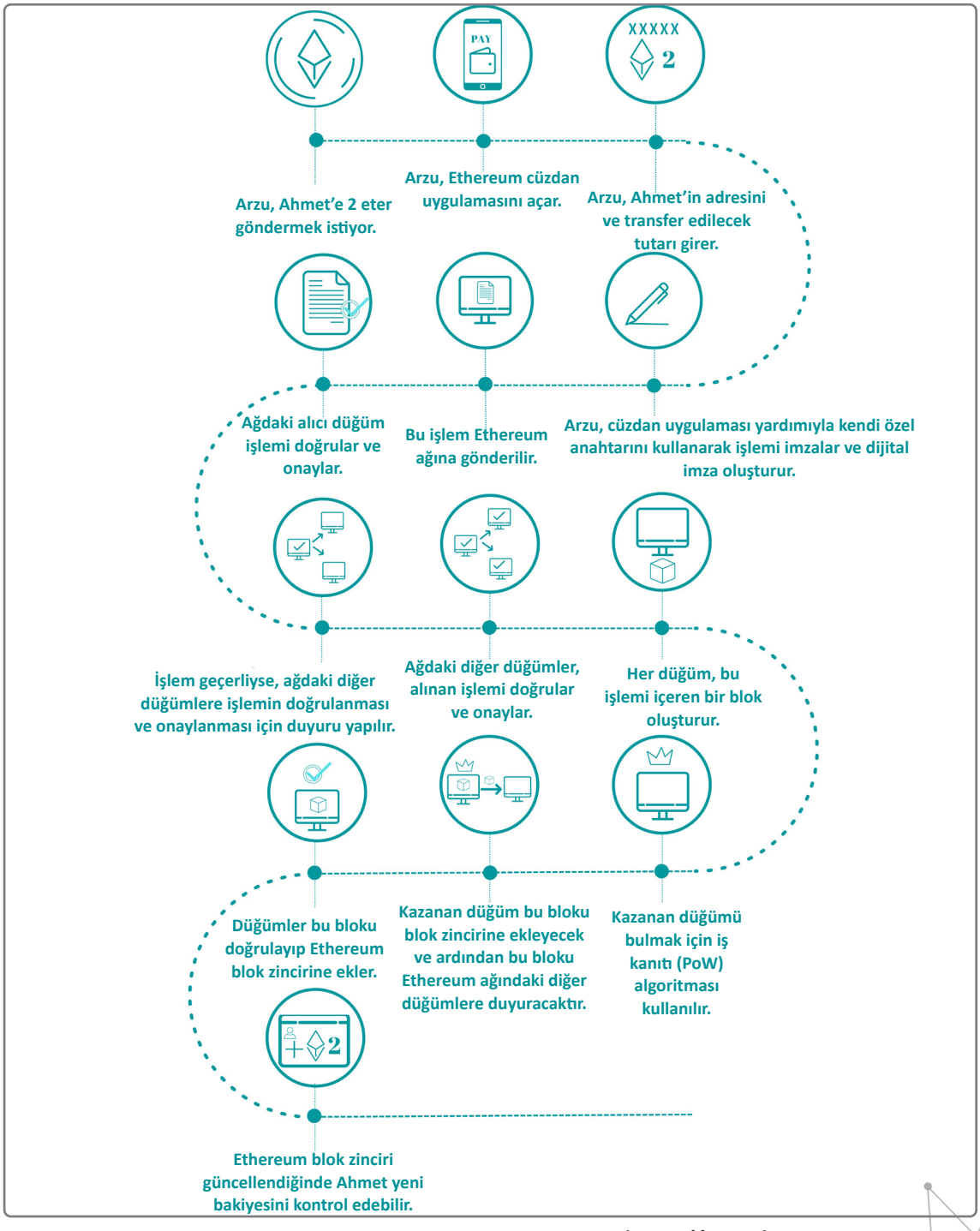
EVM; bilgisayarların Ethereum ağına dâhil olmasına, Ethereum ağına işlem gönderilmesine, akıllı sözleşmelerin yürütülmesine ve blok madenciliğine olanak sağlar.

Görsel 5.7'de Ethereum birimleri gösterilmektedir. Gas, Ethereum ağında işlemleri yürütmek için gereken hesaplama çabasının miktarını ölçmek için kullanılan birimdir. Gas, Eter'in en küçük birimi olan wei ile temsil edilir.

Birim (Unit)	Wei Değeri	Wei
wei	1 wei	1
Kwei (babbage)	1e^3 wei	1,000
Mwei (lovelace)	1e^6 wei	1,000,000
Gwei (shannon)	1e^9 wei	1,000,000,000
microether (szabo)	1e^12 wei	1,000,000,000,000
milliether (finney)	1e^15 wei	1,000,000,000,000,000
ether	1e^18 wei	1,000,000,000,000,000,000

Görsel 5.7: Ethereum birimleri

Gas ücretleri Ethereum ağını güvende tutmaya yardımcı olur. Ağdaki işlemleri yürütmek için ücret belirlenmesi, kötü niyetli aktörlerin Ethereum ağına spam göndermesini önler. Sonsuz döngülerin oluşmasını önlemeye yardımcı olur. Ayrıca geliştiricileri de EVM'de çalıştırmak için optimize edilmiş kod oluşturmaya teşvik eder. Kullanılmayan gas, kullanıcıya iade edilir. Görsel 5.8'de Ethereum işlem yaşam döngüsü gösterilmektedir.



Görsel 5.8: Ethereum işlem yaşam döngüsü (https://kba.ai)

SIRA SİZDE

Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde işlem yaşam döngüsünü araştırıp, çiziniz. Sınıfta herkesin görebileceği uygun bir yer belirleyerek çizimlerinizi sergileyiniz.

SIRA SİZDE

Ethereum ve Eşten-eşe Elektronik Nakit Ödeme Sistemi'nin işlem yaşam döngülerini sınıfta 2'şerli gruplar halinde dramatize ediniz.

5.4.2. Ether

Ether (ETH), Ethereum'un yerel para birimidir. Eter bir metrik birimdir ve farklı çeşitleri vardır. Ether'in en küçük çeşidi veya temel birim Wei olarak bilinir.

5.4.3. Hesaplar (Accounts)

Bir Ethereum hesabı ether bakiyesi tutabilir veya Ethereum blok zinciri ağındaki başka bir hesaba ether bakiyesi gönderebilir. Hesaplar, kullanıcı tarafından veya kodlarla (sözleşmeler) kontrol edilebilir.

5.4.4. Blok (Block)

Ethereum ağındaki işlemler, tek tek işletilmek yerine toplu olarak işletilir. İşlemler bloklarda tutulur. Bir blokta yüzlerce işlem bulunur. İşlem geçmişini korumak için işlemler bloklarda saklanır. Her yeni blok, önceki bloka ilişkin bir referans taşır. Referanslar, blokun verilerinden kriptografik olarak türetilen hashlerdir. Blok verilerindeki herhangi bir değişiklik hash değerini değiştirir. Bu durum da herhangi bir dolandırıcılığı önlemek için önemlidir.

5.4.5. İşlem Ücreti (Transaction Fee)

Blok zinciri ağında gerçekleştirilen her okuma veya yazma işlemi için ether olarak ödenen ve Gas maliyeti olarak bilinen bir ücreti vardır.

*Transaction Fee (paid in ether) = Gas Limit * Gas Price*

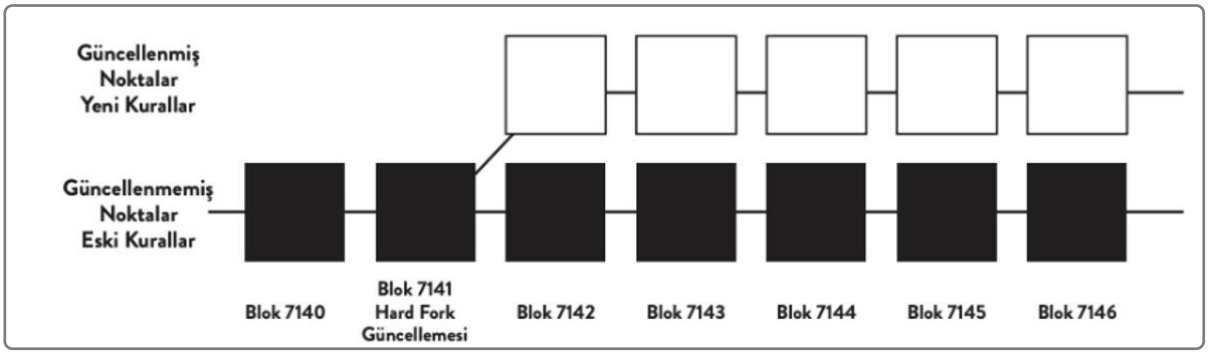
5.5. ETHEREUM ÇATALLANMA

Çatallanma, kullanıcıların ya da geliştiricilerin platformda değişiklik yapmak istemeleri durumunda gerçekleşebilir. Ethereum'da gerçekleşen büyük bir siber saldırı sonrasında nedeniyle çatallanma gerçekleşmiştir.

Blok zincirindeki gerçekleşen her sapma çatal olarak kabul edilir. İki ana çatal çeşidi bulunmaktadır. Bunlar **"Hard fork"** ve **"Soft fork"** olarak ifade edilir.

5.5.1. Sert Çatallama (Hard Fork)

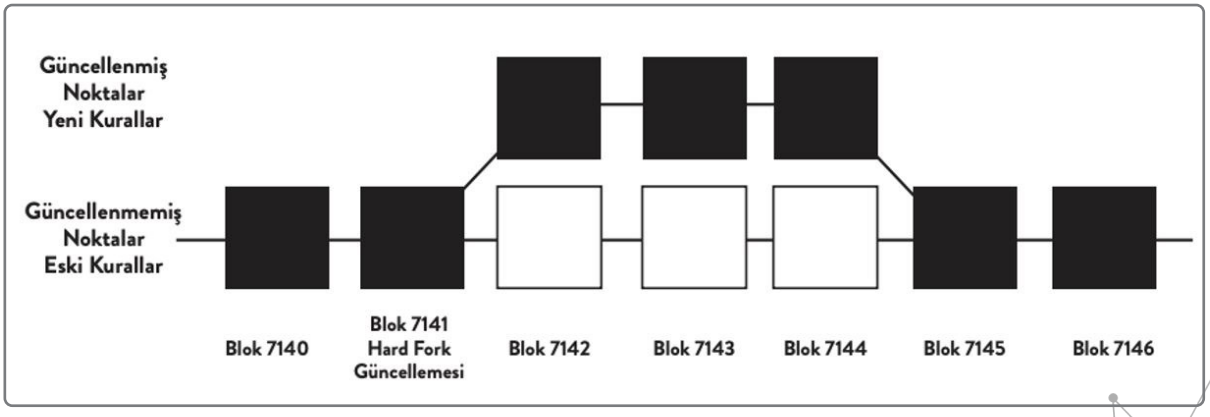
Görsel 5.9'da gösterildiği üzere hard fork geçmişe yönelik uyumluluğu bozan değişikliklerdir. Hard fork öncesi işlemleri çalıştıran cihazlarda yeni işlemler geçersiz olur.



Görsel 5.9: Hard fork (Sert Çatallama)

5.5.2. Yumuşak Çatallama (Soft Fork)

Görsel 5.10'da gösterildiği üzere soft fork geçmişe yönelik uyumlu gerçekleşen değişikliklerdir. Soft fork gerçekleştiğinde, ağa bağlı cihazlarda yeni işlemler geçerli olur.



Görsel 5.10: Soft Fork (Yumuşak Çatallama)

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

1. Ethereum ağında işlemleri yürütmek için gereken hesaplama çabasının miktarını ölçmek için kullanılan birim aşağıdakilerden hangisidir?

A) Icon B) Coin C) Dot
D) Gas E) Satoshi

2. Aşağıdaki seçeneklerden hangisi dağıtık uygulama mimarisinde bulunan bileşenlerden biri değildir?

A) Önyüz (Front End)
B) Web2
C) Akıllı Sözleşme (Smart Contract)
D) Ethereum Sanal Makinesi (EVM)
E) İşletim Sistemi (Operating System)

3. Aşağıdaki isimlerden hangisi Ethereum'u oluşturan kişidir?

A) Nick Szabo
B) Satoshi Nakamoto
C) Vitalik Buterin
D) Hal Finney
E) Satoshi Hal

4. Ether'in en küçük temel birimi nedir?

A) Avax B) Coin C) Satoshi
D) Solana E) Wei

5. Aşağıdakilerden hangisi Ethereum'un temel ilkelerinden biri değildir?

A) Basitlik
B) Çeviklik
C) Evrensellik
D) Modülerlik
E) Programlanamazlık

KONULAR

6.1. BLOK ZİNCİRİ OLUŞTURULMASI

6.2. KRİPTO PARA OLUŞTURMA

NELER ÖĞRENECEKSİNİZ?

- Blok zinciri oluşturma
- Kripto para oluşturma

ANAHTAR KELİMELER

Ethereum, Ganache, Kripto Para, Token

HAZIRLIK ÇALIŞMALARI

1. Blok zincirinde kayıt oluşturulma sürecinin nasıl işlediğini açıklayınız.

2. Blok zincirinde gerçekleşen işlemler nasıl izlenebilir? Düşüncelerinizi arkadaşlarınızla paylaşınız.

BLOK ZİNCİRİ YAZILIM GELİŞTİRME

6. ÖĞRENME BİRİMİ

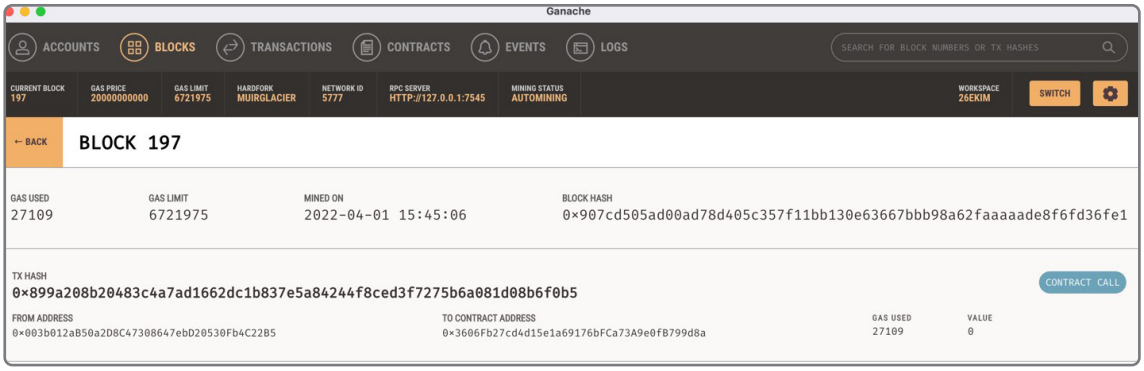
6.1. BLOK ZİNCİRİ OLUŞTURMA

Blok zinciri, bloklar ve bu blokları oluşturan kayıtlardan meydana gelir.

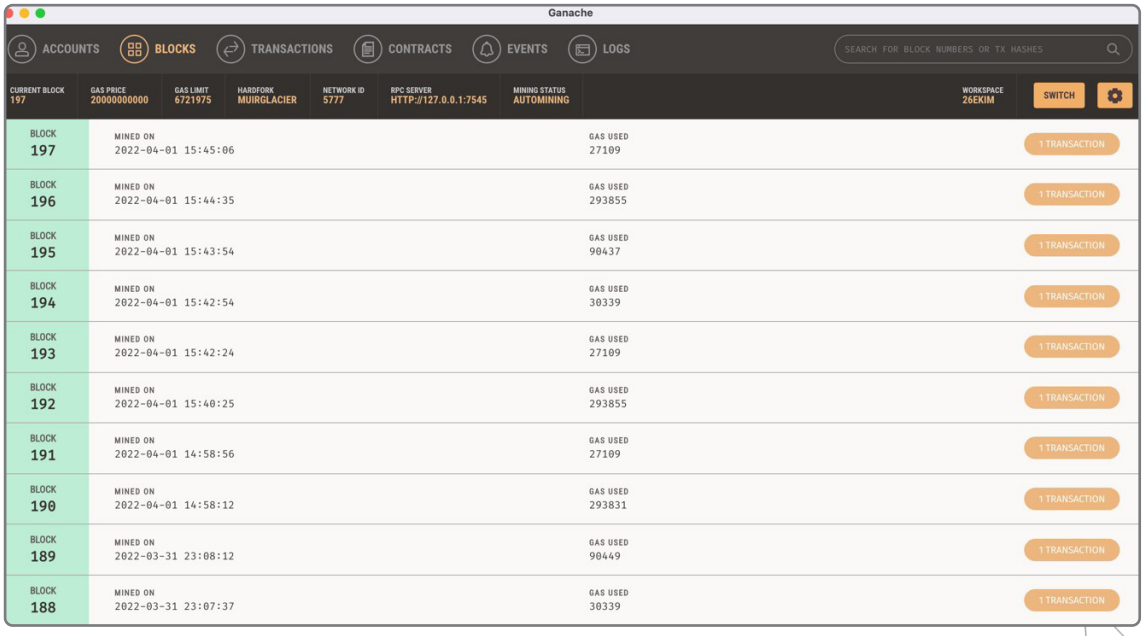
Kayıtlar: Blok zincirinde saklanmak istenen içerikler yer alır. İçeriklerde para aktarımı, görsel, ses veya video gibi değerler saklanabilir. Para transferleri için de kayıtlar tutulur. Sisteme kayıtlı olan kullanıcılar arasında gerçekleşen transferleri bu kayıtlar ile tutulur.

Bloklar: Görsel 6.1'de görüldüğü üzere kayıtlar birleştirilerek blokların içine yazılır. Blokun oluşturulmasında kriptografik özet algoritmaları ile dijital imza kullanılır.

Görsel 6.2'de görüldüğü üzere akıllı sözleşme çalıştırılması ile oluşan blokların **Ganache** ile izlenmesi yapılabilir. Ethereum için yerel bir blok zinciri geliştirme uygulaması olan Ganache ile bloklarda bulunan verileri, işlemleri ve blok zinciri üzerinde yer alan sözleşmeleri görsel olarak izlemek mümkündür. Blok zincirine eklenen yeni verileri izleyebilmek oldukça önemlidir.



Görsel 6.1: Ganache ile blok içeriğinin izlenmesi



Görsel 6.2: Ganache ile blokların izlenmesi

6.2. KRİPTO PARA OLUŞTURMA

Akıllı sözleşmeyi oluşturan kişi tarafından Ethereum ile kripto para üretilebilir. Kullanıcılar, Ethereum anahtarlarını kullanarak, herhangi bir aracıya gereksinim duymadan birbirine para gönderimi yapabilir, ödeme de alabilir.

ERC-20

Kendi bağımsız blok zinciri veya ağları üzerine oluşturulan tokenlar (jeton), birer kripto para birimidir. Ethereum blok zinciri üzerinde geliştirilen tokenlar, ERC-20 standardını kullanır.

1. UYGULAMA

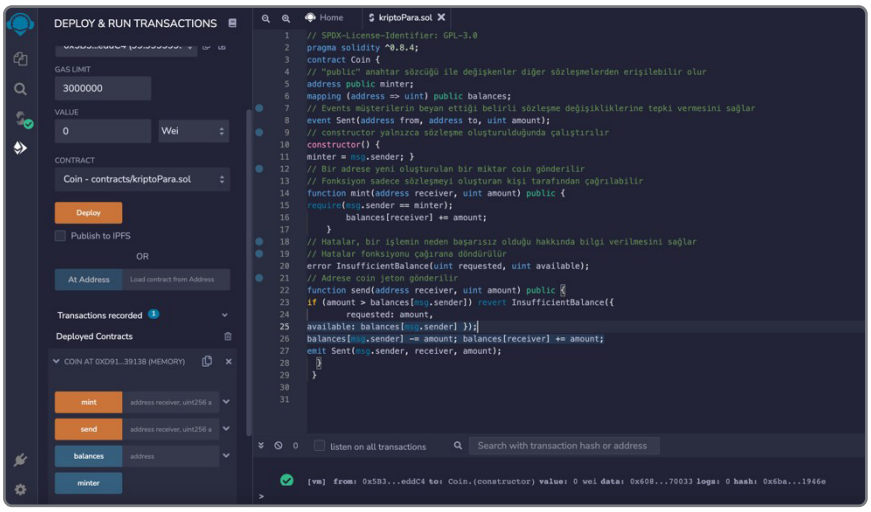
Kripto para oluşturma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Aşağıda gösterilen kripto para oluşturma uygulamasını kodlayınız (Görsel 6.3).

```
// SPDX-License-Identifier: GPL-3.0
pragma solidity ^0.8.4;
contract Coin {
// "public" anahtar sözcüğü ile değişkenler diğer sözleşmelerden erişilebilir olur
address public minter;
mapping (address => uint) public balances;
// Events müşterilerin beyan ettiği belirli sözleşme değişikliklerine tepki vermesini sağlar
event Sent(address from, address to, uint amount);
// constructor yalnızca sözleşme oluşturulduğunda çalıştırılır
constructor() {
minter = msg.sender; }
// Bir adrese yeni oluşturulan bir miktar coin gönderilir
// Fonksiyon sadece sözleşmeyi oluşturan kişi tarafından çağrılabilir
function mint(address receiver, uint amount) public {
require(msg.sender == minter);
        balances[receiver] += amount;
    }
// Hatalar, bir işlemin neden başarısız olduğu hakkında bilgi verilmesini sağlar
// Hatalar fonksiyonu çağırana döndürülür
error InsufficientBalance(uint requested, uint available);
// Adrese coin jeton gönderilir
function send(address receiver, uint amount) public {
if (amount > balances[msg.sender]) revert InsufficientBalance({
        requested: amount,
available: balances[msg.sender] });
balances[msg.sender] -= amount; balances[receiver] += amount;
emit Sent(msg.sender, receiver, amount);
  }
}
```

Görsel 6.3: Kripto para oluşturma kodları

2. Adım: Akıllı sözleşmeyi **Compile** ve **Deploy** ederek Görsel 6.4'teki ekran görüntüsünü elde ediniz.



Görsel 6.4: Akıllı sözleşme kripto para oluşturma

2. UYGULAMA

Akıllı sözleşmeyi kodlayarak ERC tokenı oluşturma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

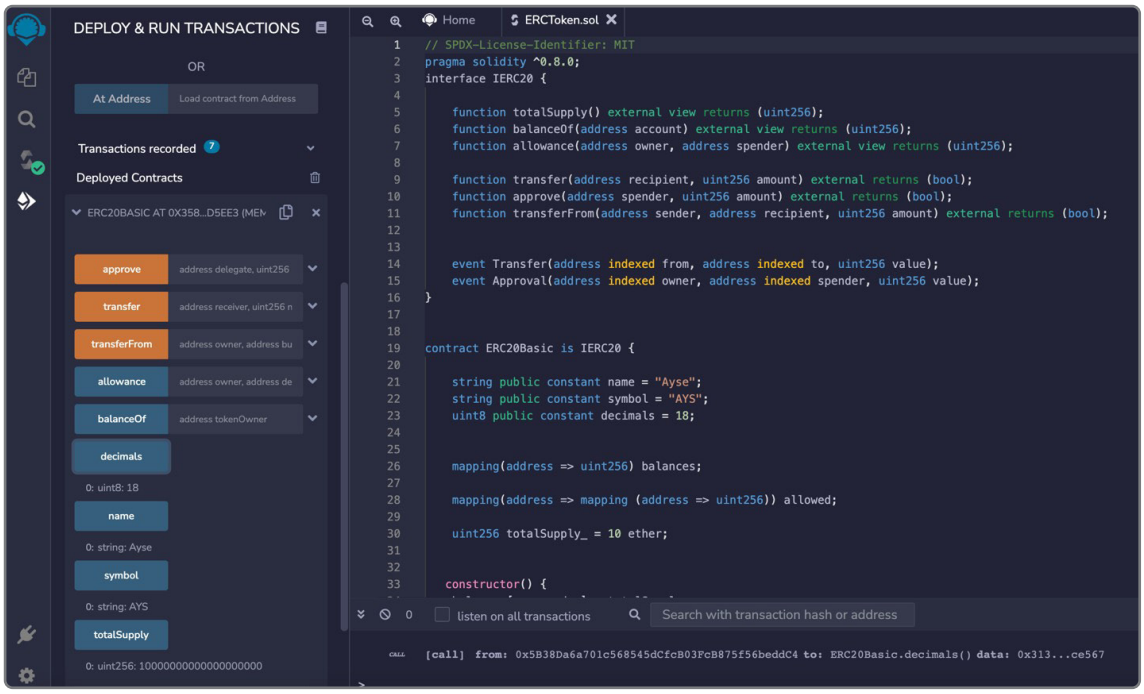
1. Adım: Ayse adında ERC tokenı oluşturmak için akıllı sözleşmeyi kodlayınız (Görsel 6.5).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
interface IERC20 {

    function totalSupply() external view returns (uint256);
    function balanceOf(address account) external view returns (uint256);
    function allowance(address owner, address spender) external view returns (uint256);

    function transfer(address recipient, uint256 amount) external returns (bool);
    function approve(address spender, uint256 amount) external returns (bool);
    function transferFrom(address sender, address recipient, uint256 amount) external returns (bool);


    event Transfer(address indexed from, address indexed to, uint256 value);
    event Approval(address indexed owner, address indexed spender, uint256 value);
}


contract ERC20Basic is IERC20 {

    string public constant name = "Ayse";
    string public constant symbol = "AYS";
    uint8 public constant decimals = 18;
```

```
  mapping(address => uint256) balances;

  mapping(address => mapping (address => uint256)) allowed;

  uint256 totalSupply_ = 10 ether;

 constructor() {
  balances[msg.sender] = totalSupply_;
  }

  function totalSupply() public override view returns (uint256) {
  return totalSupply_;
  }

  function balanceOf(address tokenOwner) public override view returns (uint256) {
      return balances[tokenOwner];
  }

  function transfer(address receiver, uint256 numTokens) public override returns (bool) {
      require(numTokens <= balances[msg.sender]);
      balances[msg.sender] = balances[msg.sender]-numTokens;
      balances[receiver] = balances[receiver]+numTokens;
      emit Transfer(msg.sender, receiver, numTokens);
      return true;
  }

  function approve(address delegate, uint256 numTokens) public override returns (bool) {
      allowed[msg.sender][delegate] = numTokens;
      emit Approval(msg.sender, delegate, numTokens);
      return true;
  }

  function allowance(address owner, address delegate) public override view returns (uint) {
      return allowed[owner][delegate];
  }

  function transferFrom(address owner, address buyer, uint256 numTokens) public override returns (bool) {
      require(numTokens <= balances[owner]);
      require(numTokens <= allowed[owner][msg.sender]);

      balances[owner] = balances[owner]-numTokens;
      allowed[owner][msg.sender] = allowed[owner][msg.sender]-numTokens;
      balances[buyer] = balances[buyer]+numTokens;
      emit Transfer(owner, buyer, numTokens);
      return true;
  }
}
```

Görsel 6.5: ERC-20 token akıllı sözleşme kodları

2. Adım: Oluşturduğunuz Ayse adındaki ERC tokenı deploy ediniz (Görsel 6.6)



Görsel 6.6: ERC-20 token

SIRA SİZDE

Tokenların kullanım alanları konulu bir sunu hazırlayınız. Gönüllü arkadaşlarınızın paylaştığı sunumları izleyerek aralarındaki ortak noktaları belirleyiniz.

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

1. I. Veri tipini belirleyen standart

II. Hafıza tipini belirleyen standart

III. Blok zincirinde kullanılan sabit değer

IV. Token'lar için Ethereum üzerinde geliştirilen standart

V. Token'lar için Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde geliştirilen standart

Yukarıda ERC-20 ile ilgili verilen bilgilerden hangisi ya da hangileri doğrudur?

A) Yalnız IV

B) Yalnız V

C) III-IV

D) III-V

E) I-II-III

2. Blok zinciri ile ilgili aşağıdaki ifadelerden hangisi yanlıştır?

A) Blok zinciri, dağıtık bir veri tabanı olarak ifade edilebilir.

B) Blok zinciri, blok ve kayıtlardan meydana gelir.

C) Eşten-eşe Elektronik Nakit Ödeme Sistemi'nde akıllı sözleşme oluşturulabilir.

D) Ethereum blok zincirinde akıllı sözleşme oluşturulabilir.

E) Akıllı sözleşmeler, yazılan kodla birlikte otomatik olarak çalışır.

3. Ganache ile ilgili aşağıda verilen bilgilerden hangisi doğrudur?

A) Akıllı sözleşme çalıştırılması ile oluşan blokların izlenmesi için kullanılır.

B) Özel ve açık kaynaklı bir blok zinciri platformu çeşididir.

C) Akıllı sözleşmeyle oluşturulan özel kripto para birimidir.

D) Akıllı sözleşmelerin hesaplama işlemlerinde kullanılan bir fonksiyondur.

E) Akıllı sözleşmelerde kullanılan değişken özellikli bir veri tipidir.

KONULAR

7.1. AKILLI KONTRAT OLUŞTURMA
7.2. SOLIDITY KULLANIMI
7.3. TEST MİMARİSİ
7.4. REMIX VE MOCHA İLE TEST YAPMA
7.5. INFURA KURMA
7.6. ETHERSCAN'DE DEPLOY GÖZLEMİ

NELER ÖĞRENECEKSİNİZ?

- Akıllı kontrat oluşturma
- Ethereum ağları ile etkileşimli akıllı kontrat oluşturma
- Solidity kullanma
- Test mimarisini açıklama

ANAHTAR KELİMELER

Akıllı kontrat, Ethereum, Remix IDE, Solidity

HAZIRLIK ÇALIŞMALARI

1. Size göre akıllı sözleşme ile geleneksel bilgisayar programı arasındaki temel fark ne olabilir?

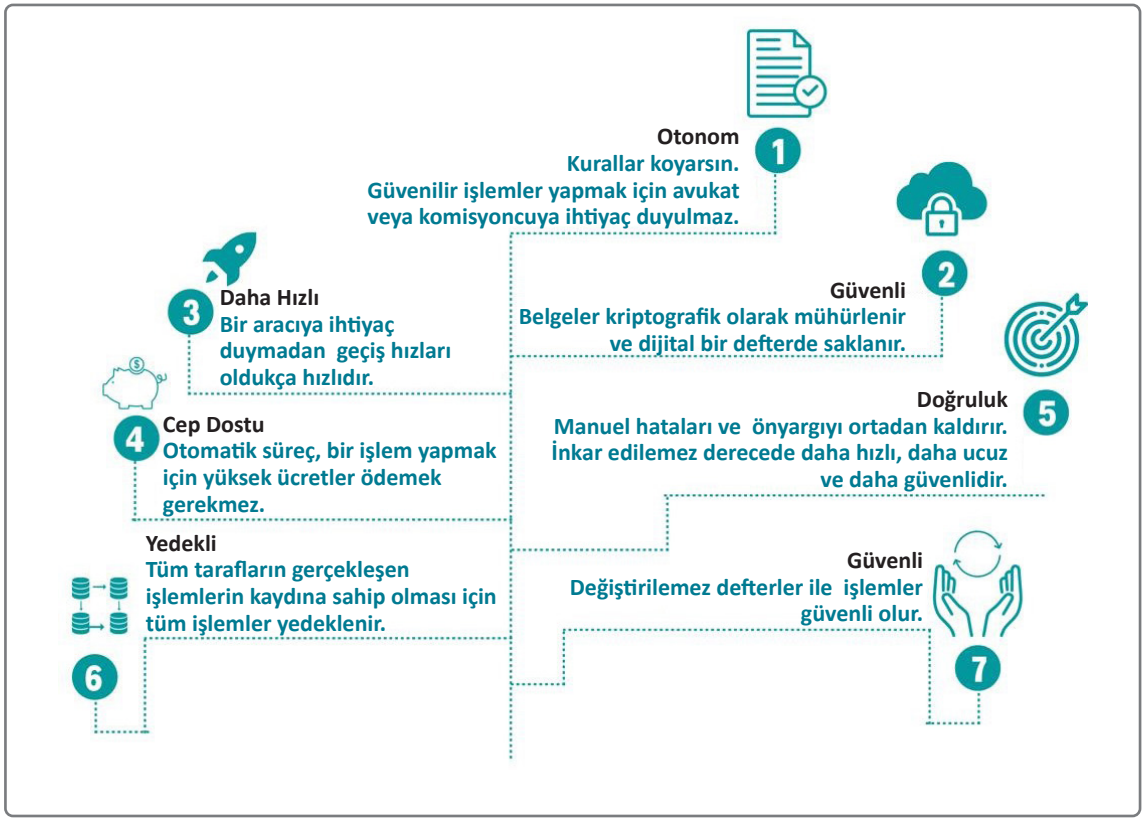
2. Ethereum ağında akıllı kontrat (sözleşme) oluşturmak için hangi programlama dilleri kullanılabilir düşüncelerinizi arkadaşlarınızla paylaşınız.

AKILLI KONTRATLAR

7.
ÖĞRENME BİRİMİ

7.1. AKILLI KONTRAT OLUŞTURMA

Bilgisayar programı belirli görev için bir dizi talimatı yerine yetirir. Akıllı kontratlar (sözleşmeler) ise iş sürecinin sorunsuz ve başarı ile yürütülmesi için gerekli yasal yükümlülükleri, kullanıcı rollerini, erişim kısıtlarını ve gerekli protokolleri kapsar. Görsel 7.1'de görüldüğü üzere akıllı sözleşmenin avantajları arasında geliştirilmiş güvenlik, düşük maliyet, anonim varlıklar arasında daha iyi güven ve aracıya ihtiyaç duyulmaması yer alır.



Görsel 7.1: Akıllı sözleşmelerin özellikleri (https://kba.ai)

7.2. SOLIDITY KULLANIMI

Ethereum ve akıllı kontrat; Solidity, C++, Python ve JavaScript dillerinden etkilenerek geliştirilmiştir. Ethereum ve farklı blok zinciri platformlarında akıllı sözleşme geliştirmek için kullanılır ve Ethereum Sanal Makinesi (EVM) üzerinde çalışır. Görsel 7.2'de görüldüğü üzere Solidity dilinde akıllı sözleşmesi yazılırken kaynak dosya için SPDX-License-Identifier etiketi kullanır. Etiket her programın başında belirtilir. Genel sözdizimi **// SPDX-License-Identifier: lisansın adı** şeklindedir. Ayrıca Compiler (derleyici) versiyonunun belirtilmesi de gerekir. Tanımlamada belirtilen versiyon, sözleşmenin uyumsuz bir

Solidity derleyicisinde derlenmemesini sağlar. **pragma** Solidity'nin derleyici sürümünü gösterir. Genel sözdizimi **pragma solidity version** şeklindedir. Sözleşmeyi adlandırırken, sözleşmenin amacını açıklayan bir ad vermek kodun okunabilirliği açısından önemlidir. Genel sözdizimi **contract [name] { }** şeklindedir. Fonksiyonun, sözleşmeyle etkileşime girecek herkes için erişebilir olması **public** anahtar kelimesiyle sağlanır.

```
// SPDX-License-Identifier: MIT
// Derleyici versiyonu 0.8.13'e eşit veya büyük ve 0.9.0'den küçük olmalıdır.
pragma solidity ^0.8.13;

contract MerhabaDunya {
   string public mesaj = "Merhaba Dünya!";
}
```

Görsel 7.2: Örnek Solidity kodu

7.3. TEST MİMARİSİ

Oluşturulan blok zinciri ağının tasarlandığı şekilde çalışıp çalışmadığı test edilerek varsa hata ve eksiklikleri tespit edildikten sonra yayınlanmalıdır. Yapılan test ile blok zinciri ağının gecikme, birim zamanda üretilebilen iş, kaynak kullanımı ve başarısız/gecikmiş işlem değerleri ölçülebilir. Test edilecek blok zinciri ağı ile iletişim kuracak en az bir adet test düğümü olmalıdır. Test düğümü, test işlemlerinin kapsamını içeren bir konfigürasyon dosyasını blok zinciri ağında çalıştırır. Konfigürasyon dosyasında uygulamaya ait akıllı sözleşme yüklenebileceği gibi varsayılan herhangi bir akıllı sözleşme de kullanılabilir. Test düğümü, belirlenen senaryoya göre sistemde çalışacak ve sonuçları gözlemlenecektir.

7.4. REMİX VE MOCHA İLE TEST ETME

Ethereum platformunda çalışmak üzere tasarlanan akıllı sözleşmeleri denemek amacıyla farklı test platformları kullanılır.

7.4.1. Truffle İle Test

Truffle, Ethereum'da uygulama geliştirmek için kullanılan bir framework'tür. Geliştiricilere otomatik test imkânı sunar. Bu durum da geliştiricilerin hızlı geliştirme yapabilmesini sağlar. Akıllı sözleşmeler derlendikten sonra test edilir. Bir testi çalıştırmak için **Truffle test** komutu yazılır. Truffle, testleri yalnızca -js, .ts, .es, .es6, .jsx ve. sol. dosya türleriyle çalıştırır.

7.4.2. Mocha İle Test

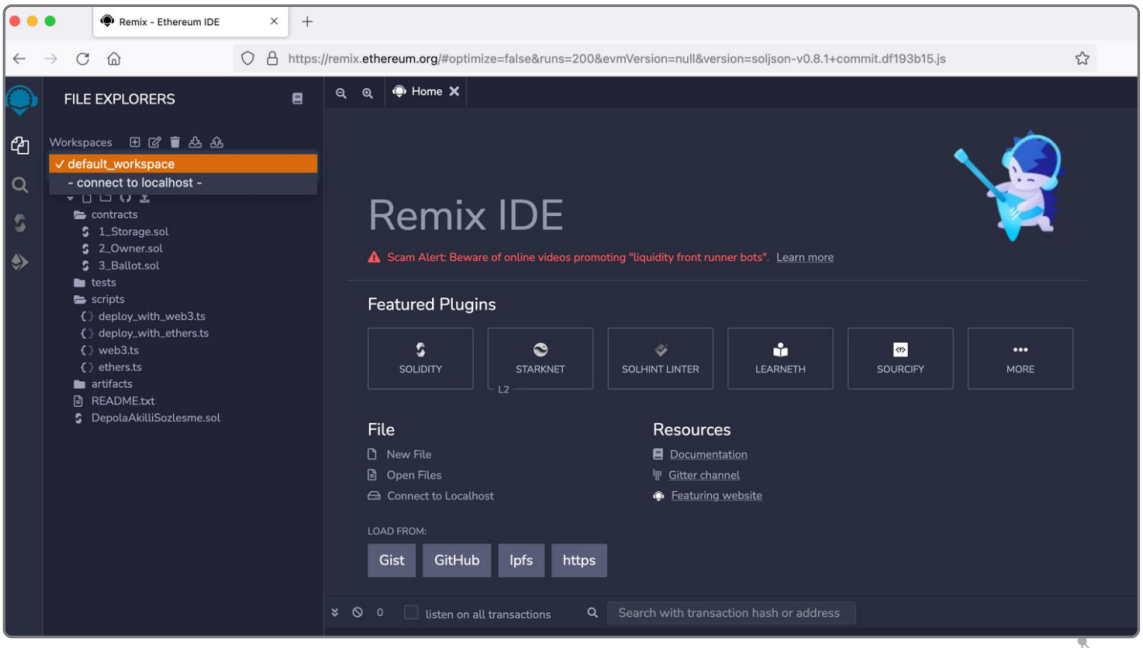
Mocha, node.js ve browser üzerinde çalışan bir test frameworküdür. Mocha onaylama kütüphanesi için Ganache ve Web3 yapıcısı gerekir.

7.4.3. Remix IDE İle Test

Ethereum platformu basit sözleşmeleri tasarlamak, yaratmak için kullanılmasının yanı sıra akıllı sözleşmeyi test etmek amacıyla da kullanılır.

7.4.3.1. Remix IDE Dosya Gezgini ve Kod Düzenleyicisi

Remix IDE [Integrated Development Environment (Tümleşik Geliştirme)], tarayıcı içinde yer alan açık kaynaklı bir IDE'dir. Tarayıcı (Browser – Solidity), solidity dili kullanılarak akıllı sözleşme yazmak ve bunları blok zinciri ağı için derlemeye ve blok zinciri ağına dağıtmak için kullanılır. JavaScript ile yazılan Remix, tarayıcıda ya da yerel (local) olarak kullanılabilir. Ayrıca Remix kodlarını test etme, hataları ayıklamayı ve çok daha fazlasını destekler. Remix IDE (web uygulaması) ve yerel bilgisayar arasında websocket bağlantısına izin verir. Remixd ise Remix IDE ile birlikte kullanılması amaçlanan bir araçtır. Remixd ile NPM paketini kullanarak dosya sistemine bağlantı kurmak mümkündür (Remix IDE Dosya Gezgini için https://remix.ethereum.org/ adresi ziyaret edilebilir.). Görsel 7.3'te görülen çalışma alanları (Workspaces), Remix IDE'deki dosyaları gruplamak için kullanılır.



Görsel 7.3: Çalışma alanları (Workspaces)

Çalışma alanı aşağıdaki seçeneklere sahiptir.

Create: Yeni bir çalışma alanı oluşturmak için kullanılır.

Rename: Mevcut bir çalışma alanını yeniden adlandırmak için kullanılır.

Delete: Mevcut çalışma alanını silmek için kullanılır.

Görsel 7.4'te görüldüğü üze Remix yazılım geliştiriciler için bazı örnek kodlar sağlamıştır. Bunlar farklı bir klasörlerde bulunur.

Contracts (Sözleşmeler)

Sözleşmeler klasöründe aşağıdaki üç farklı örnek bulunur.

Storage.sol

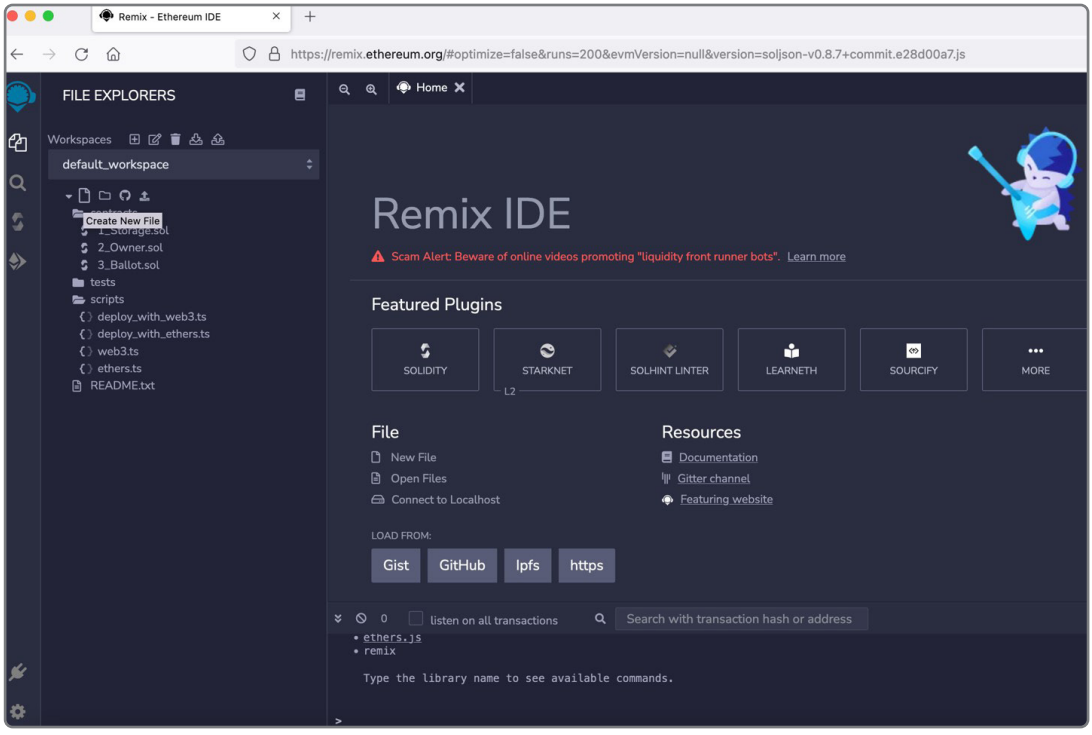
Basit olarak bir sayının saklanmasını ve okunmasını sağlar.

Owner.sol

Erişim kısıtlaması oluşturulmasını sağlar.

Ballot.sol

Akıllı sözleşme için teklif onaylamasıdır.



Görsel 7.4: Remix akıllı sözleşme kod örnekleri

Remix IDE Deploy & Run Transaction

Bu sekme aşağıdaki seçenekleri sunar.

Environment (Ortam)

Sözleşmenin uygulanacağı ortam şunlardan biri olabilir.

JavaScript VM

Remix IDE içinde test amacıyla Remix geliştiricileri tarafından Ethereum düğümünün bir simülasyonudur.

Injected Web3

Kullanıcının cüzdan uygulamalarıyla bağlantı kurmasını sağlar.

Web3 Provider (Sağlayıcı)

Bir bilgisayarda çalışan bir Ethereum düğümüne (Geth, Parity vb.) bağlanılmasını sağlar. Web3 Sağlayıcı, Ganache gibi Ethereum düğüm simülasyon araçlarını bağlamak için kullanılır.

Account (Hesap)

Bağlı ortamdaki kilitli olmayan hesapların listesini gösterir.

Gas Limit

Remix IDE'de her işlemin çalışması için belirlenen limittir.

Value (Değer)

Sözleşmeye Ether aktarmak istenildiğinde Eter değeri burada verilebilir.

NOT

Dosya Gezgini'nde listelenen tüm dosyalar, tarayıcının önbelleğinde saklanır. Dosyalar yerel olarak depolanmak isteniyorsa ana sayfada bulunan tüm dosyaları **yedek zip olarak indir** seçeneği kullanılabilir. Eğer ana sayfayı kapatılırsa sayfanın sol üst köşesinde bulunan **Remix IDE** simgesine tıklanarak sayfa yeniden başlatılabilir.

1. UYGULAMA

Blok zinciri ağında, tamsayı olan bir değeri yazmak ve okumak için solidity dilini kullanarak oluşturulan Görsel 7.5'teki ilk akıllı sözleşmeyi yazma ve test etme işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

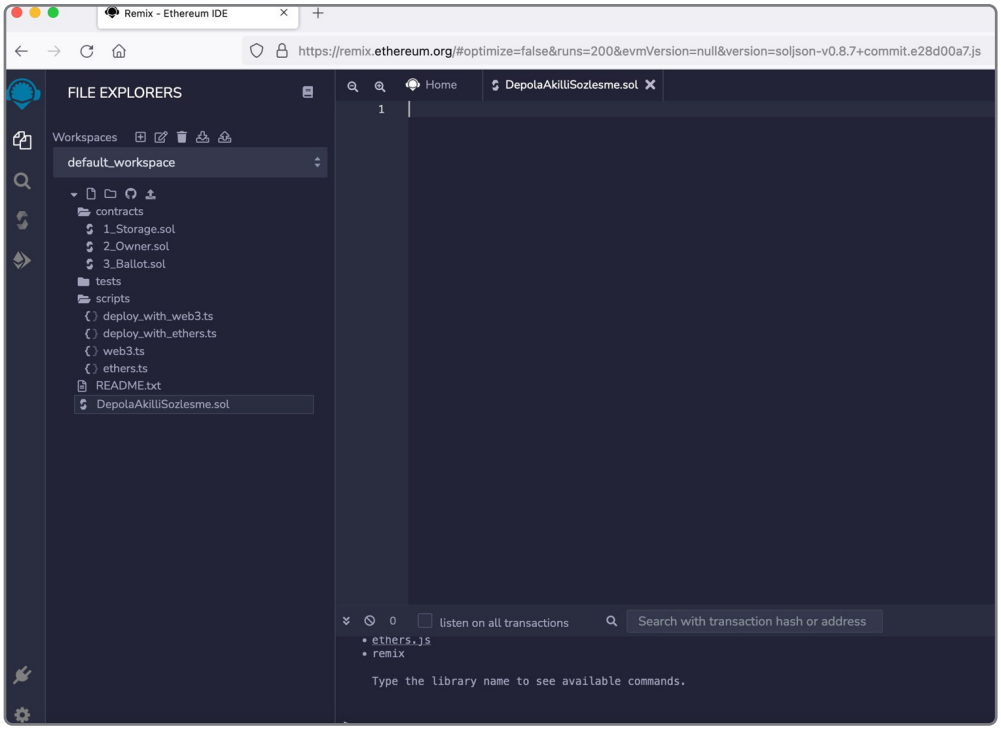
```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.1;

contract degerSaklama {

  uint256 depolananDeger;

  function store(uint256 sayi) public {
    depolananDeger = sayi;
  }

  function retreive() public view returns (uint256){
    return depolananDeger;
  }
}
```

Görsel 7.5: Akıllı sözleşme kodları

1. Adım: **https://remix.ethereum.org/** adresine gidiniz.

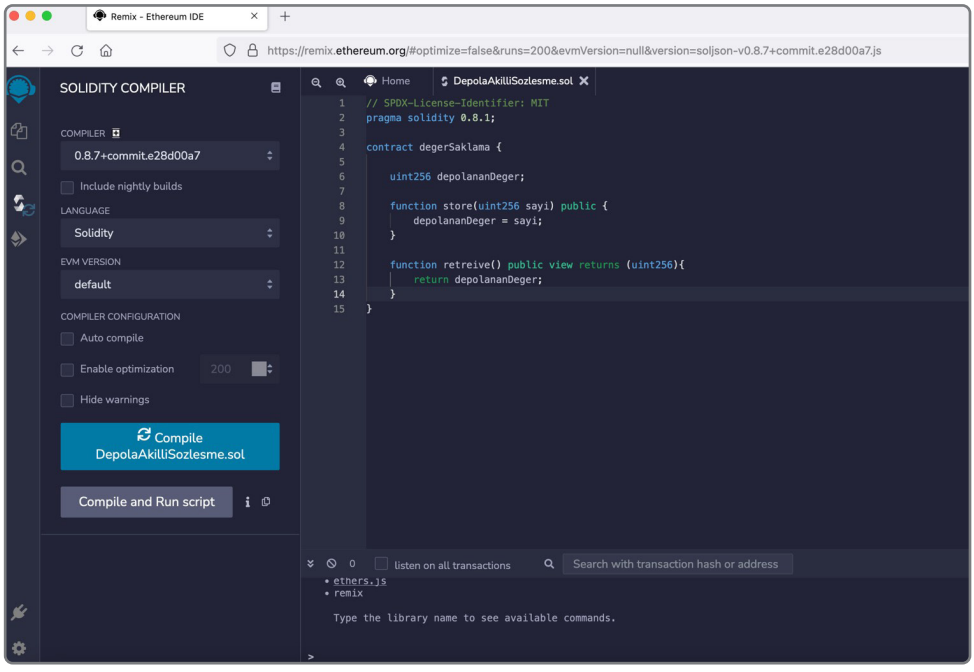
2. Adım: **Yeni dosya oluştur simgesine** tıklayarak yeni bir dosya oluşturunuz.

3. Adım: Görsel 7.6'da görüldüğü üzere dosya adını **DepolaAkilliSozlesme** olarak yazınız.



Görsel 7.6: REMIX dosya oluşturma

4. Adım: Aşağıdaki basamakları izleyerek dosyanıza Görsel 7.7' deki **Solidity** kodunu yazınız.



Görsel 7.7: Solidity kodu

a) Yorum satırı olarak **// SPDX-License-Identifier: GPL-3.0** yazınız.

b) **pragma solidity ^0.8.1** ile Solidity sürümünü yazınız.

c) **contract degerSaklama** adındaki kontratı tanımlayınız.

ç) Depolanacak sayısal değer için **uint** türünden (256 bit uzunluğunda işaretsiz tam sayı) **unit256 depolananDeger** tanımlamasını yapınız.

d) Aşağıdaki **depola** adındaki fonksiyonu yazınız.

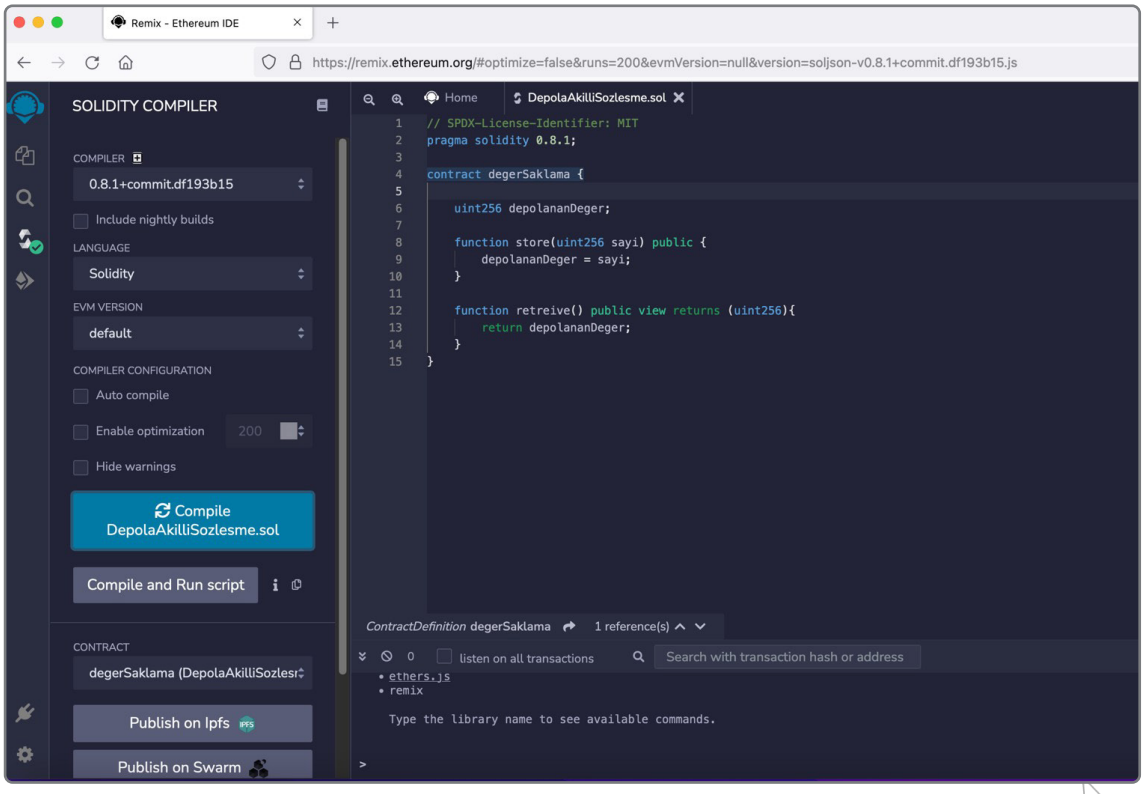
```
function depola(uint256 depolananDeger) public {
depolananDeger = sayi;
}
```

e) Sayı değişkenindeki değeri okumak için **degeriOku()** fonksiyonunu aşağıdaki gibi tanımlayınız.

```
function degeriOku() public view returns (uint256){
return depolananDeger;
}
```

5. Adım: Akıllı sözleşme kodunu aşağıdaki basamakları izleyerek çalıştırınız.

a) Görsel 7.8.'de görüldüğü üzere **SOLIDITY COMPILER** sekmesine gidiniz ve **Compile DepolaAkilliSozlesme.sol** butonuna tıklayınız.

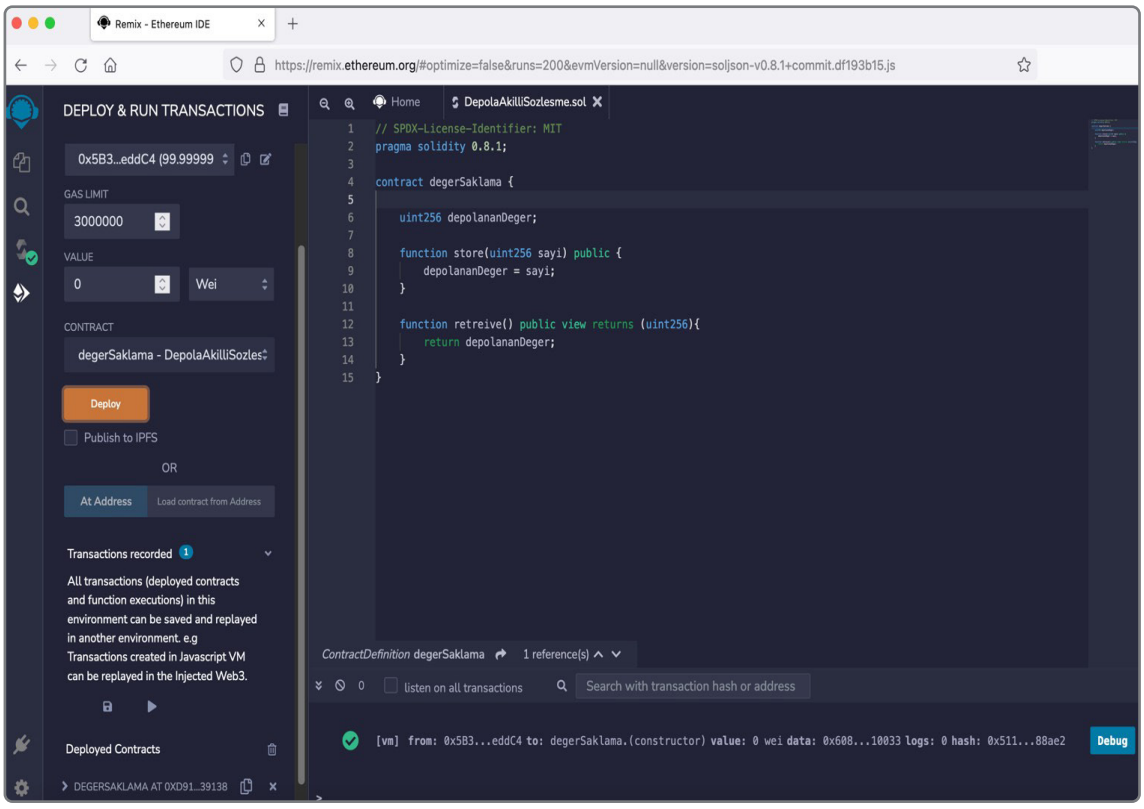


Görsel 7.8: Solidity Compiler

b) Herhangi bir hata yoksa ve derleme işlemi başarılı olarak gerçekleşmişse **SOLIDITY COMPILER** sekmesindeki buton üzerinde yeşil bir onay işareti olacaktır, kontrol ediniz.

c) Compile DepolaAkilliSozlesme.sol butonunun altında aşağıdaki sonuçları görebilirsiniz.

- Görsel 7.9'da görüldüğü üzere **DEPLOY & RUN TRANSACTIONS** sekmesine giderek **Deploy** butonuna tıklayınız.



Görsel 7.9: Deploy & Run Transactions

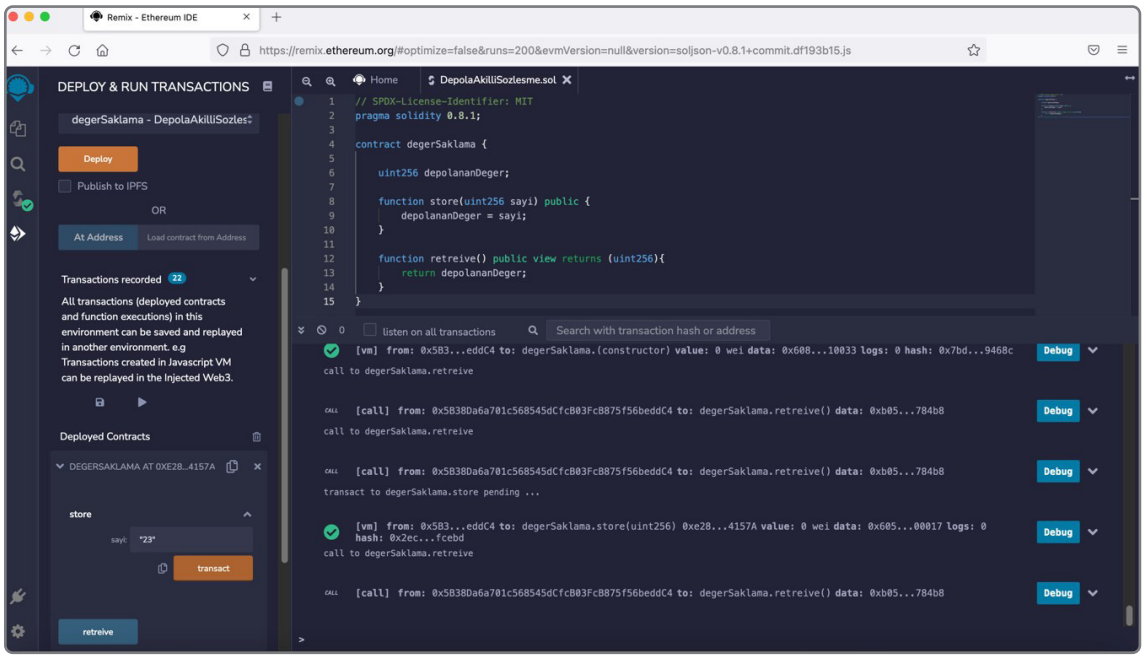
NOT

Görsel 7.9'da görüldüğü üzere herhangi bir hata yoksa sonuçları Deployed Contract bölümünde görebilirsiniz.

6. Adım: Akıllı sözleşme kodunu aşağıdaki basamakları izleyerek test ediniz.

a) Deployed Contract bölümündeki **degerSaklama** sözleşmesinin detaylarına tıklayarak akıllı sözleşmede yazılan fonksiyonlara erişiniz.

b) Görsel 7.10'da görüldüğü üzere metin kutusunu kullanarak blok zincirine "23" sayısını yazınız **store (sakla)** butonuna basınız. Blok zincirine girilen değeri okumak için **retreive (geri al)** düğmesine tıklayınız.



Görsel 7.10: Akıllı sözleşme test

SIRA SİZDE

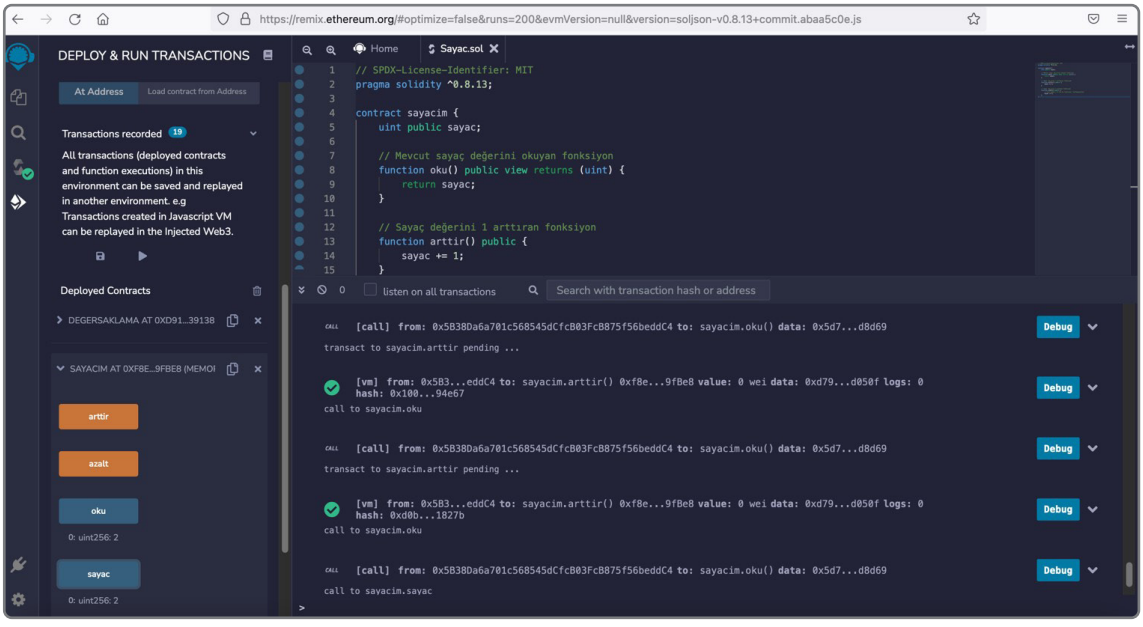
Ethereum ağları ile etkileşimli akıllı sözleşme oluşturunuz.

SIRA SİZDE

Remix IDE kullanarak Görsel 7.11'de gösterilen mevcut sayaç değerini okuyan, artıran ve azaltan aşağıda kodu verilen akıllı sözleşmeyi yazınız. Sözleşmeyi derleyiniz. Sonrasında Görsel 7.12'de gösterilen **sayacim** sözleşmesindeki **oku**, **arttir** ve **azalt** fonksiyonlarını **Remix IDE** kullanarak test ediniz.

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.13;

contract sayacim {
  uint public sayac;

  // Mevcut sayaç değerini okuyan fonksiyon
  function oku() public view returns (uint) {
    return sayac;
  }

  // Sayaç değerini 1 arttıran fonksiyon
  function arttir() public {
    sayac += 1;
  }

  // Sayaç değerini 1 azaltan fonksiyon
  function azalt() public {
    // Eğer sayac = 0 ise bu fonksiyon çalışmayacaktır.
    sayac -= 1;
  }
}
```

Görsel 7.11: Akıllı sözleşme kodu



Görsel 7.12: Akıllı sözleşme test

7.5. INFURA KURULUMU

Infura, blok zincirinde akıllı sözleşme geliştirme aşamasında test ya da ana blok zinciri ağlarına (main net) token gönderimi için API desteği sağlar. Infura'nın kurulumu, "https://infura.io/register" sitesine kaydolunarak gerçekleştirilir.

SIRA SİZDE

Infura kurulumu gerçekleştiriniz.

7.6. ETHERSCAN'DE DEPLOY GÖZLEMİ

Etherscan merkezî bir platformdur. Kullanıcılar, bu ücretsiz platformu kullanarak Ethereum blok zinciri ağında bulmak istediği veriye erişebilir. Bunun yanı sıra kullanıcılar, farklı Ethereum cüzdan adreslerindeki varlıklarını izlemek amacıyla Etherscan'i kullanabilir. Etherscan ile sahip oldukları cüzdandaki bakiyelerini ve işlem geçmişlerini gözlemleyebilir. Bu işlem için "https://etherscan.io/" adresini ziyaret edebilirler.

SIRA SİZDE

Etherscan ile cüzdan bakiyenizi ve işlemlerinizi gözlemleyiniz.

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

1. Akıllı kontrat (sözleşme) adı için aşağıdaki sözdizimlerinden hangisi doğru kullanımdır?

A) contract (name) { }

B) cont [name] { }

C) contract name { }

D) contract_name { }

E) contract [name] { }

2. Akıllı kontrat (sözleşme) ile ilgili aşağıda verilen ifadelerden hangisi yanlıştır?

A) Remix IDE, solidity dili kullanılarak akıllı sözleşme yazmak kullanılır.

B) Remix IDE, akıllı sözleşmeyi çalıştırmak için kullanılır.

C) Remix IDE, akıllı sözleşmeyi test etmek için kullanılır.

D) Remix IDE'de her işlemin çalışması için belirlenen limite CoinLimit denir.

E) Remix IDE içinde yazılım geliştiriciler için bazı örnek akıllı sözleşme kodları yer alır.

3. Aşağıdaki seçeneklerde verilen bilgilerden hangisi yanlıştır?

A) Etherscan merkezi olmayan bir platformdur.

B) Mocha, node.js ve browser üzerinde çalışan bir test frameworküdür.

C) Kullanıcılar, Ethereum cüzdan adreslerindeki varlıklarını Etherscan ile izleyebilir.

D) Infura, blok zincirinde akıllı sözleşme testi için kullanılır.

E) Etherscan ile cüzdandaki bakiye ve işlem geçmişi gözlemlenir.

KONULAR

8.1. TEMEL SOLIDITY DİLİNİN YAPISI

8.2. İLERİ DÜZEY AKILLI KONTRATLARDA HATA AYIKLAMA

8.3. ETHEREUM'DA AKILLI KONTRAT DİZAYNI

8.4. ETHEREUM'DA AKILLI KONTRAT YAZIMI

8.5. ETHEREUM PROJESİNİ TEST ETME

NELER ÖĞRENECEKSİNİZ?

- Temel Solidity dilinin yapısı ve kullanımı
- Ethereum projesi oluşturma süreci
- Ethereum projesinde kontrat yazdırılma işlemi
- Ethereum projesinde test işlemi

ANAHTAR KELİMELER

Akıllı kontrat, Ethereum, Remix IDE, Solidity

HAZIRLIK ÇALIŞMALARI

1. Yüksek seviyeli programlama dilleri ve düşük seviyeli programlama dilleri arasındaki farklar neler olabilir düşüncelerinizi arkadaşlarınızla paylaşınız.

2. Her projenin geliştirilmesinde blok zinciri teknolojisinin kullanılmasının faydaları ve zararları hakkında arkadaşlarınızla tartışınız.

İLERİ DÜZEY AKILLI KONTRATLAR

8. ÖĞRENME BİRİMİ

8.1.TEMEL SOLIDITY YAPISI

Solidity dili, düşük seviyeli bir dildir. Düşük seviyeli diller, makine diline yakın olarak ifade edilen dillerdir.

8.1.1. Veri Tipleri

Tamsayı veri türleri, Solidity sözleşmelerinde sayıları depolamak için kullanılır.

Uint: İşaretsiz tamsayı değerlerini saklamak için kullanılır. Yalnızca 0 dâhil pozitif tamsayıları kapsar. uint8 ile uint256 tipleri mevcuttur. Bir byte ve 32 byte alan kaplar.

1. UYGULAMA

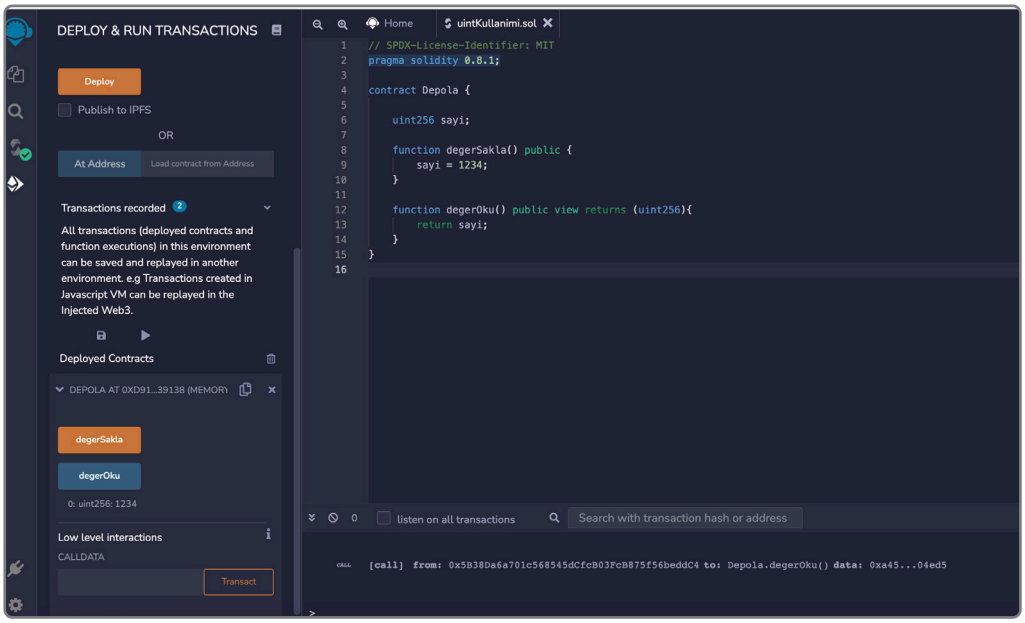
Tamsayı değerleri saklamak için uint veri tipi tanımlaması yapma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde uint tanımlamasının yapıldığı aşağıdaki uygulamayı kodlayınız (Görsel 8.1).

```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.1;

contract Depola {

   uint256 sayi;

   function degerSakla() public {
      sayi = 1234;
   }

   function degerOku() public view returns (uint256){
      return sayi;
   }
}
```

Görsel 8.1: uint değişken tanımlama kodu

2. Adım: Kodladığınız uygulamayı Görsel 8.2'deki şekilde çalıştırınız.



Görsel 8.2 : uint değişkeni kullanımı

Int: İşaretli tamsayı değerlerini saklamak için kullanılır. Bu, "hem pozitif hem de negatif tamsayı" sayıları ve sıfır anlamına gelir. 8'lik adımlarla int8 ila int256 arasındaki anahtar kelimeler, 8 bitten 256 bite kadar işaretli tamsayıları tanımlamak için kullanılır. int256, int ile aynıdır.

2. UYGULAMA

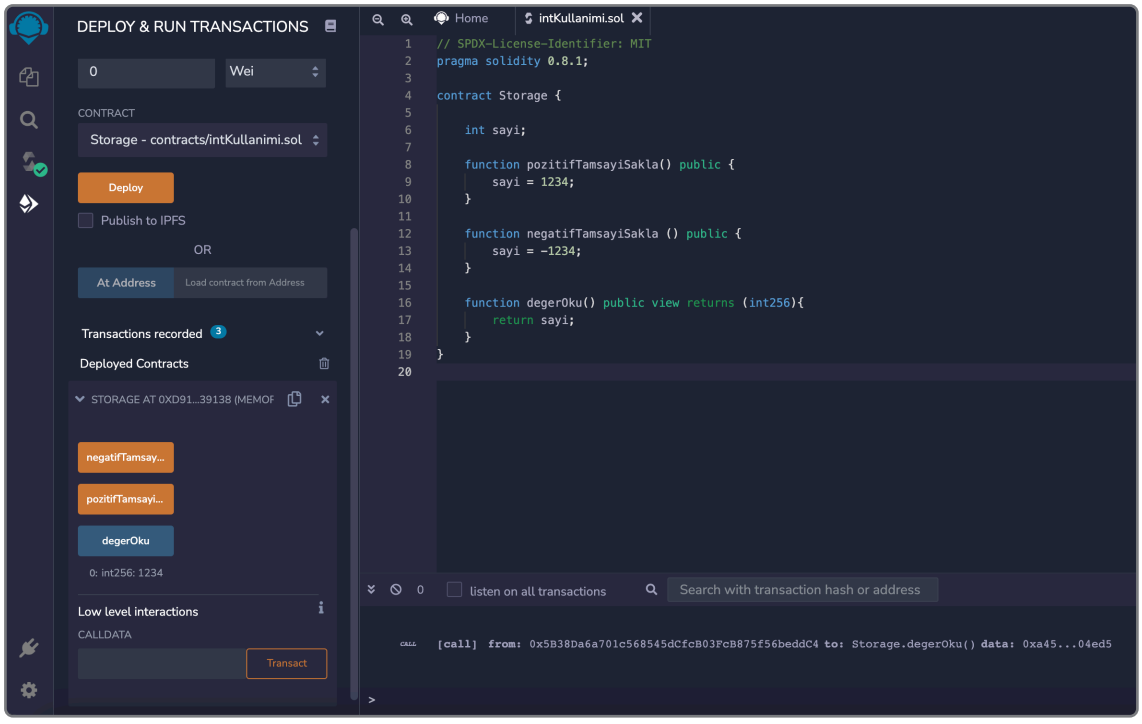
İşaretli tamsayı değerlerini saklamak için uint veri tipi tanımlaması yapma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde int tanımlamasının yapıldığı aşağıdaki uygulamayı kodlayınız (Görsel 8.3).

```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.1;

contract Storage {

  int sayi;

  function pozitifTamsayiSakla() public {
    sayi = 1234;
  }

  function negatifTamsayiSakla () public {
    sayi = -1234;
  }

  function degerOku() public view returns (int256){
    return sayi;
  }
}
```

Görsel 8.3: int değişken tanımlama kodu

2. Adım: Kodladığınız uygulamayı Görsel 8.4'teki şekilde çalıştırınız.



Görsel 8.4: int değişkeni kullanımı

Boolean (mantıksal): Solidity dilinde ikili durumlu bir senaryoyu temsil etmek için Boolean değişken türü kullanılır. İki durum doğru veya yanlış değerler alabilir. Solidity'de bool için varsayılan değer "false"tir. Bool değişkeni oluşturmak için aşağıdaki sözdizimi kullanılır.

bool değişken_adı = değer

Örnek: bool durumDegeri;

3. UYGULAMA

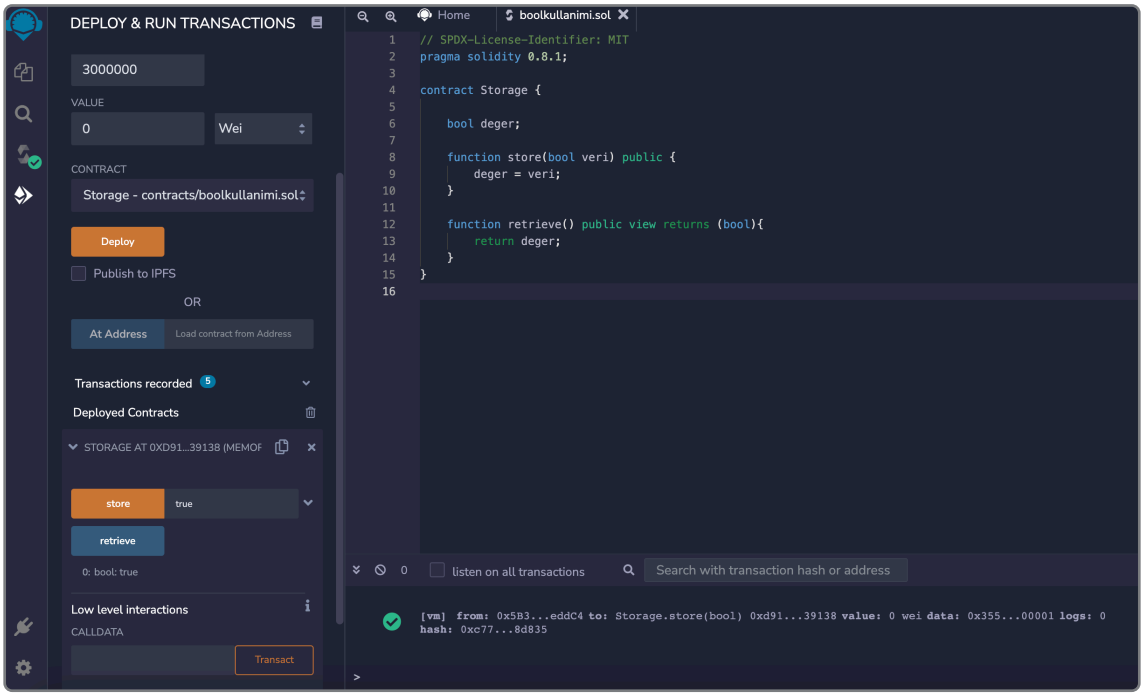
Solidty dilinde bool tanımlamasının yapıldığı uygulamayı kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Aşağıdaki uygulamayı kodlayınız (Görsel 8.5).

```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.1;

contract Storage {

  bool deger;

  function store(bool veri) public {
     deger = veri;
  }

  function retrieve() public view returns (bool){
     return deger;
  }
}
```

Görsel 8.5: Bool değişken tanımlama kodu

2. Adım: Kodladığınız uygulamayı Görsel 8.6'daki şekilde çalıştırınız.



Görsel 8.6: Bool değişkeni kullanımı

String: Solidity programı, çift veya tek tırnak ile yazılan string tanımlamalarını destekler. string değişkeni tanımlamak için string anahtar sözcüğü ardından değişkenin adı ve değişkene isteğe bağlı değer ataması yapılır. string değişkeni oluşturmak için aşağıdaki sözdizimi kullanılır.

string değişken_adı = değer

Örnek: string numarası = "123";

4. UYGULAMA

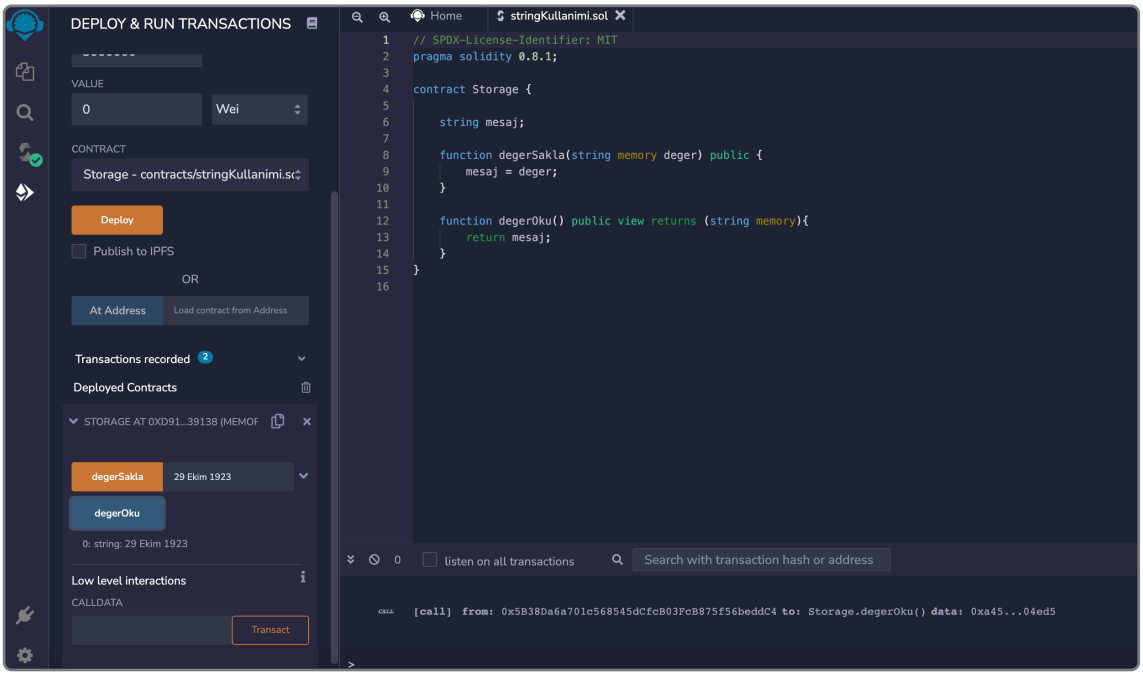
Solidty dilinde string tanımlamasının yapıldığı uygulamayı kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Aşağıdaki kodlamayı yapınız (Görsel 8.7).

```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.1;

contract Storage {

   string msgValue;

   function store(string memory data) public {
      msgValue = data;
   }

   function retrieve() public view returns (string memory){
      return msgValue;
   }
}
```

Görsel 8.7: String değişken tanımlama kodu

2. Adım: Kodladığınız uygulamayı Görsel 8.8'deki şekilde çalıştırınız.



Görsel 8.8: String değişkeni kullanımı

address: Ethereum adres tipinde 20 byte uzunluğunda değer tutulur. Ethereum üzerinde private anahtar ile oluşturulmuş **gerçek hesap adresi** ile **akıllı sözleşmelere ait adres** değerleri bulunur. İki hesap tipi de **address tipi** olarak ifade edilir. Adres tiplerinin iki üyesi bulunur.

- **Balance:** Hesabın bakiyesini tutmak için kullanılır. Eğer akıllı sözleşmeye ait bakiyeye erişilecekse **this.balance** ifadesi kullanılır ve tüm bakiyeye erişim sağlanır.
- **Transfer:** Alıcı adrese Ethereum gönderimi için kullanılmaktadır.

Bir adres değişkeni oluşturmak için aşağıdaki sözdizimi kullanılır:

adres değişkeni_adı = değer

Örnek: address admin = 0x003b012aB50a2D8C47308647ebD20530Fb4C22B5;

5. UYGULAMA

Solidty dilinde address tanımlamasının kullanıldığı uygulamayı kodlama, admin değişkenine bir adres değeri atama, degeriOku() fonksiyonunu kullanılarak adres değerini okuma, akıllı sözleşmeyi kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde **address** tanımlamasının kullanıldığı aşağıdaki uygulamayı kodlayınız (Görsel 8.9).

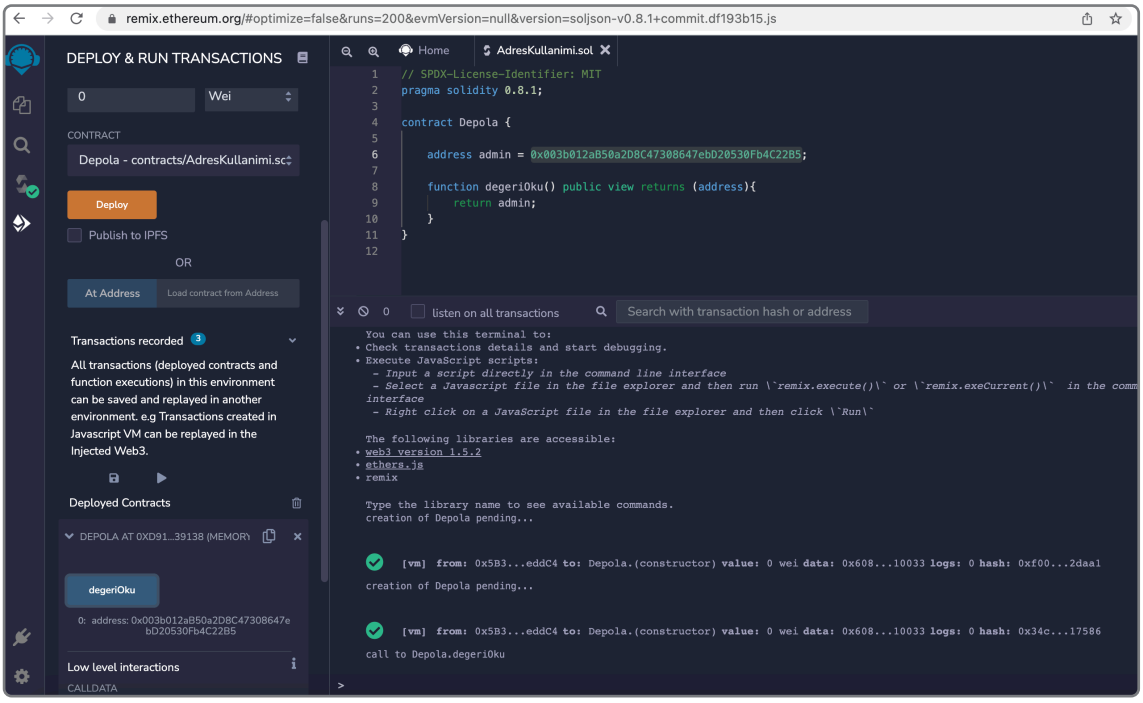
```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.1;

contract Depola {

   address admin = 0x003b012aB50a2D8C47308647ebD20530Fb4C22B5;

   function degeriOku() public view returns (address){
      return admin;
   }
}
```

Görsel 8.9: address tanımlama kodu

2. Adım: admin değişkenine bir adres değeri atayarak **degeriOku()** fonksiyonu ile adres değerini okutunuz.

3. Adım: Akıllı sözleşmeyi kodlayınız.

4. Adım: Kodladığınız akıllı sözleşmeyi Görsel 8.10'daki şekilde çalıştırınız.



Görsel 8.10: Adres değerinin okunması

Diziler (Arrays): Diğer dillerde olduğu gibi Solidity de dizileri destekler. Dizi boyutu sabit veya dinamik olabilir.

Sabit boyutlu bir dizi oluşturmak için aşağıdaki sözdizimi kullanılır.

[boyut] dizi adını yazınız

Örnek: unit [8] sayilar;

Dinamik bir dizi oluşturmak için boyut alanının boş bırakılması yeterlidir.

Tip[] array dizi adını yazınız

Örnek: unit[] sayilar;

Dizi Fonksiyonları:

Length (uzunluk): Sabit ve dinamik dizinin boyutunu almak için kullanılır.

Push: Dinamik bir diziye eleman eklemek için kullanılır.

Pop: Dinamik bir dizinin son üyesini kaldırmak için kullanılır.

• **Sabit boyutlu byte dizileri:** Boyutu kullanımından önce belirlenmiş dizilerdir. Bu diziler, bytes1 ila bytes32 olmak üzere 32 farklı tipte bulunur. Sözleşmenin yapısına göre öncelikle bytes1 ile bytes32 arasında dizi kullanımı tavsiye edilir çünkü ihtiyaç duyulan dizinin boyutunun bilindiği

durumlarda, ihtiyaç duyulan boyutta sabit dizi tanımlamasının yapılması ödenecek gas ücreti için netlik sağlayacaktır.

Bir bayt değişkeni oluşturmak için aşağıdaki sözdizimini kullanılır.

bytes32 değişken_adı = değer

Örnek: bytes32 mesaj = “23 Nisan 1920”;

6. UYGULAMA

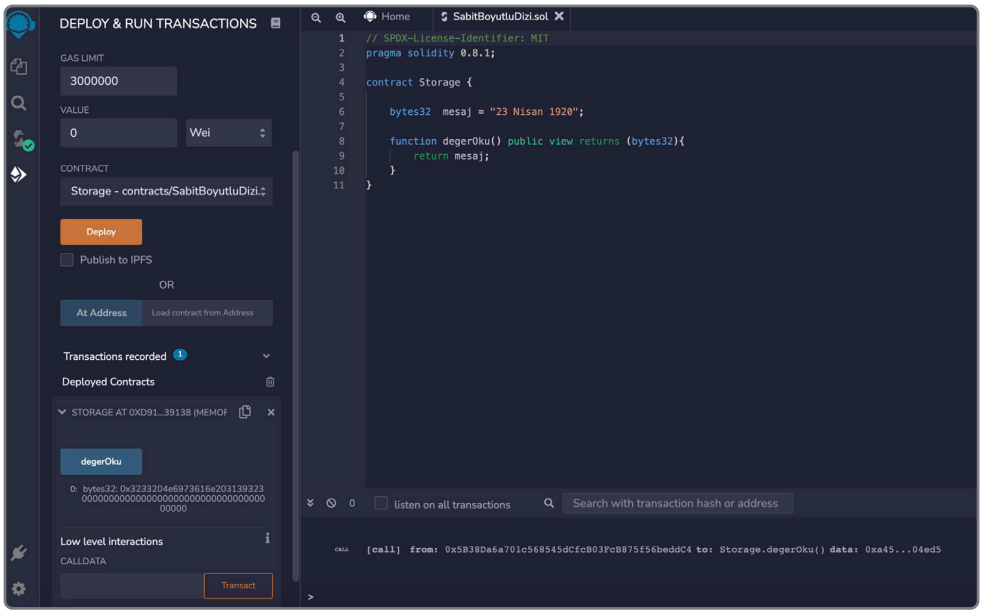
Solidty dilinde sabit uzunluklu dizi tanımlamasının kullanıldığı uygulamayı kodlama, çalıştırma ve çıktı değerini görüntüleme işlemlerini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Aşağıdaki kodlamayı yapınız (Görsel 8.11).

```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.1;
contract Storage {
   bytes32  mesaj = "23 Nisan 1920";
   function degerOku() public view returns (bytes32){
      return mesaj;
   }
}
```

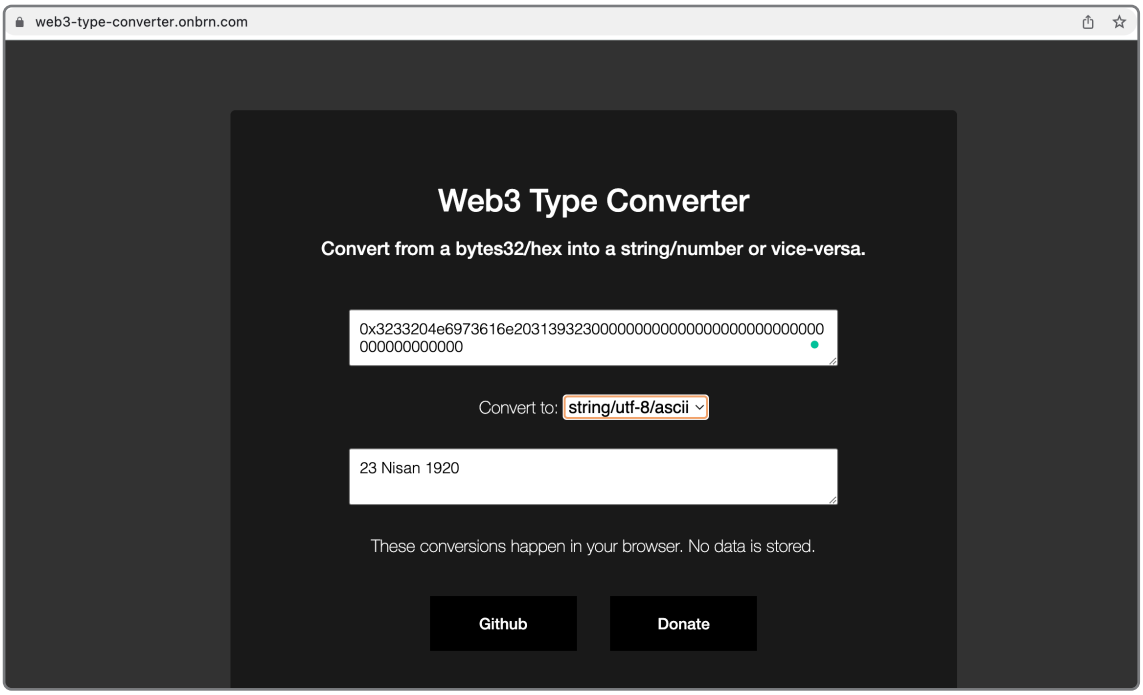
Görsel 8.11: Sabit uzunluklu dizi tanımlama kodu

2. Adım: Programı çalıştırınız. Oluşan çıktı değerinin okunabilir olmadığını gözlemleyiniz. Görüldüğü gibi çıktı değerinin okunabilir olmasını sağlamak için ilgili web sayfasında kodu Görsel 8.12’deki gibi çalıştırınız.



Görsel 8.12: Sabit uzunluklu dizi tanımlanması

3. Adım: Görsel 8.13'te görüldüğü üzere "0x3233204e6973616e203139323000000000000000000000000000000000000000" ifadesini metin kutusuna yapıştırarak "23 Nisan 1920" değerini görüntüleyiniz.



Görsel 8.13: Web3 Tip Dönüştürücü (Web3 Type Converter)

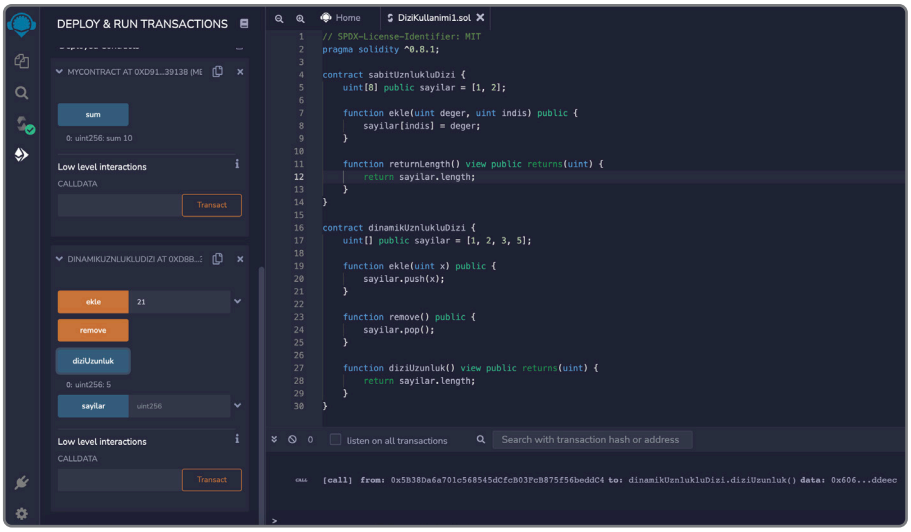
• **Dinamik boyutlu byte dizileri:** Dinamik boyutlu dizilerin **boyutu, dizi kullanımından önce belirlenmez.** Diziler **string** ve **byte** tiplerinde tanımlanabilir. Sözleşmede byte tipinin kullanılması, string tipinin kullanılmasına göre daha az **gas** ücreti gerektirir. **Byte;** boyutu dinamik belirlenen ve byte değerler içeren dizilerdir. **String** ise boyutu dinamik belirlenen ve string değer tutan dizilerdir.

7. UYGULAMA

Solidty dilinde dizi kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Aşağıdaki kodlamayı yapınız.

2. Adım: Yaptığınız kodlamayı Görsel 8.14'teki gibi çalıştırınız.



Görsel 8.14: Dizi kullanımı

8. UYGULAMA

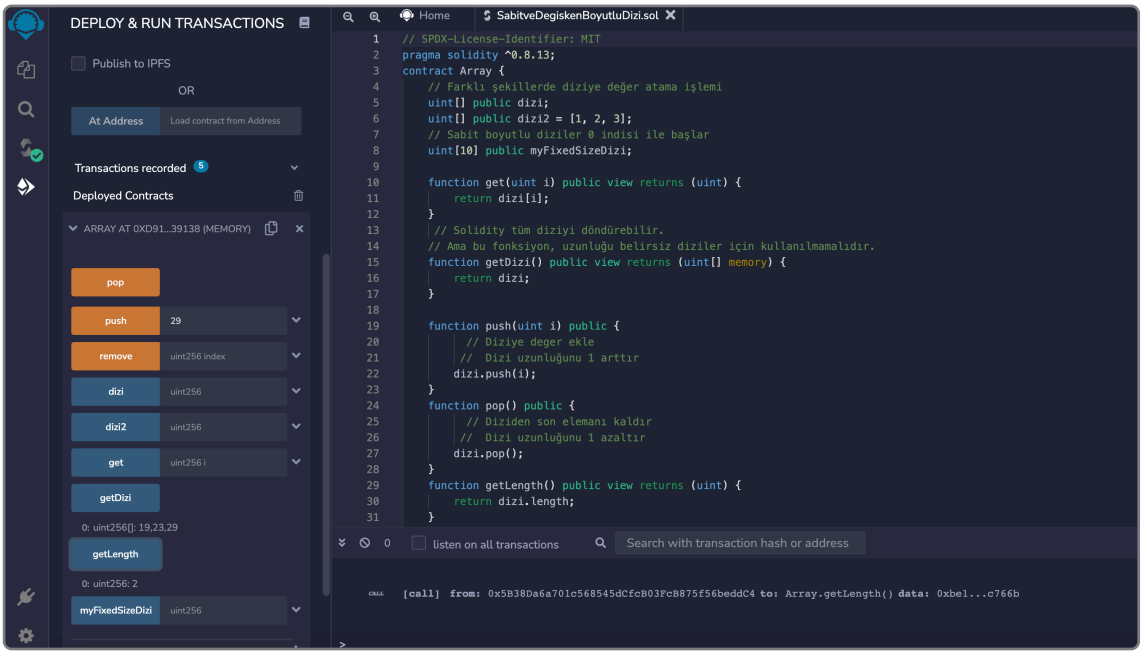
Solidty dilinde dizi kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Aşağıdaki kodlamayı yapınız (Görsel 8.15).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.13;
contract Array {
    // Farklı şekillerde diziye değer atama işlemi
    uint[] public dizi;
    uint[] public dizi2 = [1, 2, 3];
    // Sabit boyutlu diziler 0 indisi ile başlar
    uint[10] public myFixedSizeDizi;

    function get(uint i) public view returns (uint) {
        return dizi[i];
    }
     // Solidity tüm diziyi döndürebilir.
    // Ama bu fonksiyon, uzunluğu belirsiz diziler için kullanılmamalıdır.
    function getDizi() public view returns (uint[] memory) {
        return dizi;
    }

    function push(uint i) public {
         // Diziye deger ekle
        //  Dizi uzunluğunu 1 arttır
        dizi.push(i);
    }
    function pop() public {
         // Diziden son elemanı kaldır
        //  Dizi uzunluğunu 1 azaltır
        dizi.pop();
    }
    function getLength() public view returns (uint) {
        return dizi.length;
    }
    function remove(uint index) public {
        // Sil, dizi uzunluğunu değiştirmez.
        // İndex değerini varsayılan değeri sıfır yapar
        // İndex değeri 0 yapılır.
        delete dizi[index];
    }
```

Görsel 8.15: Dizi kullanım kodları

2. Adım: push butonu kullanarak 19, 23 ve 29 değerlerini dizeye yazdırınız. Görsel 8.16'daki şekilde çalıştırınız.



Görsel 8.16: Dizi kullanımı

Sabitler (Constants) : Sabit değişkenlerin değerleri değiştirilemez. "Sabit"lerin değerleri sabit olarak kodlanmıştır ve akıllı sözleşmelerde kullanılması **gas** maliyetinden tasarruf sağlayabilir.

9. UYGULAMA

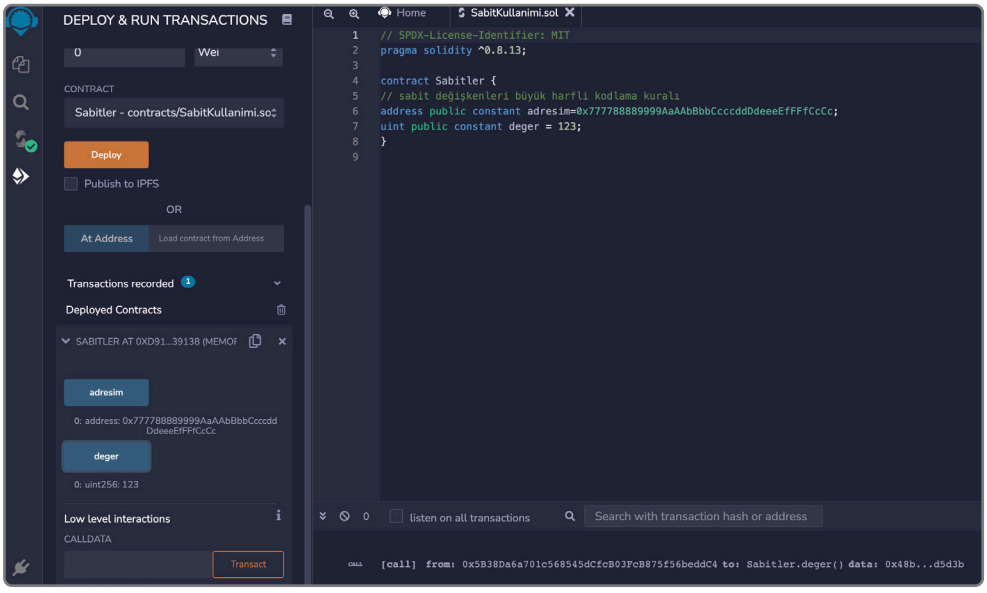
Solidty dilinde sabit (constant) veri tipinin kullanıldığı uygulamayı kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Aşağıdaki kodlamayı yapınız (Görsel 8.17).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.13;

contract Sabitler {
// sabit değişkenleri büyük harfli kodlama kuralı
address public constant adresim=0x777788889999AaAAbBbbCcccddDdeeeEfFFfCcCc;
uint public constant deger = 123;
}
```

Görsel 8.17: Sabit kullanım kodu

2. Adım: Kodladığınız uygulamayı Görsel 8.18'deki şekilde çalıştırınız.



Görsel 8.18: Sabit kullanımı

10. UYGULAMA

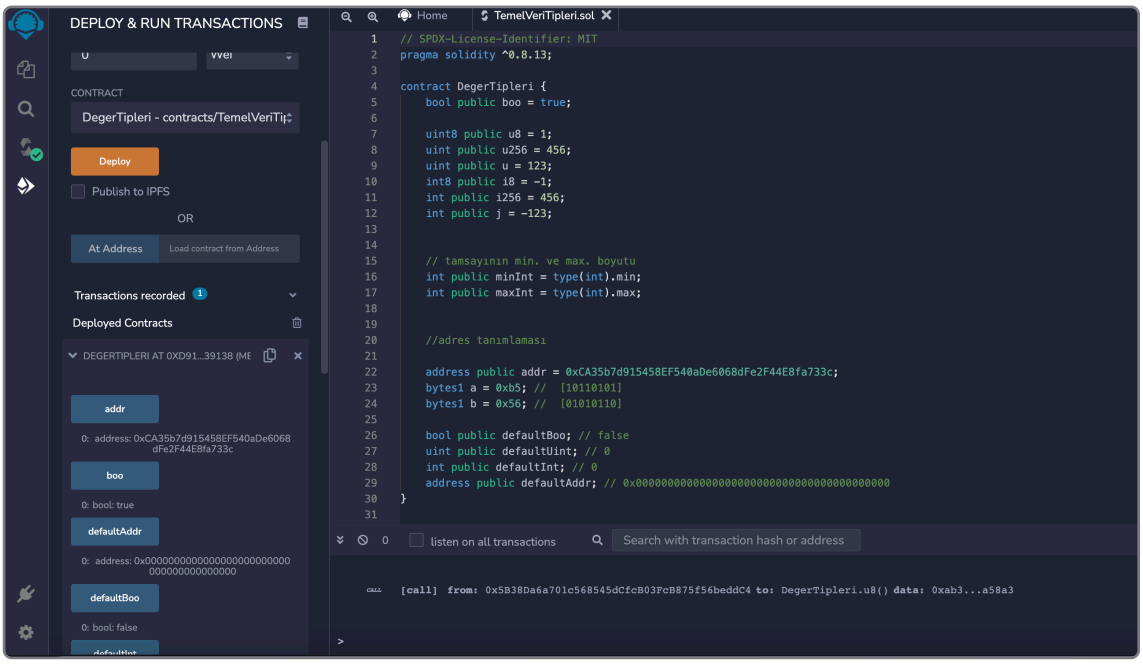
Solidty dilinde temel veri tiplerinin kullanıldığı uygulamayı kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Aşağıdaki kodlamayı yapınız (Görsel 8.19).

```
SPDX-License-Identifier: MIT
pragma solidity ^0.8.13;

contract DegerTipleri {
    bool public boo = true;

    uint8 public u8 = 1;
    uint public u256 = 456;
    uint public u = 123;
    int8 public i8 = -1;
    int public i256 = 456;
    int public j = -123;

    // tamsayının min. ve max. boyutu
    int public minInt = type(int).min;
    int public maxInt = type(int).max;

    //adres tanımlaması

    address public addr = 0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c;
    bytes1 a = 0xb5; //  [10110101]
    bytes1 b = 0x56; //  [01010110]

    bool public defaultBoo; // false
    uint public defaultUint; // 0
    int public defaultInt; // 0
    address public defaultAddr; // 0x0000000000000000000000000000000000000000
}
```

Görsel 8.19: Veri tipleri kullanım kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.20'deki şekilde çalıştırınız.



Görsel 8.20: Temel veri tipleri

Enums (Numalandırma): Kullanıcı tanımlı veri türleri oluşturmak için **enum** yapısı kullanılır. Enum kullanımı ile koddaki hata sayısının azaltılması mümkündür.

Örneğin trafik sinyal ışıklarını düzenleyen bir uygulamada sinyalleri Kirmizi, Turuncu ve Yesil olarak kısıtlamak mümkündür. Bu sayede, trafik sinyaline kontrolsüz olarak herhangi bir farklı renk eklenemez.

Bir enum kullanmak için enum anahtar sözcüğünü yazarak önceden tanımlanmış değerler kümesiyle yeni bir veri türü tanımlaması yapılır.

Sözdizimi: enum name_of_data_type {predefined_values}

Örnek:enum TrafficSignals{Kirmizi, Turuncu, Yesil}

Daha sonra yeni veri tipinde bir değişken oluşturulması gerekmektedir.

name_of_veri türü değişken_adı

Örnek: trafikSinyalleri sinyal;

11. UYGULAMA

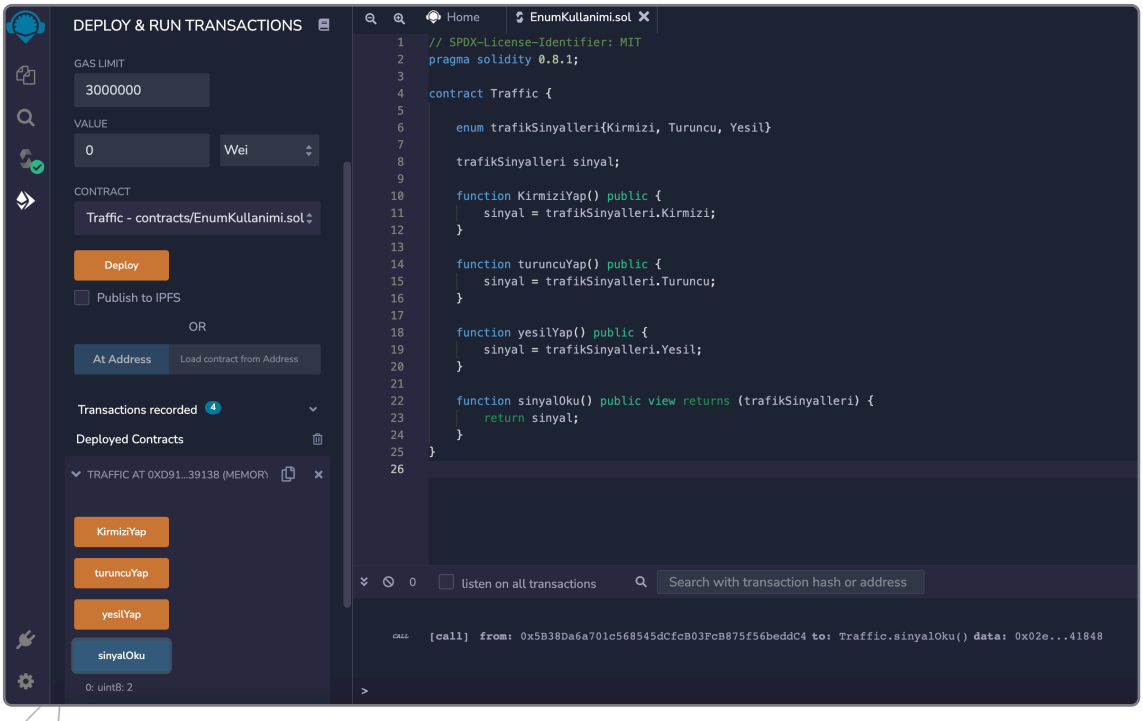
Solidty dilinde Enum yapısının kullanıldığı uygulamayı kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde enum yapısının kullanıldığı uygulamayı kodlayınız (Görsel 8.21).

```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.1;

contract Traffic {

    enum trafikSinyalleri{Kirmizi, Turuncu, Yesil}

    trafikSinyalleri sinyal;

    function KirmiziYap() public {
        sinyal = trafikSinyalleri.Kirmizi;
    }

    function turuncuYap() public {
        sinyal = trafikSinyalleri.Turuncu;
    }

    function yesilYap() public {
        sinyal = trafikSinyalleri.Yesil;
    }

    function sinyalOku() public view returns (trafikSinyalleri) {
        return sinyal;
    }
}
```

Görsel 8.21: Enum kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.22'deki şekilde çalıştırınız.



Görsel 8.22: Enum kullanımı

Structs (Yapılar): struct yapısı, kullanıcı tanımlı veri türleri oluşturmak için farklı veya aynı türdeki değişkenlerin gruplandırılmasını sağlar.

12. UYGULAMA

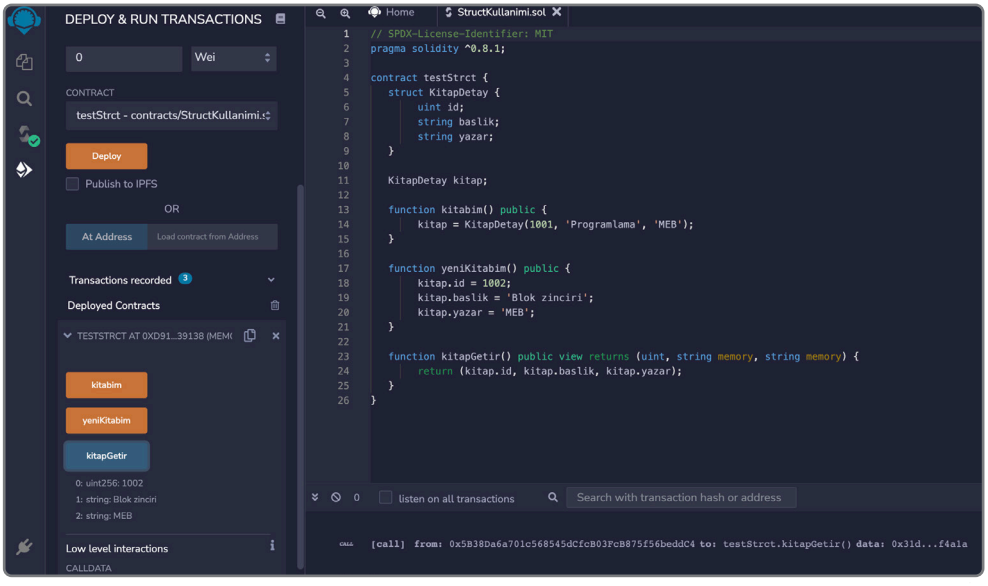
Solidty dilinde struct (yapı) kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde aşağıdaki struct (yapı) kullanım uygulamasını kodlayınız (Görsel 8.23).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.1;

contract testStrct {
  struct KitapDetay {
      uint id;
      string baslik;
      string yazar;
  }

  KitapDetay kitap;

  function kitabim() public {
      kitap = KitapDetay(1001, 'Programlama', 'MTE');
  }

  function yeniKitabim() public {
      kitap.id = 1002;
      kitap.baslik = 'Blok zinciri';
      kitap.yazar = 'MTE';
  }

  function kitapGetir() public view returns (uint, string memory, string memory) {
      return (kitap.id, kitap.baslik, kitap.yazar);
  }
}
```

Görsel 8.23: Struct (yapı) kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.24'teki şekilde çalıştırınız.



Görsel 8.24: Struct (yapı) kullanımı

Mapping (Haritalama): Mapping, bir key (anahtar) / value (değer) veri yapısıdır.

Sözdizimi: mapping (key => value) name_of_mapping

Mappingde, **key** kelimesi ile her bir anahtarın karşılık gelen bir değeri eşleştirilir. **key** ve **value** eşleşmesi için kullanılır. Özellikle DApp'larda verilerin çoğu mapping ve struct kombinasyonuyla depolanır.

13. UYGULAMA

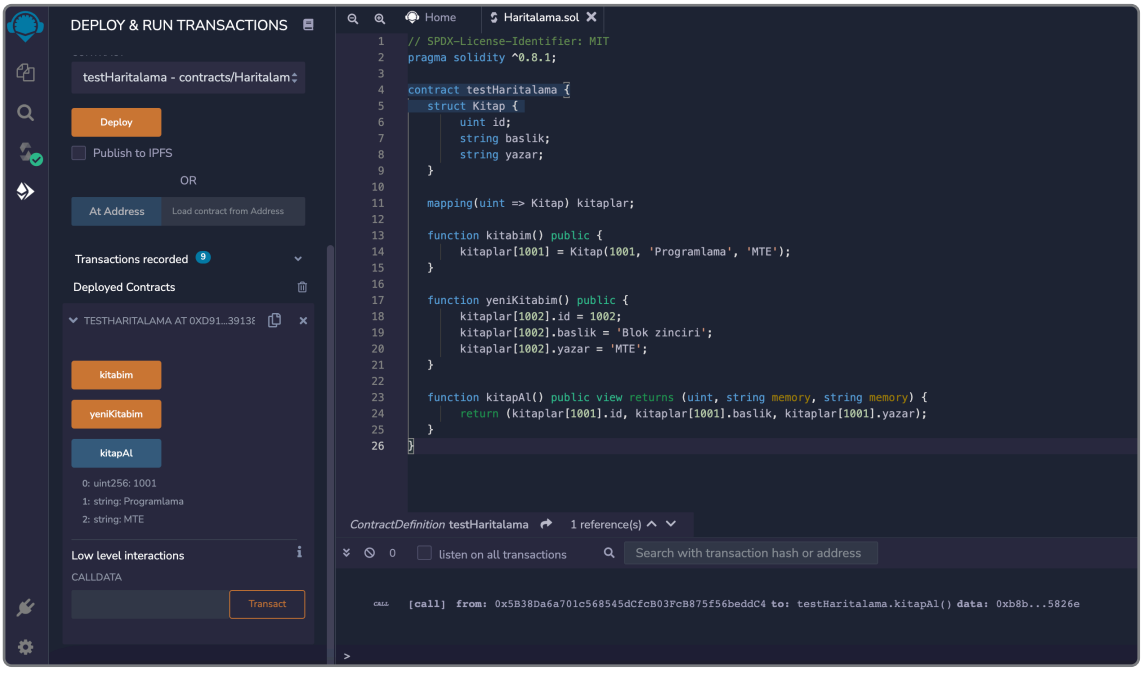
Solidty dilinde oluşturulan mapping (haritalama) kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde oluşturulan aşağıdaki mapping (haritalama) kullanım uygulamasını kodlayınız (Görsel 8.25).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.1;

contract testHaritalama {
  struct Kitap {
      uint id;
      string baslik;
      string yazar;
  }
  mapping(uint => Kitap) kitaplar;
  function kitabim() public {
      kitaplar[1001] = Kitap(1001, 'Programlama', 'MTE');
  }
  function yeniKitabim() public {
      kitaplar[1002].id = 1002;
      kitaplar[1002].baslik = 'Blok zinciri';
      kitaplar[1002].yazar = 'MTE';
  }
  function kitapAl() public view returns (uint, string memory, string memory) {
      return (kitaplar[1001].id, kitaplar[1001].baslik, kitaplar[1001].yazar);
  }
}
```

Görsel 8.25: Mapping (haritalama) kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.26'daki şekilde çalıştırınız.



Görsel 8.26:Mapping (haritalama) kullanımı

8.1.2. Kontrol Yapıları

Kontrol yapıları, akıllı sözleşme yürütme akışının kontrol edilmesine yardımcı olan özellikleri sağlar. Solidity'deki mevcut kontrol yapıları aşağıdaki başlıklardaki gibidir.

If Yapısı : Bir koşula bağlı olarak belirli bir kod bloku çalıştırmak istenildiğinde kullanılır. if, bir koşulu kontrol etmek için kullanılabilecek bir kontrol yapısıdır. Koşul sonucunun doğru olması durumunda bir kod bloku çalıştırılır. Sözdizimi aşağıdaki gibidir.

```
if (koşul) {
// kodu çalıştır
}
```

"if" koşulunun yanlış olması koşuluna bağlı olarak farklı bir kod bloku çalıştırılmak istenirse else bloku çalışır. Else, koşul yanlış olarak değerlendirildiğinde çalışır. Sözdizimi aşağıdaki gibidir.

```
if (koşul) {
// kodu çalıştır
}
else {
// kodu çalıştır
}
```

While Döngüsü: Koşul doğru olduğu sürece kodları tekrar tekrar çalıştırır. Döngüye girerken koşul kontrol edilir. Koşulun doğru olması durumunda kod bloku çalışır.

```
while (koşul) {
// kodu çalıştır
}
```

Do While Döngüsü: Koşul yanlış olana kadar, blok içindeki kodları çalıştırır. Koşul kontrolü, döngünün sonunda gerçekleşmesi nedeniyle while döngüsünden farklıdır. Bu açıdan **çıkış kontrol döngüsü** olarak da ifade edilir.

```
do {
//kodu çalıştır
} while (koşul);
```

For Döngüsü: Solidity'deki for döngüsü diğer dillere benzer. Sözdizimi aşağıdaki gibidir.

```
for (sayacı_başlat; durum_kontrolü; sayacı_artır) {
// yürütülecek kod bloku
}
```

Başlatma sayacı kod blokunun çalıştırılmasından önce yürütülür. Ardından kod blokunun çalıştırılması için koşul kontrolüne bakılır. Kod bloku her çalıştırıldıktan sonra sayaç artar.

Döngü Kontrolleri: Akıllı sözleşme yürütme akışını kontrol etmek için döngüler kullanıldığı gibi continue ve break komutları kullanılarak döngülerin yürütme akışı kontrol edilir.

continue döngüdeki geçerli tekrar durumunu atlamak için kullanılır.

break döngüyü kırmak için kullanılır.

14. UYGULAMA

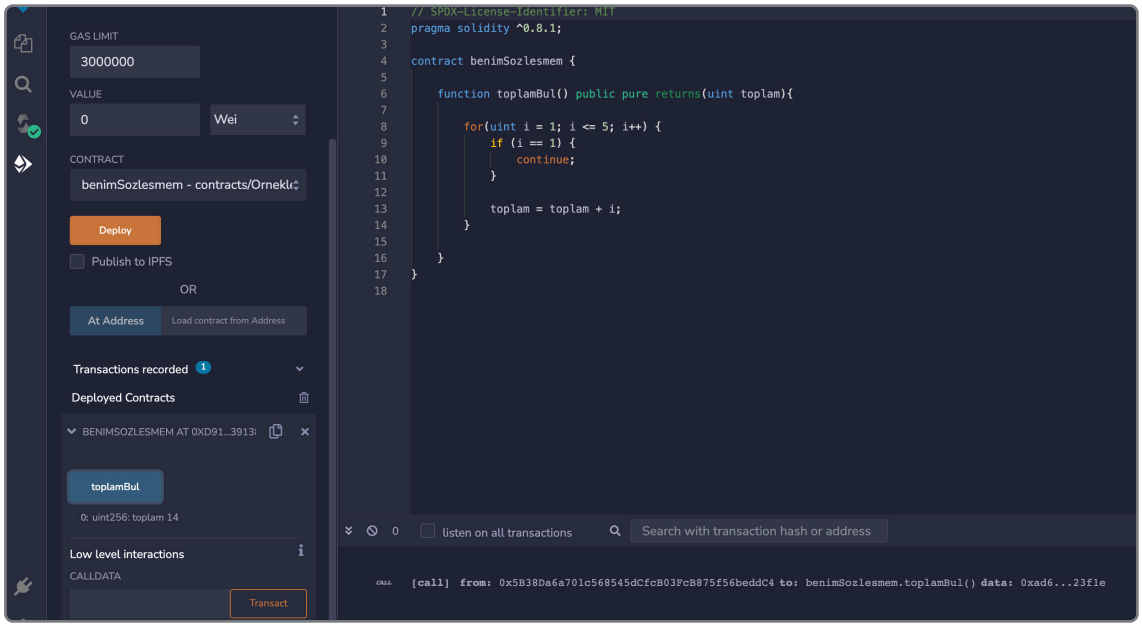
Solidty dilinde continue komutunun kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde continue komutunun aşağıdaki kullanım uygulamasını kodlayınız (Görsel 8.27).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.1;

contract benimSozlesmem {

    function toplamBul() public pure returns(uint toplam){

        for(uint i = 1; i <= 5; i++) {
            if (i == 1) {
                continue;
            }

            toplam = toplam + i;
        }

    }
}
```

Görsel 8.27: continue komut kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.28'deki şekilde çalıştırınız.



Görsel 8.28 : continue komutunun kullanımı

15. UYGULAMA

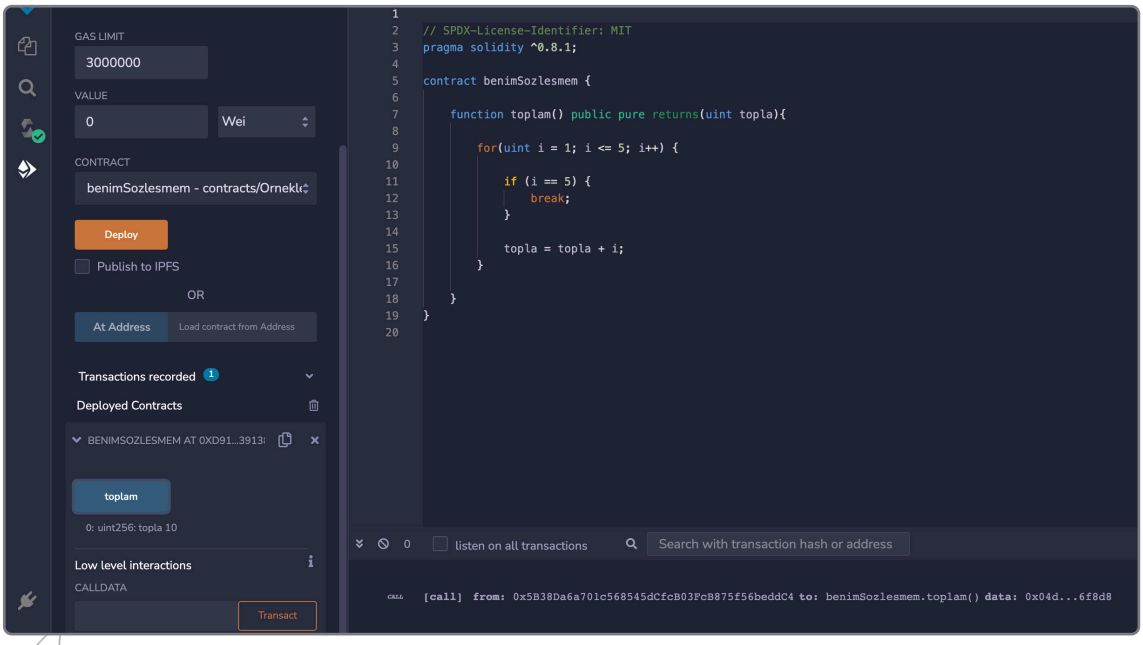
Solidty dilinde break kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde aşağıdaki break kullanım uygulamasını kodlayınız (Görsel 8.29).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.1;

contract benimSozlesmem {

   function toplam() public pure returns(uint topla){

      for(uint i = 1; i <= 5; i++) {

         if (i == 5) {
            break;
         }

         topla = topla + i;
      }

   }
}
```

Görsel 8.29: break komutu kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.30'daki şekilde çalıştırınız.



Görsel 8.30: break komutunun kullanımı

16. UYGULAMA

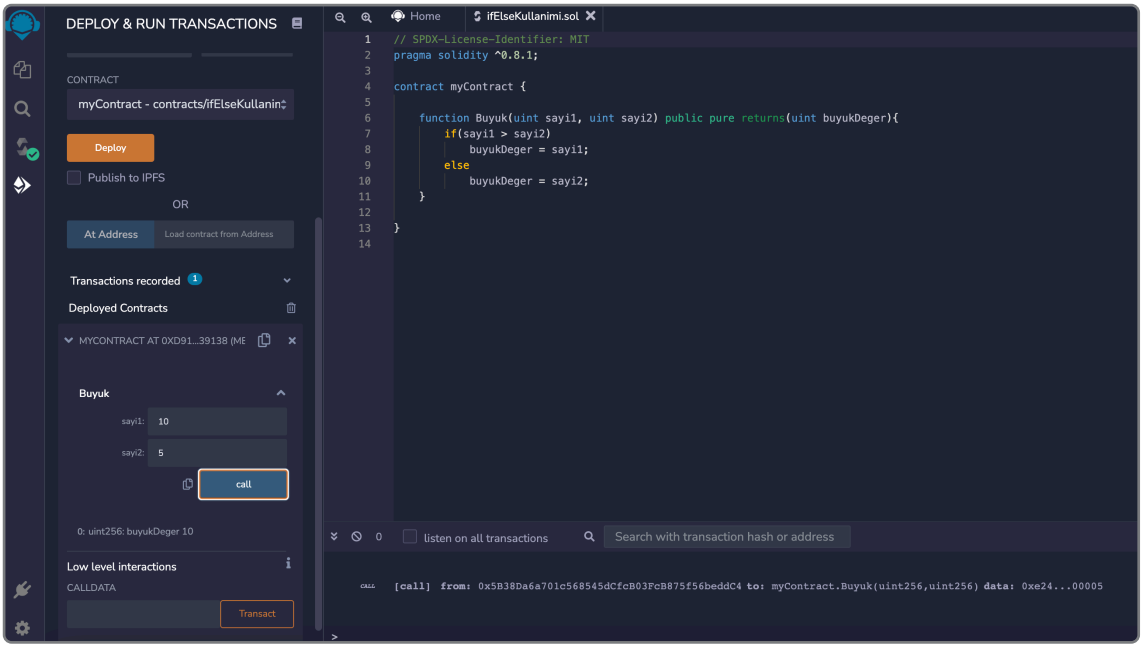
Solidty dilinde if else kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde aşağıdaki if else kullanım uygulamasını kodlayınız (Görsel 8.31).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.1;

contract myContract {

   function Buyuk(uint sayi1, uint sayi2) public pure returns(uint buyukDeger){
      if(sayi1 > sayi2)
         buyukDeger = sayi1;
      else
         buyukDeger = sayi2;
   }

}
```

Görsel 8.31: if else kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.32'deki şekilde çalıştırınız.



Görsel 8.32: If else kullanımı

17. UYGULAMA

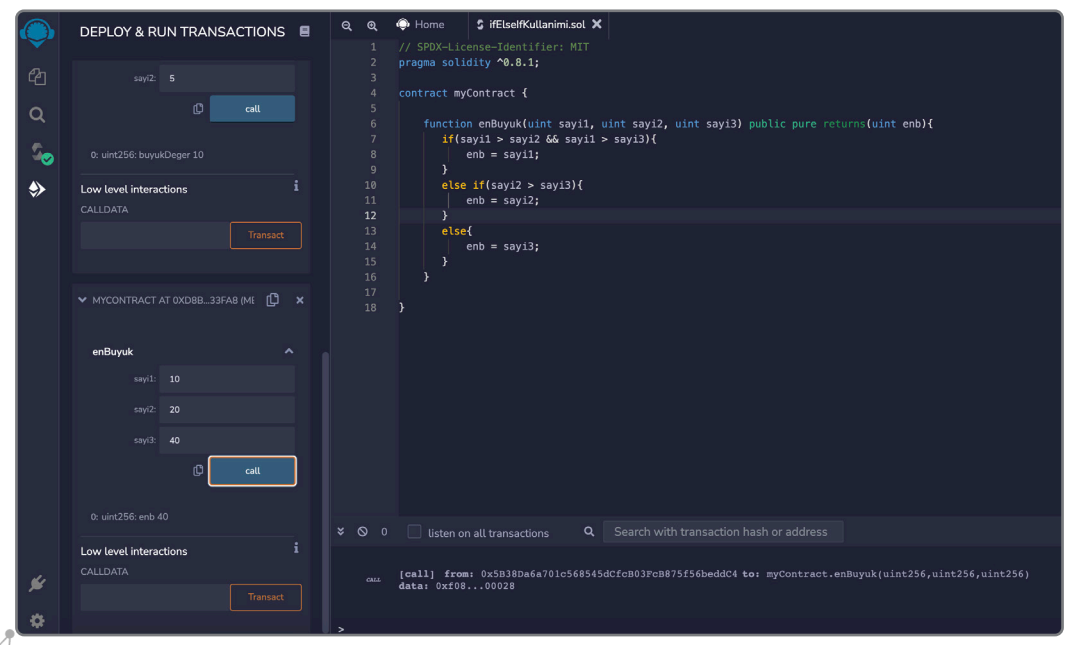
Solidty dilinde if else if kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde aşağıdaki if else if kullanım uygulamasını kodlayınız (Görsel 8.33).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.1;

contract myContract {

   function enBuyuk(uint sayi1, uint sayi2, uint sayi3) public pure returns(uint enb){
      if(sayi1 > sayi2 && sayi1 > sayi3){
         enb = sayi1;
      }
      else if(sayi2 > sayi3){
         enb = sayi2;
      }
      else{
         enb = sayi3;
      }
   }

}
```

Görsel 8.33: if else if kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.34'deki şekilde çalıştırınız.



Görsel 8.34: If else if kullanımı

SIRA SİZDE

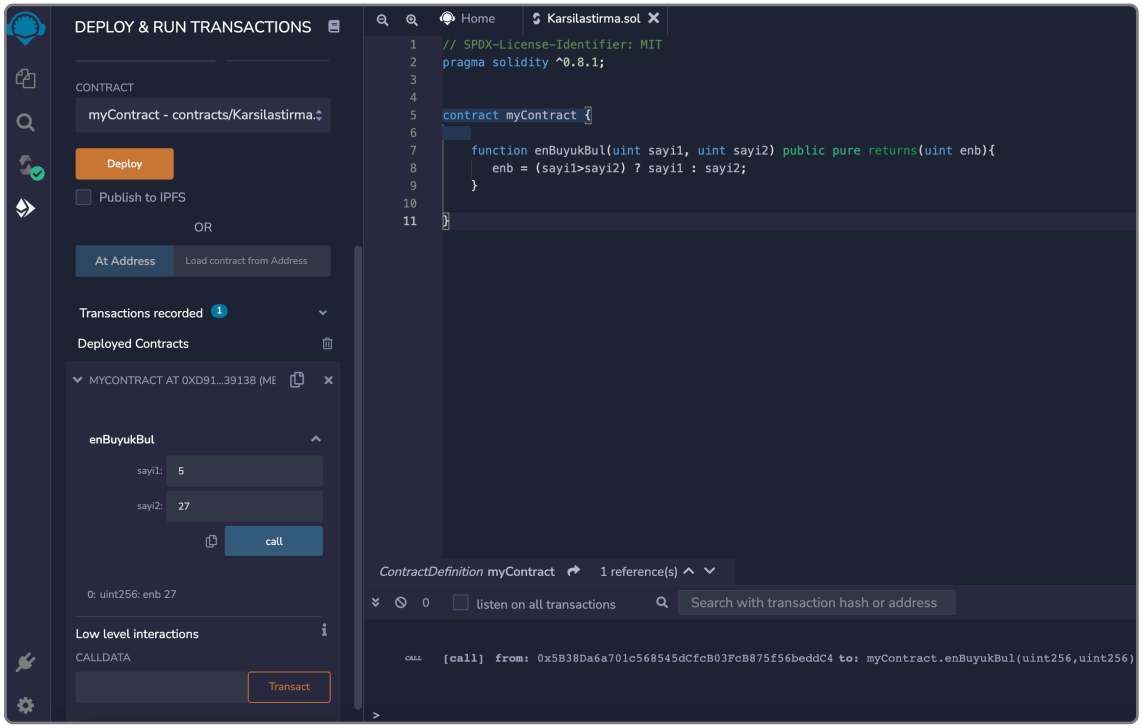
Solidty dilinde üç farklı sayıdan küçük olanını bulan uygulamayı kodlayınız.

18. UYGULAMA

Solidty dilinde koşul operatörü (ternary operator) kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde koşul operatörü (ternary operator) kullanım uygulamasını aşağıdaki gibi kodlayınız.

2. Adım: Kodladığınız uygulamayı Görsel 8.35'teki şekilde çalıştırınız.



Görsel 8.35: Koşul operatörü (ternary operator) kullanımı

19. UYGULAMA

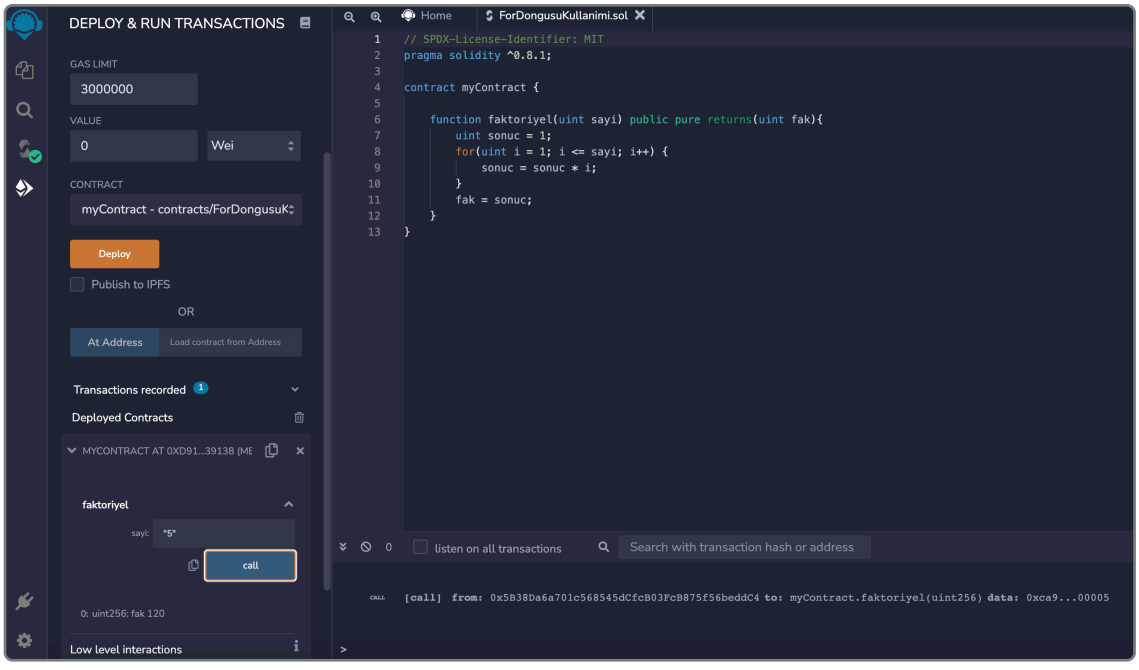
Solidty dilinde for döngüsü kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde for döngüsü kullanım uygulamasını aşağıdaki gibi kodlayınız (Görsel 8.36).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.1;

contract myContract {

   function faktoriyel(uint sayi) public pure returns(uint fak){
      uint sonuc = 1;
      for(uint i = 1; i <= sayi; i++) {
         sonuc = sonuc * i;
      }
      fak = sonuc;
   }
}
```

Görsel 8.36: For döngüsü kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.37'deki şekilde çalıştırınız.



Görsel 8.37: For döngüsü kullanımı

20. UYGULAMA

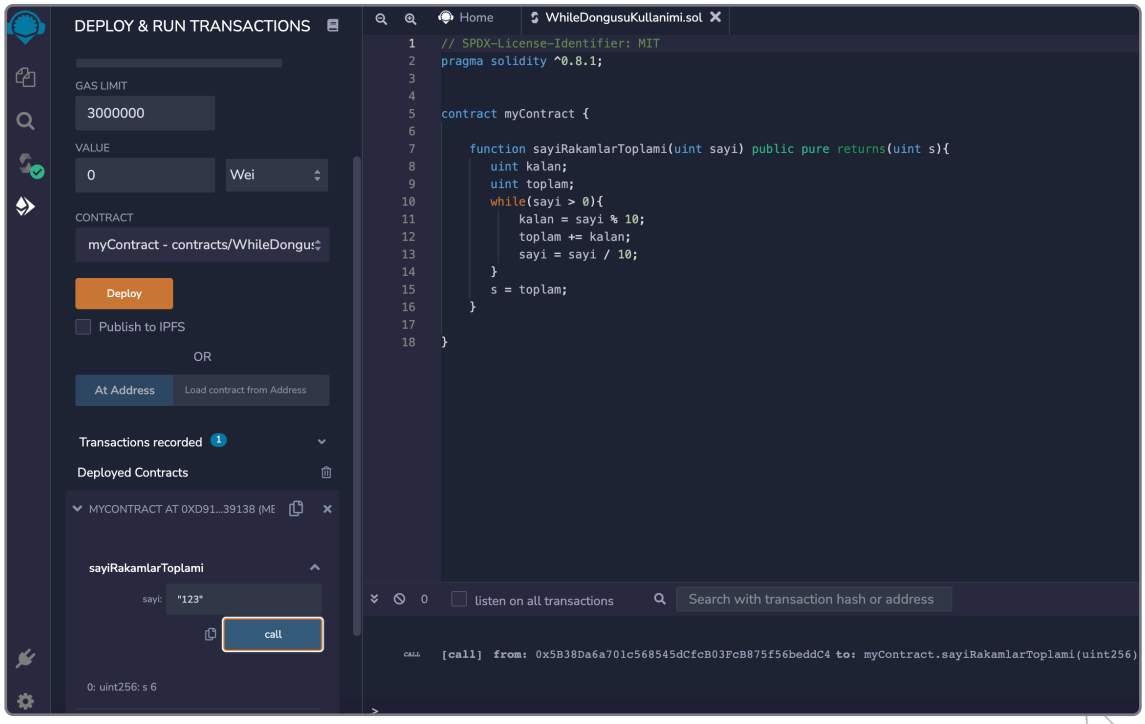
Solidty dilinde while döngüsü kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde while döngüsü kullanım uygulamasını aşağıdaki gibi kodlayınız (Görsel 8.38).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.1;
contract myContract {

  function sayiRakamlarToplami(uint sayi) public pure returns(uint s){
    uint kalan;
    uint toplam;
    while(sayi > 0){
      kalan = sayi % 10;
      toplam += kalan;
      sayi = sayi / 10;
    }
    s = toplam;
  }
}
```

Görsel 8.38: While döngüsü kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.39'daki şekilde çalıştırınız.



Görsel 8.39: While döngüsü kullanımı

21. UYGULAMA

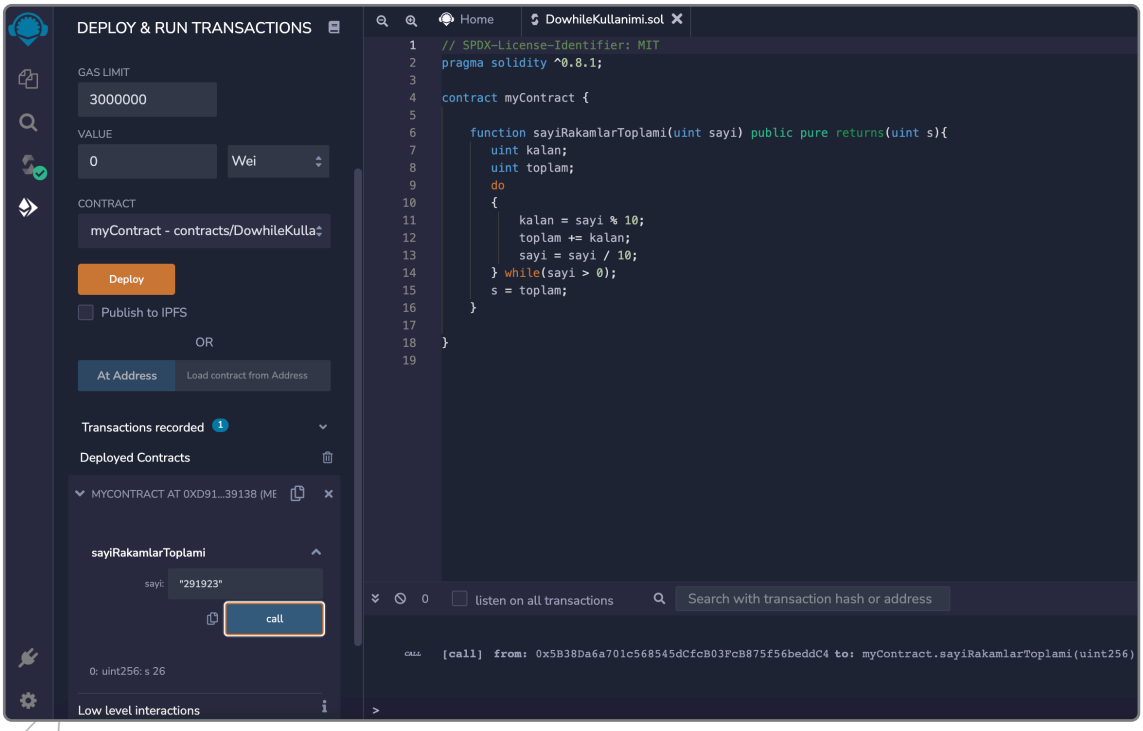
Solidty dilinde while döngüsü kullanım uygulamasını aşağıdaki gibi kodlayınız.

1. Adım: Solidty dilinde do while döngüsü kullanım uygulamasını aşağıdaki gibi kodlayınız (Görsel 8.40).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.1;

contract myContract {

   function sayiRakamlarToplami(uint sayi) public pure returns(uint s){
      uint kalan;
      uint toplam;
      do
      {
         kalan = sayi % 10;
         toplam += kalan;
         sayi = sayi / 10;
      } while(sayi > 0);
      s = toplam;
   }

}
```

Görsel 8.40: Do while döngüsü

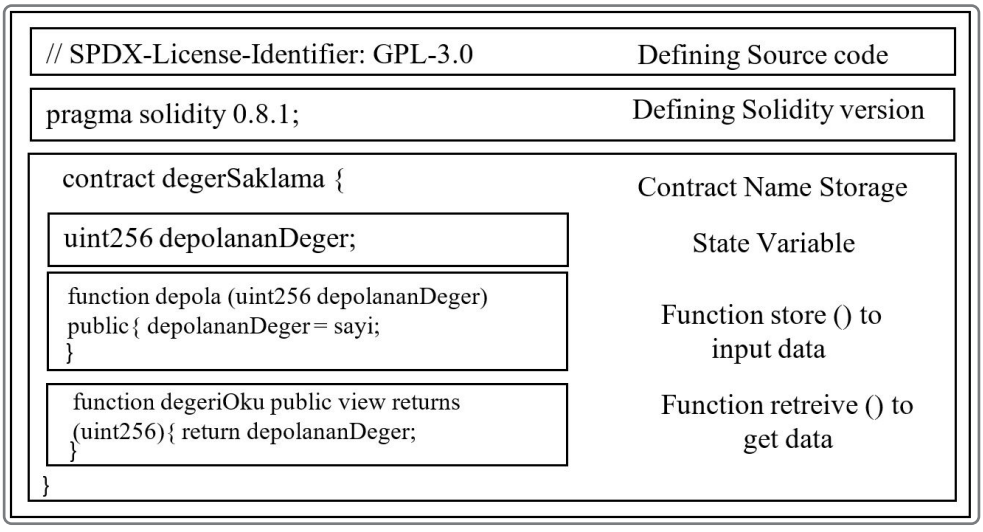
2. Adım: Kodladığınız uygulamayı Görsel 8.41'deki şekilde çalıştırınız.



Görsel 8.41: Do while kullanımı

8.1.3. Akıllı Sözleşmenin Bileşenleri

Solidity dili ile yazılmış akıllı sözleşme yapısı, Java gibi nesne yönelimli dillerdeki sınıf kullanımına benzer. Görsel 8.42'de sözleşme bileşeni gösterilmektedir. Bunlar; durum değişkenleri, yerel değişkenler, genel değişkenler, fonksiyonlar, fonksiyon değiştiriciler, yapı tipleri, enum türleri ve olaylardır (events).



Görsel 8.42: Sözleşme bileşenleri

Durum Değişkenleri (State Variables): Değerleri sözleşmede kalıcı olarak depolamak için kullanılan değişkenlerdir. Bir durum değişkenine yazmak veya değişkenin değerini güncellemek için bir işlem (transaction) gönderilmesi gerekir. Durum değişkenleri herhangi bir işlem ücreti (transaction fee) ödemeden ücretsiz olarak okunabilir. Görsel 8.42'de **depolanaDeger** durum değişkenidir.

22. UYGULAMA

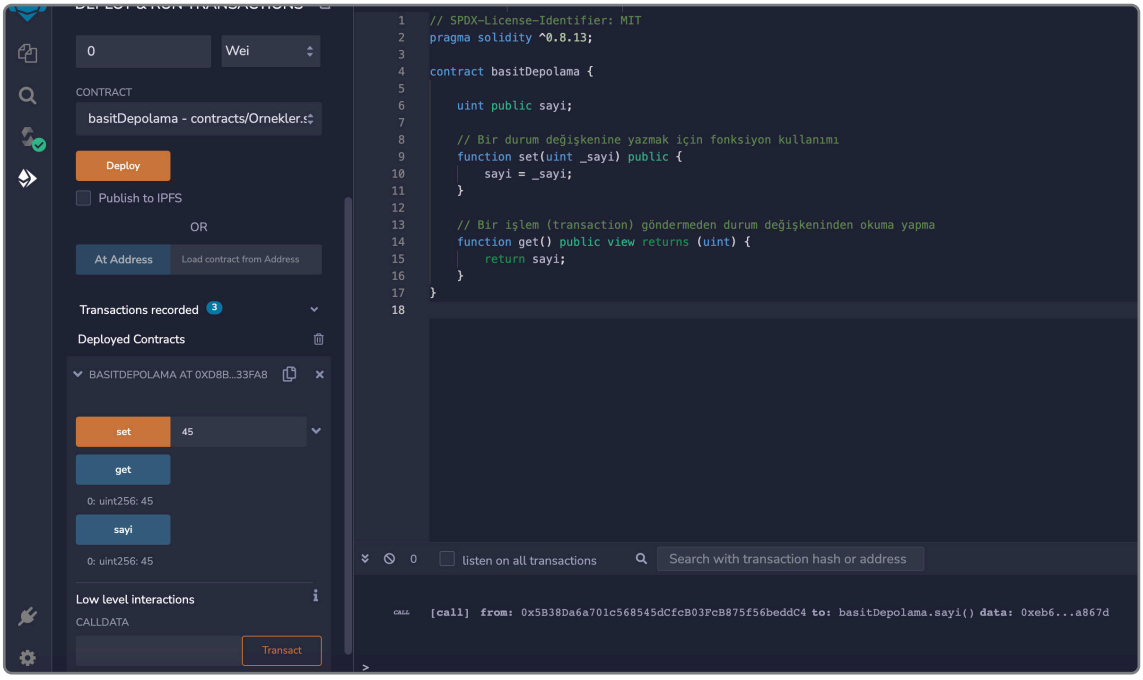
Solidty dilinde durum değişkenine okuma ve yazma işlemi için durum değişkeni (state variable) kullanım uygulamasını kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde durum değişkenine okuma ve yazma işlemi için durum değişkeni (state variable) kullanım uygulamasını aşağıdaki gibi kodlayınız (Görsel 8.43).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.13;

contract basitDepolama {

    uint public sayi;

    // Bir durum değişkenine yazmak için fonksiyon kullanımı
    function set(uint _sayi) public {
        sayi = _sayi;
    }

    // Bir işlem (transaction) göndermeden durum değişkeninden okuma yapma
    function get() public view returns (uint) {
        return sayi;
    }
}
```

Görsel 8.43: Durum değişkeni kullanım kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.44'teki şekilde çalıştırınız.



Görsel 8.44: Durum değişkeni kullanımı

Yerel değişkenler (local variables): Değerleri yalnızca tanımlandığı fonksiyon içinde kullanılabilen değişkenlerdir.

23. UYGULAMA

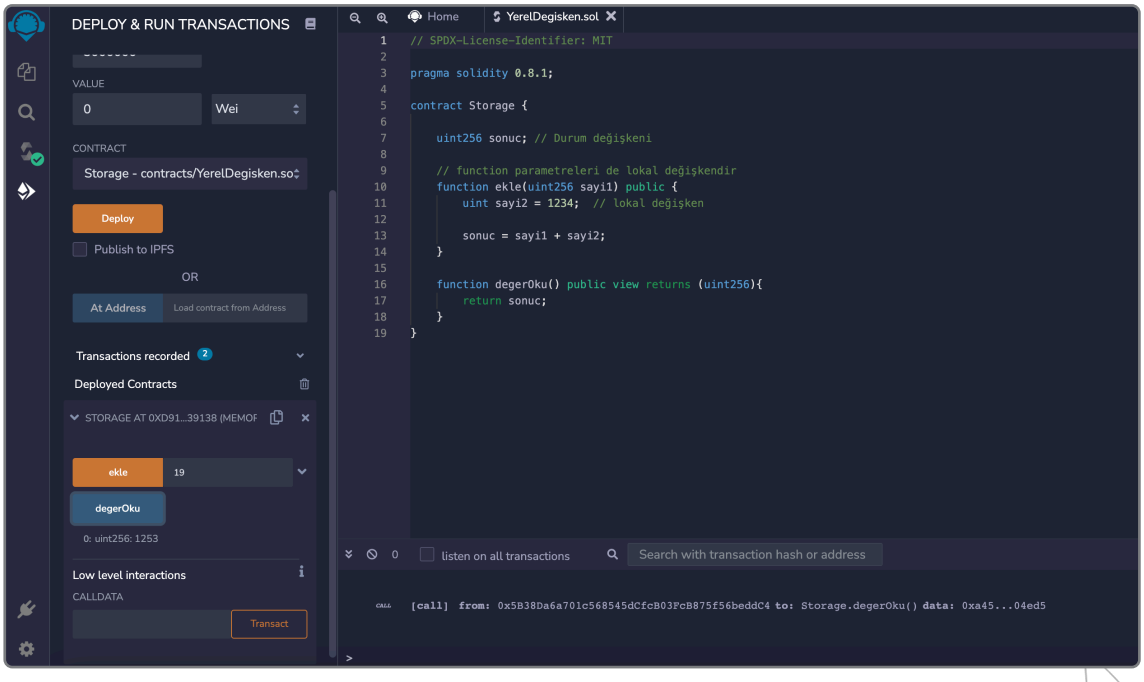
Solidty dilinde yerel değişken kullanılan uygulamayı kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde yerel değişken kullanıldığı uygulamayı aşağıdaki gibi kodlayınız (Görsel 8.45).

```
// SPDX-License-Identifier: MIT

pragma solidity 0.8.1;

contract Storage {

   uint256 sonuc; // Durum değişkeni

   // function parametreleri de lokal değişkendir
   function ekle(uint256 sayi1) public {
      uint sayi2 = 1234;  // lokal değişken

      sonuc = sayi1 + sayi2;
   }

   function degerOku() public view returns (uint256){
      return sonuc;
   }
}
```

Görsel 8.45: Yerel değişken kullanım kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.46'daki şekilde çalıştırınız.



Görsel 8.46: Yerel değişken kullanımı

Genel Değişkenler (Global Variables): Genel değişkenler, blok zinciri ve işlem özellikleri hakkında bilgi sağlar. Örneğin

msg.sender: Mesajın göndericisi olan bir adres döndürür.

now: Geçerli blok zaman damgasını UNIX zaman damgası biçimi ile gösterir.

24. UYGULAMA

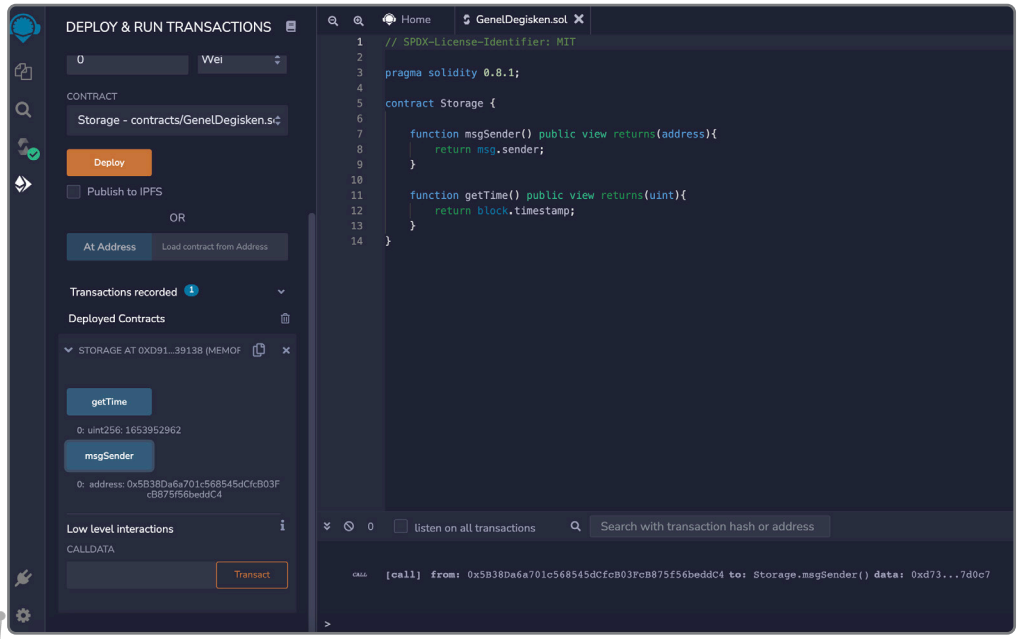
Solidty dilinde genel değişken kullanılan uygulamayı kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde genel değişken kullanılan uygulamayı aşağıdaki gibi kodlayınız (Görsel 8.47).

```
// SPDX-License-Identifier: MIT

pragma solidity 0.8.1;

contract Storage {

  function msgSender() public view returns(address){
     return msg.sender;
  }

  function getTime() public view returns(uint){
     return block.timestamp;
  }
}
```

Görsel 8.47: Genel değişken kullanım kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.48'deki şekilde çalıştırınız.



Görsel 8.48: Genel değişken kullanımı

Fonksiyonlar (functions): Fonksiyonlar, sözleşmedeki çalıştırılabilir kod yapılarıdır. Fonksiyonların dört erişim değiştiricisi vardır.

public (genel): public olarak tanımlanan fonksiyona herkes tarafından erişebilir.

private (özel): private fonksiyon, sözleşmenin kapsamı ile sınırlıdır.

internal (dahili): internal olarak bildirilen bir işleve sözleşmenin kendisi ve alt sözleşme tarafından erişilebilir.

external (harici): external olarak bildirilen bir fonksiyona, devralan bir sözleşmeyle erişilemez.

25. UYGULAMA

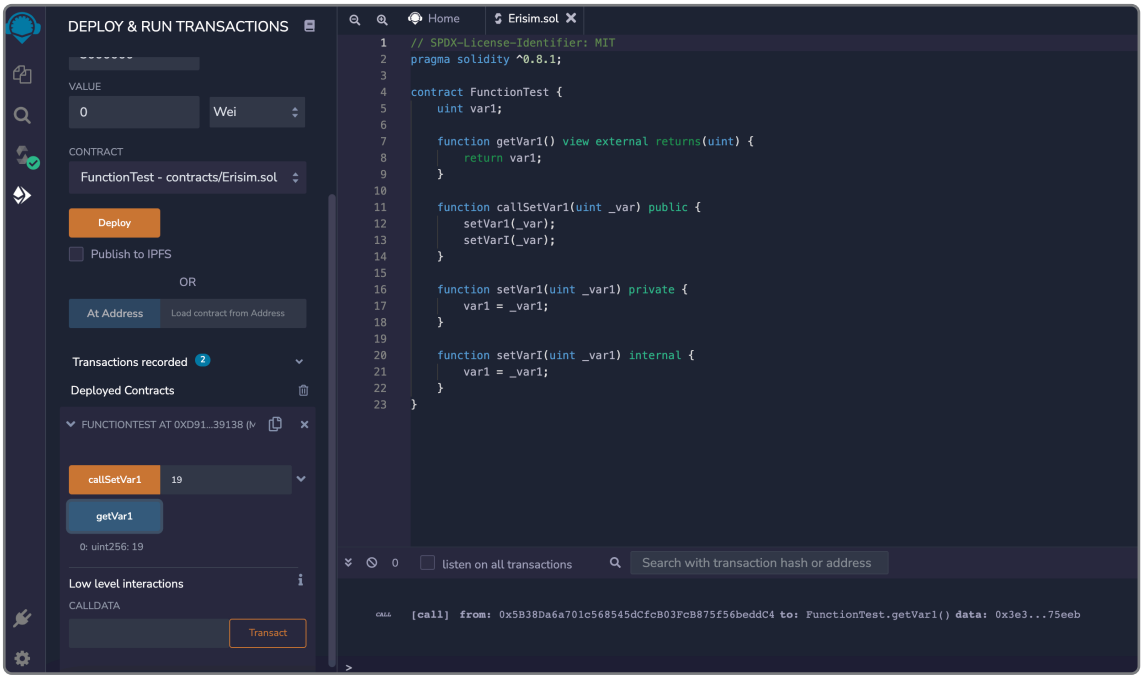
Solidty dilinde bir fonksiyonun genel, özel, dâhilî veya haricî olarak tanımlandığı uygulamayı kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde bir fonksiyonun genel, özel, dâhilî veya haricî olarak tanımlandığı uygulamayı aşağıdaki gibi kodlayınız (Görsel 8.49).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.1;

contract FunctionTest {
   uint var1;

   function getVar1() view external returns(uint) {
      return var1;
   }

   function callSetVar1(uint _var) public {
      setVar1(_var);
      setVarl(_var);
   }

   function setVar1(uint _var1) private {
      var1 = _var1;
   }

   function setVarl(uint _var1) internal {
      var1 = _var1;
   }
}
```

Görsel 8.49: Fonksiyon tanımlama kodları

2. Adım: Kodladığınız uygulamayı Görsel 8.50'deki şekilde çalıştırınız.



Görsel 8.50: Fonksiyon kullanımı

Fonksiyon Değiştiriciler (Function Modifiers): Fonksiyon değiştiriciler, bir fonksiyona ilave işlevsellik kazandırmak için kullanılır. Örneğin bir fonksiyonu çalıştırmadan önce, bir koşulu otomatik olarak kontrol etmek için bir değiştirici kullanılabilir.

Yapı Tipleri (Struct Types): Yapılar, birkaç değişkeni gruplayabilen kullanıcı tanımlı veri tipleridir.

Enum Türleri (Enums Types): Numaralandırmalar, sonlu sabit değerler kümesiyle kullanıcı tanımlı veri türleri oluşturmak için kullanılabilir.

Events (Olaylar): Olaylar, EVM günlüğü işlevleriyle etkileşim kurmak için kullanılır.

8.2. İLERİ DÜZEY AKILLI KONTRATLARDA HATA AYIKLAMA

Akıllı sözleşmelerde hata yönetimi için **metot** kullanılır. Bunlar:

assert(bool koşulu): Dâhilî hataları ayıklamak için kullanılır.

require(bool koşulu): Girişlerdeki veya haricî bileşenlerdeki hata kontrolü için kullanılır.

require(bool koşulu, string memory mesajı): Girişlerdeki veya haricî bileşenlerdeki hataların kontrolü için kullanılır ve hata mesajı verir.

revert(): Yürütmeyi durdurur ve durum değişikliklerini geri alır.

revert(string memory mesajı): Açıklayıcı bir dizi sağlayarak yürütmeyi iptal etmek ve durum değişikliklerini geri almak için kullanılır. Örneğin yanlış adrese Ethereum gönderme işlemi kontrolü için kullanılabilir.

26. UYGULAMA

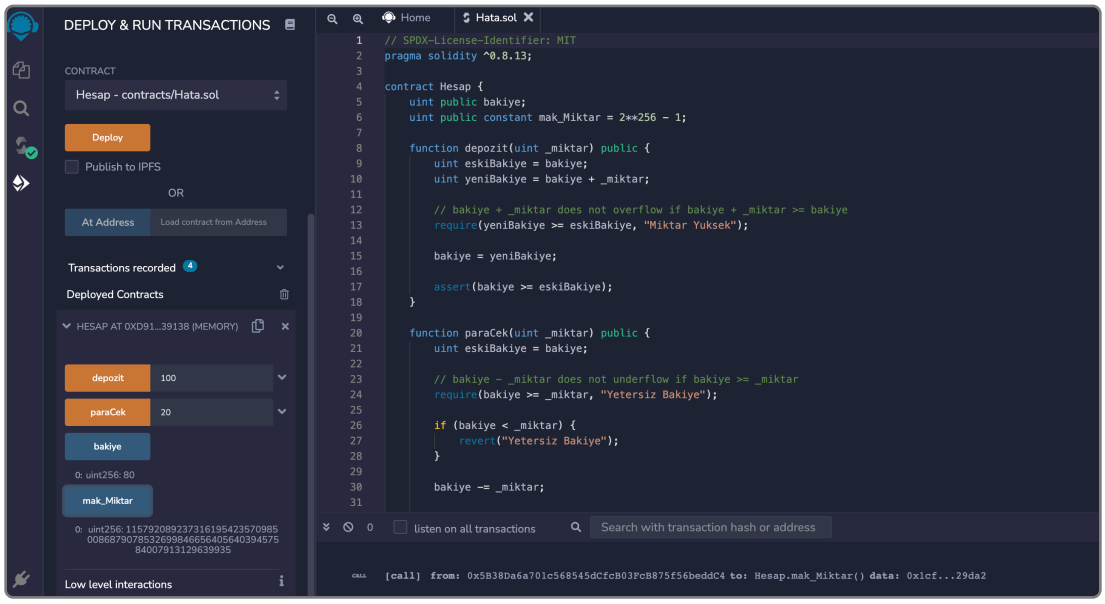
Solidty dilinde hata kontrol fonksiyonlarının kullanıldığı uygulamayı kodlama ve çalıştırma işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Solidty dilinde solidty dilinde hata kontrol fonksiyonlarının kullanıldığı uygulamayı aşağıdaki gibi kodlayınız (Görsel 8.51).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.13;

contract Hesap {
    uint public bakiye;
    uint public constant mak_Miktar = 2**256 - 1;
    function depozit(uint _miktar) public {
        uint eskiBakiye = bakiye;
        uint yeniBakiye = bakiye + _miktar;
        // bakiye + _miktar does not overflow if bakiye + _miktar >= bakiye
        require(yeniBakiye >= eskiBakiye, "Miktar Yuksek");
        bakiye = yeniBakiye;
        assert(bakiye >= eskiBakiye);
    }
    function paraCek(uint _miktar) public {
        uint eskiBakiye = bakiye;
        // bakiye - _miktar does not underflow if bakiye >= _miktar
        require(bakiye >= _miktar, "Yetersiz Bakiye");
        if (bakiye < _miktar) {
            revert("Yetersiz Bakiye");
        }
        bakiye -= _miktar;
        assert(bakiye <= eskiBakiye);
    }
}
```

Görsel 8.51: Hata kontrol fonksiyonları kodları

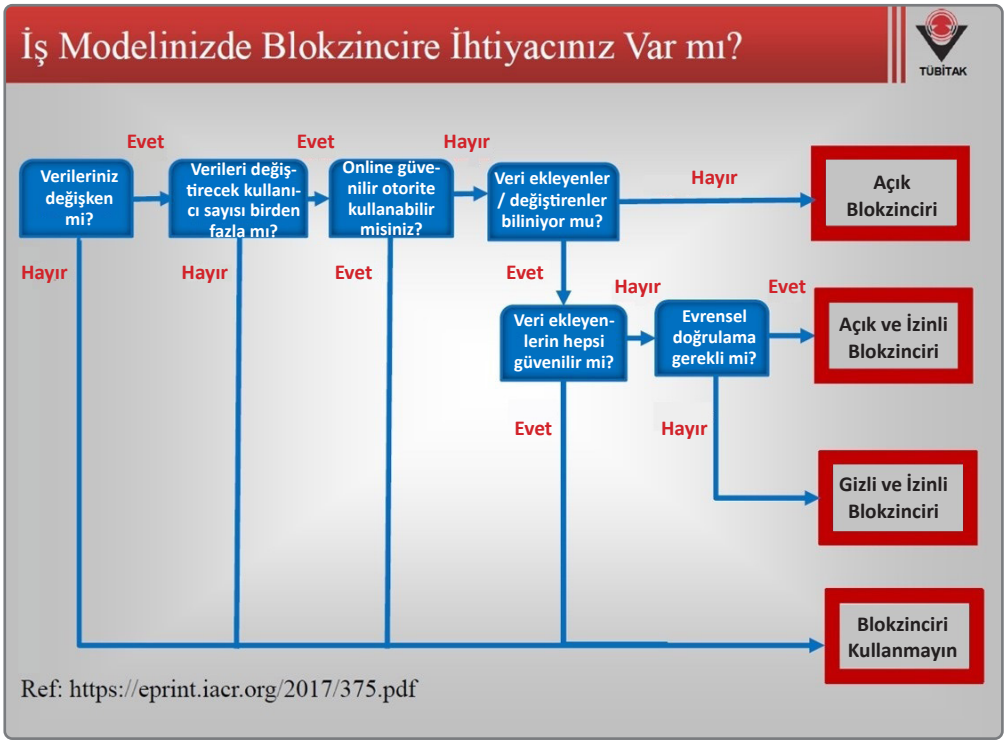
2. Adım: Kodladığınız uygulamayı Görsel 8.52'deki şekilde çalıştırınız.



Görsel 8.52: Hata kontrol fonksiyonları

8.3. ETHEREUM'DA AKILLI KONTRAT DİZAYNI

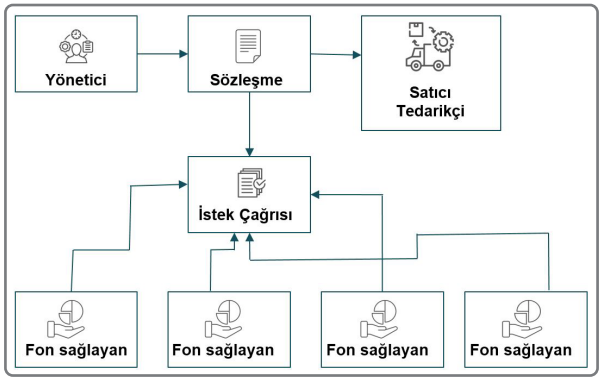
Blok zinciri projeleri geliştirmeden önce geliştirilecek projede blok zinciri gereksinimi olup olmadığı netleştirilmelidir. Blok zinciri ile proje geliştirmeden önce Görsel 8.53'te verilen blok zinciri teknolojisi kullanım gereksinimi akış şeması incelenmelidir.



Görsel 8.53 : Blok zinciri teknolojisi kullanım gereksinimi

8.4. ETHEREUM'DA AKILLI KONTRAT YAZIMI

"Akıllı Kontrat ile Kitle Fonlaması" projesinde, yönetici kitle kaynağını oluşturmak amacıyla bir araya gelen katılımcılardan fon talebi için bir akıllı sözleşme oluşturulur (Görsel 8.54). Yönetici katılımcıların çoğunluğunun (%50'sinden fazlasının) onaylaması gereken bir çağrı açar. Bir harcama talebi talep edildiğinde talep ile ilgili bilgiler yöneticiye gönderilmeden önce struct (yapısı) içine yazılır. Sonrasında ise onaylanarak satıcı hesabına gönderilir. Katılımcıların fonlarının sorumluluğu ve güvenliği garanti edilir.



Görsel 8.54: Kitle fonlaması işlem diyagramı

27. UYGULAMA

Kitle fonlamasına ilişkin akıllı sözleşmeyi Remix IDE kullanarak kodlama işlemini verilen adımlar doğrultusunda gerçekleştiriniz.

1. Adım: Kitle fonlamasına ilişkin akıllı sözleşmeyi Remix IDE kullanarak aşağıdaki gibi kodlayınız (Görsel 8.55).

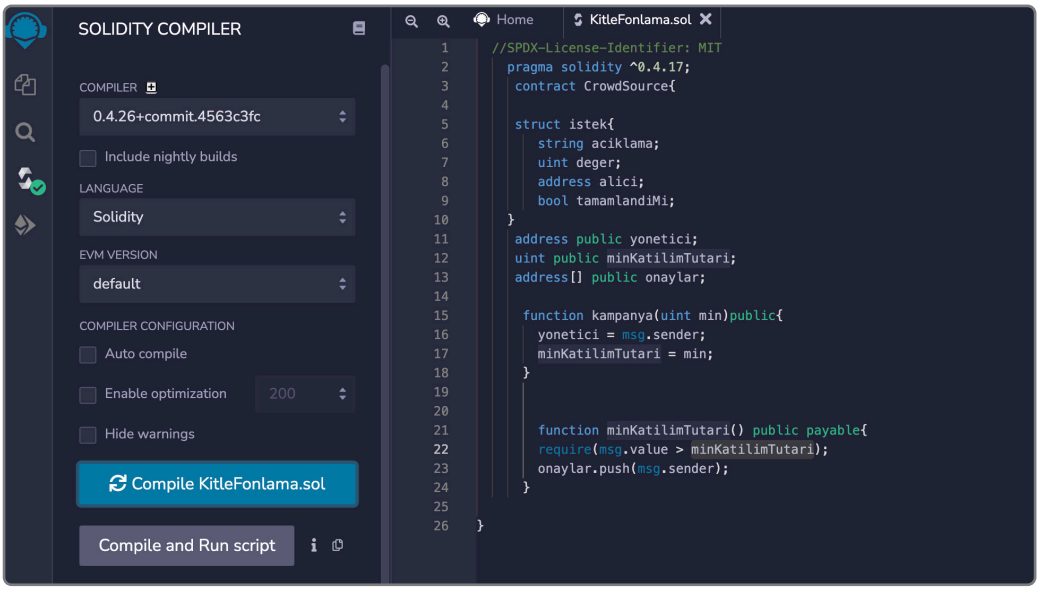
```
//SPDX-License-Identifier: MIT
pragma solidity ^0.4.17;
 contract CrowdSource{

 struct istek{
    string aciklama;
    uint deger;
    address alici;
    bool tamamlandiMi;
 }
 address public yonetici;
 uint public minKatilimTutari;
 address[] public onaylar;

  function kampanya(uint min)public{
    yonetici = msg.sender;
    minKatilimTutari = min;
  }
    function minKatilimTutari() public payable{
    require(msg.value > minKatilimTutari);
    onaylar.push(msg.sender);
  }

}
```

Görsel 8.55: Remix IDE ile akıllı sözleşme kodlama

8.5. ETHEREUM PROJESİNİ TEST ETME

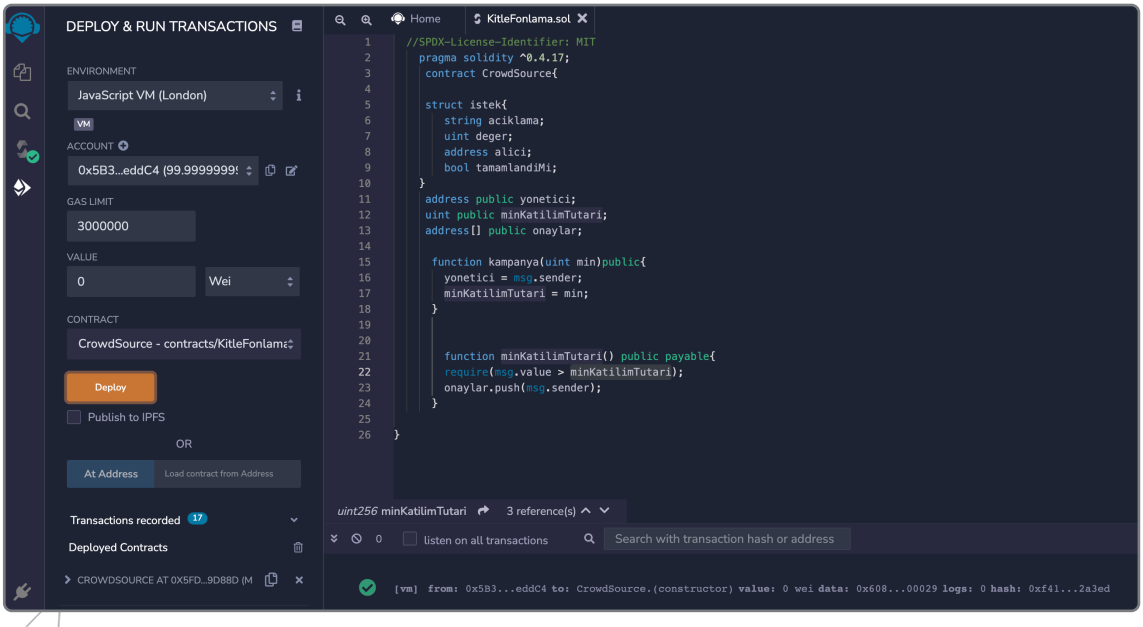
Akıllı sözleşmenin testi aşağıdaki aşamaları izleyerek gerçekleştirilir.

1. Aşama: Akıllı kontrat **Compile** edilir. Görsel 8.56'da görüldüğü üzere Compile sonrasında hata olmadığına dair yeşil tik işareti görülür.



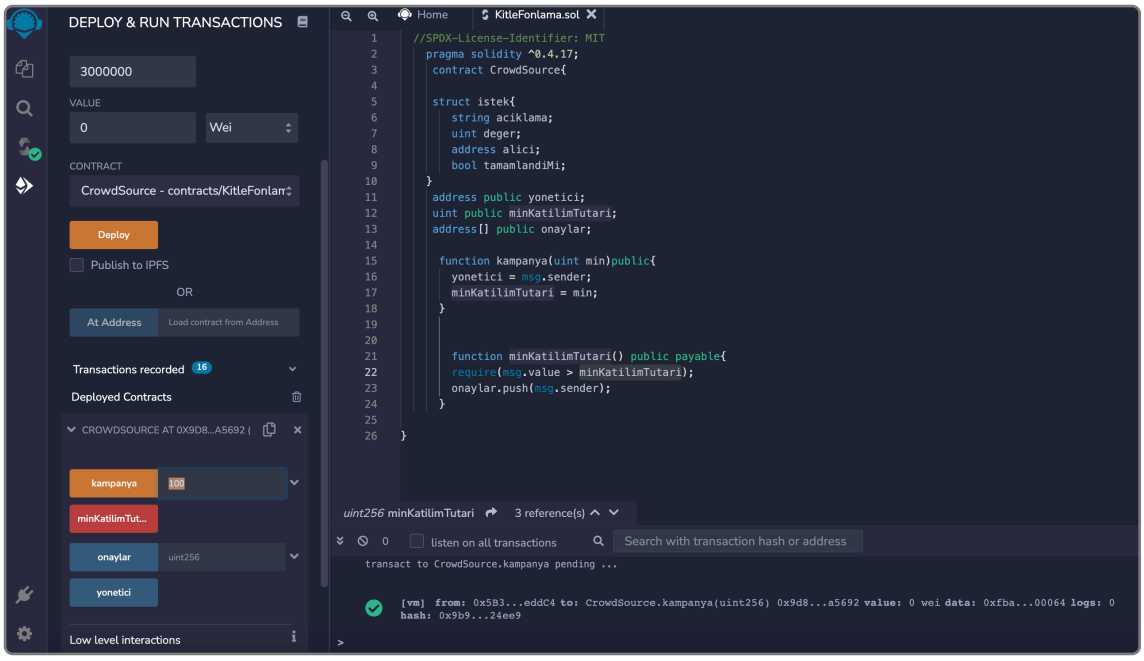
Görsel 8.56: Akıllı sözleşmenin Compile süreci

2. Aşama: Akıllı sözleşmenin **Deploy** edilir. Deploy sonrasında hata olmadığı ve gerçekleşen işlem (transaction) için yeşil tik işareti görülür (Görsel 8.57).



Görsel 8.57: Akıllı sözleşmenin Deploy işlemi

3. Aşama: Görsel 8.58'de görüldüğü üzere Deployed Contracts bölümü açılır, kampanya alanına 100 girilerek kampanya butonuna tıklanır. Sonrasında gerçekleşen işlem (transaction) için yeşil tik işareti görülür.



Görsel 8.58: Kitle fonlaması test işlemi

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

1. Aşağıdakilerden hangisi akıllı sözleşmenin bakiyesini almak için kullanılan sözdizimidir?

A) self.balance;

B) this.balance;

C) payable(this).balance;

D) address(this).balance;

E) payable.balance;

2. Kullanıcı tanımlı değişkenler oluşturmak için aşağıdaki seçeneklerden hangisi kullanılabilir?

A) bool B) enum C) string

D) struct E) uint

3. "Remix bir IDE'dir." **ifadesine göre IDE'nin anlamı aşağıdakilerden hangisidir?**

A) Internal Development Environment

B) Integrated Development Environment

C) Integrated DApp Environment

D) Integrated Development Enterprise

E) Integrated Development Engine

4. Aşağıdakilerden hangisi kullanıcı tanımlı veri türleri oluşturmak için kullanılır?

A) address B) enum C) mapping

D) struct E) type

5. Aşağıdakilerden hangisi fonksiyonların erişim değiştiricisi olarak <u>kullanılmaz</u>?

A) public (genel)

B) private (özel)

C) internal (dahili)

D) external (harici)

E) integrated (bütünleşik)

KONULAR

9.1. DAO (MERKEZİYETSİZ OTONOM ORGANİZASYON)

9.2. MERKEZİYETSİZ UYGULAMALAR

9.3. BLOK ZİNCİRİ HUKUKU

NELER ÖĞRENECEKSİNİZ?

- Merkeziyetsiz Otonom Organizasyon kavramını açıklama
- Merkeziyetsiz uygulamaları açıklama
- Blok zinciri hukukunu açıklama

ANAHTAR KELİMELER

Blok zinciri, dağıtık defter, Hash, madencilik

HAZIRLIK ÇALIŞMALARI

1. Otonom olarak çalışan sistemler hakkındaki düşüncelerinizi sınıf arkadaşlarınızla paylaşınız.

2. Hukuk kuralları olmasaydı yaşam nasıl olurdu? Fikirlerinizi arkadaşlarınızla paylaşınız.

MERKEZİYETSİZ ORGANİZASYONLAR

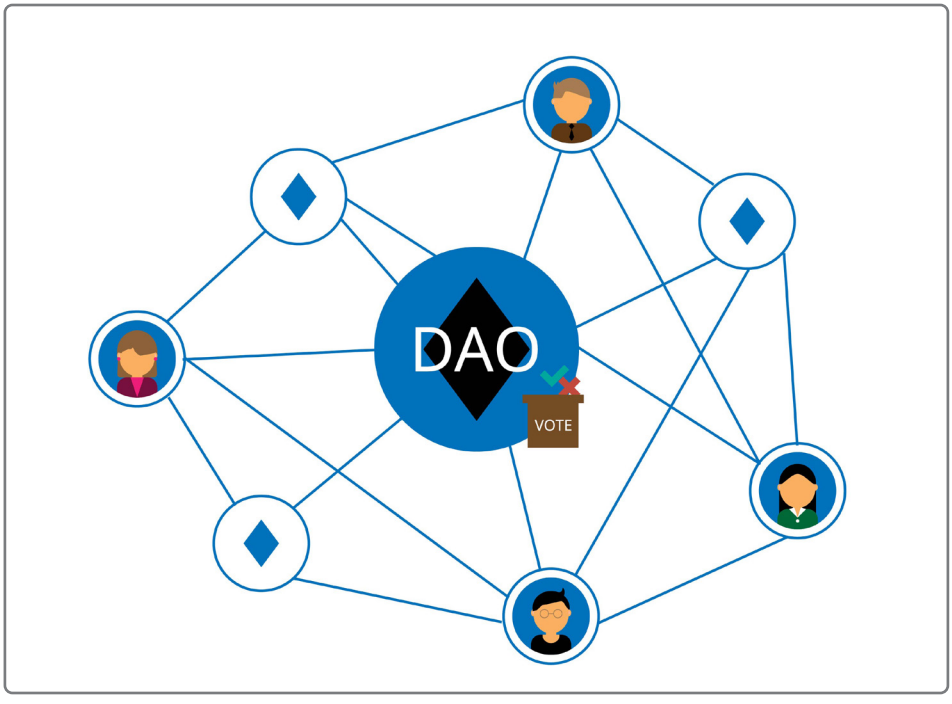
9. ÖĞRENME BİRİMİ

9.1. DAO (MERKEZİYETSİZ OTONOM ORGANİZASYON)

Merkeziyetsiz yani herhangi bir yönetim ve kontrol merkezi olmadan, kendi kendini organize ederek yöneten, idare eden; topluluk, kuruluş veya organizasyonlara **DAO [Decentralized Autonomous Organization (Merkeziyetsiz Otonom Organizasyon)]** denir. DAO, geniş kapsamlı blok zinciri tabanlı oluşumlardan biridir. DAO, Ethereum ağını kullanır ve herhangi bir merkezî otoriteyle bağlantısı bulunmamaktadır.

Geliştiriciler tarafından kararlar otomatikleştirilerek; işlemleri kolaylaştırmak, insan hatasını, özelikle yatırımcı fonlarının manipülasyonunu ortadan kaldırmak amacıyla DAO geliştirilmiştir. Açık kaynak kodlu, merkezî bir otoriteye bağlı olmadan topluluk tarafından yönetilen özerk ve şeffaf bir yapıya sahiptir. Akıllı sözleşmelerle topluluk tarafından anlaşmaya varılan kurallar uygulanır, oylamalar yapılır. Topluluk tarafından belirlenen tüm bu kuralların ve işleyişin barındığı kodlar, halka açık bir şekilde sunulur.

Hiyerarşik yapıların aksine DAO'nun her bir üyesi işleyiş protokolünü denetleyebilir. Blok zincirine DAO'nun protokolünü oluşturacak bu kurallar kaydedilir. Kurallar genelde katılımcı üyelerin oyları sayesinde belirlenir. DAO'larda karar almak için genellikle teklifler kullanılır (Görsel 9.1). Bir teklif katılımcıların çoğunluğu tarafından kabul oyu alırsa (ya da ağ mutabakat kurallarında belirlenmiş diğer bir kural grubuna uygunsa) uygulamaya konur ve blok zincirine kaydedilir.



Görsel 9.1: DAO (Merkeziyetsiz Otonom Organizasyon)

9.2. MERKEZİYETSİZ UYGULAMALAR

Merkeziyetsiz uygulamalar [Decentralised Applications (DApps)], merkezî olmayan bir ağ üzerinde çalışan uygulamalardır. DApp, merkezî olmayan uygulamalar ve sistemler bütününü ifade eder. Merkezî bir otorite tarafından denetlenmez ve ağ üzerinde işlemleri kontrol eden herhangi bir merkez yoktur. Verilerin kontrolü sistemdeki kullanıcılar tarafından yapılır. Merkeziyetsiz uygulamalar, tek bir sunucuda barındırılmak yerine ağa bağlı tüm düğümlerde çalışır ve açık kaynak kodlu yapıya sahiptir. Blok zinciri teknolojisinden faydalanan açık kaynak kodlu uygulamalar, yazılımlar DApp uygulamalarıdır (Görsel 9.2).

Görsel 9.2: DApps (Decentralised applications)

DApp, dağıtılmış bir sistem üzerinde çalışan bir protokoldür. Merkezî olan uygulamalarda olduğu gibi üçüncü bir taraf işleme dâhil olamaz. Merkezî olmayan platformlarda bir aracı olmaması nedeniyle **eşler arası ağ [Peer to Peer (P2P)]** olarak da adlandırılabilir (Görsel 9.3).

Apps	DApps
	

Görsel 9.3: Apps ve DApps karşılaştırma

DApp'ın özellikleri şunlardır:

• **Açık Kaynak Kodlu**

Merkezî olmayan uygulamaların kodları herkese açık bir yapıdadır ve herkes tarafından görülüp kullanılabilir. Uygulamada oluşacak bütün değişikliklere fikir birliği ya da kullanıcılarının çoğunun oyuyla karar verilmektedir.

• **Merkeziyetsiz**

Merkeziyetli bir yapı olmanın tehlikelerini önleyebilmek için uygulamada gelişen tüm işlemlerin kayıtlarının hepsi yine merkeziyetsiz bir blok zinciri ağındaki tüm düğümlerde depolanır.

• **Ağ Teşvikleri**

Blok zinciri doğrulayan kişilerin belirlenmiş bir değerle ödüllendirilmesiyle teşvik sağlanır.

• **Algoritması**

Merkezî olmayan uygulama, bu uygulamanın veya yapılan işlemin değerini kanıtlayabilecek özel bir kriptografik algoritma üstüne çalışmalı, anlaşmalıdır.

Merkeziyetsiz uygulamalarda akıllı sözleşme, tanımlanan algoritma doğrultusunda otonom olarak çalışır. Kullanıcılar merkeziyetsiz uygulamanın (DApps) gerçekleştirdiği işlemin, blok zinciri üzerinde güvenli biçimde gerçekleştiğini ve kalıcı biçimde kaydedildiğini görür. Merkeziyetsiz uygulamalar; özellikle merkeziyetsiz finans (DeFi) uygulamaları, kripto para borsaları, oyunlar, sosyal medya ve market uygulamaları gibi sektörlerde kullanılır (Görsel 9.4).

Görsel 9.4: DeFi (Merkeziyetsiz finans uygulamaları)

SIRA SİZDE

Küçük gruplar oluşturarak merkeziyetsiz uygulamaların (DApps) gelecekteki kullanımlarıyla ilgili bir sunum hazırlayınız. Sunumunuzu grup sözcünüz aracılığı ile sınıfta arkadaşlarınızla paylaşınız.

9.3. BLOK ZİNCİRİ HUKUKU

Blok zinciri teknolojisi yakın bir gelecekte birçok sektörde değişime ve gelişime sebep olacaktır. Bu teknolojinin yaygınlaşmasıyla birlikte, sadece teknik boyutuyla değil aynı zamanda şirketlerin ve kamusal alanlarda kullanımının insanlara, şirketlere ve kamuya etkilerinin hukuki düzenlemelerinin yapılması gerekir (Görsel 9.5).

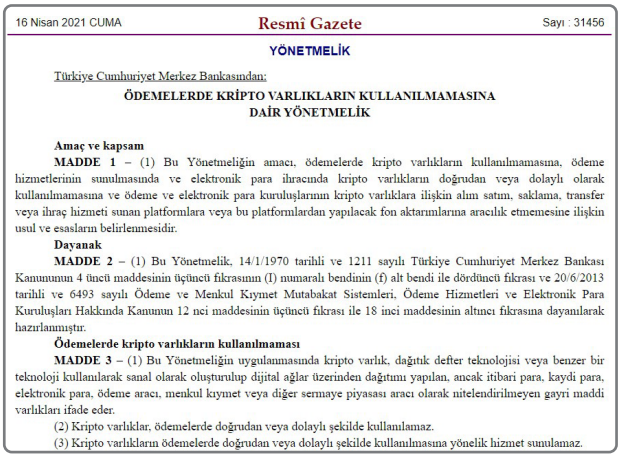
Görsel 9.5: Blok zinciri hukuku

Hukuk, TDK'ye göre "Toplumu düzenleyen ve devletin yaptırım gücünü belirleyen yasaların bütünü" olarak tanımlanır. Buna göre devletler, kanun koyucu olarak gerekli yasal düzenlemeleri yapar. Hukuki düzenlemeler, blok zinciri teknolojisinin genel hayata entegrasyonunun sağlanması açısından çok önemlidir. Diğer hukuki olaylarda olduğu gibi blok zinciri teknolojisinde de devletler arasında farklı uygulamalar bulunur.

Yapılan birçok hukuki düzenleme, blok zinciri teknolojisine yönelik değil blok zinciri uygulama alanlarına yöneliktir. Bu uygulama alanlarından en çok kripto varlıklar, kripto para birimleri, müşteri tanıma, ödeme sistemleri ve kara para aklamayla mücadele gibi belirli yasal meseleler üzerinedir.

Kripto varlıkların merkeziyetsiz olması vergilendirmenin nasıl olacağından kara para aklama gibi birçok sorunu beraberinde getirir. Merkeziyetsiz veri tabanı olan blok zinciri teknolojisinin hedefine ulaşabilmesi için uluslararası uyumun sağlanması ve bu bağlamda hukuki düzenlemelerin çıkarılması önemlidir.

Türkiye Cumhuriyet Merkez Bankası tarafından 16 Nisan 2021 tarihinde 31456 sayılı Resmî Gazete'de "Ödemelerde Kripto Varlıkların Kullanılmamasına Dair Yönetmelik" yayımlanmıştır (Görsel 9.6). Yönetmelikte kripto varlıklar; "dağıtık defter teknolojisi veya benzer bir teknoloji kullanılarak sanal olarak oluşturulup dijital ağlar üzerinden dağıtımı yapılan, ancak itibari para, kaydi para, elektronik para, ödeme aracı, menkul kıymet veya diğer sermaye piyasası aracı olarak nitelendirilmeyen gayri maddi varlıklar" olarak tanımlamıştır. Böylece ülkemizde kripto varlıkların hangi niteliğe sahip olmadıkları açıkça belirlenmiştir.



16 Nisan 2021 CUMA — Resmî Gazete — Sayı : 31456

YÖNETMELİK

Türkiye Cumhuriyet Merkez Bankasından:

ÖDEMELERDE KRİPTO VARLIKLARIN KULLANILMAMASINA DAİR YÖNETMELİK

Amaç ve kapsam

MADDE 1 – (1) Bu Yönetmeliğin amacı, ödemelerde kripto varlıkların kullanılmamasına, ödeme hizmetlerinin sunulmasında ve elektronik para ihracında kripto varlıkların doğrudan veya dolaylı olarak kullanılmamasına ve ödeme ve elektronik para kuruluşlarının kripto varlıklara ilişkin alım satım, saklama, transfer veya ihraç hizmeti sunan platformlara veya bu platformlardan yapılacak fon aktarımlarına aracılık etmemesine ilişkin usul ve esasların belirlenmesidir.

Dayanak

MADDE 2 – (1) Bu Yönetmelik, 14/1/1970 tarihli ve 1211 sayılı Türkiye Cumhuriyet Merkez Bankası Kanununun 4 üncü maddesinin üçüncü fıkrasının (I) numaralı bendinin (f) alt bendi ile dördüncü fıkrası ve 20/6/2013 tarihli ve 6493 sayılı Ödeme ve Menkul Kıymet Mutabakat Sistemleri, Ödeme Hizmetleri ve Elektronik Para Kuruluşları Hakkında Kanunun 12 nci maddesinin üçüncü fıkrası ile 18 inci maddesinin altıncı fıkrasına dayanılarak hazırlanmıştır.

Ödemelerde kripto varlıkların kullanılmaması

MADDE 3 – (1) Bu Yönetmeliğin uygulanmasında kripto varlık, dağıtık defter teknolojisi veya benzer bir teknoloji kullanılarak sanal olarak oluşturulup dijital ağlar üzerinden dağıtımı yapılan, ancak itibari para, kaydi para, elektronik para, ödeme aracı, menkul kıymet veya diğer sermaye piyasası aracı olarak nitelendirilmeyen gayri maddi varlıkları ifade eder.

(2) Kripto varlıklar, ödemelerde doğrudan veya dolaylı şekilde kullanılamaz.

(3) Kripto varlıkların ödemelerde doğrudan veya dolaylı şekilde kullanılmasına yönelik hizmet sunulamaz.

Görsel 9.6: Ödemelerde Kripto varlıkların kullanılmamasına dair yönetmelik

Kripto Varlık Yönetmeliği'nde kripto varlıkların ödemelerde doğrudan veya dolaylı biçimde kullanılamayacağı açıkça hükme bağlanmıştır. Bu hükme göre kripto varlıklar para, kredi kartı gibi ödeme aracı olarak kullanılamayacağı belirlenmiştir. Bu, Yönetmelik'in kripto varlıkları yasakladığını belirtmez. Sadece doğrudan veya dolaylı olarak alışveriş işlemlerinde kullanılamayacağını belirtir.

SIRA SİZDE

Ülkemiz ve diğer ülkelerdeki blok zinciri hukuku ile ilgili poster hazırlayınız. Sınıfınızda uygun bir yerde posterlerinizi asarak küçük bir sergi yapınız.

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

1. I. Ethereum ağını kullanır.

II. Kontrol merkezi yoktur.

III. Yönetim merkezi vardır.

IV. Otonom olarak çalışır.

Yukarıdakilerden hangileri DAO'nun özelliklerindendir?

A) I-II B) II-III C) II-IV D) I-II-IV E) I-III-IV

2. I. Açık kaynak kodludur.

II. Merkezî çalışır.

III. P2P kullanır.

IV. Blok zinciri teknolojisini kullanır.

Yukarıdakilerden hangileri DApss'ın özelliklerindendir?

A) I-II B) I-II-III C) II-IV D) I-II-IV E) I-III-IV

3. Aşağıdakilerden hangisi DApss'ın özelliklerinden <u>değildir</u>?

A) Merkezî açık bir yapıdadır.

B) Değişiklikler fikir birliğiyle yapılır.

C) Kriptografik algoritmaya sahiptir. ,

D) Ağ teşvik sistemi vardır.

E) Kodları herkese açık bir yapıdadır.

4. Aşağıdakilerden hangisi Dapss'ın uygulamalarından <u>değildir</u>?

A) Merkeziyetsiz finans (DeFi) uygulamaları B) Menkul kıymetler borsaları

C) Kripto para borsaları D) Blok zinciri tabanlı oyunlar

E) Tedarik takip uygulamaları

5. Ödemelerde Kripto Varlıkların Kullanılmamasına Dair Yönetmelik hükümlerine göre kripto varlıklarla ilgili aşağıdakilerden hangisi söylenebilir?

A) Ödemelerde dolaylı kullanılır. B) Menkul kıymet olarak kullanılır.

C) Ödeme aracı olarak kullanılmaz. D) Elektronik para olarak kullanılır.

E) Sanal para olarak kullanılır.

KONULAR

10.1. BLOK ZİNCİRİ VE WEB 3.0

10.2. İHTİYAÇ ANALİZİ

10.3. PROJE YAZIM KURALLARI

10.4. ICO, IEO, ITO

10.5. KİMLİK YÖNETİMİ

10.6. BLOK ZİNCİRİ VE İŞ DÜNYASININ DÖNÜŞÜMÜ

NELER ÖĞRENECEKSİNİZ?

- Web 1.0, Web 2.0 ve Web 3.0 Kavramları
- Blok Zinciri Teknolojisinin İhtiyaç Analizi
- White Paper Kavramı
- ICO, IEO, ITO Kavramları
- Dijital Kimlik Kavramı
- Blok Zinciri ve İş Dünyasının Dönüşümü

ANAHTAR KELİMELER

Dijital kimlik, web 1.0, web 2.0, web 3.0, White paper

HAZIRLIK ÇALIŞMALARI

1. Bir uygulamaya ihtiyacınız olup olmadığına nasıl karar verirsiniz? Fikirlerinizi arkadaşlarınızla paylaşınız.

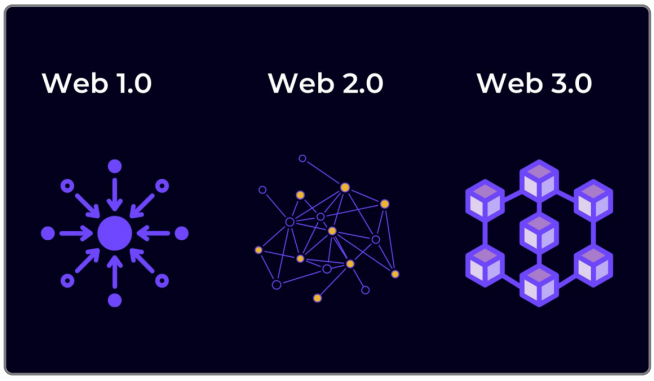
2. Gelecekte internetten beklentileriniz nelerdir? Düşüncelerinizi arkadaşlarınızla paylaşınız.

BLOK ZİNCİRİ GİRİŞİMCİLİĞİ

10.
ÖĞRENME BİRİMİ

10.1. BLOK ZİNCİRİ VE WEB 3.0

İnternet, ilk kullanılmaya başladığından şimdiye kadar büyük oranda değişmiş ve gelecekte de değişmeye devam edecektir. İnternet, özellikle world wide web (www) platformunun geliştirilmesiyle daha da yaygınlaşır. Kullanıcılarına daha pasif ve sınırlı bir kullanım imkânı sunan başlangıç dönemi **Web 1.0** olarak adlandırılır. **Web 2.0** olarak adlandırılan döneme geçilmesiyle birlikte kullanıcılar, daha aktif hâle gelir. Günümüzde ise yapay zekâ, makine öğrenmesi, blok zinciri gibi teknolojilerle **Web 3.0** dönemi başlamıştır (Görsel 10.1).



Görsel 10.1: Web platformları

Web 1.0: 1990'larda kullanıma açılan ilk ve en güvenilir fakat statik HTML sayfalarına sahip, pasif kullanıcılı bir internet yapısıdır. Sınırlı ve kullanıcıya doğru tek yönlü bir veri aktarımı vardır. Kullanıcı tarafında neredeyse hiç etkileşim imkânı yoktur. Kullanıcılara özel sayfa oluşturulması, yazılara yorum yapılması gibi seçenekler bulunmaz. Bir başka deyişle sadece okuma amaçlıdır. İçerikler ise belirli kişilerce oluşturulabilir.

Web 2.0: Kullanıcılar, Web 2.0 ile birlikte aktif bir şekilde içerik üretimine dâhil olup; oluşturulan içeriğe yorum yapabilme, düzenleme gibi işlemleri yapmaya başlar. Kullanıcılar içeriklerin çoğunu etkileşimli sosyal web araçlarını kullanarak oluşturmaya başlar. Artık HTML yerine JavaScript, HTML5 ve CSS3 gibi web teknolojileri kullanılmaya başlanır. İnternet, bu gelişmeler sayesinde daha etkileşimli hâle gelir.

Web 3.0: Anlamsal veya semantik web olarak da adlandırılır. İçerik oluşturma ve karar verme süreçleri, sadece kullanıcı insanlar tarafından değil; yapay zekâ, makine öğrenmesi, blok zinciri gibi teknolojileri kullanarak, otomatik olarak makineler tarafından gerçekleşmeye başlar. Web 3.0 ile içeriklerin kullanıcıların tercihlerine göre uyarlanarak, internete bağlanan her kişiye özel akıllı içerik oluşturulması ve dağıtılması sağlanır. Bu teknolojilerin gelişimiyle artık, bu verinin tamamı kullanıcı deneyimini iyileştirmek ve webi daha kişiselleştirilmiş hâle getirmek için birtakım algoritmalar tarafından kullanılacaktır. Web 3.0; blok zinciri, açık kaynak kod yazılım, sanal gerçeklik, nesnelerin interneti (IoT) ve bunun gibi birçok eşler arası (P2P) teknolojiyi ileriye taşıyabilecek bir gelişmedir.

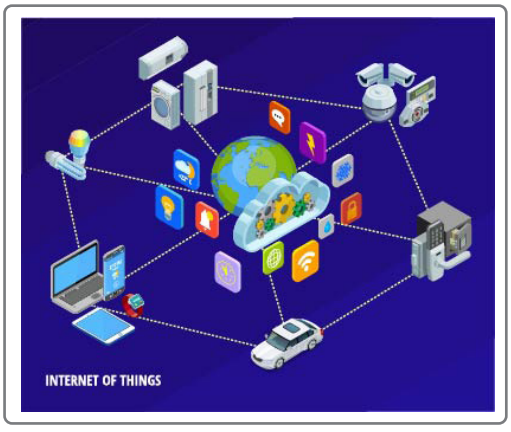
SIRA SİZDE

Web platformlarının özelliklerini karşılaştıran bir sunum hazırlayınız. Hazırladığınız sunumu sınıfta arkadaşlarınızla paylaşınız.

Web 3.0'dan önce birçok uygulama, genelde tek bir işletim sisteminin üzerinde çalışabilmekteydi. Web 3.0'da ise uygulamalar herhangi bir ek geliştirme masrafı olmadan birçok farklı türde ve alanda donanım ve yazılım üzerinde çalışabilmektedir. İnternet, Web 3.0 ile çok daha kullanıcı dostu ve merkeziyetsiz hâle gelmiştir. Mevcut internet yapısında kullanıcılar, sistemlerinden geçen verileri izleyebilmek için belirli merkezî ağ sunucularına ve hücresel sunuculara ihtiyaç duyar. Bu durum, blok zinciri teknolojilerinin geliştirilmesiyle merkeziyetsiz olacaktır. Bu ağların temel yeniliği, tek bir şirket veya kuruluşun kontrolünde olamayan, herkesin güvenebileceği platform ve ortamların oluşturulmasıdır. Blok zinciri teknolojisiyle birlikte Web 3.0 kullanıcıları, verileri üzerinde mutlak kontrol ve mülkiyet hakkına sahip olur.

10.1.1. Web 3.0 Uygulamaları

Web 3.0 uygulamalarının ortak özellikleri, genellikle büyük miktardaki verilerin işlenmesi, gerçek bilginin sunulması ve ihtiyaca uygun şekilde kullanılmasıdır. Zamanla bu uygulamalarda çok daha fazla güncelleştirme ve iyileştirme yapılacaktır. Web 3.0 kullanımı yaygınlaştıkça akıllı cihazların kullanımının artmasıyla da internet, günlük hayata daha entegre (bütünleşik) hâle gelecektir. Buzdolabı, fırın, elektrikli süpürge ve hatta otomobil gibi normalde çevrim dışı olan birçok nesnenin neredeyse hepsi; nesnelerin interneti (IoT) ekosisteminin bir parçası olacaktır (Görsel 10.2). Hem otonom sunucular hem de merkeziyetsiz uygulamalar (DApp) ile etkileşime girerek blok zinciri ve dijital varlık sistemlerinden faydalanacaktır.



Görsel 10.2: IoT ekosistemi

Akıllı telefonlarda kullanılan ses kontrollü yapay zekâ asistanı, Web 3.0 teknolojilerinden faydalanan uygulama örneği olarak gösterilebilir. Karmaşık ve kişiselleştirilmiş komutları yerine getirebilmek için yapay zekâ ile birlikte konuşma tanıma teknolojisi kullanılır. Yapay zekâlı asistanlar, "en yakın müze nerede" ya da "yarın sabah 9:30 için göz doktorumdan randevu al" gibi talepleri anlayabildiği gibi bunlarla ilgili gerekli işlemi gerçek zamanlı biçimde yapabilir.

Bir diğer uygulama örneği ise **gelişmiş arama motorlarıdır**. Bunlar, diğer geleneksel arama motorlarının yaptığı gibi web sayfalarının listesini çıkarmanın aksine soruları doğrudan hesaplama yoluyla yanıtlayan "hesaplama bilgi motoru" olarak çalışır. Geleneksel arama motoruyla "Türkiye

ve Azerbaycan" araması yapılsa hatta daha sonra aramaya "futbol" anahtar kelime olarak dâhil edilse bile son yapılan karşılaşma sonucunu verir. Gelişmiş arama motorlarında yapılan aramada ise iki ülke arasındaki birçok özelliklerin detaylı bir karşılaştırmasını sunar.

SIRA SİZDE

Blok zinciri teknolojileri kullanan Web 3.0 araçları hakkında bir poster hazırlayınız. Hazırlanan posterleri sınıfta uygun bir köşede sergileyiniz.

10.2. İHTİYAÇ ANALİZİ

Blok zinciri teknolojilerinin hızlı bir şekilde gelişip yaygınlaşması bu teknolojilerin tüm sektörlerde, kurumlarda kolaylıkla uygulanıp sektöre, kuruma katkı sağlayabileceği anlamına gelmez. Teknolojik gelişmeleri her zaman sektöre uyarlayıp kurumlar ya da şirketlerde uygulamak gerekli olmayabilir. Kurum ya da şirketler gelişen teknolojileri; araştırmalar, çalışmalar ve denemeler yaparak bunların sonuçlarına göre ihtiyaçlarını ve maliyetini değerlendirerek gerekli ise mevcut altyapılarına dâhil eder. Bu değerlendirmeleri yaparken ihtiyaç analizi, blok zinciri teknolojisine doğru şekilde tanımlanmalıdır. Bu ihtiyaç analizinde; ilgili firma, tedarik zinciri ve sektör bazında kullanım potansiyeli ölçülmelidir (Görsel 10.3). Gerekli fizibilite çalışmaları objektif ve doğru bir şekilde yapılmamış girişimler, başarısızlıkla sonuçlanır.

Görsel 10.3: İhtiyaç analizi

İhtiyaç analizini yaparken sektörde ve uygulamada blok zinciri teknolojisinin merkeziyetsiz, eşten eşe (aracısız), değişmez, anonim, izlenebilir, şeffaf, gizlilik ve güvenlik, maliyet, akıllı sözleşme gibi özelliklerinden ne kadar faydalanılabileceği ve projenin bunlara ihtiyacı belirlenmelidir.

Birçok sektörde ve uygulamada olduğu gibi blok zincirinin ihtiyaç analizinde en önemli ölçütlerden biri **maliyettir**. Genel olarak herhangi bir sektörde blok zinciri teknolojisinin uygulanması için maliyeti düşürme potansiyeline bakılır. Maliyetlerin azaltılmasında, özellikle blok zinciri temelli uygulamanın verimliliğinin artırılması ve genel iş yükünün bir bölümünü oluşturan yönetim çabaları, ara çalışma basamaklarının ortadan kaldırılması etkili olur.

Her teknolojik gelişimin olduğu gibi blok zinciri teknolojisinin de fırsatları, güçlü yönleri bulunduğu kadar zayıflıkları ve tehditleri de bulunur. İhtiyaç analizini yaparken tüm bu yönleri detaylıca değerlendirmek gerekir.

Blok zinciri teknolojisinin **güçlü yönleri**; merkeziyetsiz ağ yapısı, genişletilebilir ve değiştirilemez veri tabanı, dağıtılmış esneklik ve kontrol, güvenli şifreleme ve modern kriptografi, uygulama verimliliği ve bilgi paylaşım kolaylığı, akıllı sözleşmeler, büyük veri projeleri için uygun bir platform, şeffaf, hızlı, ucuz ve açık kaynak kodlu olması olarak sıralanabilir.

Blok zinciri teknolojisinin **zayıf yönleri**; regülasyon eksikliği, müşteri memnuniyetsizliği ve zayıf kullanıcı deneyimi, teknolojinin test edilme zorluğu, akıllı sözleşme programlama tecrübesi eksikliği, cüzdan ve anahtar yönetimi, zayıf geliştirici deneyimi, blok zincirinin imajı ve güven sorunları olarak sıralanabilir.

Blok zinciri teknolojisinin **fırsatları**; IoT aygıtları arasında güvenli iletişim, düşük işlem ücretleri, hızlı ve verimli iş süreçleri, dolandırıcılığı azaltması, sistemsel riskleri düşürmesi, yeni iş modeli etkinleştirme potansiyeli olarak sıralanabilir.

Tehditler; yasal yargı engelleri, vergilendirme sorunları, devletlerin olumsuz bakışı, teknolojik problemler, kurumsal adaptasyon sorunları, zayıf yönetim ve uzlaşma modelleri olarak sıralanabilir.

10.3. PROJE YAZIM KURALLARI

Şirket, kuruluş veya kişi tarafından bir fikri, çözümü, ürünü veya hizmeti gerçekleştirmek için bir proje hazırlanır. Hazırlanan bu projenin teknik özelliklerini tanıtmak amacıyla proje hakkında bilgilendirici bir belge hazırlanır. Bu belgeye **white paper [beyaz kâğıt (izahatname)]** denir (Görsel 10.4).

Görsel 10.4: White paper (İzahatname)

10.3.1. White Paper

White paper (izahatname), yeni geliştirilen herhangi bir projenin veya ürünün amaç, kapsam, getireceği yenilik, felsefi ve teknik detaylarının yazılarak yayınlanan belgedir. İçeriği ve amacı gereği akademik ve teknik bir dille yazılmalıdır. Blok zinciri teknolojisinde geliştirilen projeler için white paper oldukça önemlidir. White paper, geliştirilen projenin amacını, hangi problemleri nasıl çözdüğüne dair matematiksel hesapların, grafiklerin ve çizimlerin verildiği; projenin ayrıntılı tanımı ve mimarisini içerir. Ayrıca transfer işlemlerini ve bu işlemlerin doğrulanmasında kullanılacak uzlaşı (konsensüs) mekanizması, üretim, dağılım, ödüllendirme sistemi gibi bilgileri içermelidir.

Böylece projeyle ilgilenecek yatırımcılar, teknoloji grupları, bireysel kullanıcılar ve işlemleri doğrulayacak madencilere bu projeyi neden kullanmaları gerektiği hakkında somut bilgiler veren teknik bir belge oluşturulmuş olur. Aynı zamanda oluşturulan bu teknik belge, şeffaf bir şekilde herkesin rahatlıkla ulaşabileceği şekilde yayınlanmalıdır (Görsel 10.5).

Görsel 10.5: Blok zinciri projesi white paper yayını

10.3.2. White Paper Nasıl Yazılır?

Bir blok zinciri projesine yönelik white paper incelemelerinde bulunması gereken genel unsurlar aşağıdaki gibi sıralanabilir.

- Başlık ve giriş
- Özet
- Feragatname
- Projenin amacı ve çözdüğü problem
- Projenin çözüm metodu, nasıl çözdüğü
- Projenin kapitalizasyonu (finansmanı, minimum sermaye, maksimum sermaye, geliştirme aşamaları vb.)
- Geliştirme için yol haritası veya son tarihler

White Paper Başlığı ve Girişi: Blok zinciri projesinin tam adı ve kısaltma olarak kullanılacak adı veya kodunu yer almalıdır.

Özet: Projenin amacı, çözdüğü problem ve nasıl çözdüğü hakkında özet bilgi bu alanda verilebilir.

Feragatname: Eğer blok zinciri projesiyle ilgili gereken yasal bir açıklama ve uyarılar bu bölümde verebilir. Bir kısım blok zinciri projelerinin, özellikle kripto para gibi finansal projelerin, bazı ülkelerin yasaları gereği çeşitli kısıtlamaları bulunur. Birçok ülkede yasalar gereği yatırım tavsiyesi vermenin suç olması sebebiyle projeye yatırım yapılması durumunda kâr etme garantisi verilmediği ve kullanıcının veya yatırımcının kendi tercihine bırakıldığıyla ilgili uyarılar verilebilir.

Projenin Amacı ve Çözdüğü Problem: White paperın bu kısmında; projesinin amacı, çıkış noktasını oluşturan detaylar, projenin uygulanacağı alanın, sektörün tanımı ve sektörde projenin sunduklarına dair eksiklikler verilebilir. Geliştirilen proje alanında ilk ve tek olabileceği gibi muadili olan fakat diğerlerine göre fark oluşturan bir proje de olabilir. Bu farkın ve öne çıkacağı, çözüm bulduğu noktalar da burada belirtilebilir.

Projenin Problem Çözüm Metodu: Geliştirilen projenin amacında belirtilen problemin çözümünde uygulanacak metotlar, metotların teknik detayları, gerekli matematiksel hesaplamaları, grafikleri, kullanılan kodları ve şifreleme yöntemleri bu bölümde verilebilir.

Projenin Kapitalizasyonu: Geliştirilen projenin finansmanı ile ilgili teknik detaylara bu bölümde yer verilebilir. Geliştirilen proje özellikle kripto para birimiyse bu kripto para biriminin arzı ve kuralları verilebilir. Bazı kripto para birimleri arzı sınırlı olarak oluşturulur. Bazıları ise belirli kurallar çerçevesinde sınırsız arza sahiptir. Bu arza dair detaylar da gerekirse ayrı bir başlık altında yatırımcılara ve kullanıcılara sunulabilir.

Yol Haritası: Projenin çıkışı sonrası kısa, orta ve uzun vadede nasıl bir yol haritasına sahip olduğu; ileride yapılabilecek geliştirmeleri, gerekli güncellemeleriyle projenin güçlü yanlarını gösterir. Bu nedenle proje ekibi tarafından belirlenmiş olan yol haritası genel hatlarıyla paylaşılabilir.

Kaynakça: Proje hazırlanmasında yararlanılan kaynakların yer aldığı liste verilebilir.

Bunlarla sınırlı kalmayıp proje içeriği hakkında gerekli görülen her türlü teknik detay, bilgilendirme bir başlık altında açık ve şeffaf bir şekilde verilmelidir. Bu detay ve açıklamaların bulunması proje için olacağı kadar hem yatırımcılar hem de projenin teknik kısımlarıyla ilgilenenler için faydalı olur.

SIRA SİZDE

Yayınlanan blok zinciri projelerine ait white paperları inceleyerek projeler hakkında teknik bir sunum hazırlayınız. Gönüllü arkadaşlarınızın paylaştığı sunumlar üzerinden değerlendirmeler yaparak teknik bir sunumun nasıl olması gerektiğini değerlendiriniz.

SIRA SİZDE

Sınıfta arkadaşlarınızla bir proje ekibi oluşturarak geliştireceğiniz bir proje için white paper hazırlayınız. Hazırladığınız white paper belgenizi sınıfta arkadaşlarınızla paylaşınız.

10.4. ICO, ITO, IEO

ICO (Initial Coin Offering), ilk para (coin) teklifi anlamına gelir. Blok zinciri tabanlı projeler, kaynak oluşturmak amacıyla oluşturdukları kripto paraları satışa sunarak fon toplayabilir. ICO,

şirketlerin hisse senetlerini halka arz etmelerine benzer. ICO etkinliklerinde toplanan fonlar genelde izin verilen kripto paralar aracılığıyla alınabilir. ICO etkinliklerinde, fon toplamak amacıyla coinler satışa sunulur. Coinler, kendi bağımsız blok zincirlerinde çalışan para birimleridir.

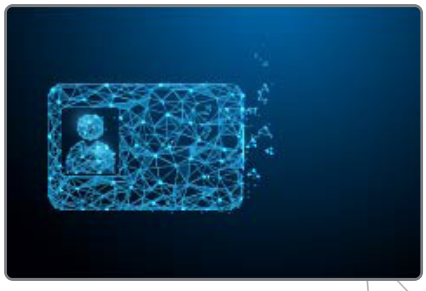
ICO etkinlikleri, sermaye toplamanın ve projeye finansman sağlamak için kullanılan etkili bir yöntemidir. Projelere bu sayede erken aşamalarda finansal destek sağlamış olur. Yatırımcılar ise genellikle, coinlerin ve bunu çıkaran şirketin başarılı olacağı umudu ve beklentisiyle ICO etkinliklerine katılır. ICO etkinliğinden önce projenin white paperı yayınlanır. Gerekli fonun toplanabilmesi amacıyla bu white paperda, proje detaylıca yer almalı, gerekli tüm bilgiler teknik açıklamalar verilmelidir. ICO etkinliği sonucunda belirlenen fon miktarına ulaşılamazsa yatırımcılara paraları iade edilir.

ITO (Initial Token Offering), ilk jeton (token) teklifi anlamına gelir. ICO ile benzer bir yapı olmasına rağmen önemli farkları vardır. ICO yeni bir kripto para birimi için fon toplamaya odaklanmıştır fakat ITO genellikle kanıtlanmış yapısal bir kullanıma sahip jetonları sermaye piyasasına sunmaya odaklanır. Böylece yatırımcılara abonelik yoluyla platforma erişim hakkı verilmesini ve jeton sahiplerinin ekosistem içindeki özel hizmetlerden faydalanmasını sağlar. Jetonlar, coinler gibi bağımsız blok zinciri yapılarına sahip değildir. Mevcut bir blok zinciri üzerinde çalışır.

IEO (Initial Exchange Offering), ilk değişim teklifi anlamına gelir. IEO, ICO ve ITO gibi işlemlerin kripto borsası tarafından gerçekleştirilmesidir. ICO ve ITO işlemlerinin çoğalması ve bunların bir bölümünün dolandırıcılıkla ilgili olması bir çok yatırımcı ve yeni projeler için sorun teşkil eder. Bu sorun çözümü ICO ve ITO işlemlerinin, projenin geliştiricileri tarafından kendi web sayfaları üzerinden gerçekleştirmeleri yerine kripto borsası tarafından kontrol edilerek piyasaya çıkışının gerçekleşmesidir. IEO'nun tüm yönetiminden, IEO'yu gerçekleştiren borsa sorumludur. Kaynak toplama esnasında satışı gerçekleştirilen kripto para, jeton veya dönüşüm işlemleri, bu borsa üzerinden yapılır. Kripto para borsaları, kendi platformları üzerinden listelenecek kripto para ve jetonlar için yatırımcılara çeşitli garantiler verir. IEO'larda süreç şeffaf bir şekilde yürütüldüğü için yatırımcılara güven verir.

10.5. KİMLİK YÖNETİMİ

Kamu ve özel sektör tarafından sunulan eğitim, sağlık, finans, oy kullanma ve sosyal yardımlardan faydalanma gibi birçok hizmete erişebilmek için insanların kim olduklarını kanıtlamaları gerekir. Bunun için de resmî bir kimlik belgesi sahibi olmaları gerekir (Görsel 10.6). İnsan Hakları Evrensel Beyannamesi'ne göre herkesin her nerede olursa olsun, hukuksal kişiliğinin tanınması hakkına sahip olduğu belirtilmektedir.



Görsel 10.6: Kimlik belgesi

İnsanların hastanede muayene olmak, araç kullanmak, banka hesabı açmak, oy vermek, kamu hizmetlerine erişmek gibi birçok haktan faydalanması için kimliğini yasal olarak ispatlaması gerekir (Görsel 10.7). "Dünya Bankası Gelişim İçin Kimliklendirme Girişimi 2018 Yılı Faaliyet Raporu"na göre Dünya üzerinde 1 milyardan fazla insanın kimliklerini ispat edecekleri herhangi bir belgeleri bulunmamaktadır. Bu insanlar sağlık, eğitim, finans hizmetleri gibi en temel insani gereksinimlere erişim sağlayamamaktadır.

Görsel 10.7: Yasal kimlik ispatlama

Teknoloji, hızla ilerlemesiyle gündelik hayatın her alanında kullanılmaya başlanmıştır. Kamu ve özel sektörün sunduğu birçok hizmet ve iş süreci de gelişen teknolojilerden faydalanılarak dijitalleşmiştir. Bu hizmetlerden biri de kimliklerin dijitalleşmesidir (Görsel 10.8).

Görsel 10.8: Dijital kimlik

Devletlerce resmî olarak sağlanan kimlik belgelerinin yanı sıra, dijital olarak sunulan birçok hizmete erişim için dijital kimliklerin kullanılması gerekir. Dijital kimlik, bir kişinin internet üzerinden yapacağı bazı işlemler için kullanmaları gereken, resmî fiziksel kimliklerinin bir karşılığıdır. Fiziksel kimlik ile dijital kimlikler arasında güvenilir bir bağlantı kurularak, insanların dijital dünyada gerçekten iddia ettikleri kişi olduklarını doğrulamaları gerekir.

Blok zinciri teknolojisiyle kullanıcıların kimlik bilgileri, dijital ortamda güvenli ve şifreli olarak saklanabilmektedir. Güvenlik, geleneksel yönteme göre daha güçlüdür. Kullanıcıların kimliklerinde yer alan bilgilerin gizliliği blok zincirinin kriptografik özelliğiyle korunabilmektedir.

Dijital kimliğin içerisinde isim, soy isim, ev adresi, mail adresi, kimliği veren kuruluşun adı ve bir seri numarası gibi belirlenen bilgiler yer alır. Hizmetlere giriş, sertifika güvenliği ve dijital imzalar için kullanılır. Erişim ayrıcalığı gerektiren durumlarda kullanıcıların tanımlanmasını sağlar. Dijital kimliğin belirli bir zaman aşımı süresi vardır.

Kullanıcı bilgilerinin doğrulanması işlemleri, kurumlar için oldukça maliyetlidir. Blok zinciri teknolojisiyle kimlik bilgileri, güvenli dijital ortama aktarıldığından doğrulama işlemleri için herhangi bir merkeze ihtiyaç duymadan otomatik olarak gerçekleştirilir. Böylece doğrulama işlemi geleneksel yöntemden daha hızlı, güvenli ve ekonomik olur.

Blok zinciri teknolojisi, dijital kimliklerin oluşturulmasında ve yönetilmesinde kullanılabileceği gibi bu kimliklerin birçok farklı özelliğinin yönetiminde de kullanılabilir. Örneğin doğum kayıtları, evlilik cüzdanları, pasaport ve vize bilgileri, hastane kayıtları gibi veriler de yönetilebilir. Blok zincirine dayalı kimlikler, hizmetlerin daha kolay ve maliyetinin daha düşük hâle getirmesinin yanı sıra özellikle çok faktörlü doğrulama veya biyometrik kontrolle birleştirildiğinde kimlik hırsızlığının da önüne geçebilir (Görsel 10.9).

Görsel 10.9: Dijital kimlik ispatı

SIRA SİZDE

Sanal ağ üzerindeki işlemlerde gerçek kimliğin doğrulanmasının nasıl yapıldığı hakkında bir poster hazırlayınız. Hazırladığınız posterleri karşılaştırarak benzerlikleri ve farklılıkları hakkında değerlendirme yapınız.

10.6. BLOK ZİNCİRİ VE İŞ DÜNYASININ DÖNÜŞÜMÜ

Blok zinciri teknolojisine olan ilgi gün geçtikçe artmaktadır. Bu teknoloji; merkeziyetsiz, dağıtık veri tabanı mimarisi, şeffaflığı, otomatik algoritma yapısı, güvenli olması ve eşten eşe iletişim sağlaması gibi avantajları nedeniyle birçok sektör ve işletme tarafından tercih edilmektedir. Blok zinciri teknolojisi işletme içinde verilerin doğrulanması, saklanması, karar verme gibi süreçlerinde bürokrasinin azaltılmasında faydalı olacaktır. Müşterileriyle olan işlemler de dâhil olmak üzere, işletme dışından paydaşlarla olan işlemlerde de şeffaflık sağlayarak, blok zinciri teknolojisinin değişmezlik özelliğiyle güven oluşturmada etkili olacaktır. Blok zinciri teknolojisiyle iş dünyasında yaşanan dönüşüm sayesinde, daha önce güvensizlik sebebiyle ortaklığa girmeyen ulusal ve uluslararası şirketler, yeni işbirliği modelleri üzerinde çalışma ve yeni pazarlara ulaşma fırsatı bulacaktır.

İş dünyasında dijital dönüşümün en önemli avantajlarından biri, iş süreçlerini çok hızlı bir hâle getirmesidir. İş dünyasında dijital dönüşüm açısından en çok dikkat edilecek hususlardan biri de güvenliktir. Blok zinciri teknolojisi, ağ içindeki işlemlerin hem çok hızlı hem çok güvenli bir şekilde gerçekleştirilebilmesini sağlar. Blok zinciri sayesinde iş süreçlerinin tam otomatik hâle getirilmesi üretkenliği ve verimliliği artırır. Blok zinciri, bu avantajları sayesinde iş dünyasının dijital dönüşümünde kullanacağı en önemli teknolojilerdendir. Devletler tarafından yapılacak yasal düzenlemeler, blok zinciri teknolojisinin iş dünyasında kullanımına yön vermesi açısından önemlidir.

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

1. I. Pasif kullanım

II. Aktif içerik ekleyebilme

III. Sınırlı deneyim

IV. Belirli içerik ekleyici

Yukarıda verilenlerden hangileri Web 1.0 teknolojisinin özelliklerindendir?

A) I-II B) I-III

C) II-III D) I-III-IV

E) I-II-IV

2. Blok zinciri uygulaması için ihtiyaç analizi yapılırken aşağıdaki yönlerden hangisini dikkate almaya gerek yoktur?

A) Merkeziyetsiz ağ yapısına B) Şeffaf ve hızlı olmasına

C) Yasal düzenlemelere D) Değiştirilemez veri tabanına

E) Açık kaynak kodlu olmasına

3. I. Başlık ve giriş

II. Feragatname

III. Proje amacı

IV. Çözüm metodu

V. Bitiş

Blok zinciri projesine yönelik white paper hazırlanırken yukarıda belirtilen unsurlardan hangileri bulunmalıdır?

A) I-II B) III-IV-V

C) II-III D) I-II-IV

E) I-II-III-IV

4. Dijital kimliğin geleneksel kimliğe göre avantajlarından değildir?

A) Şifreli ve güvenli olması B) Kopyalanıp çoğaltılması

C) Farklı bilgiler eklenebilmesi D) Doğrulamanın hızlı olması

E) Maliyetinin düşük olması

KONULAR

11.1. TEDARİK ZİNCİRİ

11.2. SAĞLIK ALANINDA BLOK ZİNCİRİ

11.3. ENERJİ SEKTÖRÜNDE BLOK ZİNCİRİ

11.4. EMLAK PİYASASINDA BLOK ZİNCİRİ

11.5. SİGORTACILIK ALANINDA BLOK ZİNCİRİ

11.6. TELİF HAKLARI

11.7. KAMU YÖNETİMİNDE BLOK ZİNCİRİ

NELER ÖĞRENECEKSİNİZ?

- Blok zinciri teknolojisinin tedarik zincirindeki yeri ve proje örnekleri
- Blok zinciri teknolojisinin sağlık alanındaki yeri ve bu alandaki girişimleri
- Blok zinciri teknolojisi kullanmanın enerji sektöründeki yeri ve proje örnekleri
- Emlak piyasalarında blok zinciri kullanmanın yararları
- Sigortacılık alanında blok zincirinin önemi ve bu alandaki girişimleri
- Blok zincirinde telif hakkının yeri ve önemi
- Kamu yönetiminde blok zinciri kullanımı

ANAHTAR KELİMELER

Blok zinciri, emlak, enerji, NFT, sağlık, tedarik zinciri, yenilenebilir enerji

HAZIRLIK ÇALIŞMALARI

1. Blok zinciri teknolojisinin potansiyel kullanım alanları neler olabilir? Arkadaşlarınızla değerlendiriniz.

2. Size göre blok zinciri teknolojisinin farklı alanlarda kullanımı ne gibi avantajlar sağlayabilir? Düşüncelerinizi arkadaşlarınızla paylaşınız.

PROJE ÖRNEKLERİ VE DÜNYADAN GİRİŞİMLER

11. ÖĞRENME BİRİMİ

11.1. TEDARİK ZİNCİRİNDE BLOK ZİNCİRİ

Blok zinciri projeleri dünyada birçok ülkede olduğu gibi ülkemizde de çeşitli alanlarda yerini almış ve gelişmeye devam eder. Blok zinciri teknolojisi ile hem finansal alanlar hem de finansal olmayan alanlarda farklı çalışmalar yapılır. Bunlardan bazıları; tedarik zinciri, fikrî mülkiyet hakları, tapu/emlak, bankacılık, finans, politika, bağış sistemleri, güvenli bulut bilişim depolama sistemleri, yapay zekâ ve IoT sistemleri, sağlık, eğitim, gıda ve kimlik yönetimidir.

Görsel 11.1: Tedarik zinciri

Tedarik zinciri (Supply chains) bir ürünün veya hizmetin, ham madde tedarikinden müşteriye ulaştırılıncaya kadar olan süreç içindeki üretim merkezleri, kişiler, bilgi, teknoloji, etkinlik ve kaynakları içeren ve bunların koordine edilmesini sağlayan sistemin bütünü olarak ifade edilir. Tedarik zinciri yönetim sürecinde, zincir içinde yer alan ilk tedarikçiden son kullanıcıya kadar tüm kişi, kurum ve kuruluşlar koordine edilerek etkili bir şekilde ve tek bir işletmeymiş gibi yönetilir (Görsel 11.1). Bu sayede işletmelerin maliyetleri azalır ve müşterilerin memnuniyeti de artar.

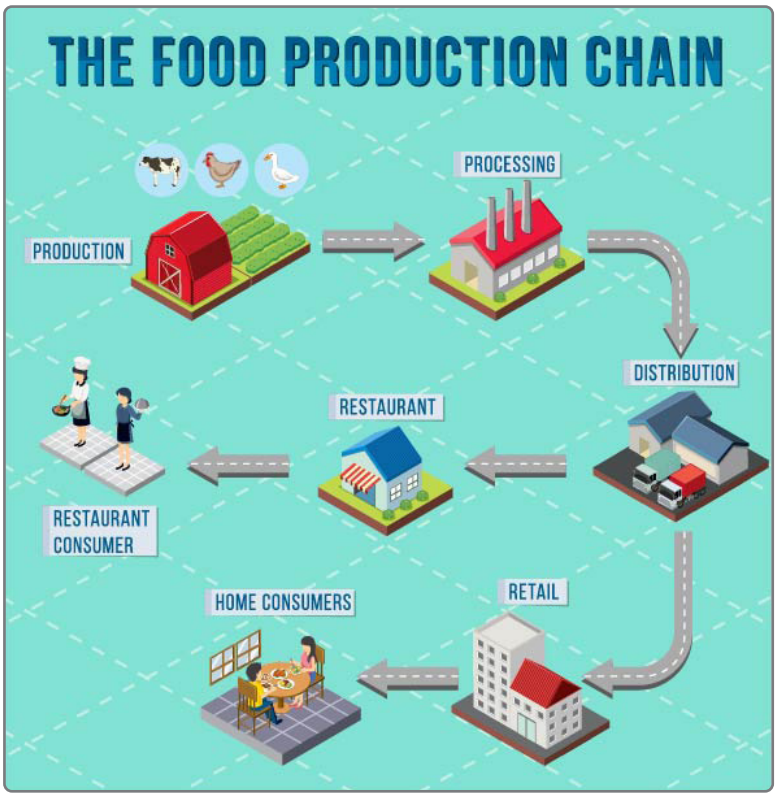
Tedarik zincirlerinin şeffaf olmayan ve oldukça karmaşık bir yapısı vardır. Tüketiciler, satın aldıkları ürün hakkında herhangi bir bilgiye sahip değildir. Bir ürünün orijinal olup olmadığı, hangi kaynaktan geldiği, sağlıklı mı yoksa sağlıksız mı olduğunu bilinmez. Bu anlamda tedarik zinciri, blok zincirinin en fazla yarar sunabileceği alanların başında gelir.

OECD ve EUIPO tarafından yapılan bir araştırmada "sahte ve korsan mal imalatının küresel ithalatın yaklaşık yüzde 2,5'i değerinde olduğu" saptanır. 2013 yılında, sığır eti olarak reklamı yapılıp piyasaya sürülmüş olan yiyeceklerin, beyan edilmemiş et içerdiği tespit edilir. At eti skandalı olarak tarihe geçmiş bu ve buna benzer gıda kaynaklı hastalık vakaları nedeniyle müşterinin güveni sarsılır. Bir ürünün kaynağını takip etmek için bazı yollar olsa da bunların maliyeti yüksek ve zaman alıcıdır.

Blok zinciri teknolojisi mevcut güvenli yapısı ve izlenebilirliği sayesinde, tedarikçiden müşteriye kadar olan zincir boyunca şeffaflığın artmasına katkı sağlar. Blok zinciri teknolojisi köken takibi ve maliyetsiz doğrulama süreci ile bir işlemin güvenliğini sağlamaya yardımcıdır. Herhangi bir yanlış faaliyet, topluluk tarafından tespit edilebilir, izlenebilir ve engellenebilir.

Blok zinciri teknolojisi kullanan tedarik zinciri uygulamaları sayesinde, tükettiğimiz gıdaların ham maddesi, hangi şartlar altında üretildiği, içeriğindeki bileşenleri gibi gıda üretim aşamaları gözler önüne serilerek insan ve hayvanların sağlığının güvence altına alınması sağlanır.

Örneğin bir yiyecek veya içecek, üretiminden evdeki dolaba girene kadar birçok aşamadan geçer. Gıdanın tedarik edilmesi, fabrikaya götürülmesi, fabrikada çeşitli işlemlerden geçmesi ve ardından araçlara yüklenerek marketlere dağıtılması gerekir (Görsel 11.2).



Görsel 11.2: Gıda tedarik zinciri

Gıda tedarik zinciri sürecinde kontrollerdeki ve ekipmanlardaki eksiklik, sağlığı riske atacaktır. Günümüzde gıda alanındaki hilelerin çok fazla olması blok zincirinin gıda alanında kullanılmasını sağlar. Bu teknoloji ile birlikte; gıda işleme belgelerinde sahtecilik yapılıp yapılmadığı, gıda işleme ve paketlemesinin ne kadar zaman aldığı, gıdanın kaç gün boyunca nakliye aracında bulunduğu, marketlerde kaçıncı gününü doldurduğu ve gıdalara ne tür bir paketleme yapıldığı şeffaf olarak görülebilir.

Güncel blok zinciri gıda uygulamalarında, telefona barkod okutularak gıdalar izlenebilir. Bazı firmalar dünya üzerindeki tüm müşterilerine, sipariş verdikleri yiyeceklerin kaynağını görebilme şeffaflığını sunar. Blok zinciri tabanlı teknolojiler, gıda güvenliğini iyileştirmek, gıda kaynaklı hastalıkları izlemek ve gıda tedarik zincirine şeffaflık getirmek amacıyla kullanılır.

Avustralyalı bir otomobil üreticisi, bazı tedarikçilerine ödeme yapmak için kripto para kullanır. Ayrıca firma İsrail ile Tayvan'daki üç müşterinin ödemeleri için kripto para kullanımını kabul eder. Böylelikle uluslararası ödemelerde para transfer ücreti giderlerinin ortadan kaldırılması sağlanır.

Bir madencilik devi, tedarikçilerini doğrulamak için teknolojiyi kullanarak ve tedarik zinciri boyunca çevresel, sosyal ve yönetişim gereksinimlerinin karşılanmasını sağlayarak operasyonlarını blok zinciri üzerinden dijitalleştirir.

Elmas devi bir şirket, taşları çıkarıldıkları yerden müşterilere satıldıkları ana kadar takip etmek için blok zinciri teknolojisini kullanır.

Bir araba firması, blok zinciri teknolojisiyle takip edilen ve geri dönüşümlü kobalt maddesi kullanılan ilk araçları ürettir. Firma, blok zinciri teknolojisinin sunduğu dağıtık defter teknolojisinin tedarik zincirlerinde hesap verilebilirliği ve şeffaflığı artırdığını tespit eder.

Ülkemizde ise bir market zinciri, blok zinciri teknolojisiyle tedarik zincirini güçlendirerek müşterilerinin ürünlerin tazeliğinden ve üretim şartlarından şüphe etmelerini ortadan kaldıran bir girişimde bulunur. Bu girişim ile logolu meyve ve sebzeler merkezîyetsiz ve uçtan uca şifreli işlem takibi sağlayan bir kayıt defterine eklenir. Kayıt defterinde yer alan veriler değiştirilmez ve geriye dönük işlem yapılamaz olduğundan güvenilirlik sağlanmış olur.

11.2. SAĞLIK ALANINDA BLOK ZİNCİRİ

Blok zinciri teknolojisi sağlık sektöründe oldukça geniş uygulama alanına ve işlevlere sahiptir. Bu teknoloji, sunmuş olduğu güvenlik, şeffaflık ve gerçek zamanlı takip özellikleriyle sektördeki birçok sorunun çözülmesine katkı sağlar.

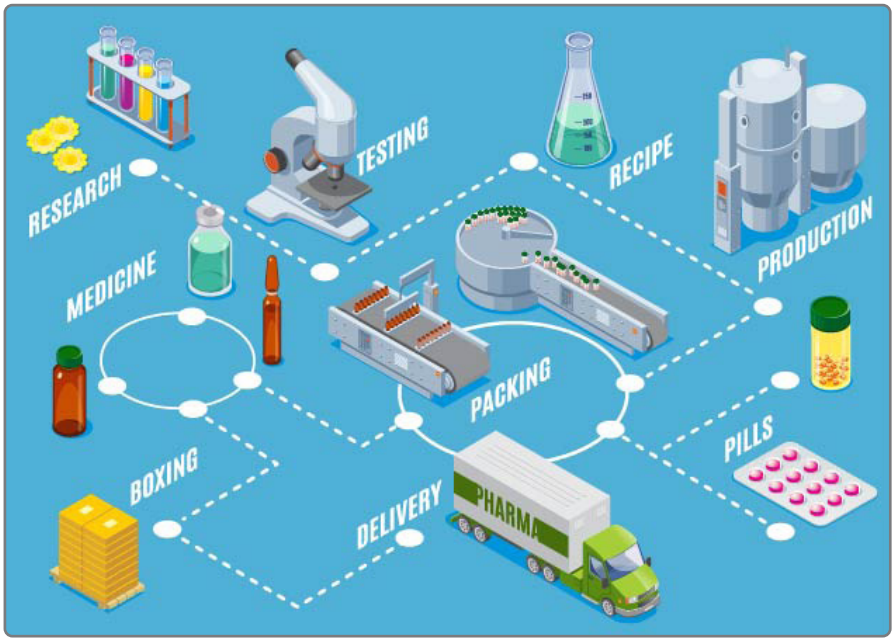
Sağlık sektöründe blok zinciri teknolojisi; kişisel kayıtların yönetimi, uzaktan hasta takip ve yönetimi, tıbbi cihaz ve ilaç tedarikinde sahtecilikle mücadele etme, reçetelerin takibi, sigorta geri ödeme süreçlerinin denetimi, sağlık verilerinin klinik araştırmalarda güvenli kullanımı gibi birçok alanda kullanılmakta ve bu alanlarda çalışmalara devam edilmektedir (Görsel 11.3).

Görsel 11.3: Blok zinciri teknolojisi ve sağlık alanının ilişkisi

Elektronik ve kişisel sağlık kayıtlarının tutulması esnasında, hasta verilerinin blok zincirine aktarılarak saklanabilmesi sayesinde sağlık personelleri, hastanın önceki sağlık verilerine doğrulanmış erişim yoluyla rahatça erişebilir. Blok zincirinde depolanmış olan hasta verileri sayesinde ilaç şirketleri, hastanın tüm sağlık durumunu ve kullandığı ilaç geçmişini görebilir ve hastaya özel ilaçlar yapabilir.

Sağlık sektöründe kullanılan uygulamalardan biri de **ilaç tedarik zinciridir**. İlaç ve aşılarla ilgili veriler, blok zincirinde güvenli bir şekilde saklanır ve teknolojinin sunduğu gerçek zamanlı takip yeteneği sayesinde müşteri kaynağı geriye doğru izleyebilir. Aşı ve ilaçların üreticiden son

kullanıcıya hangi şartlarda ulaştığı gözlenerek sahte veya olumsuz şartlardan dolayı risk yaratabilecek ilaçların kullanılmasının önüne geçilir (Görsel 11.4).



Görsel 11.4: İlaç tedarik zinciri

Türkiye'de ilaç tedarik zincirinin takibi ve izlenmesi için kullanılan uygulama Sağlık Bakanlığına bağlı olan ve merkezî bir sistem olan **İTS (İlaç Tedarik Sistemi)** teknolojisidir. Türkiye'de ilaç tedarikinde henüz blok zinciri teknolojisi kullanılmamaktadır ancak tedarik zincirinde blok zinciri teknolojisi tabanlı uygulamaların merkezî olmayan özelliklerinden dolayı İTS'ye göre daha şeffaf olduğu bilinmekte ve bu konularda çalışmalar yapılmaktadır. İTS uygulamalarında, paydaşların tedarik zincirindeki tüm verileri görememesinden dolayı şeffaflık sorgulanmaktadır. Blok zincirinin merkezî olmayan özelliği, tüm paydaşların birbirleriyle iş birliği yapmalarını sağlar. Akıllı sözleşmeler sayesinde ise her paydaş, tedarik zincirinde kendisine tanımlanan işi yapar ve diğer paydaşlar işleyişi izleyebilir. İlaç Takip Sistemi Teknolojisi ile Blok Zinciri İlaç Tedarik Teknolojisinin karşılaştırılmalı özellikleri Tablo 11.1'de verilmiştir.

Tablo 11.1: İlaç Takip Sistemi Teknolojisiyle Blok Zinciri İlaç Tedarik Teknolojisinin Karşılaştırılması

Özellikler	İlaç Takip Sistemi	Blok Zinciri İlaç Takip Sistemi
Merkezî olmama	Hayır	Evet
Şeffaflık	Hayır	Evet
Takip ve izlenebilirlik	Evet	Evet
Güvenlik	Hayır	Evet
Esneklik	Düşük	Yüksek
Uygulanabilirlik	Yüksek	Düşük
Teknik altyapı ihtiyacı	Düşük	Yüksek

Sağlık sektöründe blok zinciri projelerine dünyada birçok örnek mevcuttur. Tıbbi faturalama ve sağlık odaklı bir Bilişim Teknolojileri (BT) şirketi, birden çok elektronik sağlık kaydı sistemi (EHR) arasında, izinlerle değiş tokuş edilen, tıbbi kayıtların referanslarını tutan bir blok zinciri ağı uygular. Bu uygulama ile hastalar, kayıtlarına erişimi kontrol edebilir ve EHR'si ağa katılan herhangi bir sağlayıcıya izin verebilir.

Çin menşeli bir çalışma grubu, geliştirdiği **Healthcare Data Gateway** isimli proje ile hastaların blok zinciri ağına bağlanıp kendi sağlık bilgilerini görüntüleyebilmelerini ve yine hastaların, kendi sağlık bilgisine kimin erişebileceğini kontrol edebilmesine olanak sağlar.

11.3. ENERJİ SEKTÖRÜNDE BLOK ZİNCİRİ

Her geçen gün artan dünya nüfusu ve enerji tüketimi ile enerji yönetimi ve enerji verimliliği konuları, daha fazla ön plana çıkar. Enerji ticareti, akıllı ev sistemleri, yeşil tarife ve yenilenebilir enerji, şebeke yönetim sistemleri, e-mobilite vb. blok zinciri teknolojisinin enerji sektöründeki kullanım alanlarından bazılarıdır (Görsel 11.5).

Görsel 11.5: Dünya, enerji, çevre ve blok zinciri teknolojisi ilişkisi

Blok zinciri teknolojisinin, nesnelerin interneti (IoT) teknolojisi ile birlikte kullanılabilir bu sayede enerji tasarrufu sağlamaya yönelik çeşitli uygulamalar geliştirilebilir. Üretici ve tüketicilerin birbirleri ile doğrudan enerji ticareti yapmaları buna örnek olarak verilebilir.

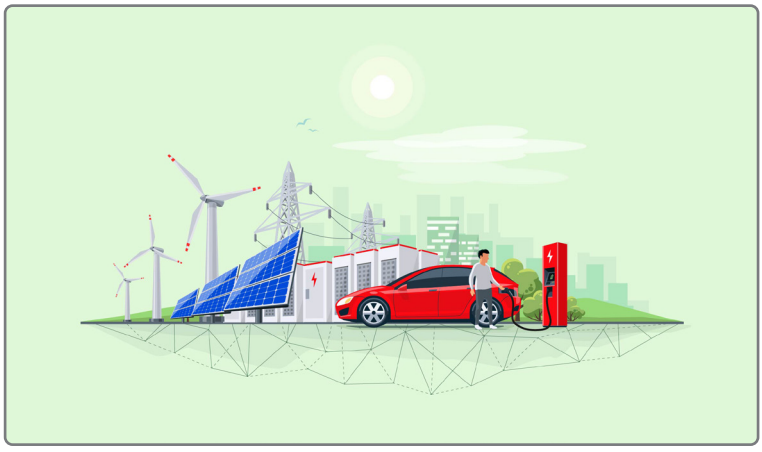
Blok zinciri teknolojisinin eşler arası (Peer to peer) işlem yapma imkânlı enerji ticareti, kendi enerjisini üreten bireylerin ihtiyaçlarından fazla olan enerjiyi satarak enerji ticareti yapmalarına olanak sağlar.

Blok zinciri temelli bir elektrik şebeke veri yönetimi, enerji kullanım verilerinin güvenli ve gerçek zamanlı takibini yaparak oluşabilecek sorunların ortadan kalkmasına yardımcı olur. Verilerin blok zinciri üzerine kaydedilmesiyle oluşabilecek mali zararlar engellenir ve verilerin şeffaf bir şekilde paylaşılması sağlanır.

Emtia ticareti, birçok farklı alanda alım satımları ve el değiştiren varlıkları kaydeden büyük bir kaydın tutulmasını gerektirir. Blok zinciri teknolojisini emtia ticaretine uygulamak, mevcutta kullanılan sistemlerden daha ucuz ve verimli bir çözüm sunulmasına destek olur.

Türkiye'de güneş enerjisi sistemleri, enerji verimliliği uygulamaları, elektrikli araç şarj istasyonu, yeşil enerji sertifikasyonu gibi birçok çevre dostu ve sürdürülebilir enerji uygulamaları bulunur. Bununla birlikte enerji piyasaları için birçok yeni iş modeli uygulamaları da mevcuttur. Gerçek zamanlı veri yönetimi ve paylaşımı, yenilenebilir enerji sertifikaları ve karbon kredilerinin yaygın olarak kullanılması bunlardan bazılarıdır.

Blok zinciri ve enerji denildiğinde genellikle ilk akla gelen konulardan biri, fosil enerji kaynaklarının yaygın kullanımı ve sürdürülebilirlik konularıdır. Küresel iklim değişikliği ve çevre kirliliğini önlemek için fosil yakıt tüketiminin sonlandırılması gerekir. Elektrikli ve hibrit araçların yaygın olarak kullanılmaya başlanması bu yönde atılan öneli bir adımdır. Elektrikli araçların batarya dolumu için elektrikli araba şarj üniteleri (eşarj) kullanılır (Görsel 11.6). Türkiye'nin 81 ilinde eşarj istasyonları mevcut olup her geçen gün sayıları artmaktadır. Bunun yanı sıra bireysel kullanım amacı ile kişilerin evlerine de eşarj üniteleri kurulabilir.



Görsel 11.6: Elektrikli araba şarj ünitesi

1 Haziran 2021 tarihinde uygulamaya konan Y**enilenebilir Enerji Kaynak Garanti Sistemi (YEK-G)**, ülkemiz enerji kaynaklarının kullanılarak üretilen enerjinin, üreticiden tüketiciye kadar olan sürecini güvenli ve sürdürülebilir olarak takibini sağlamak amacıyla **EPİAŞ** tarafından devreye alınmış enerji piyasalarının **ilk**, **yerli** blok zinciri tabanlı ağdır. Bu sistem ile yenilenebilir enerji kaynaklarının kullanımının yaygınlaştırılması, çevrenin korunması ve yenilenebilir enerjiyi herkes için ulaşılabilir hâle getirmek amaçlanmıştır.

Türkiye'de bir elektrik dağıtım şirketi, **İşimin Enerjisi** projesi kapsamında, birçok sektör ve kamu kurum ve kuruluşu için güneş enerjisi çözümleri, LED aydınlatma dönüşümü uygulamaları, yeşil enerji sertifikasyonu ve elektrikli araç şarj istasyonu yönetimi gibi pek çok alanda sürdürülebilirlik hizmeti sunar. Yine sektörde bir **ilk** olan ve **blok zinciri tabanlı**, **Enerji Güven Endeksi** projesi de sektörde uygulanan bir projedir.

Bu proje, serbest tüketici statüsüne sahip müşterilerin fatura ödeme alışkanlıklarına göre oluşturulan skorlarının sektör genelindeki tüm şirketlerle birlikte takip edilmesini sağlar.

Avrupa Komisyonu tarafından 2019 yılı Şubat ayında onay almış olan **Flexi_Grid Projesi** ile elektrikli araçların ve hava şartlarına bağlı üretim kapasiteleri olan rüzgâr ve güneş enerjisi santralleri gibi dağıtık üretim kaynaklarının yaratabileceği şebekesel sıkıntıların yönetilebilmesi amaçlanır.

Enerchain, farklı platformlar ve veri iletişim süreçleri ile alım satım yapan enerji tüccarlarına hizmet sunarak dünyadaki ilk blok zinciri tabanlı toptan enerji ticareti projesidir. 2019 yılı Mayıs ayında devreye alınır. Enerchain altapısıyla gelecekte kömür, petrol ve diğer endüstriyel ürünler için de merkezîyetsiz pazarlar yaratmak amacıyla kullanılması hedeflenir.

Brooklyn Microgrid Projesi (BMG), üreten tüketicilerin (yani konut ve ticari güneş paneli sahipleri) ürettikleri fazla güneş enerjisini, fosil yakıt yerine yenilenebilir enerji kaynakları ile üretilen elektrik kullanmayı tercih eden New York sakinlerine satmalarına olanak tanıyan blok zinciri tabanlı bir projedir. Kullanılan akıllı sayaç ile enerji fazlalığı tespit edilerek fazla enerji, pazara sunulabilir hâldedir.

11.4. EMLAK PİYASASINDA BLOK ZİNCİRİ

Gayrimenkul alanında geleneksel alım satım işlemlerinin yerini blok zinciri teknolojileri almaya başlar (Görsel 11.7). Geleneksel emlak alım satım işlemleri sırasında, aynı mülkün birden fazla kişiye satılması gibi sahtekârlıklar veya alım satım sürecinde birçok aracı (bankalar, noterler, değerleme uzmanları vb.) ile birebir uğraşmanın getirdiği birtakım zorluklar yaşanır. Yatırımların global olması durumunda ise bu zorluklara farklı zorlular eklenir. Her ülke veya bölgenin farklı uygulamaları olabilir, farklı para birimleri kullanabilir.

Görsel 11.7: Blok zinciri teknolojisi ve gayrimenkul alımı

Blok zinciri teknolojisi ile varlıklar dijitalleştirilerek ve tokenleştirilerek, emlak şirketlerinin varlıkları kolayca satması ve yönetmesi sağlanır. **Tokenizasyon işlemi**, fiziksel varlıkları dijital varlıklara dönüştürmeyi sağlayan, gayrimenkulde çok popüler olan bir blok zinciri uygulamasıdır.

Diğer bir deyişle menkul kıymetlerin, alternatif varlıkların ve finansal araçların dijitalleştirilmesidir.

Blok zinciri teknolojisinin emlak alanında kullanılmasının kısmi mülkiyet, aracısız işlem, likiditenin artması, azalan maliyetler gibi avantajları vardır. Bunların yanında aşağıdaki gibi yararları da sayılabilir:

A) Gayrimenkul işlemleri genellikle alıcı, satıcı, değerleme uzmanı, bankalar vb. birçok kişiyi içerir. Blok zinciri teknolojisi ile işlemlerin tüm kritik verileri blok zinciri veri tabanında saklanır. Böylelikle alım satım işlemleri daha kolay ve hızlı bir şekilde gerçekleşir.

B) Gayrimenkul işlemleri söz konusu olduğunda dolandırıcılık yaygın bir olgudur. Blok zinciri, alıcının rahatlıkla erişebileceği sahiplik sertifikası içeren veri tabanı sunarak bunu ortadan kaldırır.

C) Blok zincirindeki tüm bilgiler alıcı ve satıcı tarafından erişilebilir durumdadır. Bilgilerde herhangi bir değişiklik durumunda her iki taraf da bunu görebildiğinden taraflar arasında şeffaflık sağlanır.

Gayrimenkul alanında blok zinciri teknolojisinin kullanılmasıyla alıcı ve satıcılar, işlemlerini avukatlar, komisyoncular, banka gibi aracılar olmadan gerçekleştirebilir. **ATLANTA** projesi, emlak alanında blok zinciri uygulamasına örnektir.

Bir evin satışını kabul etmekle işlemi tamamlamak arasında geçen sürede, taraflar arasında yoğun trafik (telefon, e-posta vb.) olabilir; bu yoğunluk, tarafların gergin ve yorucu zamanlar geçirmesine sebep olabilir. Günümüzde bu tür durumları ortadan kaldırmayı hedefleyen bir blok zinciri girişimi, gayrimenkul alanında faaliyet gösteren işletmelerin herhangi bir mülkün satışıyla ilgili verilerini paylaşması imkânını sağlayan bir dağıtık defter teknoloji (DLT) ağını piyasaya sürer ve satış sözleşmesi gibi belgelerin paylaşım ve iletişimini sağlar. Bu girişim, ev satışlarında katılımcıların etkileşimde bulunabilmesi amacıyla bir mesajlaşma uygulamasını kullanır.

11.5. SİGORTACILIK ALANINDA BLOK ZİNCİRİ

Birçok sektörde olduğu gibi sigorta şirketleri de faaliyetleri sırasında çeşitli sorunlarla karşılamaktadır. Bu sorunların başında insan kaynakları ve teknoloji ile ilgili sorunlar gelmektedir. Veri saklama ve gizlilik, bilgi teknolojileri güvenliği, dijital kimlik bunlardan bazılarıdır (Görsel 11.8).

Görsel 11.8: Sigorta sektörü ve blok zinciri

Blok zinciri teknolojisinin sunduğu eş zamanlı veri paylaşımı ile sigorta poliçesi sahipleri, riskleri ve varlıkları daha doğru şekilde değerlendirir, hatalı hasar bildirimleri azalır, maliyet ve zamandan tasarruf edilir. Çok taraflı verilere otomatik bir şekilde ulaşılacağından güven sorunu ve suistimal azalır.

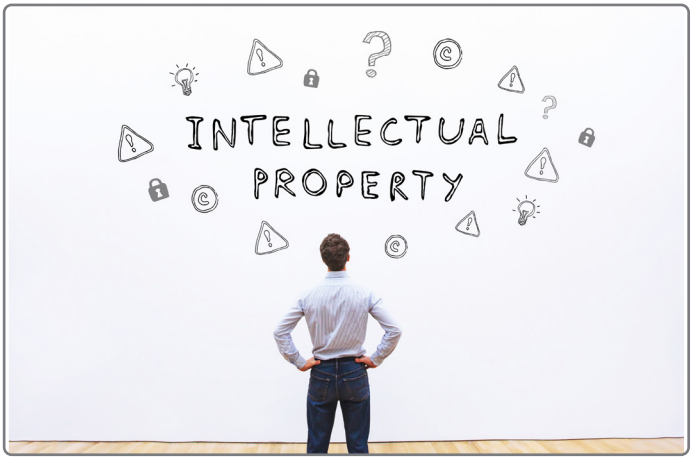
Etherisc, blok zinciri tabanlı sigorta platformu geliştiren bir girişimdir. **Uçuş Gecikme Sigortası (Flight Delay Insurance)** ise uçuş gecikmelerine veya iptallerine karşı koruma sağlayan ilk merkezî olmayan sigorta uygulamasıdır. Uçuşların takip edilmesini sağlayan bu girişimde yolcular, kripto para veya geleneksel para kullanarak bir sigorta poliçesi satın alabilir. Böylece uçuşta meydana gelen gecikme veya uçuşun ertelenmesi durumunda, bilet parasının dakikalar içinde sigortalıya iade edilmesi sağlanır. Etherisc tarafından uygulamaya konan ve lisanslı olan diğer bir sigorta mahsul sigortasıdır. Bunların yanı sıra kasırga koruması, Kripto Destekli Krediler İçin Teminat Koruması, sosyal sigorta ve kripto para cüzdan sigortası çözümleri üzerinde de çalışmalar yapar.

Singapur temelli bir platform, bankacılık sistemi dışındaki kişilere sigorta hizmetleri sağlamak ve gelişmekte olan ülkelerdeki tüketiciler ile sigorta sağlayıcıları arasındaki uçurumu kapatmayı hedefleyen bir proje geliştirmiştir.

Amerikan menşeli bir sigorta şirketi, çiftçilere uygun fiyatlı iklim sigortası sağlamak için blok zinciri teknolojisi ve kripto para birimi kullanacak parametrik iklim risk transferi girişimi başlatır. İklim sigortası merkezî olmayan uygulama olarak tasarlanarak böylece çiftçilerin küresel sabit paralar ile yerel para birimlerini kullanarak cep telefonlarından kolaylıkla ödeme yapmaları ve almaları hedeflenir.

11.6. TELİF HAKLARININ KORUMASINDA BLOK ZİNCİRİ

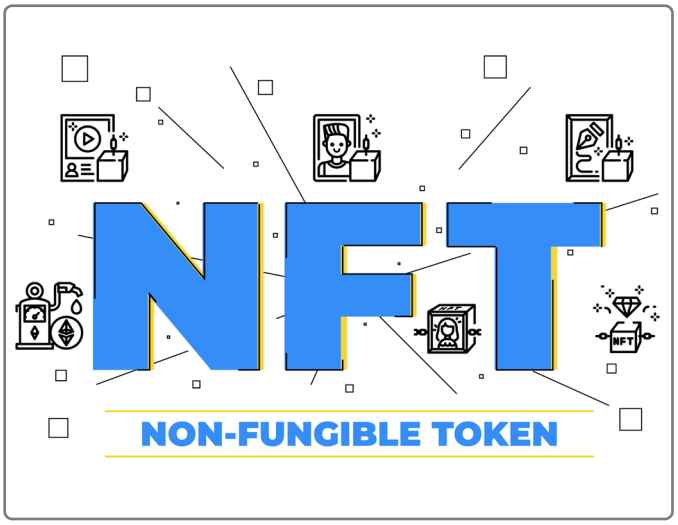
Blok zinciri teknolojisinin kullanıldığı bir diğer alan ise fikrî mülkiyettir. Telif hakları veya fikrî haklar olarak söz edilen **fikrî mülkiyet**, bir eser üzerinde sahip olunabilecek maddi ve manevi hakların tamamını ve komşu haklarını ifade eder. Fikrî mülkiyet, bir kişiye veya kuruluşa ait olan fikir ürünüdür (Görsel 11.9).



Görsel 11.9: Fikrî mülkiyet

Eser sahibi eserini paylaşmak veya kullanımını belirli biçimlerde kontrol etmek isteyebilir. Bu da eserin veya fikrin çalınması, başka kişiler tarafından hak iddia edilmesi riskini taşır. Telif haklarında blok zinciri teknolojisinin kullanılması ile telif hakkı ihlalleri güvence altına alınır. Blok zincirinin değiştirilemez sahiplik geçmişi sağlama özelliği, eser sahibi dışındaki kişilerin o eser üzerinde hak iddia etmesini engeller.

Nitelikli Fikrî Tapu [Non-fungible token'lar (NFT)], günümüzde sıkça kullanılan kripto paralardan bir tanesidir. NFT'ler değiştirilemez varlıklardır. Diğer kripto para birimlerinden farkı kendilerine ait blok zincirlerinin olmamasıdır. NFT'ler resim, ses, video vb. herhangi bir dijital çalışmanın değiştirilemez formatını temsil etmek amacıyla kullanılır. Bu dijital dosyalarla bir eserin diğer kopyalarından ayrılması sağlanır (Görsel 11.10).



Görsel 11.10: NFT

Dijital ortamdaki sanat eseri sahiplerinin karşılaştığı sorunlar; esere ve sahibine ilişkin bilgilerin şeffaf olmaması, sahtecilik ve eser sahiplerine ödenen ücretlerin tahsilatının zorluğudur. Blok zinciri, bu sorunlara eser ve sahibi hakkında şeffaf bilgiler sunarak, eserlerin dijital kopyaları üzerinde kontrol sağlayarak, otomatik ödeme imkânı sunarak ve lisanslama işlemlerini basitleştirerek çözüm getirir. Blok zinciri sisteminde, kişilerin ürettikleri sanat eserleri üzerindeki eser sahipliği aynı anda kayıt altına alınır ve sistemi kullanan herkes eser sahibinin bilgisine ulaşabilir.

NFT üzerine işlenen grafiklerden dolayı telif hakkı ile korunur. Telif hakkına tabi olan sanat eserleri de NFT'ler üzerine işlenebilir. NFT'ler sayesinde eseri satın alan kişi, eserin orijinal olduğundan ve eseri sahibinden satın aldığından emin olur.

SIRA SİZDE

Küçük gruplar oluşturarak NFT pazaryerleri ile ilgili sunu hazırlayınız. Hazırladığınız sunuyu arkadaşlarınızla paylaşınız.

İki Türk girişimci tarafından **nitelikli otorite** ve **blok zinciri protokollerini** tek bir platformda toplayan, global ve yerel yasal deliller oluşturulmasına yardımcı olan, notere gitmeye gerek kalmadan her türlü eser ve çalışmayı güvence altına almayı sağlayan bir proje oluşturmuştur. Proje; fikir, makale, şiir, fotoğraf, ses kaydı vb. çalışmaların dünya çapında güvence altına alınmasını sağlar. Eserin kime ait olduğu, ne zaman yapıldığı gibi bilgileri tüm dünyaya kanıtlayan belgeye sahip bir projedir.

SIRA SİZDE

Bir dijital varlığa (resim, fotoğraf, karikatür, blog yazısı vb.) ilişkin kendinize ait bir NFT oluşturunuz.

İtalyan Yazarlar ve Yayıncılar Derneği (SIAE), bir blok zinciri platformu ile bir işbirliği yaparak telif hakları ile ilgili bir sistem geliştirilmeye başlar. Projede Roma Üniversitesi ile bir danışmanlık şirketin katkıları yer alacaktır. Blok zinciri platformunun ağında yer alacak bu yeni sistemle telif hakları ve korsan yayıncılık ile daha iyi mücadele edilmesi sağlanabilir.

Blok zincirindeki şeffaflık ve güvenilirlik, blok zinciri altyapısından yararlanılan dijital sanat eserleri için de değer kazanır. Amerikalı bir dijital sanatçıya ait eser, blok zinciri üzerine kaydedilmiş eserlere örnek verilebilir. Sanatçı, 5.000 gün boyunca çizdiği resimleri JPG formatında bir araya getirerek yaklaşık 69 milyon dolara satmıştır. Eserin NFT olması onun pek çok kopyası oluşturulsa da diğerlerinden ayırt edilmesine imkân sunar. Yani Dolar, kripto para veya altın gibi değişebilen varlıkların aksine "değiştirilemez" varlıkların her biri benzersiz ve bir tane olarak yaratılır. Dolayısıyla birbirinden farklı olan bu varlıkların takası da mümkün olmaz. Bu durumda, üzerinden ne kadar süre geçerse geçsin, sanatçıya ait olan eser orijinalliğini yitirmeyip her zaman ayırt edilebilir kalır.

SIRA SİZDE

Ülkemizde ve dünyadaki NFT sanatçıları ile ilgili bir sunum hazırlayınız. Sunumlarınızı sınıfta arkadaşlarınızla paylaşarak bulduğunuz isimleri listeleyiniz.

11.7. KAMU YÖNETİMİNDE BLOK ZİNCİRİ

Dünyadaki teknolojik gelişmelerle birlikte kamu yönetimi alanında değişim ve dönüşüm daha fazla önem kazanır. Devletlerin dijitalleşmesiyle birlikte kamu yönetimi alanında e-devlet, e-eğitim, akıllı şehirler, büyük veri yönetimi gibi kavramların önemi artmıştır. E-devlet sistemleri, devlet veri tabanlarını kullanır. Bu sistemler, içeriğindeki zenginleştirilmiş verilerden dolayı siber saldırıların başlıca hedeflerinden olabilir. Bunun yanı sıra e-devlet otomasyon sistemleri kamu görevlilerinin iş yükünü hafifletmeye ve bürokratik süreçleri hızlandırmaya destek olur. Blok zinciri, dağıtık değişmez bir defter teknolojisi olma özelliğiyle hükümetlerin özellikle e-devlet projelerini uygulayabilmeleri için uygun bir teknolojidir. Akıllı şehirler oluşturma, şeffaflık ve hesap verebilirlik konularında birçok kolaylık sağlar (Görsel 11.11).

Görsel 11.11: Kamu yönetiminde blok zinciri teknolojisi

Sanayi ve Teknoloji Bakanlığının 2019 yılında yayınladığı, **Sanayi ve Teknoloji Stratejisi 2023** raporunda, ulusal blok zinciri altyapısının geliştirilmesi ve blok zinciri tabanlı ağ oluşturulmasının teşvik edilmesi kararlaştırılır ve önceliğin blok zinciri altyapısına taşınabilecek kamu merkezli uygulamalardan (tapu kayıt, diploma, gümrük uygulamaları vb.) seçilmesine değinilir.

Türkiye Strateji ve Bütçe Başkanlığı tarafından hazırlanan **Türkiye 11. Kalkınma Planı'nda**; elektrik şebekeleri, elektrik santralleri, donanım ve yazılım gibi blok zinciri altyapısı için gerekli yatırımların yapılacağı belirtilir. Ayrıca, blok zincirinin uygulanması konusunda, gümrük hizmetleri ve ulaşım alanlarına öncelik verilmesi ve Merkez Bankası Dijital Para biriminin piyasaya sürülmesinin planlaması yapılır.

SIRA SİZDE

Beşer kişilik gruplar oluşturarak dijital para ile kripto para arasındaki farklılıkları anlatan bir sunum hazırlayınız ve arkadaşlarınızla paylaşarak ortak bir liste oluşturunuz.

Türkiye Ticaret Bakanlığı, bir kamu kurumu olarak Blockchain Türkiye Platformu'nun ilk üyesidir. 2019 yılında Bakanlık bünyesinde kurmuş olduğu ilk resmî blok zinciri birimi ile Bakanlığın, özel sektörün uygulamış olduğu yeni teknolojik gelişmelerden geri kalmamasını hedefler. Blok zinciri teknolojileriyle kurumlar arası koordinasyonun da yeni bir boyut kazanması sağlanacaktır. Blok zinciri birimi, ithalat ve ihracat uygulamalarına öncelik verecektir.

Kamu yönetiminde blok zinciri projelerinden biri de **akıllı şehir** projeleridir. Ülkemizde akıllı şehirlerle ilgili birçok proje mevcuttur. Bir telekomünikasyon firması ve bir bilişim çözümleri firmasının Karaman'daki projesi olan akıllı şehir projesi veya Sakarya, Kayseri ve Gaziantep illerini akıllı şehirlere dönüştürmeyi hedefleyen **2023'e Kadar Türkiye'de Üç Akıllı Şehir Oluşturma** projeleri bunlardan bazılarıdır.

Hükümetler, gerek devlet veri tabanlarının siber saldırıların bir numaralı hedefi olması nedeniyle gerek vatandaşlarının mahremiyetlerini korumak maksadıyla blok zinciri veri tabanından yararlanmaya çalışır. Kamu yönetiminde blok zinciri kullanımı alanında Kore Cumhuriyeti, Estonya, Avustralya, Birleşik Krallık ve İsrail öncü ülkelerdir. Estonya ise blok zinciri teknolojisini ulusal düzeyde kullanan ilk ülkedir.

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerde parantez içine yargılar doğru ise "D", yanlış ise "Y" yazınız.

1. ( ) Tedarik zincirlerinin şeffaf olmayan bir yapısı vardır.

2. ( ) Blok zinciri teknolojisini kullanan gıda uygulamaları ile gıdanın kaynağına ulaşmak mümkündür.

3. ( ) Blok zinciri İlaç Takip Sistemi düşük bir güvenliğe sahiptir.

4. ( ) NFT'ler değiştirilebilir bir yapıdadır.

B) Aşağıdaki cümlelerde boş bırakılan yerlere doğru sözcükleri yazınız.

5. Türkiye'de ilaç tedarik zincirinin takibi ve izlenmesi için kullanılan ..... teknolojisi Sağlık Bakanlığına bağlı bir uygulamadır.

6. Elektrikli araçların batarya dolumu için ..... kullanılır.

KONULAR

12.1. HYPERLEDGER PLATFORMU

12.2. NEO PLATFORMU

NELER ÖĞRENECEKSİNİZ?

- Hyperledger platformunu kullanma
- Neo platformunu kullanma

ANAHTAR KELİMELER

Hyperledger, Neo

HAZIRLIK ÇALIŞMALARI

1. Özel blok zinciri platformuna neden ihtiyaç duyulmuş olabilir? Düşüncelerinizi arkadaşlarınızla paylaşınız.

2. Özel blok zinciri platformlarında bildiğiniz hangi programlama dilleriyle akıllı sözleşme yazılabilir? Arkadaşlarınızla tartışınız.

YENİ NESİL BLOK ZİNCİRİ PLATFORMLARI

12. ÖĞRENME BİRİMİ

12.1. HYPERLEDGER PLATFORMU

Hyperledger 2016 yılında yayınlanan açık kaynak kodlu bir blok zinciri projesidir. 12 alt projeden oluşmaktadır. Bunlar; Burrow, Fabric, Grid, Indy, Iroha, Sawtooth, Caliper, Cello, Composer, Explorer, Quilt, Ursa'dır. Projenin üzerinde en yoğun çalışılan projesi ise Hyperledger Fabric'dir.

12.1.1. Hyperledger Fabric Mimarisi

Hyperledger Fabric izinli blok zincirlerini çalıştıran, modüler ve dağıtık bir işletim sistemi olarak ifade edilir**. Hyperledger fabric**'in, dağıtık yapısı üzerinde çalışan akıllı sözleşmeler çalıştırabilir. Modüler yapısı sayesinde ise birçok fonksiyon ve uygulamaları içermektedir.

Hyperledger Fabric izinli, dağıtık defter platformunda kimlik doğrulama ve yetkilendirme için güçlü bir güvenlik altyapısı bulunmaktadır.

12.1.2. Üst Seviye Mimari ve Teknolojiler

Linux Vakfı'nın desteği ile oluşturulan Hyperledger Fabric platformunun yapısı izinli bir blok zinciridir. Sistemdeki tüm katılımcıların tanımlandığı ve sorunları çözmek için uygun bir yönetim şekli ile bir araya geldiği bir sistemdir. Hyperledger Fabric, ortak amaç sahibi, birbirine tam olarak güvenmeyen katılımcılar arasında güvenli etkileşim sağlar. Karşılıklı çalışan eşlerin kimliklerini kullanarak, geleneksel Bizans hata toleransı (BFT) konsensüsünü kullanır.

Hyperledger fabric dağıtık defter yapısı eşler arası çalışır. Bunlar doğrulayıcı ve doğrulayıcı olmayan eşlerdir.

- **Doğrulayıcı Eş:** Konsensüsün yürütülmesinden, işlemlerin doğrulanmasından ve defterin güncellenmesinden sorumlu olan düğümdür.
- **Doğrulayıcı Olmayan Eş:** Alıcıları doğrulama yapan çiftlere bağlamak için vekil olarak çalışan düğümdür. Doğrulayıcı olmayan eş işlemleri gerçekleştirmez fakat onları doğrulayabilir.

Doğrulayıcı eşler şu BFT konsensüs protokolünü yürütür:

1. İşlemi Uygula

Blok zinciri kodu eşlere kurulur ve bu kod çağrılmaya hazırdır. Akıllı bir sözleşme kodunu uygulanır.

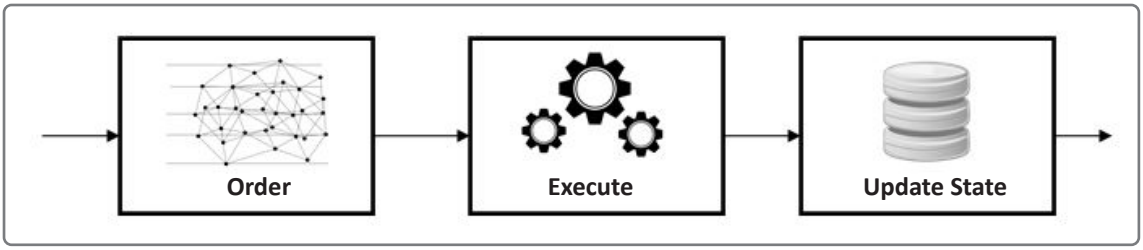
2. İşlem Çağırma

Daha önce yüklenen işlemi belirli bir zincir kodundan çağırır. İşlemler, işlemin türüne özeldir.

3. Sorgu İşlemi

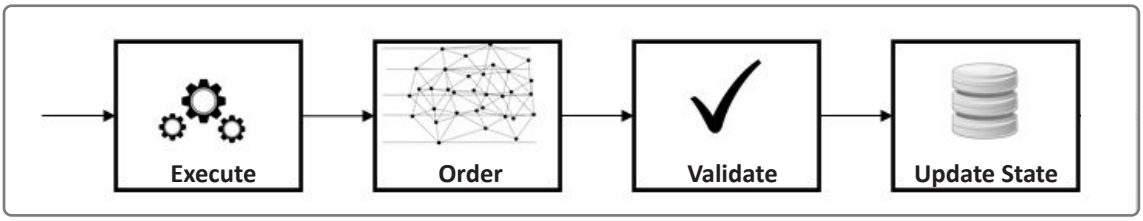
Eşin kalıcı durumunu doğrudan okuyarak durumun kaydı döndürülür.

Görsel 12.1'de gösterilen geleneksel işlem mimarisindeki işlem akışında gerçekleştirilecek işlemler önce sıralanıp, sonrasında çalıştırılır. Esnek çoğaltılmış hizmetler oluşturmak için geleneksel blok zinciri mimarisi kullanılır.



Görsel 12.1: Geleneksel işlem akışı

Görsel 12.2'de gösterilen Hyperledger Fabric İşlem Mimarisi'nde ise işlemler çalıştırılır sonrasında sıralıp doğrulanır ve durum güncellemesi yapılır.



Görsel 12.2: Hyperledger Fabric işlem akışı

SIRA SİZDE

HyperledgerPlatformu ile geliştirilmiş projeleri araştırınız ve projelerin kullanım amaçlarını içeren sunum hazırlayınız.

12.2. NEO PLATFORMU

Neo platformu akıllı sözleşmeler (NeoContracts) kullanarak yazılımları yürütmek ve merkezi olmayan uygulamalar (DApps- Decentralised Applications) tasarlamak için kullanılır. Ether para biriminin Ethereum ağı üzerinde kullanılması gibi, GAS da NEO ağında kullanılmaktadır. NeoContracts, geliştiricilerin yeni bir dil öğrenmek yerine mevcut dilleri (C#, Python, Go, TypeScript, and Java) kullanarak uygulama oluşturabilmelerini sağlar. Bu açıdan diğer akıllı sözleşme tabanlı protokollerden farklıdır. NeoContract ile farklı programlama dillerinde, DApp'ler oluşturulur. Geliştirici havuzu bu nedenle büyüktür.

12.2.1. Neo Platformu Pyhton Akıllı Sözleşme

Görsel 12.3'te görüldüğü gibi Neo platformunda Pyhton dili kullanılarak akıllı sözleşme oluşturmak mümkündür.

```
from boa3.builtin import NeoMetadata, metadata, public
from boa3.builtin.interop import storage

@public
def Main():
        storage.put('hello', 'world')

@metadata
def manifest() -> NeoMetadata:
      meta = NeoMetadata()

       meta.author = "COZ in partnership with Simpli"
       meta.email = "contact@coz.io"
       meta.description = 'This is a contract example'
       return meta
```

Görsel 12.3: Python akıllı sözleşme örneği

SIRA SİZDE

Neo platformunda Phyton dili kullanılarak yazılmış akıllı sözleşme uygulamalarını araştırıp, bulduğunuz örnek uygulamaları kendi bilgisayarınızda çalıştırınız.

12.2.2. Neo Platformu C# Akıllı Sözleşme

Görsel 12.4'te görüldüğü gibi Neo platformunda C# dili kullanılarak akıllı sözleşme oluşturmak mümkündür.

```
using Neo.SmartContract.Framework;
using Neo.SmartContract.Framework.Native;
using Neo.SmartContract.Framework.Services;
using System;
using System.Numerics;

namespace Neo.SmartContract.Examples
{
    [ManifestExtra("Author", "Neo")]
    [ManifestExtra("Email", "dev@neo.org")]
    [ManifestExtra("Description", "This is a NEP17 example")]
    [SupportedStandards("NEP-17")]
    [ContractPermission("*", "onNEP17Payment")]
    public partial class NEP17Demo : Nep17Token
    {
        [InitialValue("NhGobEnuWX5rVdpnuZZAZExPoRs5J6D2Sb",  ContractParameterType.
        Hash160)]
        private static readonly UInt160 owner = default;
        // Prefix_TotalSupply = 0x00; Prefix_Balance = 0x01;
        private const byte Prefix_Contract = 0x02;
        public static readonly StorageMap ContractMap = new StorageMap(Storage.
        CurrentContext, Prefix_Contract);
        private static readonly byte[] ownerKey = "owner".ToByteArray();
        private static bool IsOwner() => Runtime.CheckWitness(GetOwner());
        public override byte Decimals() => 8;
        public override string Symbol() => "NEP17";

        public static void _deploy(object data, bool update)
        {
            if (update) return;
            ContractMap.Put(ownerKey, owner);
        }

        public static UInt160 GetOwner()
        {
            return (UInt160)ContractMap.Get(ownerKey);
        }
```

```
        public static new void Mint(UInt160 account, BigInteger amount)
        {
            if (!IsOwner()) throw new InvalidOperationException("No Authorization!");
            Nep17Token.Mint(account, amount);
        }

        public static new void Burn(UInt160 account, BigInteger amount)
        {
            if (!IsOwner()) throw new InvalidOperationException("No Authorization!");
            Nep17Token.Burn(account, amount);
        }

        public static bool Update(ByteString nefFile, string manifest)
        {
            if (!IsOwner()) throw new InvalidOperationException("No Authorization!");
            ContractManagement.Update(nefFile, manifest, null);
            return true;
        }

        public static bool Destroy()
        {
            if (!IsOwner()) throw new InvalidOperationException("No Authorization!");
            ContractManagement.Destroy();
            return true;
        }
    }
}
```

Görsel 12.4: C# akıllı sözleşme örneği

SIRA SİZDE

Neo platformunda C# dili kullanılarak yazılmış akıllı sözleşme uygulamaları araştırıp, bulduğunuz örnek uygulamaları kendi bilgisayarınızda çalıştırınız.

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

1. Aşağıdaki seçeneklerden hangisi Hyperledger Fabric işlem akışında yer almaz?

A) Çalıştır

B) Doğrula

C) Güncelle

D) Kopyala

E) Sırala

2. I. Akıllı sözleşme oluşturulabilir.

II. Geliştirici havuzu geniştir.

III. Merkezi uygulamalar geliştirilebilir.

IV. C# dili kullanılarak akıllı sözleşme yazılabilir.

V. Gas kullanılmaz.

NEO platformu için yukarıdaki ifadelerden hangileri doğrudur?

A) II-III B) III-V

C) I-II-V D) I-III-IV

E) I-II-IV

3. Geleneksel blok zinciri mimarisindeki işlem akışı nedir?

A) Sırala-Çalıştır-Güncelle

B) Kopyala-Çalıştır-Güncelle

C) Sırala-Kopyala-Çalıştır

D) Güncelle-Doğrula-Çalıştır

E) Doğrula-Kopyala-Çalıştır

KAYNAKÇA

Bilişim Teknolojileri Alanı Çerçeve Öğretim Programı

Abreu, P. W., Aparicio, M., & Costa, C. J. (2018, June). Blockchain technology in the auditing environment. In 2018 13th Iberian Conference on Information Systems and Technologies (CISTI) (pp. 1-6). IEEE.

Arvas, İ. S. (2022). Gutenberg galaksisinden meta evrenine: Üçüncü kuşak internet, Web 3.0. AJIT-e: Bilişim Teknolojileri Online Dergisi, 13(48), 53-70.

Avşar, İ. İ. (2020). Blok zinciri tabanlı yeni sosyal medya yaklaşımı. International Journal of Arts and Social Studies, 3 (4), 17-33.

Aydar, M., & Çetin, S. C. (2020). Blokzincir Teknolojisinin Sağlık Bilgi Sistemlerinde Kullanımı. Avrupa Bilim ve Teknoloji Dergisi, (19), 533-538.

Ayhan, E., Aytekin, M., & Güvener, H. (2021). Türkiye'de ilaç tedarik zincirinde kullanılan ilaç takip sistemi ile blok zincir tabanlı ilaç tedarik zinciri uygulamalarının karşılaştırılması (ss. 318-326). 7. International Istanbul Scientific Research Congress, December 18-19, 2021.

Babaoğlu, C., & Karasoy, H. (2022). Kamu Yönetiminde Blokzincir: Kullanım alanları ve örnek uygulamalar. Sosyoekonomi, 30(52), 283-297.

Bakan, İ., & Şekkeli, Z. H. (2019). Blok zincir teknolojisi ve tedarik zinciri yönetimindeki uygulamaları. OPUS International Journal of Society Researches, 11(18), 2847-2877.

Bankalararası Kart Merkezi. (2020). Herkes için blokzincir. BKM Yayınları. https://bkm.com.tr/wp-content/uploads/2015/06/herkes_icin_blokzincir_2020_web.pdf.

Barnett, B. J., Barrett, C. B., & Skees, J. R. (2008). Poverty traps and index-based risk transfer products. World Development, 36(10), 1766-1785. doi:10.1016/j.worlddev.2007.10.016.

Blockchain. (2021). Blockchain Boyutu (MB). https://www.blockchain.com/charts/blocks-size, Aralık 10, 2021, kaynağından alınmıştır.

Blockchain Türkiye Platformu, (2019). Bitcoin: Eşten-eşe Elektronik Nakit Ödeme Sistemi. https://bctr.org/wpcontent/uploads/2019/03/t%C3%BCrk%C3%A7e_bitcoin.pdf, Mayıs 10, 2022, kaynağından alınmıştır.

Blockchain Türkiye Platformu, Hukuk, Düzenlemeler ve Kamu İlişkileri Çalışma Grubu. (2019). Dünyada blokzinciri regülasyonları ve uygulama örnekleri karşılaştırma raporu - ŞUBAT 2019.

Blockchain Türkiye Platformu, Enerji Çalışma Grubu. (2019). Enerji sektöründe blokzinciri gelişmeleri- KASIM 2021. Türkiye Bilişim Vakfı. https://bctr.org/dokumanlar/Enerji_Sektorunde_Blokzinciri_Gelismeleri.pdf, Nisan 16, 2022, kaynağından alınmıştır.

Buterin, V. (2014). A next-generation smart contract and decentralized application platform. White paper. https://www.ethereum.org/ pdfs/EthereumWhitePaper.pdf/, Nisan 16, 2022, kaynağından alınmıştır.

Brownworth, A. (2021) SHA256 Hash [table]. https://andersbrownworth.com/blockchain/hash, Aralık 15, 2021, kaynağından alınmıştır.

Cleanpng, homepage. (2022). https://www.cleanpng.com/, Nisan 24, 2022, kaynağından alınmıştır.

CryptoKitties, homepage. (2022). https://www.cryptokitties.co/, Mart 28, 2022, kaynağından alınmıştır.

Cumhuriyeti Merkez Bankası (2018). Kâğıt Paranın Tarihçesi. https://www.tcmb.gov.tr/wps/wcm/connect/d189b219-fe71-40bf-9754-6a5f7d0a65eb/KagitParaTarihce. pdf?MOD=AJPERES, Mart 21, 2022, kaynağından alınmıştır.

Çekin, M. S. (2019). Borçlar Hukuku ile Veri Koruma Hukuku açısından blockchain teknolojisi ve akıllı sözleşmeler: Hukuk düzenimizde bir paradigma değişimine gerek var mı?. İstanbul Hukuk Mecmuası, 77(1), 315-341. doi: 10.26650/mecmua.2019.77.1.0012.

EPİAŞ. (2021). Çevresel piyasalar: YEK-G sistemi ve organize YEK-G piyasası tanıtımı. https://www.epias.com.tr/yek-g-piyasasi/yek-g-sistemi-ve-organize-yek-g-piyasasi-tanitimi/, Nisan 22, 2022, kaynağından alınmıştır.

Ethereum. (2022). https://ethereum.org/en/eth/, Ocak 30, 2022, kaynağından alınmıştır.

Ethereum. (2022). Blocks. https://ethereum.org/en/developers/docs/blocks/, Ocak 30, 2022, kaynağından alınmıştır.

Ethereum. (2022). Solidity Documentation: Release 0.8.16. https://buildmedia.readthedocs.org/media/pdf/solidity/develop/solidity.pdf, Nisan 24, 2022, kaynağından alınmıştır.

Ethereum. (2022). Sunumlar: ERC-20 token akıllı sözleşmesini anlamak. https://ethereum.org/tr/developers/tutorials/understand-the-erc-20-token-smart-contract/, Nisan 30, 2022, kaynağından alınmıştır.

Etherum. (2022). Solidity DocumentationRelease 0.8.17.https://buildmedia.readthedocs.org/ media/pdf/solidity/develop/solidity.pdf, Ağustos, 16, 2022, kaynağından alınmıştır.

Hackmamba, homepage. (2022). https://hackmamba.io/, Nisan 5, 2022, kaynağından alınmıştır.

Hyperledger Foundation, homepage. (2022). https://www.hyperledger.org/, Şubat 4, 2022, kaynağından alınmıştır.

IBM. (2022). IBM Blockchain. https://www.ibm.com/blockchain, Mart 4, 2022, kaynağından alınmıştır.

IBM Research Editorial Staff. (2022). Blockchain: Behind the architecture of hyperledger fabric. https://www.ibm.com/blogs/research/2018/02/architecture-hyperledger-fabric/, Şubat 4, 2022, kaynağından alınmıştır.

İslam, A. (2019). Blok zinciri teknolojisi ve kripto paralar: Mevcut durum, potansiyel ve risk analizi (Yayımlanmamış yüksek lisans tezi). Marmara Üniversitesi Sosyal Bilimler Enstitüsü, İstanbul.

Kapan, K., & Üncel, R. (2020). Gelişen web teknolojilerinin (Web 1.0, Web 2.0, Web 3.0) Türkiye turizmine etkisi. Safran Kültür ve Turizm Araştırmaları Dergisi, 3(3), 276-289.

Karahan, Ç., & Tüfekci, A. (2020). Blokzincir teknolojisinin dijital kimlik yönetiminde kullanımı: Bir sistematik haritalama çalışması. Politeknik Dergisi, 23(2), 483-496.

Kaya, H. (2019). Sektörel ve operasyonel blokzincir uygunluk analizlerinde kullanılacak kriterlerin belirlenmesi (Yayınlanmamış yüksek lisans tezi). Ankara Yıldırım Beyazıt Üniversitesi Sosyal Bilimleri Enstitüsü, Ankara.

Kerela Blockchain Academy. (2022). All courses. https://learn.kba.ai/course/, Mart 28, 2022, kaynağından alınmıştır.

KindPNG, homepage. (2022). https://www.kindpng.com/, Nisan 24, 2022, kaynağından alınmıştır.

Koç Sistem. (2022). Blockchain. https://kocsistem.com.tr/trending-subjects/blockchain/, Nisan 10, 2022, kaynağından alınmıştır.

Köse, B. Ö. (2021). Sağlıkta blok zincir. İçinde N. Bozbuğa & S. Gülseçen (Eds.), Tıp Bilişimi [Medical Informatics] (ss. 367-398). İstanbul: İstanbul Üniversitesi Yayınevi. doi: 10.26650/B/ET07.2021.003.19.

Lamport, L., Shostak, R., & Pease, M. (2019). The Byzantine generals problem. In D. Malkhi (Ed.), Concurrency: the works of leslie lamport (pp. 203-226). doi: org/10.1145/3335772.3335936 Loom Network. (2022). The curriculum. https://cryptozombies.io/en/course, Nisan 4, 2022, kaynağından alınmıştır.

Mattila, J. (2016, Mayıs). The blockchain phenomenon – the disruptive potential of distributed consensus architectures. (ETLA Working Papers No 38). http://pub.etla.fi/ETLA-Working-Papers-38.pdf, Aralık 1, 2021, kaynağından alınmıştır.

Mechkaroska, D., Dimitrova, V., & Popovska-Mitrovikj, A. (2018, November). Analysis of the possibilities for improvement of blockchain technology. In 2018 26th Telecommunications Forum (TELFOR) (pp. 1-4). IEEE.

Mükerrer S: 31456. https://www.resmigazete.gov.tr/eskiler/2021/04/20210416-4.htm, Mart 7, 2022, kaynağından alınmıştır.

Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. Decentralized Business.

Narin, İ. C., & Gürüf, E. (2022). Turkey: NFT Sanat Eserlerinin Telif Hukuku Kapsamında Değerlendirilmesi [online]. https://www.mondaq.com/turkey/copyright/1164424/nft-sanat-eserlerinin-telif-hukuku-kapsam305nda-de287erlendirilmesi, Nisan 22, 2022, kaynağından alınmıştır.

Neo, homepage. (2022). https://neo.org/, Şubat 4, 2022, kaynağından alınmıştır.

Ödemelerde kripto varlıkların kullanılmamasına dair yönetmelik. R.G.: 16 Nisan 2021, Mükerrer S: 31456. https://www.resmigazete.gov.tr/eskiler/2021/04/20210416-4.htm, Mart 7, 2022, kaynağından alınmıştır.

Osborne, M. J. (2004). An introduction to game theory. New York: Oxford University Press.

Özaltın, O., & Ersoy, M. (2020). Kamu yönetiminde blokzincir kullanımı: D5 örneği. Nevşehir Hacı Bektaş Veli Üniversitesi SBE Dergisi, 10(2), 746-763. doi: 10.30783/nevsosbilen.748379.

Pekdemir, E. (2021). The use of blockchain technology in public administration: Implications for Turkey (Yayımlanmamış yüksek lisans tezi). Graduate School of Social Sciences, Middle East Technical University, Ankara.

Pngwing, homepage. (2022). https://www.pngwing.com/, Nisan 24, 2022, kaynağından alınmıştır.

Proofstack, anasayfa. (2022). https://tr.proofstack.io/index.html, Mart 13, 2022, kaynağından alınmıştır.

Remix IDE, homepage. (2022). https://remix.ethereum.org/, Mayıs 3, 2022, kaynağından alınmıştır.

Review [online]. https://bitcoin.org/bitcoin.pdf, Kasım 4, 2021, kaynağından alınmıştır.

Soliditylang. (2022). Units and globally available variables. https://docs.soliditylang.org/en/v0.4.24/units-and-global-variables.html, Nisan 4, 2022, kaynağından alınmıştır.

Solidity by Example. (2022). https://solidity-by-example.org/,Nisan 4, 2022, kaynağından alınmıştır.

Taşdemir, T. (2021). Yenilenebilir Enerji Kaynak Garanti (YEK-G) Sistemi ve Organize YEK-G Piyasası [Powerpoint sunum]. https://www.epias.com.tr/wp-content/uploads/2021/03/EPIAS-YEK-G-Sunum_ETD-Toplanti-8.3.2021.pdf, Nisan 17, 2022, kaynağından alınmıştır.

Topcu, B. A., & Sarıgül, S. S. (2020). Dünyada ve Türkiye'de blok zinciri teknolojisi: Finans sektörü, dış ticaret ve vergisel düzenlemeler üzerine genel bir değerlendirme. Avrupa Bilim ve Teknoloji Dergisi, Ejosat Special Issue 2020 (ARACONF), 27-39. doi: 10.31590/ejosat.araconf5

TUBİTAK Bilgem. (2017). Blokzincir teknolojiler. http://blokzincir.tubitak.gov.tr/blok-zincir.html, Kasım 5, 2021, kaynağından alınmıştır.

Tunca, S., & Sezen, B. (2020). Sigorta işlemlerinde blokzincir (blockchain) teknolojisi uygulamaları. Bankacılık ve Sigortacılık Araştırmaları Dergisi, (14), 13-25.

Türk Dil Kurumu, noktalama işaretleri (açıklamalar). (t.y.). https://www.tdk.gov.tr/icerik/yazim-kurallari/noktalama-isaretleri-aciklamalar/.

Türk Dil Kurumu, sözlükleri. (t.y.). https://sozluk.gov.tr/.

Türk Dil Kurumu, yazım kuralları. (t.y.). https://www.tdk.gov.tr/kategori/icerik/yazim-kurallari/.

Türkiye Bilişim Vakfı. https://bctr.org/dokumanlar/Dunyada_Blokzinciri_Regulasyonlari.pdf, Nisan 16, 2022, kaynağından alınmıştır.

Türkiye Cumhuriyeti Merkez Bankası (2018). Kâğıt Paranın Tarihçesi. https://www.tcmb.gov.tr/wps/wcm/connect/d189b219-fe71-40bf-9754-6a5f7d0a65eb/KagitParaTarihce.pdf?MOD=AJPERES, Mart 21, 2022, kaynağından alınmıştır.

Türkiye Cumhuriyeti Sanayi ve Teknoloji Bakanlığı. (2019). 2023 sanayi ve teknoloji stratejisi. https://www.sanayi.gov.tr/assets/pdf/SanayiStratejiBelgesi2023.pdf, Nisan 22, 2022, kaynağından alınmıştır.

Türkiye Cumhuriyeti Ticaret Bakanlığı. (2019). Fikri mülkiyet hakları. https://ticaret.gov.tr/gumruk-islemleri/sikca-sorulan-sorular/ticari/fikri-mulkiyet-haklari, Şubat 23, 2022, kaynağından alınmıştır.

Usta, A., & Doğantekin, S. (2018). Blockchain 101v2. https://bctr.org/dokumanlar/Blockchain101v2r2.pdf, Mart 28, 2022, kaynağından alınmıştır.

Ünal, E. (2018). Bitcoin ve Blockchain Nedir? Nasıl Çalışır?_ [blog]. https://enginunal.medium.com/bitcoin-ve-blockchain-nedir-nas%C4%B1l-%C3%A7al%C4%B1%C5%9F%C4%B1r-78d5c9e28095, Şubat 5, 2022, kaynağından alınmıştır.

Yener, E. (2020). Dijital girişimcilikte blok zincir teknolojilerinin rolü ve bir model önerisi: Blok zincir tabanlı ikinci el araç alım satım platformu (Sechandchain). (Yayınlanmamış yüksek lisans tezi). İstanbul Medipol Üniversitesi Sosyal Bilimler Enstitüsü, İstanbul.

*** Kaynakça, APA6 referanslama sistemi kullanılarak oluşturulmuştur.**

GENEL AĞ KAYNAKÇASI VE GÖRSEL KAYNAKÇASI

http://kitap.eba.gov.tr/karekod/Kaynak.php?KOD=2421

CEVAP ANAHTARLARI

1. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** D **2.** Y **3.** Y **4.** Y **5.** D **6.** D

B) **7.** E **8.** D **9.** C **10.** A **11.** D **12.** B **13.** D

2. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** D **2.** D **3.** Y **4.** Y

B) **5.** A **6.** D **7.** B **8.** C **9.** E **10.** C

3. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** B **2.** B **3.** C **4.** A **5.** E

4. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** Y **2.** D **3.** Y **4.** D

B) **5.** C **6.** B **7.** D

5. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** D **2.** B **3.** C **4.** E **5.** E

6. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** A **2.** C **3.** A

7. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** E **2.** D **3.** A

8. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.B** **2.** D **3.** B **4.** B **5.** E

9. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** D **2.** E **3.** A **4.** B **5.** C

10. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** D **2.** C **3.** E **4.** B

11. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** D **2.** D **3.** Y **4.** Y

B) **5.** İTS (İlaç Tedarik Sistemi) **6.** EŞARJ

12. ÖĞRENME BİRİMİNİN CEVAP ANAHTARI

A) **1.** D **2.** E **3.** A