

Snovanje in razvoj NFT (non fungible token)

Vid Keršič, Urban Vidovič, Andraž Vrečko, Muhamed Turkanović



Univerza v Mariboru

Fakulteta za elektrotehniko,
računalništvo in informatiko



INŠTITUT ZA
INFORMATIKO

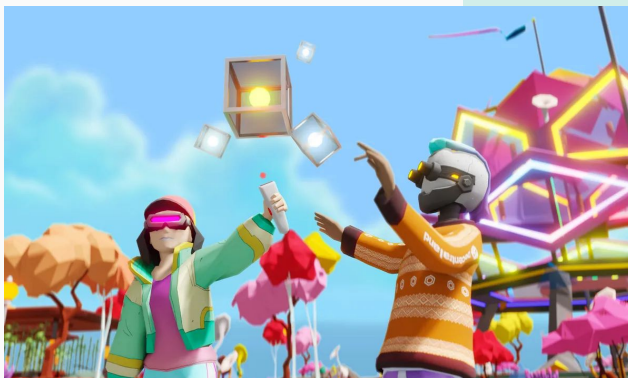
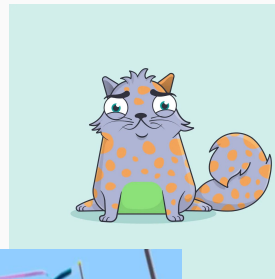


Agenda

1. Uporaba NFT
2. Kaj so NFT-ji?
3. Tehnologija veriženja blokov
4. Kriptografija, kriptovalute, transakcija, blok
5. Pametne pogodbe
6. Standardi in tehnologije, ki omogočajo NFT-je
7. Zakaj deluje?
8. Prednosti (in slabosti)
9. Prihodnost
10. Praktična delavnica

Uporaba NFT

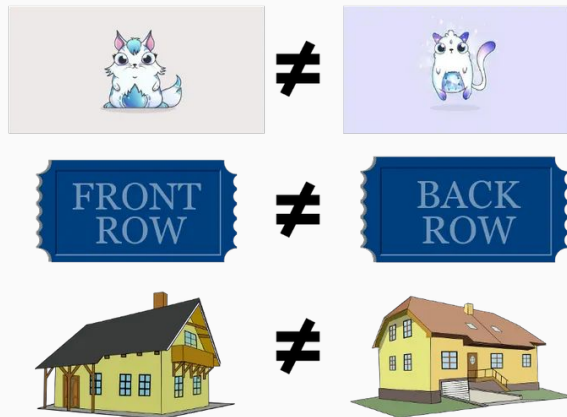
- Igre
- Umetnost
- Članstvo (v klubu, v restavraciji, ...)
- Delno lastništvo
 - Podjetja
 - Nepremičnin
- Potrdilo o udeležbi
- Metaverse



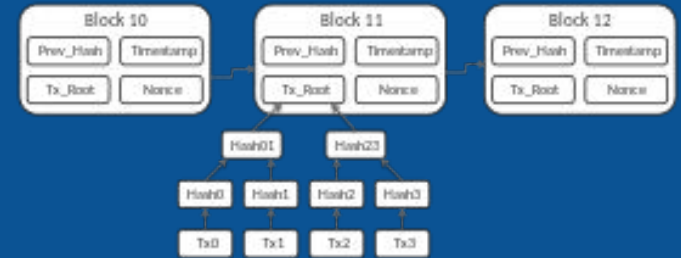
Kaj so NFT-ji?

- **Non-Fungible Tokens**, nezamenljivi žetoni
- Dokazilo o (digitalnem) lastništvu
 - Fizičnega ali digitalnega produkta
- Podatki/programska koda shranjena na blockchain-u
 - Zaščiteno s tehnologijo veriženja blokov
 - Samo uporabnik lahko "dostopa" do žetonov
- Vsak žeton je unikatnen
 - Različna slika
 - Različna zaporedna številka (ID)

Non-fungible



Tehnologija veriženja blokov

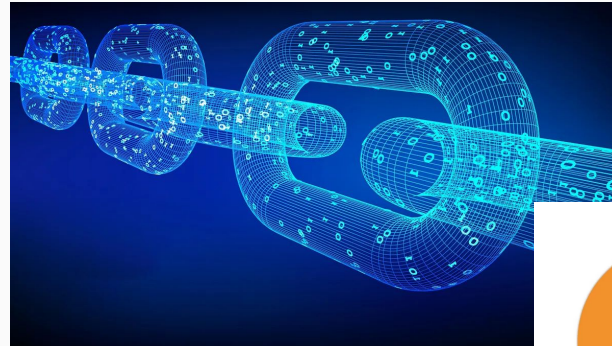


- Porazdeljena knjiga transakcij

- Bloki, ki so povezani v verigo
- Kriptografsko zaščitena
- Nespremenljiva
- Javna in transparentna
- Replikacija podatkov
- Odporna na "zlonamerne" akterje

- Generacije blockchain-ov

- **1. generacija:** Bitcoin (2008)
- **2. generacija:** Ethereum (2013)
- **3. generacija (?)**: Ethereum 2.0 + L2, Cardano, IOTA, Cosmos, Polkadot, ...

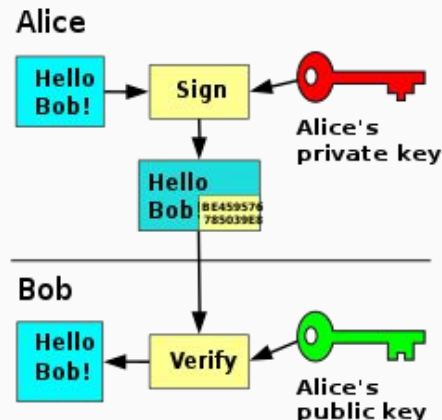


Tehnologija veriženja blokov

- Sestavni deli
 - Kriptografija
 - Kriptovaluta
 - Naslov
 - Transakcija
 - Blok
 - Omrežje

Kriptografija

- Asimetrična kriptografija
 - Zasebni in javni ključ
 - Digitalno podpisovanje podatkov: podpišem z zasebnim ključem, vsak lahko preveri z mojim javnim ključem
- Zgoščevalne funkcije
 - Enosmerna funkcija
 - Enolični izhod funkcije glede na vhodne vrednosti
 - V obratno smer (izhod - > vhod) računsko neizvedljivo
 - Rezultat - zgoščena vrednost (hash)
- Javni naslov
 - “poštni nabiralnik” blockchain-a
 - Izračun iz vrednosti javnega ključa



Kriptovaluta

- Kriptovaluta, digitalni denar, digitalni žeton
- Dva poglobitna namena:
 - Prenos vrednosti
 - Plačilo provizije (angl. transaction fee or gas)
- Cena v evrih/dolarjih nima vpliva na delovanje samega blockchain-a



Naslov

- Angl. *account*
- Izračunan iz javnega ključa
- Na blockchain-u zapisano, kaj ima določen naslov v lasti
 - Kriptovalute
 - NFT
 - ...

Transakcija

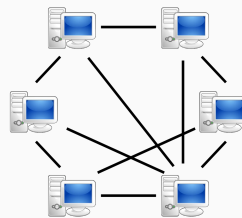
- Oseba A je poslala X žetonov osebi B
- Oseba A digitalno podpiše transakcijo
- Transakcija
 - Prenos "native" kriptovalute
 - Izvedba programske kode na pametni pogodbi
- Dodana provizija
- Transakcije različnih uporabnikov združene v blok

Blok

- Skupek transakcij
- Povezava na prejšnji blok
- Nagrada za izdelavo bloka v kriptovaluti
- Ostali podatki: težavnost, število transakcij, nagrada, rudar (angl. *miner*)

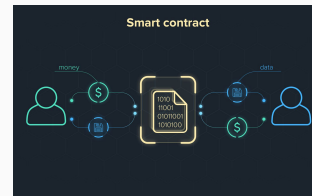
Omrežje

- Distribuirano/decentralizirano omrežje
 - Skupek vozlišč
 - Vozlišče = računalnik/strežnik
 - Vsako vozlišče hrani repliko vseh blokov in trenutnega stanja naslovov
- Varnost omrežja dosežena z algoritmom soglasja
 - Kako se vozlišča dogovorijo o zaporedju blokov?
 - Kako preprečiti dvojno porabo kriptovalute (angl. double-spending problem)?
 - Proof of work in proof of stake
 - Proof of work: iskanje takšne zgoščene vrednosti, ki se bo začela z x 0, pri čemer bo vhodna vrednost blok in naključno število (potrebno je preizkusiti VELIKO različnih možnosti)
 - Proof of stake: izdelovalci blokov se menjavajo, vsi, ki založijo za določen čas kriptovaluto lahko sodelujejo
- Glavno omrežje, testno omrežje



Pametne pogodbe

- Programska koda, ki “živi” na blockchain-u
- Stanje (navadno mapping (address → value)) in funkcije s programsko kodo
- Enaka programska koda se izvede na vseh vozliščih v omrežju
 - Koda povzroči veljavno spremembo stanja -> sprememba sprejeta, sicer ne
- Najbolj znan virtualni stroj za izvajanje pametnih pogodb EVM (Ethereum Virtual Machine)
 - Določene osnovne operacije (na nivoju zbirnega jezika - angl. assembler) ki se lahko izvedejo
 - Za vsako osnovno operacijo je določena provizija (glede na zahtevnost) - od skupne zahtevnosti operacij je odvisna provizija transakcije
 - Visokonivojsko programiranje → Solidity
 - Objektno orientirano programiranje
 - Jezik podoben JavaScript-u
 - Remix IDE



Pametne pogodbe

- Uporaba
 - Objava na blockchain (angl. *deploy*)
 - Transakcije vsebujejo ime funkcije in parametre
- Decentralizirana aplikacija (dApp)
 - Namesto podatkovne baze in spletnega strežnika -> pametna pogodba in blockchain stanje
- Programska koda je javna!
 - Napaka v programski kodo navadno vodi v nepravilno/nepredvideno delovanje in navadno v izgubo sredstev
- Pomembna učinkovita implementacija!!!
 - Manjši stroški za deploy in končne uporabnike (gas)



Standardi

- Programiranje decentraliziranih aplikacij je zlaganje “lego” kock
 - Transparentnost blockchain-a
 - Povezovanje pametnih pogodb
 - Grajenje na obstoječih aplikacijah
 - Interoperabilnost
- Interoperabilnost/povezovanje možno zaradi standardov
 - Standardni postopki implementacije
 - Pametne pogodbe za isti namen uporabe imajo enake funkcije in strukturo
- ERC (Ethereum Request for Comments) in EIP (Ethereum Improvement Proposals)
 - ERC20: standard za kriptožetone
 - ERC721: standard za NFT-je

ERC-721



ERC721/NFT

- Izšel leta 2018 (EIP 721)
- Drugi najbolj uporabljen standard
- Stanje pametne pogodbe
 - Lastništvo (mapping (id → address))
 - Metapodatki (angl. *metadata*): vrednosti, ki jih žeton predstavlja
 - Slike
 - Tekst
 - Video
 - 3D modeli, ...
 - Funkcije: mint, transfer, burn, ...
- OpenZeppelin - implementacija ERC721 standarda (dedovanje iz *interface*)

```
1 pragma solidity ^0.5.0;
2 //import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
3
4 contract my721 is ERC721{
5     mapping(address => uint) tokens;
6     function approval(address _owner, address _approved,uint _tokenId){
7         require(tokens[_owner]==_tokenId);
8         tokens[_approved]=_tokenId;
9     }
10    function transfer(address _to, uint _amount) public payable{
11        require(_amount <= tokens[msg.sender]);
12        tokens[msg.sender]-=_amount;
13        tokens[_to]+=_amount;
14    }
15    function balanceOf(address _owner) public view returns (uint){
16        return tokens[_owner];
17    }
18    function ownerOf(uint _tokenId) public view returns(address){
19        return tokens[_id].address;
20    }
21    function TransferFrom(address _from, address _to, uint _tokenId) payable{
22        require(tokens[_from]==_tokenId);
23        tokens[_from]=0;
24        tokens[_to]=_tokenId;
25    }
26    function approve(address _approved, uint _tokenId) payable{
27        require(tokens[msg.sender]==_tokenId);
28        tokens[_approved]=_tokenId;
29    }
30    function mint(address _to, uint _tokenId,) public{
31        tokens[_to] = 'mytoken '+str(uint(blockhash(block.number - 1)));
32    }
33 }
```


IPFS

- NFT-ji “zapisani” na porazdeljenem omrežju
- Kaj pa metapodatki?
 - Na blockchain ne, ker lahko veliko zasedajo → visoke provizije/fee-ji
- IPFS - InterPlanetary File System
 - P2P distribuiran datotečni sistem
 - Dostop do datotek preko zgoščenih vrednosti vsebine
 - Dve datoteki z isto vsebino \longleftrightarrow ista zgoščena vrednost
 - Metadata za NFT vsebuje zgoščeno vrednost



Zakaj deluje?

- Digitalno lastništvo mora biti suvereno in ne sme biti preprečeno
 - Decentralizirano omrežje - noben ne more ugasniti omrežja oz. omejiti dostopa
- Transparentnost
 - Vsak lahko preveri, kdo je trenutni lastnik digitalnega izdelka, kdo ga je izdelal in kdo vse so bili lastniki v preteklosti
 - Vsak lahko preveri programsko kodo
- Kriptografija
 - Samo lastnik privatnih ključev ima “dostop” do digitalnega izdelka
- Copy/paste?
 - Original od izumitelja bo vedno več vreden kot kopija (enako kot pri realnih izdelkih/slikah, ponaredki ...)
 - Za realne dogodke bodo implementirana “digitalna vrata”, samo lastnik (potrjen s kriptografijo) bo imel vstop

Prednosti (in slabosti)

- Prednosti

- Licenciranje digitalnih izdelkov (nobena entiteta nima v lasti omrežja - podobno kot interneta)
- Transparentnost in sledljivosti
 - Dokazljiva količina digitalnega izdelka
- Interoperabilnost med izdelki/projekti

- Slabosti

- Vpliv tehnologije na okolje
 - Ethereum 2.0, alternativni blockchain sistemi, ...
- Lažje izpeljive prevare
- Trgovalni mehurček?

Prihodnost

- Metaverse
 - Vse več storitev in načinov zabave je digitalnih
- Sprememba lastništva obstoječih digitalnih izdelkov
 - Na primer skin-ov v video igrah
- Promoviranje znamk in grajenje skupnosti
 - Znane (modne) znamke izdajo NFT žetone in gradijo znamko na NFT kolekcijah
 - Uporabniki (finančno) povezani z znamko
 - Z NFT kovanci lahko uporabnik prejme popuste v trgovini, posebne izdelke ...
- Članstvo v klubih in restavracijah
 - V letu 2023 se bo v New Yorku odprla restavracije le za imetnike določene NFT kolekcije
 - “prestiž”

Praktična delavnica