

Telepítési és konfigurálási útmutató

Megjegyzés: ez a típusú szoftver a számítógép és a böngésző felkészítése nélkül sajnos nem futtatható. A lokális tesztelés oka a mainnet-re való deployolás költsége. (Lásd a szakdolgozat 4.2.1 fejezetét.)

Megjegyzés: a konfigurálás végigkövethető az alábbi linken: <https://youtu.be/kJgh9KVk9SQ>

1. Futtassa a főmappában található NPM_Install.bat fájlt!
2. Nyisson egy Node.js konzolt, navigáljon a projekt főmappájába!
3. A Node.js konzolban futtasson *ganache-cli -a 5 -p 7545 -i 5777 -e 1000000 -m [...]*
allow try bridge height car train apple heist attack increase pillow --allowUnlimitedContractSize 100000000000 --db ganache-config parancsot, ahol a [...] helyére helyettesítsen egy tetszőleges angol szót, mely nem azonos az utána következő 11 egyikével sem! Ez határozza meg a mnemonik egyediségét! Ez az összetett parancs elindít egy lokális Ethereum node-ot a számítógépen. A parancs paramétereinek értelmezéséhez lásd: <https://www.npmjs.com/package/ganache-cli>
4. Nyisson egy új Node.js konzolt, navigáljon a projekt főmappájába, és futtasson *truffle migrate* parancsot! A *Deploying 'ArtworkToken'* blokkból másolja ki a *contract address*-t, és deklarálja az *src/app/services/web3-init-service.ts* 19. sorában az *artworkTokenAddress* változó értékeként!
5. Nyisson egy olyan böngészőt, amelyhez telepíthető a MetaMask bővítmény (például Google Chrome, Vivaldi) és telepítse a MetaMask bővítményt: <https://chrome.google.com/webstore/detail/metamask/nkbihfbeamoeaohlefnkodbefgpgknn>
6. Hozzon létre egy új pénztárcát! A biztonság kedvéért jegyezze fel a 12 szavas mnemonikot!
7. Kattintson a hálózatok legördülő menüjére és válassza az *Egyedi RPC* opciót! A hálózat neve legyen *Art Tokenization*, az RPC URL legyen *http://127.0.0.1:7545*, a lánc azonosítója legyen *1337*, a szimbólum pedig legyen *ETH*! A létrehozást követően a MetaMask automatikusan erre a hálózatra vált. Ez a hálózat jelképezi a lokális Ethereum node hálózatát, vagyis a tesztelésre használt Ethereum blokkláncot.
8. Kattintson a jobb felső sarokban lévő ábrára és válassza a *Fiók importálása* opciót! Végezze el az importálást mind az öt, második Node.js konzolban található cím esetén az ott található privát kulcsok segítségével! Amennyiben nem látja a címeket, görgessen fel a konzolban a 7. pontban kiadott parancsig!

9. Az első Node.js konzolban látható, hogy a deployer az öt közül legelső cím volt, ezért ő lesz az okos szerződés tulajdonosa. Őt a MetaMask-ban nevezze el Owner-nek, a többi tetszőleges. Én Auctioneer 1, 2 és Investor 1, 2-nek neveztem el őket. A cím nevét az importálás után a három függőleges pontra kattintva, majd a *Fiókadatok* opciót választva lehet módosítani!
10. Az első Node.js konzolban futtasson `ng serve` parancsot, majd nyissa meg a `localhost:4200`-at MetaMask bővítménnyel rendelkező böngészőben! Az oldalhoz történő csatlakozást követően a MetaMask jelezni fog (előhoz egy oldalt, vagy megjelenik az ikonján egy kék körben egy fehér 1-es szám), hogy csatlakozni lehet az oldalhoz. Jelölje ki az összes importált címet (tehát az eredeti Account 1-et ne) és csatlakozzon az oldalhoz!
11. Nyomjon egy F5-öt, és innentől használható az okos szerződés!

Megjegyzés: Ha az okos szerződés a mainnet-en futna, a MetaMask úgy működne, hogy egy felhasználó csak a saját címeihez fér hozzá a saját számítógépét használva. Most azonban lokális node-on fut, ezért egyetlen, tesztelésre szánt MetaMask fiókhoz kell hozzáadni az összes címet és elképzelni, hogy ők független szereplők saját MetaMask fiókokkal. Így csak a fiókon belüli címeket kell váltogatni, amikor árverőből befektetővé válunk, ami gyorsabb, mint ki-be jelentkezni a fiókokba.

Sajnos a program felülete nem képes érzékelni, ha egy MetaMask fiókon belül címet váltunk, ezért érdemes minden váltásnál frissíteni az oldalt. Így ha a Tulajdonos felületen váltottunk Owner-ről Investor 1-re, akkor frissítés után az Investor 1-gyel már nem fogjuk látni a tulajdonos felületet. (Ez a probléma csak azért jelentkezik, mert egyetlen MetaMask fiókot használunk!)

Opcionálisan regisztrálhat minden szereplőnek külön MetaMask fiókot, de ez nehezkesebbé teszi a tesztelést.