



## Упутство за Ethereum

**Студенти:**

Кристина Бојовић  
Андрија Трошић  
Никола Миловановић  
Дарко Бјеличић  
Лазар Беслаћ

**Ментори:**

Др Бобан Стојановић  
Лазар Крстић  
Андрија Живић  
Филип Бојовић

## Садржај

|                                  |   |
|----------------------------------|---|
| Упутство за Ethereum .....       | 1 |
| Шта је Ethereum? .....           | 3 |
| Како Ethereum функционише? ..... | 3 |
| Предности Ethereum-а? .....      | 3 |
| Недостаци Ethereum-а? .....      | 3 |
| Инсталација .....                | 4 |
| 1. Node.js .....                 | 4 |
| 2. Ganache .....                 | 4 |
| 3. MetaMask .....                | 5 |
| 4. Solidity.....                 | 6 |
| Прављење пројекта.....           | 7 |

## Шта је Ethereum?

Ethereum је јавно дистрибуирана софтверска платформа отвореног кода заснована на blockchain технологији, која дозвољава програмерима да граде и развијају децентрализоване апликације.

## Како Ethereum функционише?

Ethereum се заснива на Bitcoin протоколу и његовом blockchain дизајну, али је подешен тако да апликације изван новчаних система могу бити подржане. Ethereum памти читаву историју трансакција. Поред трансакција, сваки node на Ethereum мрежи такође треба да преузме најновије стање или тренутне информације о сваком паметном уговору унутар мреже, биланс сваког корисника и све кодове паметних уговора и податак о томе где су ускладиштени.

Ethereum blockchain се може описати као state machine заснован на трансакцијама. State machine се дефинише као уређај способан да чита серију улаза (input) и прелазак у ново стање на основу тих улаза. Ове трансакције су груписане да формирају “блокове”, при чему сваки блок буде повезан са свим претходним блоковима. Али пре него што се трансакција може додати у регистар, она мора бити потврђена од свих чворова који се налазе у мрежи, то се обавља кроз процес који се зове рударење.

## Предности Ethereum-а?

- Ethereum платформа користи све карактеристике blockchain технологије на којој ради.
- Потпуно је имуна на све интервенције треће стране, што значи да све децентрализоване апликације, које се налазе у мрежи, не може нико да контролише.
- Свака blockchain мрежа се формира на принципу консензуса, што значи да се сви чвориви унутар система морају слижити око сваке промене која треба да се направи унутар њега. Ово елиминише могућност преваре, корупције и чини мрежу отпорну на кварове.
- Целокупна платформа је децентрализована, што значи да не постоји једна тачка неуспеха.
- Све апликације ће увек остати на мрежи и никада се неће искључити.
- Децентрализована природа и криптографска сигурност чине Ethereum мрежу добро заштићену од многих хакерских напада.

## Недостаци Ethereum-а?

- Паметни уговори су намењени да заштите мрежу од кварова, али они су добри само онолико колико су добри људи који пишу код за њих. Увек постији простор за људску грешку.

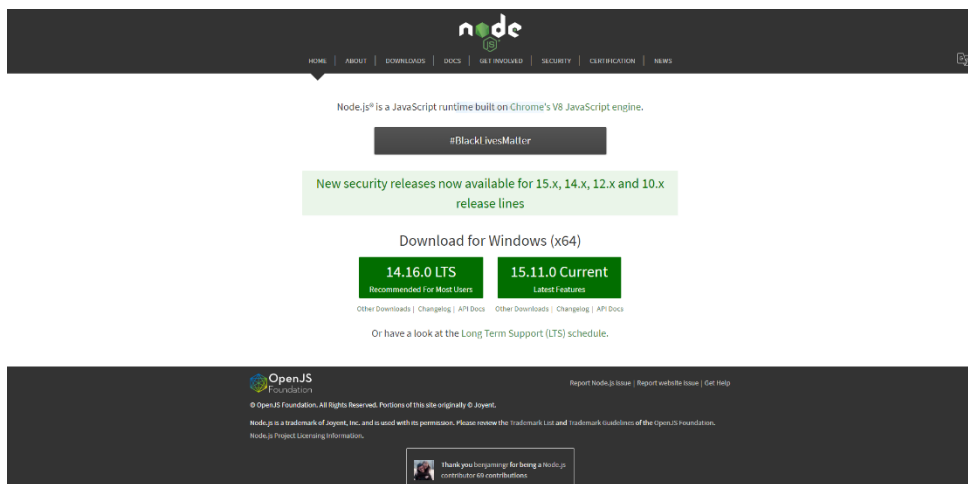
## Инсталација

За коришћење Ethereum мреже неопходно је инсталирати:

### 1. Node.js

Линк са кога можете преузети node.js: <https://nodejs.org/en/>

Кликом на 14.16.0 LTS преузеће се последња стабилна верзија node.js-a.

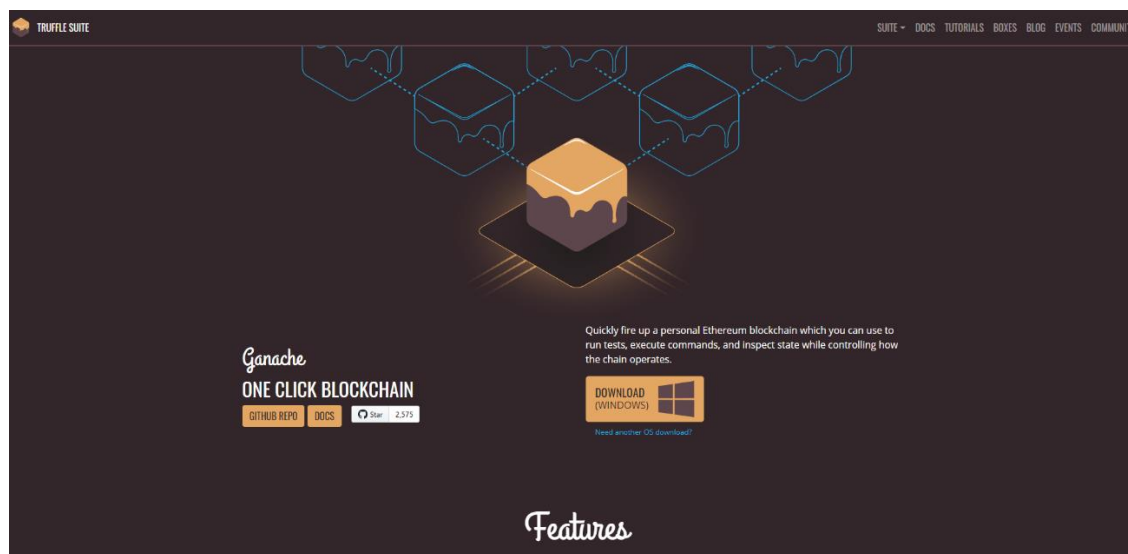


Након преузимања проћи кроз *Node.js Setup Wizard*. На овај начин ће се инсталирати *Node.js*.

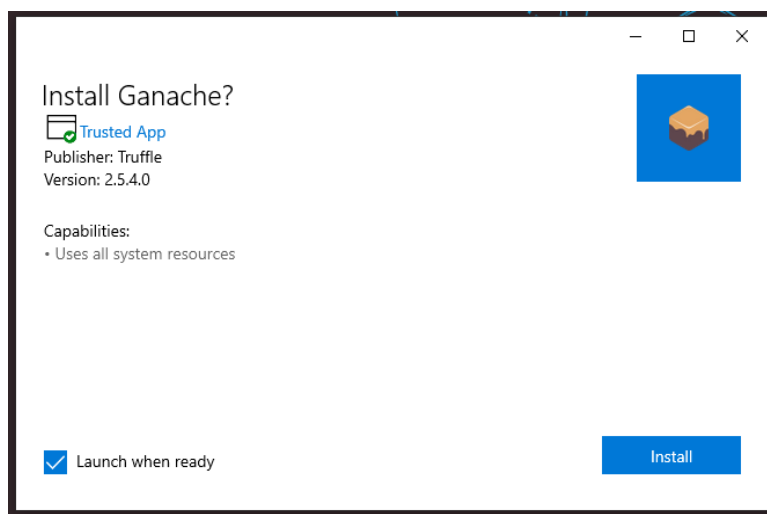
### 2. Ganache

Линк са кога можете преузети ganache: <https://www.trufflesuite.com/ganache>

Кликом на дугме DOWNLOAD преузеће се ganache.



Након отварања преузетог фајла отвориће се прозор:

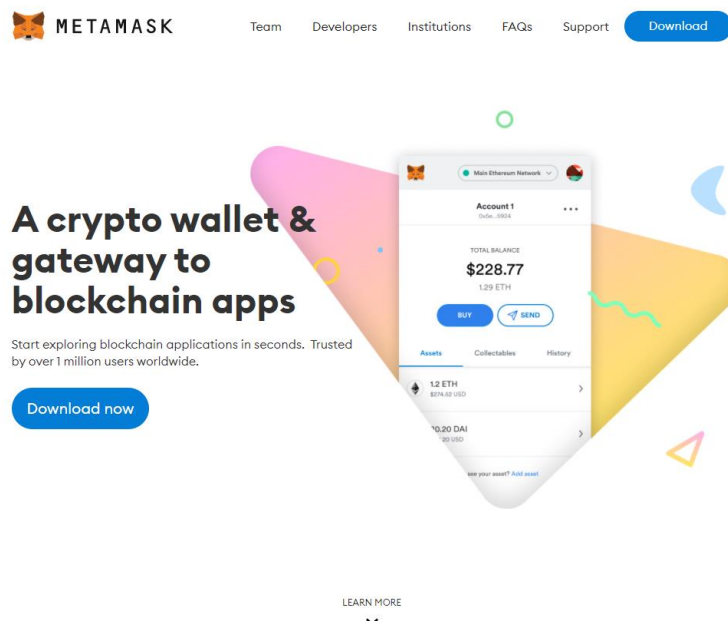


Потребно је само кликнути Install и Ganache ће бити инсталиран.

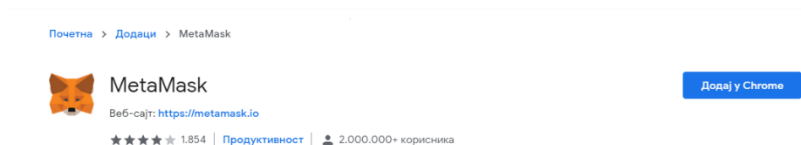
Након ове инсталације треба ући у PowerShell као администратор и искуцати следећу команду: ***npm install –global truffle*** и тиме ће бити инсталирани сви алати потребни за паметне уговоре.

### 3. MetaMask

Линк за инсталацију је: <https://metamask.io/>



Кликом на дугме *Download now* отвара се следећа страница за преузимање где треба кликнути *Install MetaMask for Chrome*. Када се инсталира отвориће се страница на којој треба додати MetaMask у Chrome (Додај у Chrome).



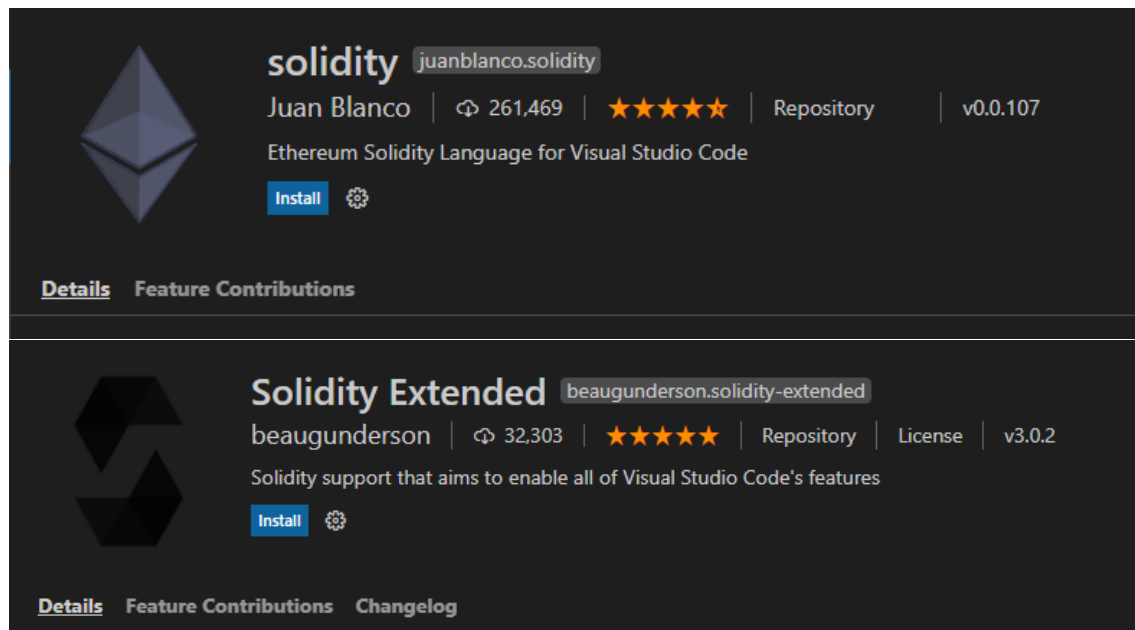
Када додамо у Chrome, завршена је инсталација. И MetaMask ће се налазити на Chrome-у

#### 4. Solidity

*Solidity* је објектно-оријентисан језик на високом нивоу за примену паметних уговора. Паметни уговори су програми који управљају понашањем рачуна на Ethereum мрежи.

Програмски језик *Solidity* ћемо инсталирати из *Visual Studio Code*-а. Линк за инсталацију *Visual Studio Code*-а: <https://code.visualstudio.com/>. Кликом на Download преузеће се *Visual Studio Code*, потребно је само проћи кроз *VSC Setup Wizard*.

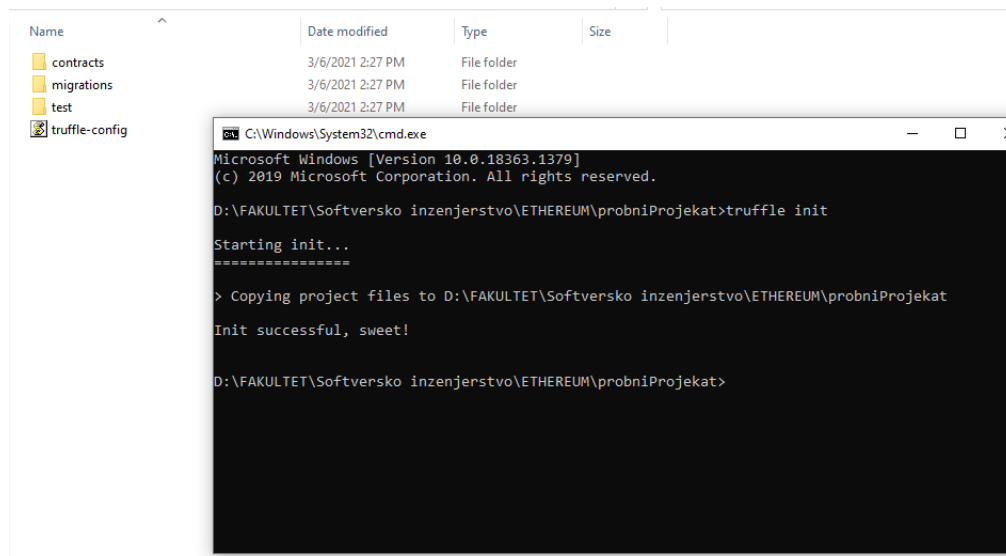
У Visual Studio Code – у кликнути на дугме Extensions и инсталирати следеће две ствари:



Овие смо завршили инсталацију

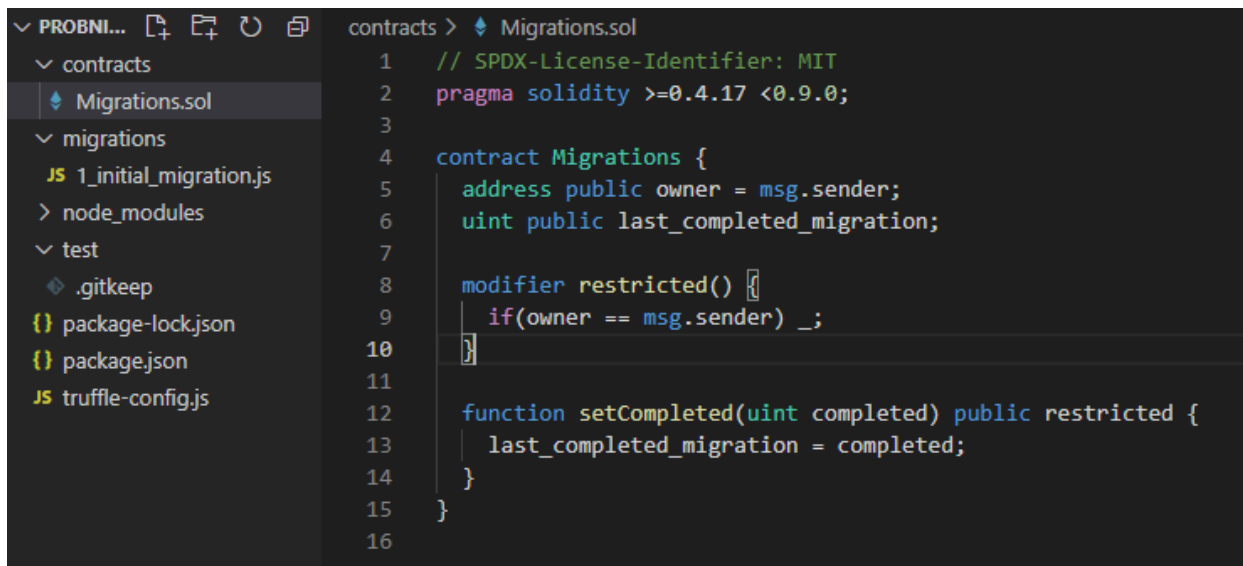
## Прављење пројекта

Отворимо фолдер где хоћемо да сместимо наш пројекат. Онда у том фолдеру отворимо cmd и њему куцамо следећу командз за иницијализацију пројекта(truffle init).



Онда овај фолдер отворимо у Visual Studio Code-у. У Visual Studio Code-у отворимо терминал и куцамо `npm init`, да би направили `package.json` филе. Када формирамо `package.json` можемо почети са радом.

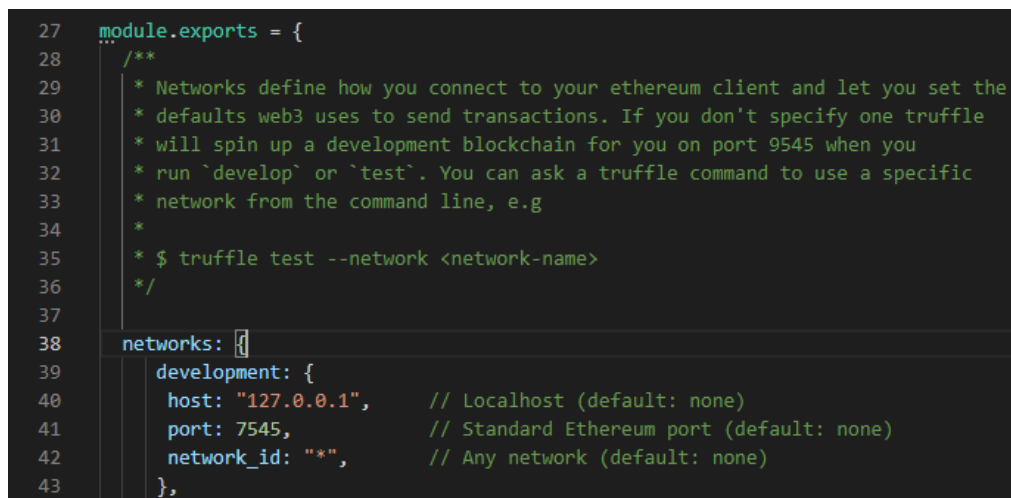
Овако изгледа иницијализован пројекат



The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar displays a project structure with folders `contracts`, `migrations`, `node_modules`, and `test`. Files include `1_initial_migration.js`, `package-lock.json`, `package.json`, and `truffle-config.js`. The main editor area shows the `Migrations.sol` file, which contains a Solidity contract. The contract is named `Migrations` and includes a `restricted` modifier and a `setCompleted` function.

```
contracts > Migrations.sol
1  // SPDX-License-Identifier: MIT
2  pragma solidity >=0.4.17 <0.9.0;
3
4  contract Migrations {
5      address public owner = msg.sender;
6      uint public last_completed_migration;
7
8      modifier restricted() {
9          if(owner == msg.sender) _;
10     }
11
12     function setCompleted(uint completed) public restricted {
13         last_completed_migration = completed;
14     }
15 }
16
```

Да би покренули пројекат неопходно је у `truffle-config.js` откоментарисати следеће ствари, и поставити порт на 7545. И тај фајл треба да изгледа овако:



The screenshot shows a portion of the `truffle-config.js` file. It displays the `module.exports` object and a commented-out section for network configuration. The `networks` object is defined with a `development` network configuration, including `host`, `port`, and `network_id` properties.

```
27 module.exports = {
28   /**
29    * Networks define how you connect to your ethereum client and let you set the
30    * defaults web3 uses to send transactions. If you don't specify one truffle
31    * will spin up a development blockchain for you on port 9545 when you
32    * run `develop` or `test`. You can ask a truffle command to use a specific
33    * network from the command line, e.g
34    *
35    * $ truffle test --network <network-name>
36    */
37   networks: {
38     development: {
39       host: "127.0.0.1",    // Localhost (default: none)
40       port: 7545,          // Standard Ethereum port (default: none)
41       network_id: "*",    // Any network (default: none)
42     },
43   },
44 }
```