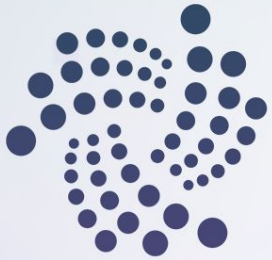
Kaip pasidaryti IOTA Tangle įrenginį

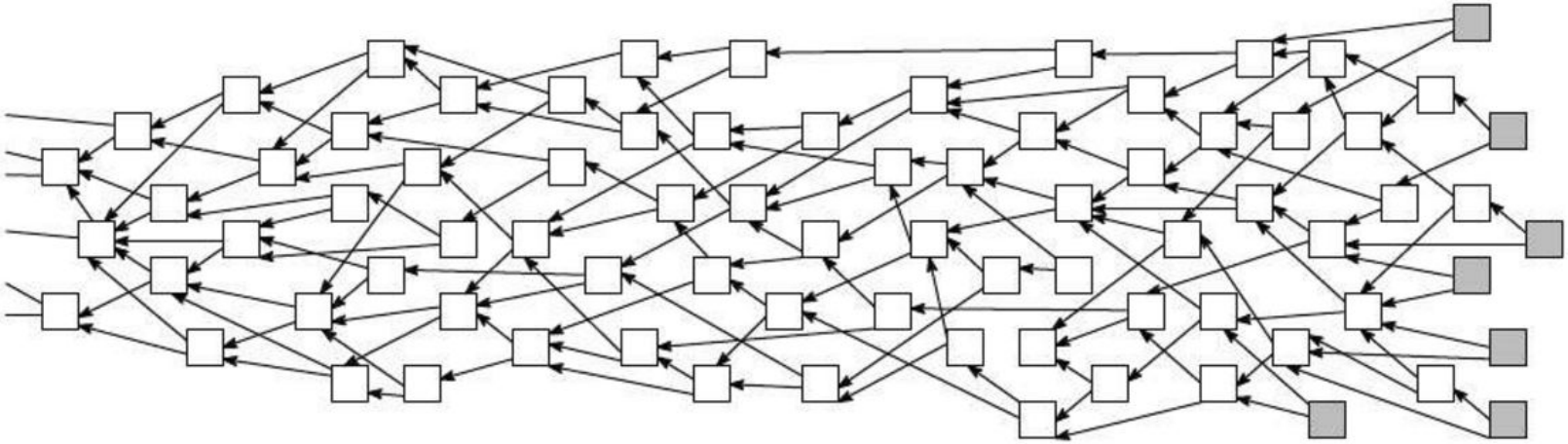
Dviejų temperatūrų stotelė blockchaine

Nerijus M., Maker.lt, 2018



IOTA

The Backbone of IoT



Nėra blokų

Nėra grandinių

Nėra kasėjų

Kuo daugiau nodų - tuo greitesnis tinklas

Nemokamos tranzakcijos

Višiskai decentralizuota

Trinarė sistema

Viskas IOTA yra per trinarę sistemą

Tritai ir Trytai - analogija bitams ir baitams

Simboliai - 9ABCDEFGHIJKLMNOPQRSTUVWXYZ

Funkcijos - *toTrytes* ir *fromTrytes*

Visi adresai, tagai, hashai, bundlai sudaromi iš trytų

Seedai ir adresai

Seedas yra tavo piniginė. Gali būt bet kas sudarytas iš trytų iki 81 simbolio ilgio

Geriau susigeneruoti lokaliai, nenaudoti webinių generatorių (dingo keli milijonai €)

Adresų gali turėti kiek nori ir kokių nori, iki 81 simbolio ilgio

Į adresą gauti gali kiek nori kartų

Iš adreso siųsti geriau vieną ar du kartus - dėl saugumo.

Field	Description	Type	Length (if string)
address	In case this is an *output*, then this is the address of the recipient. In case it is an *input*, then it is the address of the input which is used to send the tokens from (i.e. address generated from the private key)	`String`	27-trytes
attachmentTimestamp	Timestamp after POW	`Int`	
attachmentTimestampLowerBound	Lower Bound of timestamp	`Int`	
attachmentTimestampUpperBound	Upper bound of timestamp	`Int`	
branchTransaction	Transaction being approved	`String`	81-trytes
bundle	bundle hash, which is used for grouping transactions of the bundle together. With the bundle hash you can identify transactions which were in the same bundle.	`String`	81-trytes
currentIndex	Index in Bundle	`Int`	
hash	Transaction hash	`String`	81-trytes
lastIndex	Last index in bundle	`Int`	
nonce	The nonce is required for the transaction to be accepted by the network. It is generated by doing Proof of Work (either in IRI via the attachToTangle API call, or with one of the libraries such as ccurl).	`String`	27-trytes
obsoleteTag	User-defined tag (removed soon)	`String`	27-trytes
signatureMessageFragment	Signature message fragment. In case there is a spent input, the signature of the private key is stored here. If no signature is required, it is empty (all 9's) and can be used for storing the message value when making a transfer. More to that later.	`String`	2187-trytes
tag	User-defined tag	`String`	27-trytes
timestamp	Timestamp (not-enforced, can be arbitrary)	`Int`	
trunkTransaction	Transaction being approved	`String`	81-trytes
value	Transaction value	`Int`	

Tranzakcija

Adresas (*address*) - adresas, kur bus saugomos visos tranzakcijos

Laikas (*attachmentTimestamp)* - laikas, kada buvo atliktas PoW

Tagas (*tag*) - tagas, pvz: sensoriaus ID

Žinutė (*signatureMessageFragment)* - laukelis, kuriame siunčiama visa informacija

Vertė (*value*) - tranzakcijos vertė IOTA (1 IOTA yra 1/1000000 IOTA)

IRI

IRI - IOTA node serveris

Raspberio kol kas neužtenka

Minimalūs reikalavimai:

- 4Gb RAM, 60Gb SSD, JAVA core

Galima naudoti kitų serverius

- http://iota.fm

Hardwaras

Raspberry Pi

DS18B20 termometrai

SD kortelė

Maitinimas

Wifi / Ethernet / 3G

Softwaras

Node.js

IOTA API biblioteka

DS18D20 biblioteka

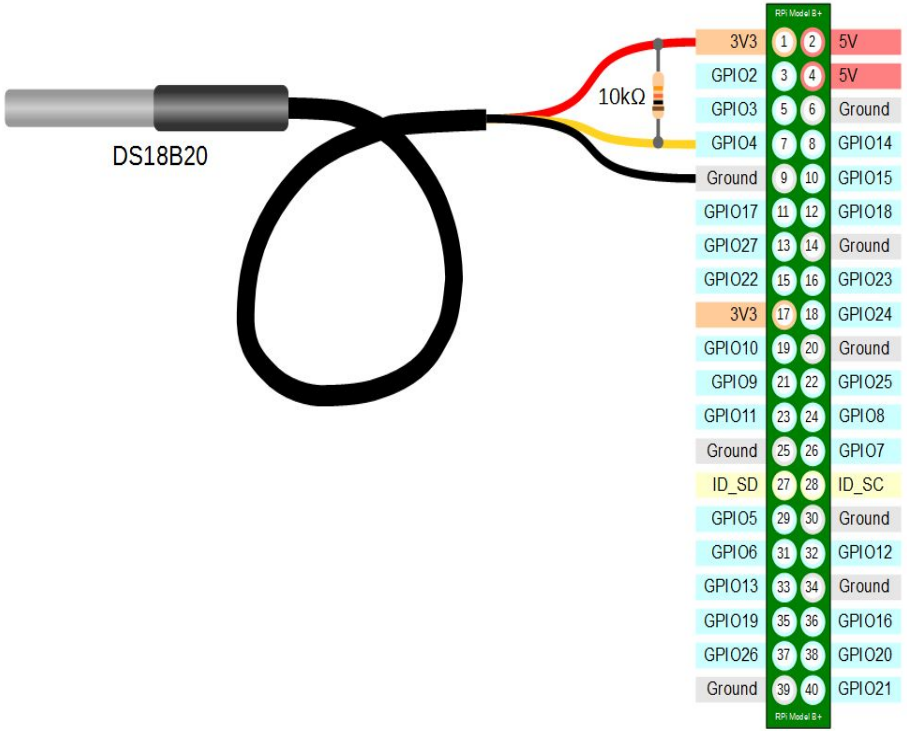
Forever biblioteka

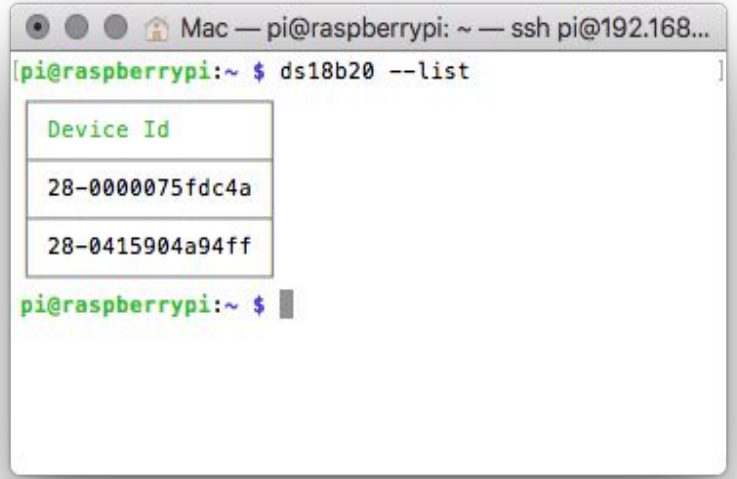
Raspberris

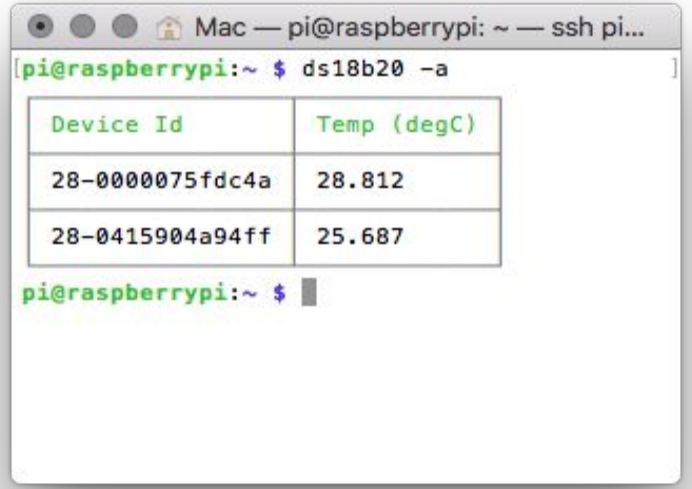
1Wire pajungimas

Reikia tik papildomos 10K varžos

Galima jungti bet kiek sensorių



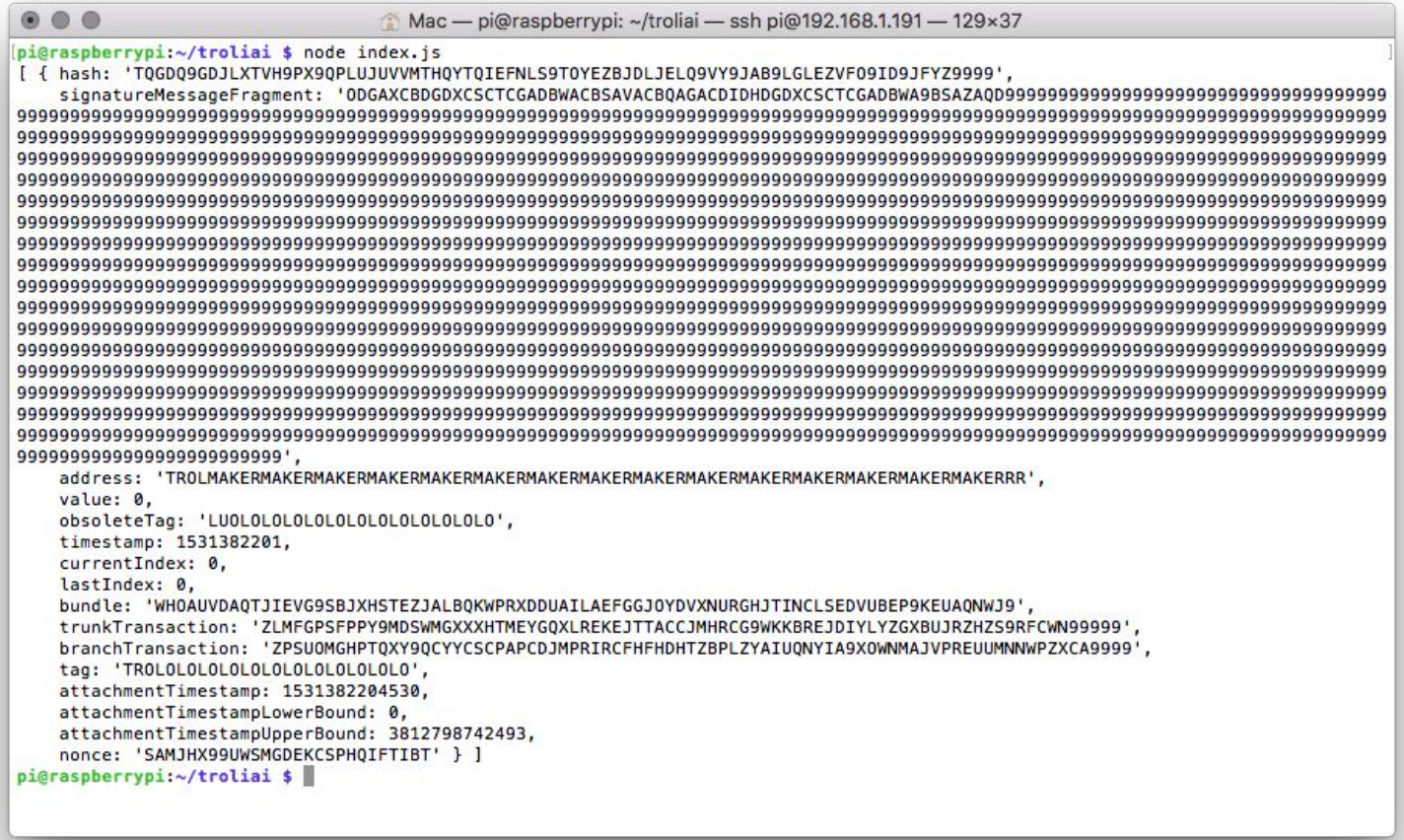

Mac — pi@raspberrypi: ~ — ssh pi@192.168...
pi@raspberrypi:~ $ ds18b20 --list
Device Id
28-0000075fdc4a
28-0415904a94ff
pi@raspberrypi:~ $


Mac — pi@raspberrypi: ~ — ssh pi...
pi@raspberrypi:~ $ ds18b20 -a
Device Id
Temp (degC)
28-0000075fdc4a
28.812
28-0415904a94ff
25.687
pi@raspberrypi:~ $

```
const iota = new IOTA({
    'provider': "http://nodes.iota.fm:80"
});

const iotaAddress =
"TROLMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKER
MAKERMAKERMAKERMAKERRR";
const iotaTag = "TROLOLOLOLOLOLOLOLOLOLOLLLL";
const insideSensor = "28-0000075fdc4a";
const outsideSensor = "28-0415904a94ff";
```

```
function logTempIOTA() {
    const sensorsData = {
        inside: sensor.readC(insideSensor, 2),
        outside: sensor.readC(outsideSensor, 2)
    }
    const message = iota.utils.toTrytes(JSON.stringify(sensorsData));
    const transfer = [{
        value: 0,
        tag: iotaTag,
        address: iotaAddress,
        message: message
    }];
    iota.api.sendTransfer(iotaAddress, 9, 14, transfer, (error, result) => {
        if (error) {
            console.log(error)
        } else {
            console.log(result)
        }
    });
}
```

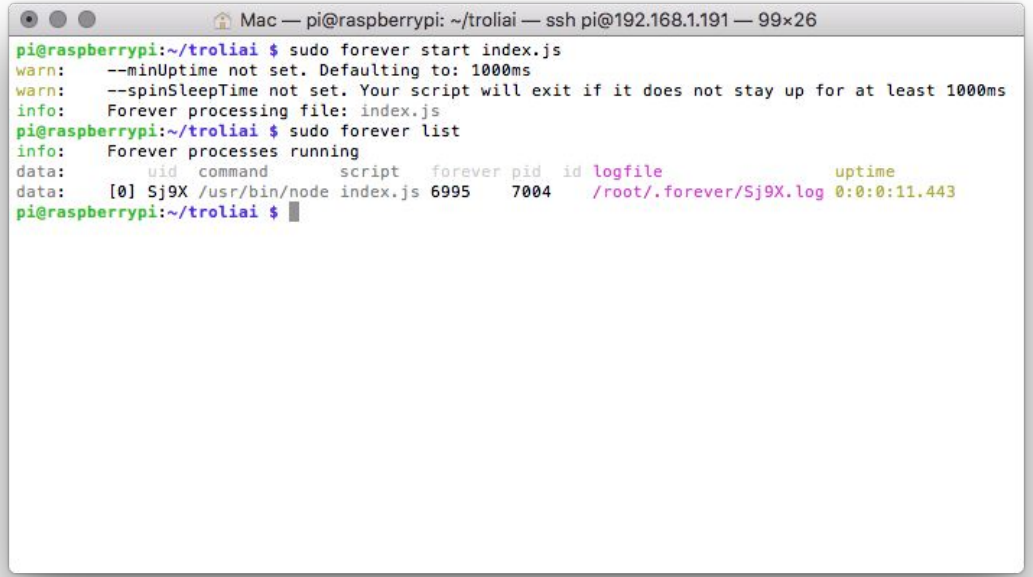

Mac — pi@raspberrypi: ~/troliai — ssh pi@192.168.1.191 — 129×37
pi@raspberrypi:~/troliai $ node index.js
[{ hash: 'TQGDQ9GDJLXTVH9PX9QPLUJUVVMTHQYTQIEFNLS9TOYEZBJDLJELQ9VY9JAB9LGLEZVFO9ID9JFYZ9999',
signatureMessageFragment: 'ODGAXCBDGDXCSCTCGADBWACBSAVACBQAGACDIDHDGDXCSCTCGADBWA9BSAZAQD9999999999999999999999999999999999999999
address: 'TROLMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERRR',
value: 0,
obsoleteTag: 'LUOLOLOLOLOLOLOLOLOLOLOLO',
timestamp: 1531382201,
currentIndex: 0,
lastIndex: 0,
bundle: 'WHOAUVDAQTJIEVG9SBJXHSTEZJALBQKWPRXDDUAILAEFGGJOYDVXNURGHJTINCLSEDVUBEP9KEUAQNWJ9',
trunkTransaction: 'ZLMFGPSFPPY9MDSWMGXXXHTMEYGQXLREKEJTTACCJMHRCG9WKKBREJDIYLYZGXBUJRZHZS9RFCWN99999',
branchTransaction: 'ZPSUOMGHPTQXY9QCYYCSCPAPCDJMPRIRCFHFHDHTZBPLZYAIUQNYIA9XOWNMAJVPREUUMNNWPZXCA9999',
tag: 'TROLOLOLOLOLOLOLOLOLOLOLO',
attachmentTimestamp: 1531382204530,
attachmentTimestampLowerBound: 0,
attachmentTimestampUpperBound: 3812798742493,
nonce: 'SAMJHX99UWSMGDEKCSPHQIFTIBT' }]
pi@raspberrypi:~/troliai $

Raspberris

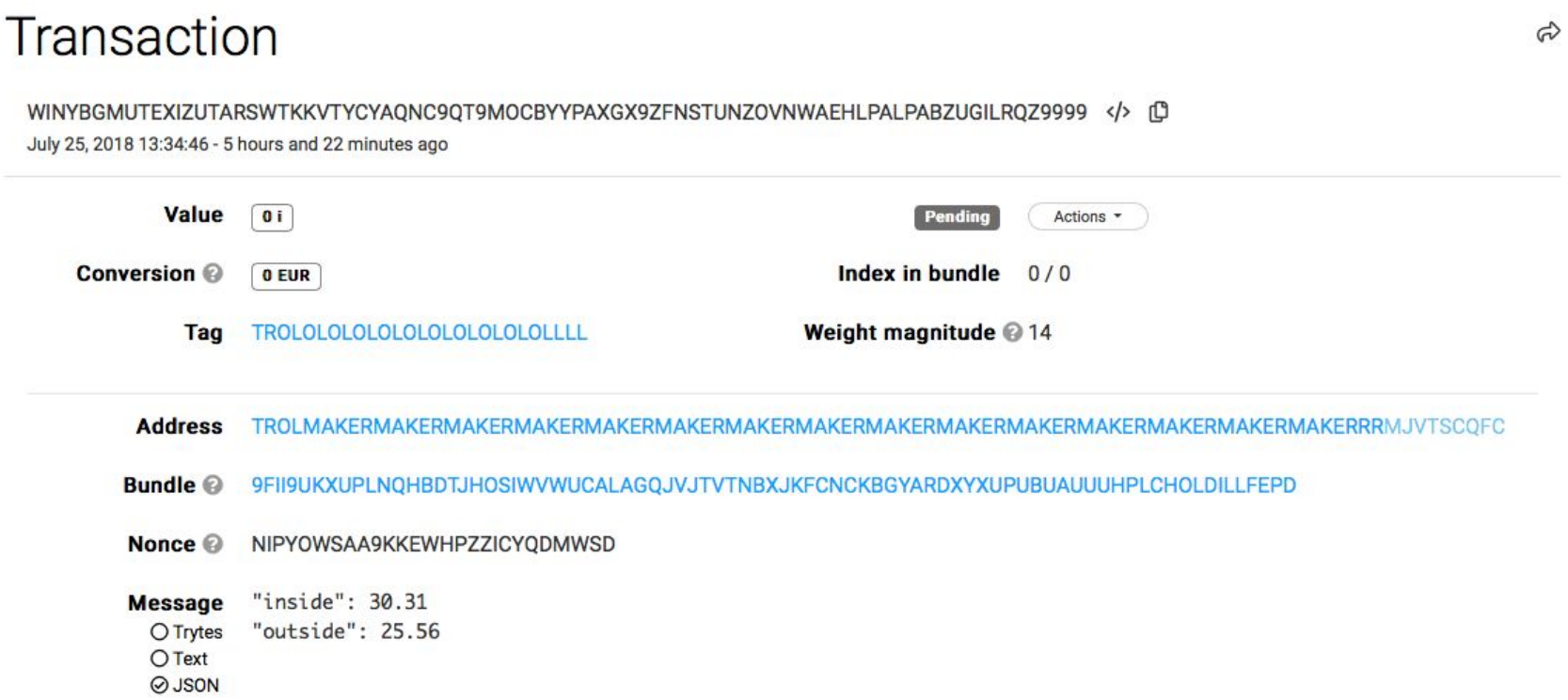
Terminale suvedam: crontab -e

Apačioj įrašom tokią eilutę:
@reboot sudo forever start
/home/pi/troliai/index.js

Raspberris amžinai suks *index.js* failą ir taip darys IOTA tranzakcijas su temperatūra



Tangle Explorer



Frontendas

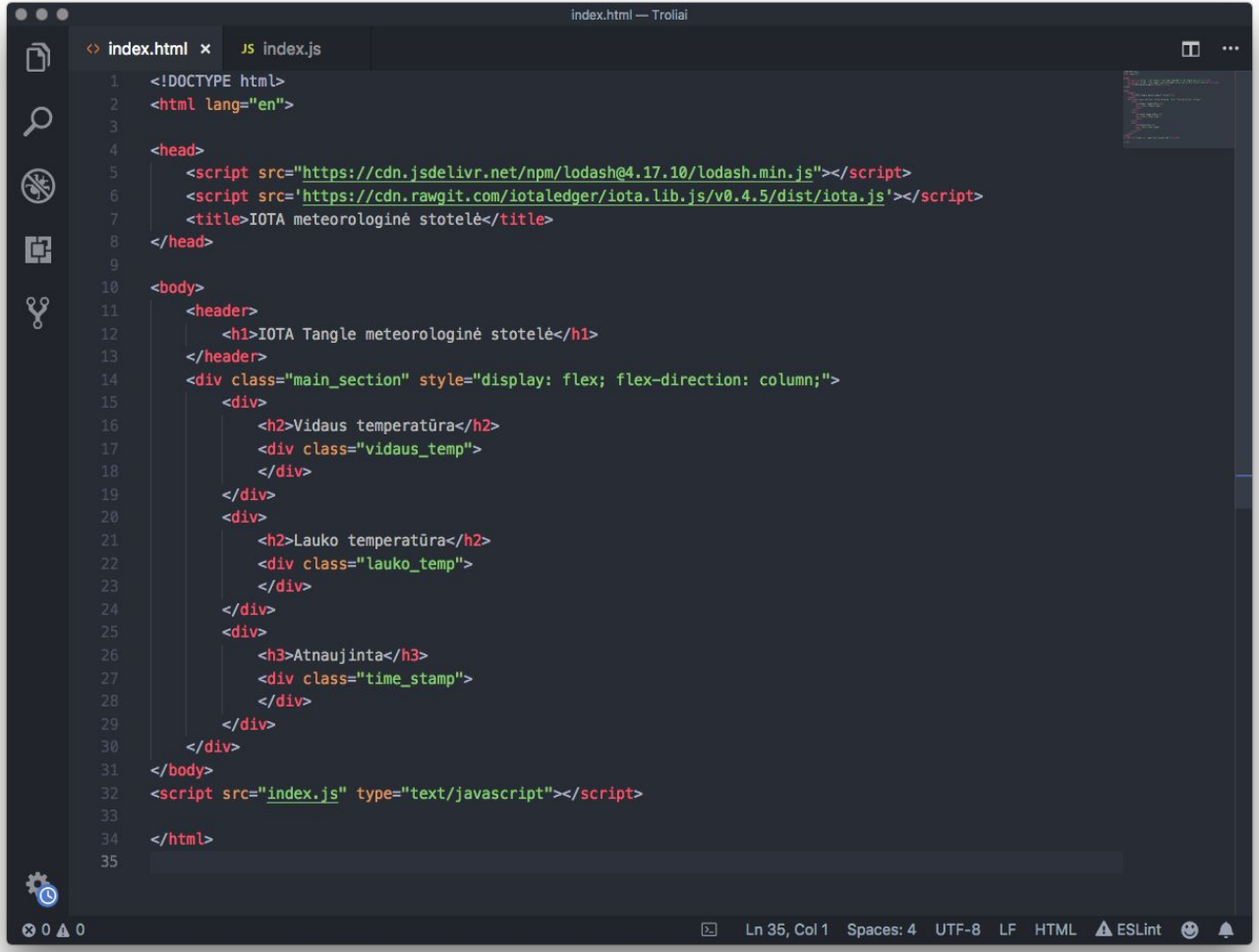
Naudojamos bibliotekos

- Lodash
- IOTA

IOTA setupas toks pat kaip *Node.js* - adresas ir tagas. Serveris gali būti ir kitas.

HTML

```
<div class="main_section" style="display: flex; flex-direction: column;">
        <div>
            <h2>Vidaus temperatūra</h2>
            <div class="vidaus_temp">
            </div>
        </div>
        <div>
            <h2>Lauko temperatūra</h2>
            <div class="lauko_temp">
            </div>
        </div>
        <div>
            <h3>Atnaujinta</h3>
            <div class="time_stamp">
            </div>
        </div>
    </div>
```


index.html — Troliai
index.html
index.js
<!DOCTYPE html>
<html lang="en">
<head>
<script src="https://cdn.jsdelivr.net/npm/lodash@4.17.10/lodash.min.js"></script>
<script src='https://cdn.rawgit.com/iotaledger/iota.lib.js/v0.4.5/dist/iota.js'></script>
<title>IOTA meteorologinė stotelė</title>
</head>
<body>
<header>
<h1>IOTA Tangle meteorologinė stotelė</h1>
</header>

<h2>Vidaus temperatūra</h2>

<h2>Lauko temperatūra</h2>

<h3>Atnaujinta</h3>

</body>
<script src="index.js" type="text/javascript"></script>
</html>
0 0
Ln 35, Col 1
Spaces: 4
UTF-8
LF
HTML
ESLint

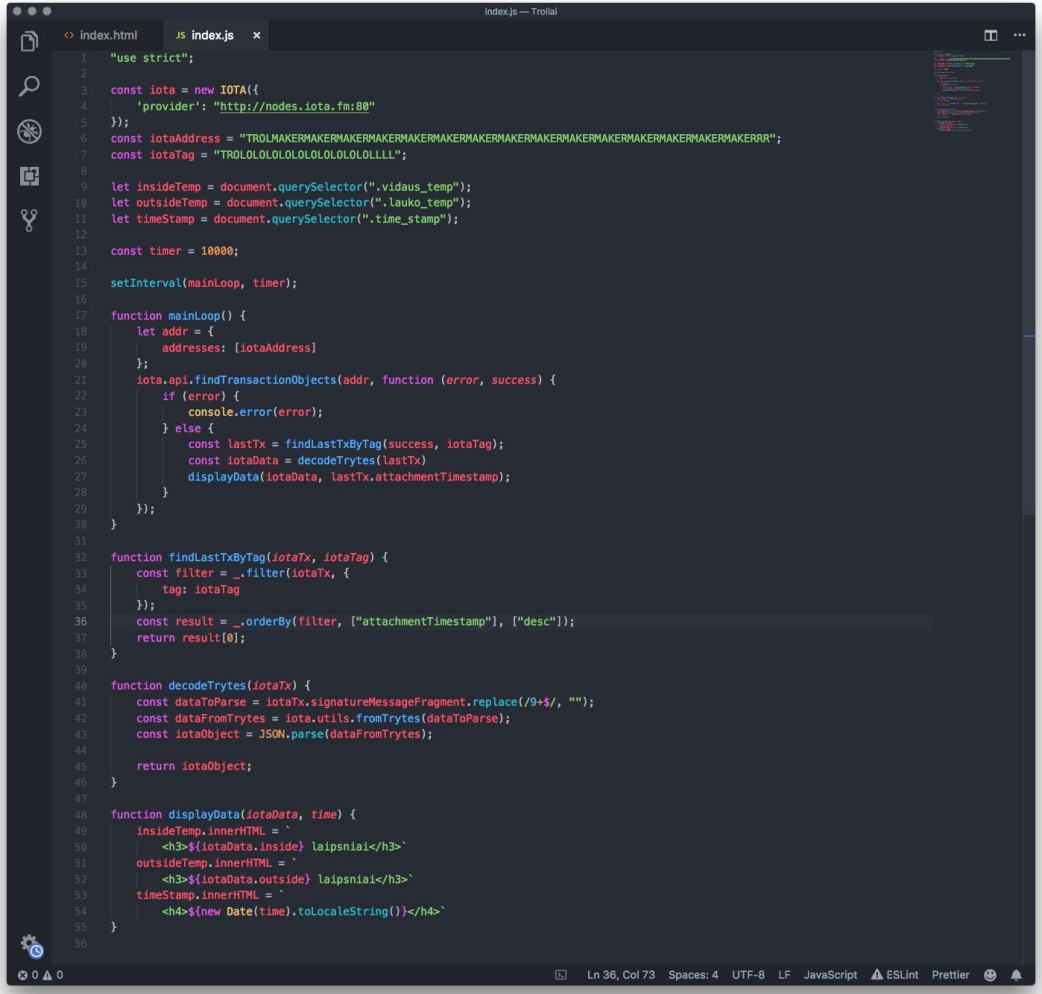

IOTA meteorologinė stotelė
Nerijus
file:///Users/Mac/Dropbox/Troliai/frontend/index.html
IOTA Tangle meteorologinė stotelė
Vidaus temperatūra
30.31 laipsniai
Lauko temperatūra
25.56 laipsniai
Atnaujinta
7/25/2018, 1:34:46 PM

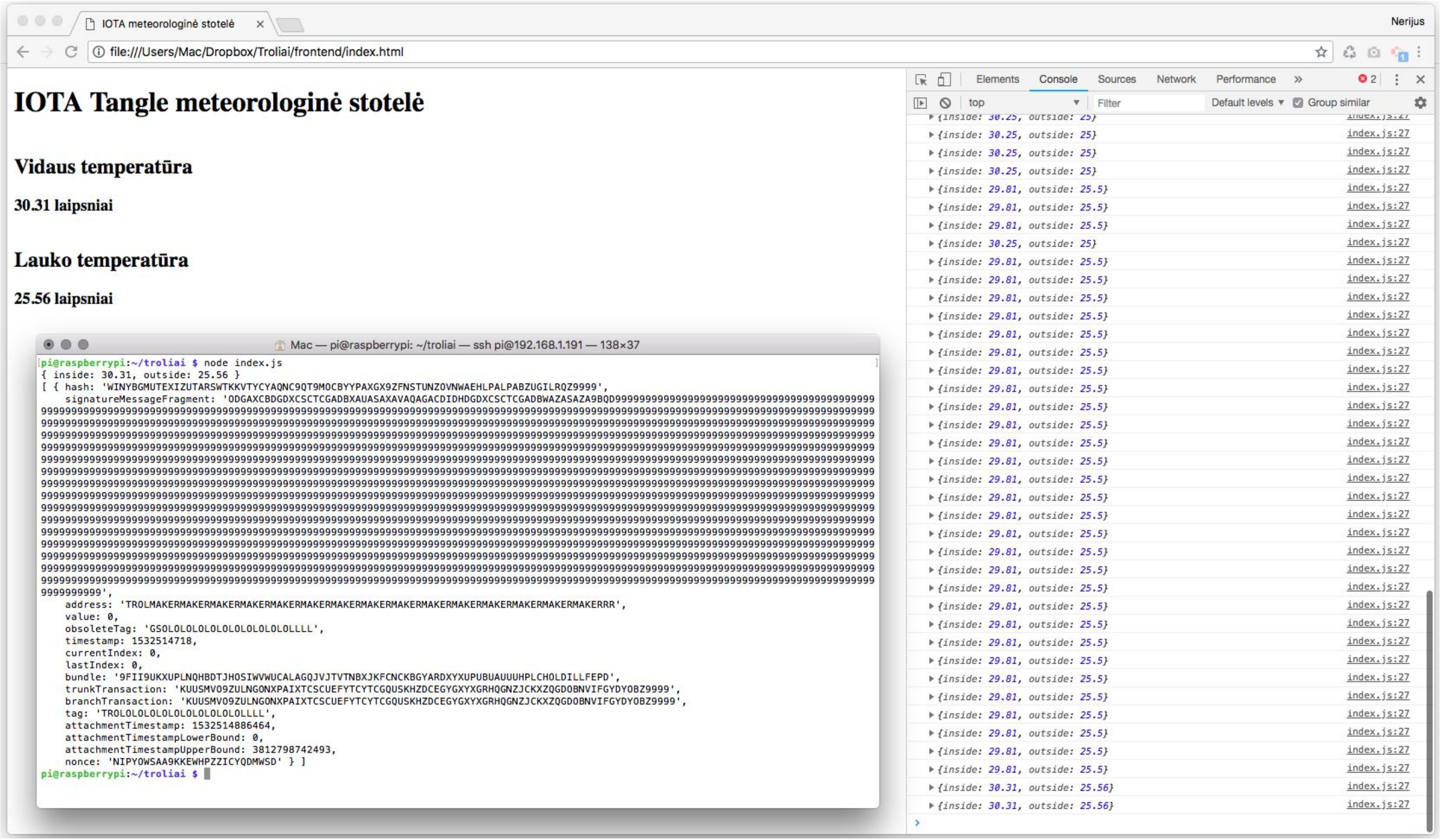
JavaScript

```
function mainLoop() {
    let addr = {
        addresses: [iotaAddress]
    };
    iota.api.findTransactionObjects(addr, function (error, success) {
        if (error) {
            console.error(error);
        } else {
            const lastTx = findLastTxByTag(success, iotaTag);
            const iotaData = decodeTrytes(lastTx)
            displayData(iotaData, lastTx.attachmentTimestamp);
        }
    });
}
```

```
function findLastTxByTag(iotaTx, iotaTag) {
    const filter = _.filter(iotaTx, {
        tag: iotaTag
    });
    const result = _.orderBy(filter, ["attachmentTimestamp"], ["desc"]);
    return result[0];
}
function decodeTrytes(iotaTx) {
    const dataToParse = iotaTx.signatureMessageFragment.replace(/9+$/, "");
    const dataFromTrytes = iota.utils.fromTrytes(dataToParse);
    const iotaObject = JSON.parse(dataFromTrytes);

    return iotaObject;
}
function displayData(iotaData, time) {
    insideTemp.innerHTML = `
        <h3>${iotaData.inside} laipsniai</h3>`
    outsideTemp.innerHTML = `
        <h3>${iotaData.outside} laipsniai</h3>`
    timeStamp.innerHTML = `
        <h4>${new Date(time).toLocaleString()}</h4>`
}
```


index.js — Trolial
index.html
index.js
"use strict";
const iota = new IOTA({
'provider': "http://nodes.iota.fm:80"
});
const iotaAddress = "TROLMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERRR";
const iotaTag = "TROLOLOLOLOLOLOLOLOLOLLLL";
let insideTemp = document.querySelector(".vidaus_temp");
let outsideTemp = document.querySelector(".lauko_temp");
let timeStamp = document.querySelector(".time_stamp");
const timer = 10000;
setInterval(mainLoop, timer);
function mainLoop() {
let addr = {
addresses: [iotaAddress]
};
iota.api.findTransactionObjects(addr, function (error, success) {
if (error) {
console.error(error);
} else {
const lastTx = findLastTxByTag(success, iotaTag);
const iotaData = decodeTrytes(lastTx)
displayData(iotaData, lastTx.attachmentTimestamp);
}
});
}
function findLastTxByTag(iotaTx, iotaTag) {
const filter = _.filter(iotaTx, {
tag: iotaTag
});
const result = _.orderBy(filter, ["attachmentTimestamp"], ["desc"]);
return result[0];
}
function decodeTrytes(iotaTx) {
const dataToParse = iotaTx.signatureMessageFragment.replace(/9+$/, "");
const dataFromTrytes = iota.utils.fromTrytes(dataToParse);
const iotaObject = JSON.parse(dataFromTrytes);
return iotaObject;
}
function displayData(iotaData, time) {
insideTemp.innerHTML = `
<h3>${iotaData.inside} laipsniai</h3>`
outsideTemp.innerHTML = `
<h3>${iotaData.outside} laipsniai</h3>`
timeStamp.innerHTML = `
<h4>${new Date(time).toLocaleString()}</h4>`
}
Ln 36, Col 73
Spaces: 4
UTF-8
LF
JavaScript
ESLint
Prettier


IOTA meteorologinė stotelė
Nerijus
file:///Users/Mac/Dropbox/Troliai/frontend/index.html
IOTA Tangle meteorologinė stotelė
Vidaus temperatūra
30.31 laipsniai
Lauko temperatūra
25.56 laipsniai
Mac — pi@raspberrypi: ~/troliai — ssh pi@192.168.1.191 — 138×37
pi@raspberrypi:~/troliai $ node index.js
{ inside: 30.31, outside: 25.56 }
[{ hash: 'WINYBGMUTEXIZUTARSWTKKVTYCYAQNC9QT9MOCBYYPAXGX9ZFNSTUNZOVNWAEHLPALPABZUGILRQZ9999',
signatureMessageFragment: 'ODGAXCBDGDXCSCTCGADBXAUASAXAVAQAGACDIDHDGDXCSCTCGADBWAZASAZA9BQD99999999999999999999999999999999999999999999
9999999999',
address: 'TROLMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERMAKERRR',
value: 0,
obsoleteTag: 'GSOLOLOLOLOLOLOLOLOLOLOLLLL',
timestamp: 1532514718,
currentIndex: 0,
lastIndex: 0,
bundle: '9FII9UKXUPLNQHBDTJHOSIWVWUCALAGQJVJTVTNBXJKFCNCKBGYARDXYXUPUBUAUUUHPLCHOLDILLFEPD',
trunkTransaction: 'KUUSMVO9ZULNGONXPAIXTCSCUEFYTCYTCGQUSKHZDCEGYGXYXGRHQGNZJCKXZQGDOBNVIFGYDYOBZ9999',
branchTransaction: 'KUUSMVO9ZULNGONXPAIXTCSCUEFYTCYTCGQUSKHZDCEGYGXYXGRHQGNZJCKXZQGDOBNVIFGYDYOBZ9999',
tag: 'TROLOLOLOLOLOLOLOLOLOLOLLLL',
attachmentTimestamp: 1532514886464,
attachmentTimestampLowerBound: 0,
attachmentTimestampUpperBound: 3812798742493,
nonce: 'NIPYOWSAA9KKEWHPZZICYQDMWSD' }]
pi@raspberrypi:~/troliai $
Elements
Console
Sources
Network
Performance
top
Filter
Default levels
Group similar
{inside: 30.25, outside: 25} index.js:27
{inside: 30.25, outside: 25} index.js:27
{inside: 30.25, outside: 25} index.js:27
{inside: 29.81, outside: 25.5} index.js:27
{inside: 29.81, outside: 25.5} index.js:27
{inside: 29.81, outside: 25.5} index.js:27
{inside: 30.25, outside: 25} index.js:27
{inside: 29.81, outside: 25.5} index.js:27
{inside: 30.31, outside: 25.56} index.js:27
{inside: 30.31, outside: 25.56} index.js:27

Pabaiga

Visas kodas ir lengvas tutorialas - https://github.com/NerijusWeb/weatherIOTA

IOTA tranzakcijas gali daryti beveik su bet kuo - JavaScript, Python…

Paprastas API

Neribotas kiekis tranzakcijų - pagrindas IoT daiktams

nerijus@maker.lt

github.com/NerijusWeb/weatherIOTA