DOI 10.34185/1562-9945-1-150-2024-14
UDC 004.415.2

I.V. Ponomarev

CREATION OF A DECENTRALIZED APPLICATION FOR SALES OF IMAGES USING IPFS

Abstract. Decentralized applications are more secure and reliable than centralized applications because they do not have a single point of failure or control. However, storing big data on the Ethereum blockchain is very expensive and, therefore, it is necessary to use special solutions for this. The development of a decentralized application for selling images using IPFS decentralized storage is considered.

Keywords: Decentralized application, blockchain, Ethereum, server, smart contract, Web 3.0, URL, IPFS, Web3, MetaMask.

Formulation of the problem. An image selling application must be always available to customers, reliable, protected from attacks and falsifications. Centralized applications have a single central server for processing information and storing images, which is the main point of failure of the entire system.

Therefore, for reliable operation, a decentralized application is needed, based on blockchain technology without a single master node, with control distributed among many nodes of the user network. Creating such an application is possible using the existing Ethereum blockchain platform based on smart contracts. Contracts are stored inside the Ethereum Virtual Machine (EVM) and can be accessed simply by accessing the blockchain network node [1, 2].

The decentralized application can be accessed through a web page that accesses the smart contract using a special library.

The web page itself, as well as all its components, such as images, CSS or JavaScript files, can be placed in storage such as the user's local computer or centralized hosting. At the same time, such storage remains a possible point of failure for the entire application, which is not completely decentralized.

And the problem of fault tolerance of the entire application remains unresolved. However, there are decentralized information storage and transmission environments, such as IPFS, Swarm or Filecoin, that make the application completely decentralized [3].

Purpose of the research. It is necessary to consider the possibilities of creating a completely reliable application based on the Ethereum platform. To do this, the image selling application must use the decentralized IPFS information storage and transmission environment.

IPFS provides opportunities for decentralized and free storage of files of any type. Therefore, the application logic will be located inside the Ethereum blockchain, and the data will be in IPFS, which will lead to a greater degree of decentralization and independence of the application.

When developing an application, it is necessary to take into account the features of connecting and working with IPFS.

Main part. There are two main approaches for creating web applications: centralized, often called Web 2.0 architecture, and decentralized, which is based on Web 3.0 architecture.

Web 2.0 architecture contains a client and a server. The client sends an HTTP request to the server and receives a generated HTML page in response. When interacting with page elements, it can re-initiate requests to the server. The server, in turn, receives requests and processes them in accordance with its algorithms. All information of such a web application is located in a database located on the server.

Web 3.0 represents the next generation of the World Wide Web, which is based on blockchain technology. The main feature of Web 3.0 architecture is decentralization and tokenization. Intermediaries are eliminated, such as centralized servers that store program data or logic. Instead, the application logic is represented by one or more smart contracts. Smart contracts are stored within the blockchain and can be accessed simply by accessing a node on the blockchain network.

In a centralized architecture, most information found on the Internet is hosted on centralized servers. To get information, you need to indicate where to look for it. For example, when a user wants to download an image file, he must specify the exact Uniform Resource Locator (URL) of that file. The disadvantage of centralized storage is that the server may be unavailable or have limited access.

Although the Ethereum blockchain supports large storage capacity per block, storing large files requires high costs. These shortcomings have limited the application of many blockchain applications.

Therefore, a new data storage technology was developed, the Interplanetary File System (IPFS), a peer-to-peer distributed file storage system and transfer protocol for file sharing. Instead of specifying where to look for information, the user specifies what he wants to find. For each file, when uploaded to the network, its unique hash is calculated, which is used to access this resource. This mechanism has the following positive properties:

- hash search is fast,
- the file cannot be changed without the knowledge of other users and
- no duplicates of the same file.

IPFS can store any type of data, from a simple test file to full-fledged programs. Typically, the hash code is stored on the blockchain via a smart contract and then provided to the user.

It should also be noted that, like blockchain, files uploaded to the IPFS network cannot be modified. After all, after the change, it will be a completely different file, with a different hash. When you make changes to a file, a new file is downloaded, which will link to its previous version, which also remains on the network. IPFS works in such a way that if, when calculating the hash of a downloaded file, it was discovered that the file had already been downloaded previously, the system simply returns a link to the file that has already been downloaded to the network, without creating duplicates. This approach is certainly justified, because the network is not clogged with hundreds of copies of the same data [3].

Let's look at the features of working with decentralized IPFS storage.

The algorithm for accessing a decentralized application can be described as follows. When a user in a browser wants to access an application, he must contact the IPFS node from where he gets the application's web page. It contains control elements, upon interaction with which the corresponding JavaScript scripts are activated.

For interaction between the browser and the blockchain, the Web3.js library is used, which allows you to work with Ethereum network nodes using the Remote Procedure Call (RPC) protocol [3]. Thanks to the Web3.js library, you can use JavaScript language syntax to call smart contract functions. The library packages calls to the smart contract into transactions, which are then transmitted to the service provider, which in most cases will be the MetaMask cryptocurrency wallet. Next, the service provider can directly contact the Ethereum blockchain and transfer the finished transaction to it. In response, the client side receives data from the smart contract, which then, displaying the result, will make requests to the IPFS node to obtain the data (Fig. 1).

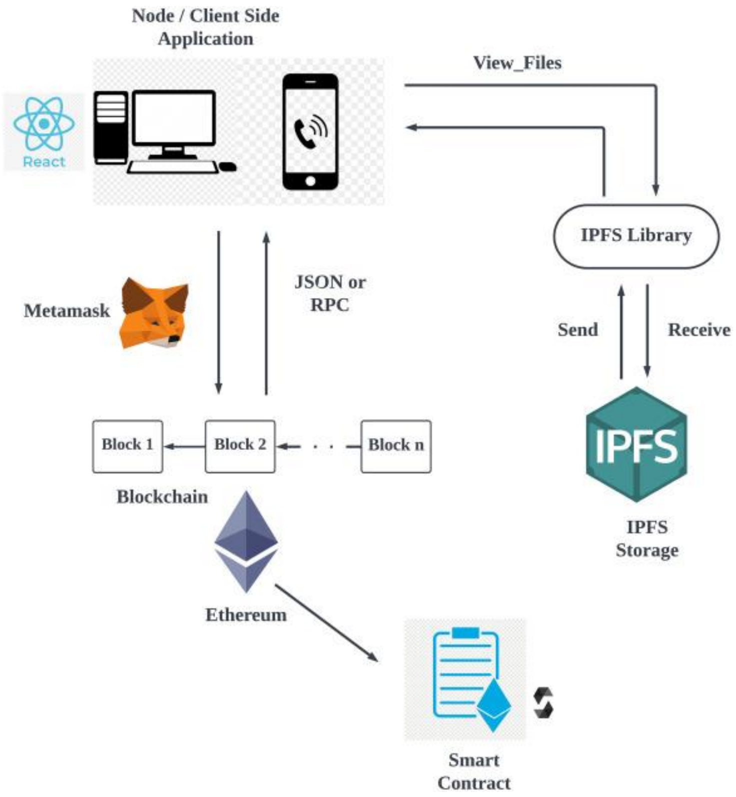

Figure 1 – Architecture of a decentralized application on the Ethereum blockchain using IPFS storage

An Image structure is created in the smart contract to store data about the image loaded into the application. This structure defines a cid field that contains the hash of the image returned after loading it into IPFS. When you add it to a URL with the structure "http://localhost:8080/ipfs/", you can get this image.

Since the application data will be located on the IPFS network, you must first configure a local IPFS node in the client part of the application. When a new image is uploaded to the application, the onSubmitHandler() function is called.

```
async onSubmitHandler(event) {
    event.preventDefault();
    const form = event.target;
    const files = form[3].files;
    const file = files[0];
    const ipfs = create({ url: "http://127.0.0.1:5001" });
    const result = await ipfs.add(file);
    this.props.uploadImage(
      result.path,
      parseInt(this.price.value),
      this.description.value
    );
    form.reset();
}
```

First, the image file is received from the form. An object is created to interact with IPFS. To do this, first import the "ipfs-http-client" library, then call the create({url:""}) method, which is passed the address listened to by the IPFS node. This method returns the ipfs object used to load data into the IPFS network. The ipfs object calls the add() method, which is passed the image file received from the form. The add() method returns the hash of the image.

Next, a call is made to the smart contract function uploadNewImage(), which records this hash and other parameters of the new uploaded image. The advantage is that the smart contract stores a unique hash of the image, so there cannot be two copies of the same image.

Conclusions. The features of creating a decentralized application for selling images on the Ethereum blockchain platform using decentralized IPFS storage are considered. Availability and reliability of the application is achieved by placing smart contracts with the logic of the application on network nodes in the blockchain, and the client interface, data and images in IPFS storage.

LITERATURE

1. I.V. Ponomarev Development of a decentralized voting application using blockchain technology - Системні технології. Регіональний міжвузівський збірник наукових праць. Випуск 1(126). – Дніпро, 2020, - с. 104-110.

2. I.V. Ponomarev Features of creating a voting system using the Ethereum blockchain platform - Системні технології. Регіональний міжвузівський збірник наукових праць. Випуск 1(132). – Дніпро, 2021, - с. 124-129.

3. Narayan Prusty. Building Blockchain Projects: Building decentralized Blockchain applications with Ethereum and Solidity. — Packt Publishing, 2017. — 268 p.

REFERENCES

1. I.V. Ponomarev Development of a decentralized voting application using blockchain technology – System technologies: Regional interuniversity compendium of scientific works. Issue 1(126). – Dnipro, 2020, - P. 104-110.

2. I.V. Ponomarev Features of creating a voting system using the Ethereum blockchain platform - System technologies: Regional interuniversity compendium of scientific works. Issue 1(132). – Dnipro, 2021, - c. 124-129.

3. Narayan Prusty. Building Blockchain Projects: Building decentralized Blockchain applications with Ethereum and Solidity. — Packt Publishing, 2017. — 268 p.

Received 05.02.2024.
Accepted 09.02.2024.

Створення децентралізованого додатку продажу зображень з використанням IPFS

Додаток продажу зображень завжди має бути доступним для клієнтів, надійним, захищеним від атак та фальсифікацій. Централізована архітектура для додатку не підходить, оскільки має єдиний центральний сервер для обробки інформації та зберігання даних, який є основною точкою відмови всієї системи. Тому, для надійної роботи, потрібен децентралізований додаток, заснований на технології блокчейн без єдиного головного вузла, з управлінням, розподіленим між багатьма вузлами мережі користувачів.

Створення такого додатку можливе за допомогою блокчейна Ethereum на базі смарт-контрактів, які зберігаються всередині віртуальної машини Ethereum та доступні при зверненні до вузла блокчейн мережі. Доступ до децентралізованого додатку можна отримати через веб-сторінку, яка за допомогою спеціальної бібліотеки звертається до смарт-контракту.

Сама веб-сторінка, як і всі її складові, такі як зображення, CSS або JavaScript файли можуть бути розміщені в таких сховищах як локальний комп'ютер користувача або централізований хостинг. Такі сховища залишається можливою точкою відмови всього додатка, і проблема стійкості до відмови залишається не вирішеною.

Для вирішення цієї проблеми пропонується використовувати децентралізоване середовище зберігання та передачі даних IPFS.

IPFS надає можливості децентралізованого зберігання файлів будь-якого типу. При розробці додатку враховуються особливості підключення та роботи з IPFS. Для кожного файлу при завантаженні в мережу обчислюється його унікальний хеш, яким відбувається звернення до цього ресурсу. Хеш-код зберігається в блокчейні за допомогою смарт-контракту, а потім надається користувачеві.

Доступність та надійність децентралізованого додатку продажу зображень досягається за рахунок розміщення смарт-контрактів з логікою роботи на вузлах мережі в блокчейні, а клієнтського інтерфейсу, даних та зображень у сховищі IPFS.

Пономарьов Ігор Володимирович - кандидат технічних наук, доцент кафедри електронних обчислювальних машин факультету фізики, електроніки та комп’ютерних систем Дніпровського національного університету ім.О.Гончара.

Ponomarev Igor Volodimirovich - candidate of technical sciences, associate professor of the department of electronic computers of the faculty of physics electronics and computer systems ofthe Oles Honchar Dnipro National University.