We announce the launch of series of articles about NFT technologies, in which we will describe the NFT project from the point of view of the developers. Now everyone is talking about web3.0, crypto technologies, blockchain and NFT. The ability to make big and even huge money with little investment has attracted a huge number of people and projects, which has led to various types of fraud in the creation and promotion of projects, which has also created many difficulties for new projects.

We will talk about our motivation, our view of the industry and its development over the past year, we will talk about financial investments and "tricks" of developers. We will start with more general topics and definitions and gradually move on to more complex, concrete examples from our experience.

Blockchain

Let's start with the basic definition. What is Blockchain? Many mistakenly equate specific networks such as Bitcoin or Ethereum with this concept. However, they only use the idea of blockchain. Blockchain is the combination of blocks of data into a chain. Each block contains a cryptographic hash of the previous block. A hash is a small number that maps to a large amount of data. Any change in the data in the block also changes its hash. Let's consider an example of several consecutive elements:

The zero block is special because there are no blocks before it. For it the first entry might be:

```
"B0": {
        "prev_hash": "0",
        "data": "..."
}
```

The first block will contain the hash from zero block:

```
"B1": {
        "prev_hash": hash(B0),
        "data": "..."
}
```

The second block will contain the hash from the first block:

```
"B2": {
        "prev_hash": hash(B1),
        "data": "..."
}
```

etc.

If we change the data in the block, its hash will change. The next block contains the hash of the previous one, so it will also transform. Because of what, its hash will also change.
Which will again entail a change in the block following it and its hash. That is, when data in a block changes, this block and all subsequent ones change.

That's the whole idea behind blockchain. It turns out a kind of chain, if we replace the concept of "change data" with "pull for a chain element", then pulling a certain block, all subsequent blocks are pulled.

Smart contracts and what is it about?

NFT technology is a type of smart contract. A smart contract is a program that allows you to sell, resell, donate, destroy NFT and more.

There are various marketplaces for selling NFTs. We chose https://opensea.io/ as one of the most popular. NFT creators who don't need any special features of the smart contract use the standard smart contract provided by the site.

You can also create your own smart contract for your collection and add more features to it. For example we added the following features to our smart contract:

- Half of the value of the NFT is saved in the smart contract wallet.
- Refund of half the value of the NFT when it is destroyed.
- Every time you resell NFT 2.5% goes into the total reserved amount of the entire collection. When you burn the NFT, you get your share of that amount. For example, if there are 10 Alvara NFT left, you get a tenth of that reserved amount.

So your NFT will never drop below half of its original value on opensea. Also you can always burn your NFT and get back half of its value and a share of all resales of the collection. If the part of your collection is burned, the value of the remaining pictures increases as the number of pictures decreases and the collection becomes more unique. Pretty nice, isn't it?

Types of data storage systems

Storing data in one place has its disadvantages. The owner can change the data, delete it, prevent any transactions, impose censorship, some restrictions. For example, let's look at electronic money and cash.

Centralized electronic systems are controlled by the owner of the central hub and they can impose any restrictions on your transactions, whereas with cash it's much harder. Also, centralized systems are easier to hack, change, and delete.

The advantage of a centralized system is its speed. It keeps everything in one place and doesn't have to deal with the task of synchronizing data from different storage locations. As an example, let's take a group of people to the movie theater.

If it is an educator and two children, the educator decides for everyone in what theater they will go and what movie they will watch, but in this case not necessarily every child will be satisfied. If three friends go to the movie, they spend some time negotiating, but everyone is satisfied with the result.

So distributed decentralized data storage is the storage of data on multiple computers. Each computer is the same as all the others, they all have the same program installed on them all.

Why Trust Blockchain?

Is the idea of blockchain enough to create a distributed decentralized system like Bitcoin?

Suppose we have an application that we can run on our computer locally and add some information (at the moment it doesn't matter what this information is) into a chain of blocks. Suppose this application has a possibility to publish its changes, so it is possible to press a button and then local blocks become visible to everyone else.

If the application works locally without access to the global network, then the information can be

trusted, no one can fake it. So what happens if you connect all this to the global network? Someone could take the block we created earlier and tamper with the data in it, then they would recreate all subsequent blocks with new hashes and publish their block sequence.

We would know that the newly published chain is a fake, but a third party seeing two versions of the block chains would not be able to figure out which one is correct. To make the system work, we need some mechanisms to solve such problems.

One way to solve the problem is to isolate one of the computers and give it special properties. We can make this computer a server for storing reliable information. Each network client publishing new data could send it to the server, it would have to read the data from this server too. This is how centralized systems work, it turns out that the trust in the whole system is the trust in the central server.

We consider complete decentralization when all are equal in the network and there is only one small but very important assumption, that there are more than half of honest users. This is where consensus algorithms (a way to agree on several programs) appear, and Proof of Work is the most famous one.