Blockchain

**PREPARED FOR**

Erik van der Schriek

Niko Kuijpers

Gertjan Schouten

**PREPARED BY**

Maarten Blömer

Nick Krijgsman

Faruk Aydin

Jursley Gonzalez

Vincent Andersen

# Table of Contents

[**Table of Contents**](#_z6lhuq9emniv) **1**

[**Case Study**](#_k6jx82pctwds) **2**

[Case goal:](#_bpxkra1gvikb) 2

[Case questions:](#_3vn6lbsuph60) 2

[Implementation](#_58tw4qf8lsdh) 2

[Brainstorming:](#_llbf4q7qnu37) 3

[**Independent study**](#_zc6h8lveooxw) **4**

[Cryptocurrency (Faruk)](#_eaou76j8xww7) 4

[How is blockchain used in cryptocurrency?](#_6gtzx46sm5xe) 4

[What determines the price?](#_s2ga9sq9i56b) 5

[Why is cryptocurrency so popular?](#_10tfezkw32ke) 6

[Permissionless blockchain (Nick)](#_y08rzwni0erd) 7

[Smartcontract](#_94vl5wypqjhi) 7

[What would a permissionles/permissioned blockchain architecture look like? (Jursley)](#_152e5vqtnyfs) 8

[What are permissioned blockchains](#_z8fb1phdf06o) 8

[What are permissionless blockchains](#_7ecvubo2dl54) 8

[Characteristics of Permissioned Blockchains](#_aggwvtplr50g) 8

[Characteristics of Permissionless Blockchains](#_buxlwbue8fgk) 9

[Use-cases of Permissioned Blockchain](#_246msq1xb7yl) 10

[Use-cases of Permissionless Blockchain](#_p32a00e2ob64) 10

[Basic Workflow and Architecture](#_un6crw355j7p) 10

[Implementation (Vincent)](#_fjh71osn0g1t) 13

[Implementation of a blockchain](#_6uo1ihf60uyt) 13

[Blockchain.cs](#_4cbok3kpwm95) 15

[Block.cs](#_3q4ydq9185j) 16

[Program.cs](#_4f7h38mm4rjo) 17

[**Sources**](#_qwibiw7iqp2l) **18**

# Case Study

## Case goal:

Gain more knowledge about emerging software trends in the group

## Case questions:

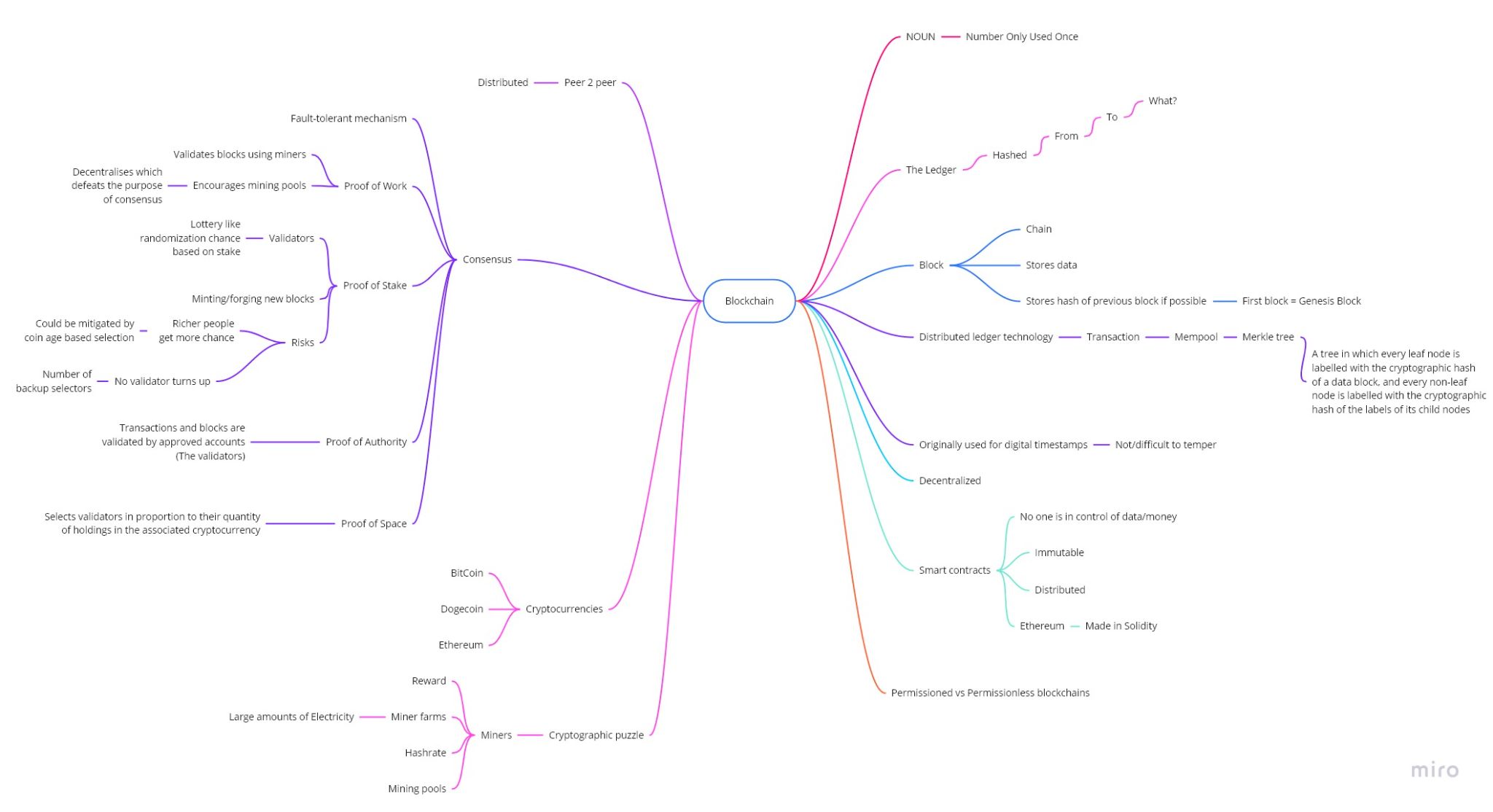
* **What would a permissionless/permissioned blockchain architecture look like?**
  + Differences in type of use cases
  + Architecture in practice
* **Permissionless Blockchain**
  + Which data or smartcontract should be verified in the ledgers?
  + In case of a smartcontract: when is the contract finalized.
  + Is the data GDPR proof?
* **Permissioned Blockchain**
  + Organize a coalition/alliance.
  + Agree on permissioned or permissionless blockchain
  + Design you application as usual
  + Decide on what the shared interest is.
  + Choose your blockchain wisely (ethereum, IOTA, lisk, hyperledger,....)
  + Design the smartcontract or datastructure.

## Implementation

* **Implementation of a blockchain**

In the implementation we will create a blockchain in order to gain understanding of how the theory comes together in practice. The code will feature some explanation given the research we have done beforehand and offer some of the basic functionalities of the inner workings of the blockchain. In the refinement we will discuss which other implementations of blockchain technology would be good to implement.

## Brainstorming:



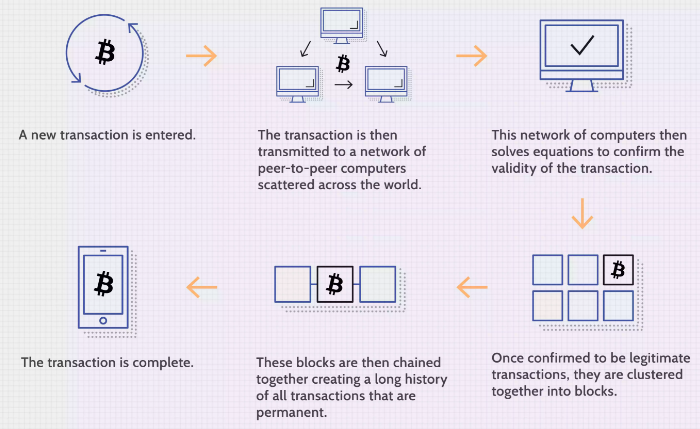
# Independent study

## Cryptocurrency (Faruk)

“Cryptocurrency is value in the form of an amount in a digital currency. This is a kind of digital currency, which is often used as an alternative money system for the regular currencies. The world's most famous cryptocurrency is bitcoin. The extent to which cryptocurrency is actually money is controversial. In the Netherlands, for example, bitcoin has the legal status of a medium of exchange, but not that of money.”

### How is blockchain used in cryptocurrency?

Many cryptocurrencies rely on blockchains, which are organizational mechanisms for ensuring the integrity of transaction data.



Blockchain technology is central to the appeal and functionality of Bitcoin and other cryptocurrencies, as it is used to keep an online ledger of all transactions that have ever been conducted, thus providing a data structure for this ledger that is quite secure and is shared and agreed upon by the entire network of an individual node, or computer maintaining a copy of the ledger. Every new block generated must be verified by each node before being confirmed, making forging transaction histories nearly impossible.

**Secure?**

Blockchain technology is responsible for the problems of security and trust in several ways. First, new blocks are always stored linearly and chronologically. They are always added to the "end" of the blockchain. If you look at the Bitcoin blockchain, you will see that each block has a position on the chain, called a 'height'.

It is extremely difficult to go back and change the contents of a block once it has been placed to the end of the blockchain unless the majority has agreed to do so. That's because each block has its own hash, as well as the hash of the block preceding it and the timestamp specified before. A mathematical procedure that turns digital data into a hash code generates hash codes.

Although cryptocurrency blockchains are extremely secure, other components of the cryptocurrency ecosystem, including exchanges and wallets, are vulnerable to hacking.

### What determines the price?

Because bitcoin is neither issued by a central bank or backed by a government, it is not affected by monetary policy, inflation rates, or economic growth indicators that affect the value of traditional currencies. Cryptocurrency prices, on the other hand, are impacted by the following factors:

* The market's demand for cryptocurrency and its supply.
* The cost of creating a cryptocurrency via the mining process.
* Cryptocurrency miners are rewarded for validating transactions on the blockchain.
* The number of cryptocurrencies in competition.
* Regulations governing its sale on the exchanges on which it trades.
* Its internal governance.

### 

### Why is cryptocurrency so popular?

One of the main reasons cryptocurrency is so popular around the world is that it has very few fees associated with it. Another reason is that cryptocurrencies are not linked to any world governments. This means that cryptocurrencies have the ability to remain steady even when a country is in turmoil.

The potential to make profit from cryptocurrencies is, of course, another major cause. Cryptocurrency is much more secure than many other traditional payment options to pay for things online. This also makes it appealing to anyone with the wrong motives.

Finally, many people believe that cryptocurrencies are the way of the future when it comes to money. Cryptocurrency users are also adopting important technology advancements such as blockchain. This allows you to be on the cutting edge, and many people appreciate the opportunity to be ahead of the game. Blockchain technology is expected to affect the world in a variety of ways, including making trading much more transparent.

## 

## Permissionless blockchain (Nick)

*“A blockchain network where the network is publicly available for participation.”*

## Smartcontract

* + Which data or smartcontract should be verified in the ledgers?
  + In case of a smartcontract: when is the contract finalized.

Is the data GDPR proof?

## 

## What would a permissionles/permissioned blockchain architecture look like? (Jursley)

### What are permissioned blockchains

These blockchains, also known as private blockchains, can be thought of as closed ecosystems that can only be accessed by those who are allowed access. Anyone who is interested in validating transactions or viewing data on the network needs to get approval from a central authority. This is useful for companies, banks, and institutions that are comfortable to comply with the regulations and are very concerned about having complete control of their data. Ripple is a perfect example of a permissioned blockchain.

### 

### What are permissionless blockchains

Popular blockchains such as **Bitcoin, Ethereum, Litecoin, Dash, and Monero fall under this category.** Also known as public blockchains, these allow anyone to transact and join as a validator. The data on these blockchains is publicly available, and complete copies of the ledgers are stored across the globe. This is what makes it hard to censor or hack these systems. This blockchain does not have anyone who controls it, and one can remain relatively anonymous as there is no need for identifying themselves to get an address and perform transactions.

### 

### Characteristics of Permissioned Blockchains

1. **Transparency and Anonymity**

These blockchains need not be transparent, but they can choose to be so, depending on the inner organization of businesses. In terms of privacy, these are not needed on a central level and can be determined on a user-case basis. These blockchains store an extensive amount of data relating to the operations and transactions carried out by users.

2. **Varying Decentralization**

Members of this blockchain are free to negotiate and come to a decision about the level of decentralization the network can have. Private blockchains can be fully centralized or partially decentralized. These are free to choose the consensus algorithms they wish to employ.

3. **Governance**

Here, governance is decided by the members of the business network. There are various dynamics to determine how decisions are made on a central level. But here, there is no need for consensus-based mechanisms where the entire network must agree to a change.

4. **Efficient**

Every single data on the platform is immutable that means you can’t change them anytime.

### Characteristics of Permissionless Blockchains

1. **Digital Assets**

In a permissionless network, there is a user-incentivizing token that can increase or decrease in value based on the relevancy and state of the blockchain they belong to. Depending on the purpose, these blockchains either employ monetary or utility tokens.

2. **Transparency**

According to default design, these are bound to be transparent. Users of the network must be incentivized to trust the network. A transparent network must give users access to all information apart from just the private keys, and this can include addresses, freedom to see transactions processed by the network, and the way in which transactions are processed into blocks.

3. **Decentralization**

These are decentralized as there is no central entity that shuts down the network, changes its protocols, or edits the ledger. These are based on consensus protocols. This means that network changes of any type can be achieved only if 51% of the users agree to it.

4, **Immutable**

Every single data on the platform is immutable that means you can’t change them anytime.

### Use-cases of Permissioned Blockchain

There are multiple use-cases of permissioned blockchain including

* Research
* Food tracking
* Banking and payments
* Asset ownership
* Internal voting
* Supply chain management

### 

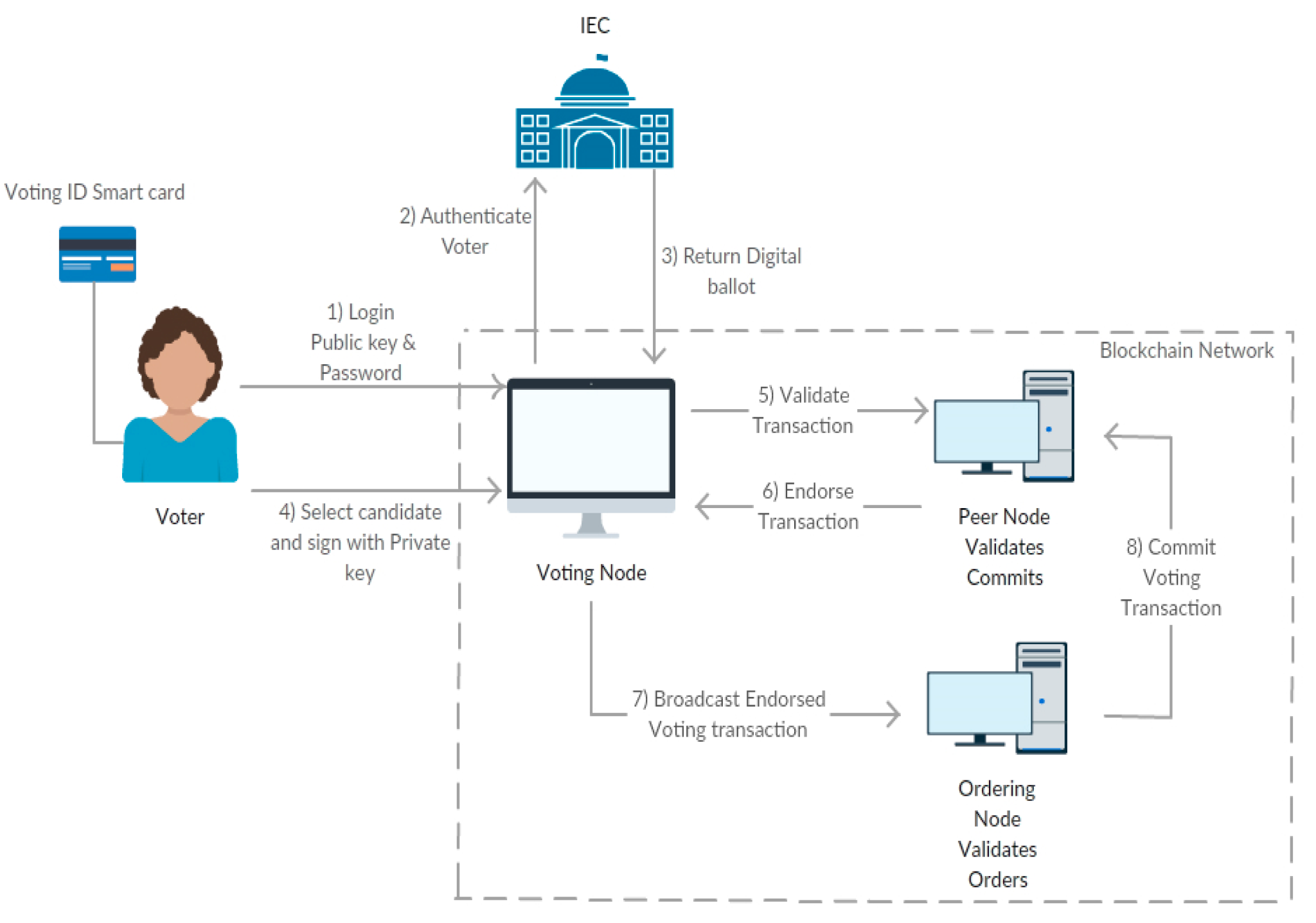
### Use-cases of Permissionless Blockchain

There are multiple use-cases of permissionless blockchain. Some of them include the following

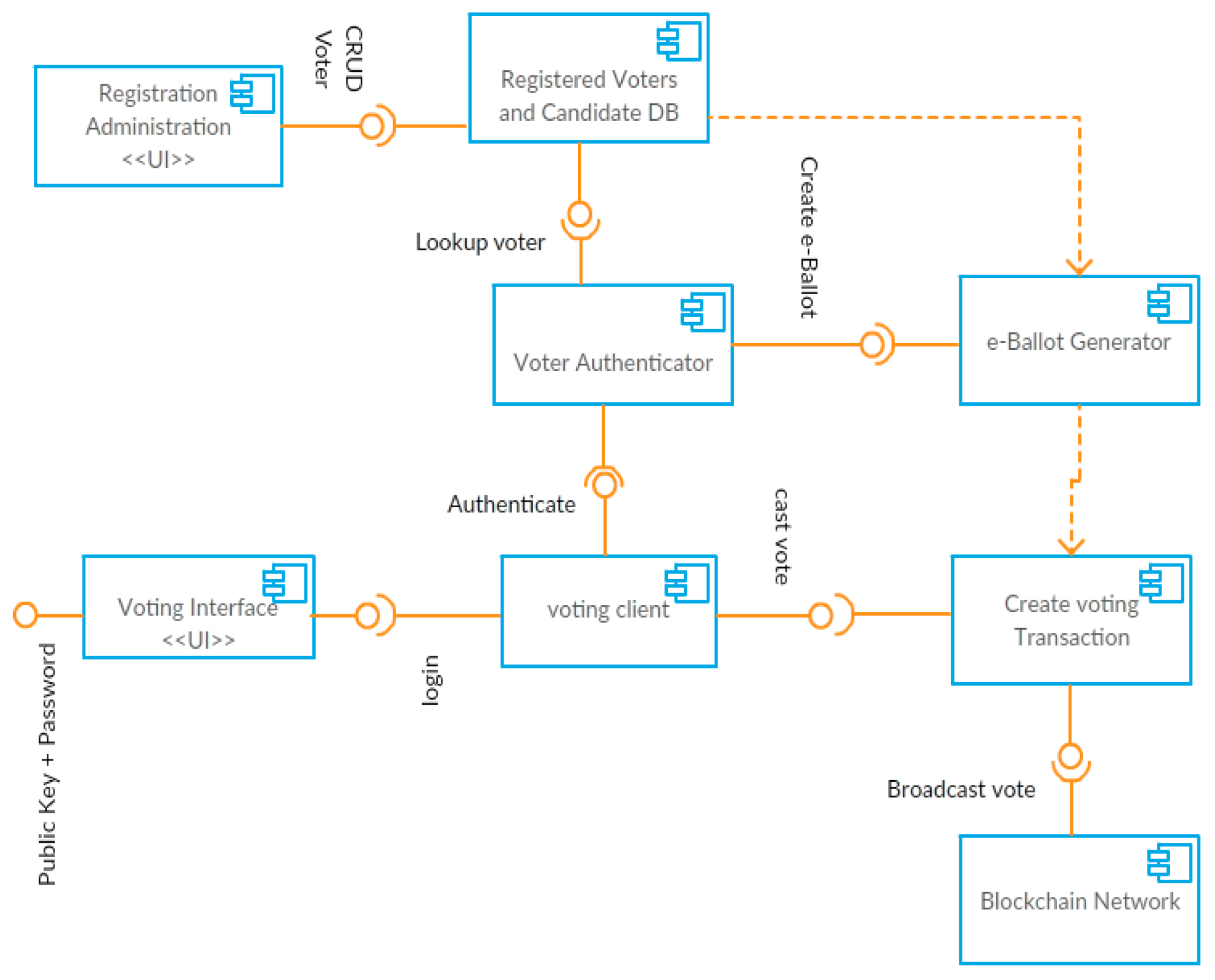
* Digital Identity
* Voting
* Fundraising

### Basic Workflow and Architecture

**Workflow**



**Architecture**

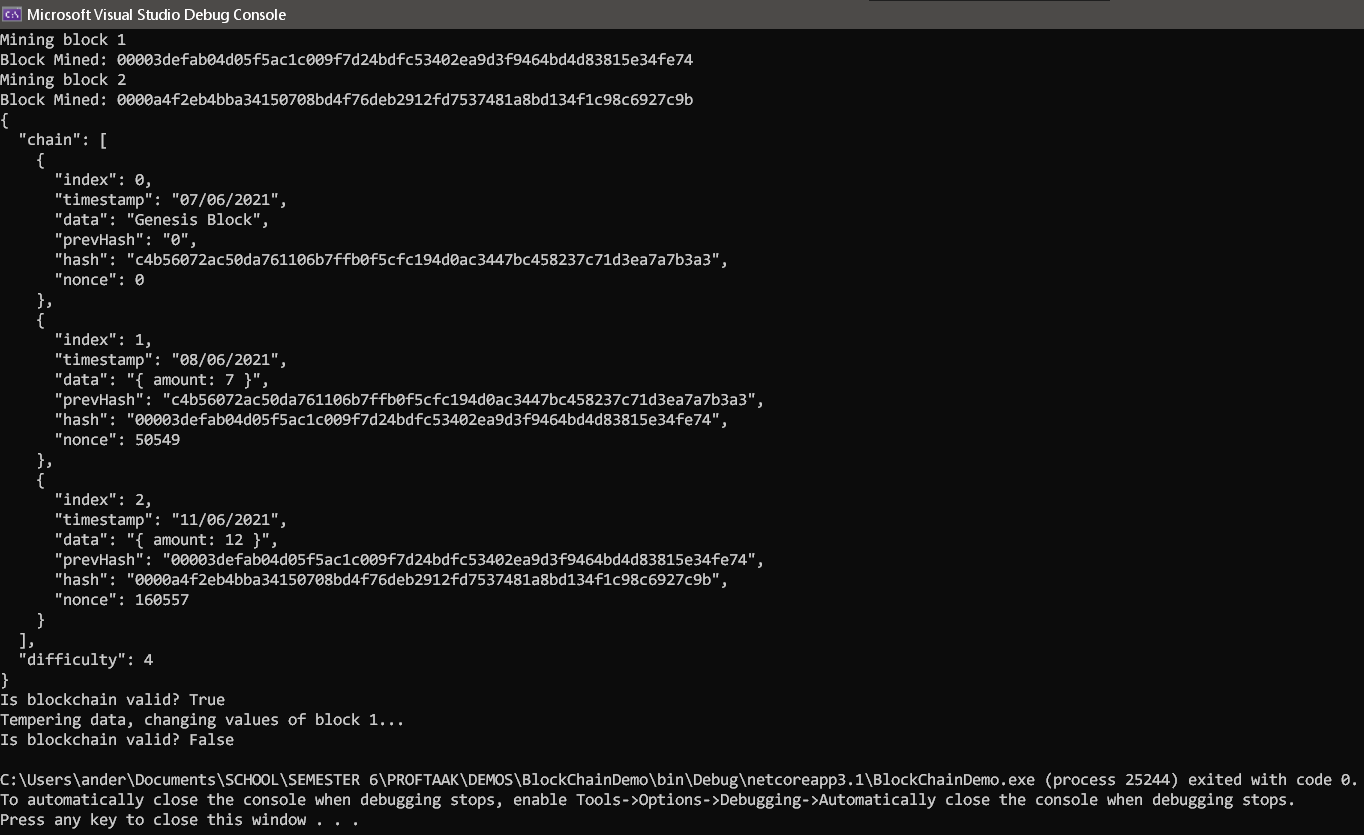


## 

## Implementation (Vincent)

### Implementation of a blockchain

The blockchain in and of itself is not the most complicated thing, the whole other structures around the blockchain that can be added makes it varyingly more complex based on the requirements that the developer wants to fulfill. In order to prove that making a simple blockchain is simple enough, we will be exploring the making of a simple blockchain program.



In this Console log file you can see it starts the chain with adding blocks. These blocks have a difficulty of 4 which makes them relatively easy to mine before they’re added to the blockchain. The chain is then open to see with the given data put in them along with hardcoded timestamps.

The interesting part here is that you are able to verify the validity of the block 1 and 2 as they hold the previous hash value, which makes them valid.

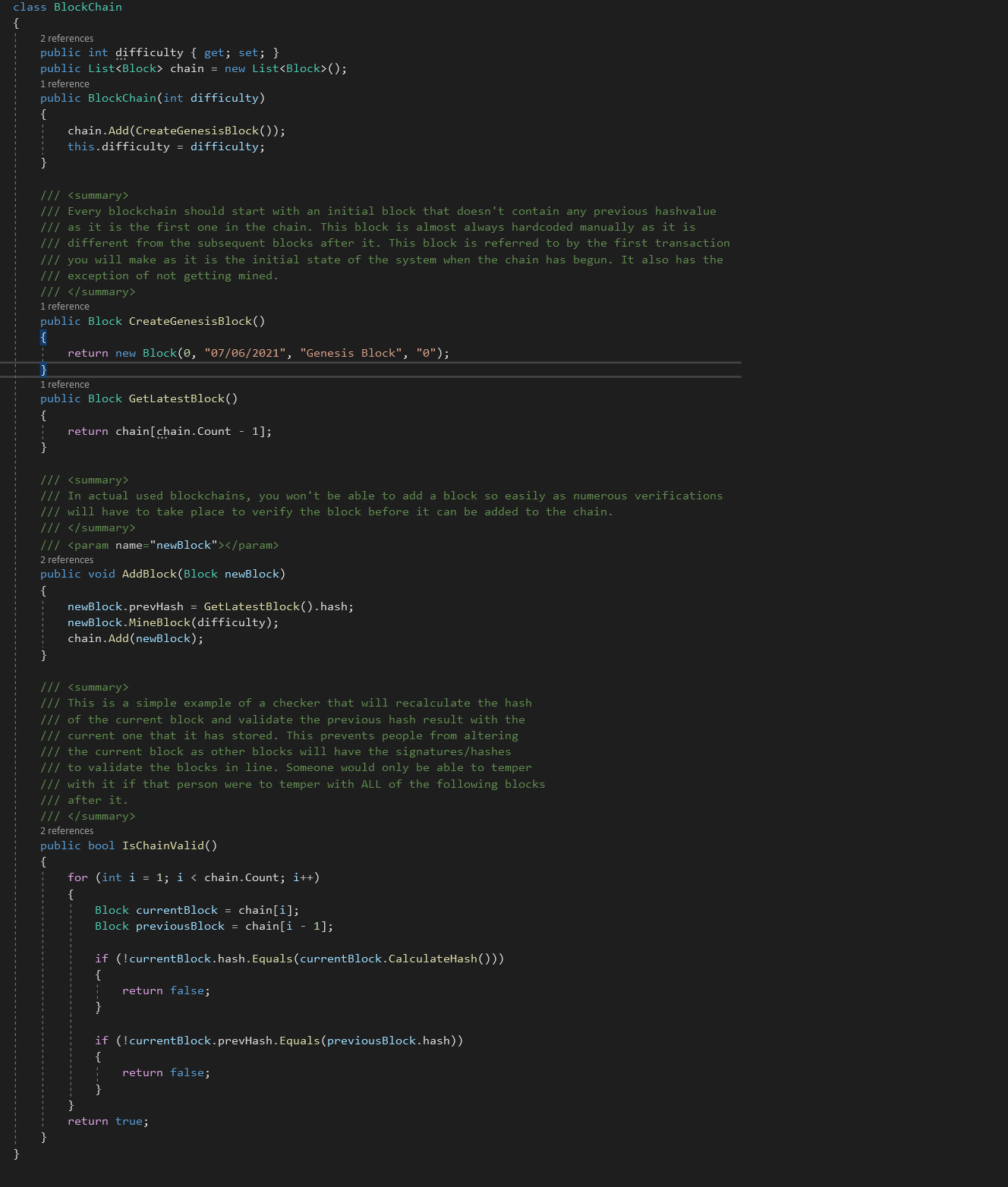
At the end we verify the blockchain by checking the hash value of the current block and hash value of the previous block in order to see if they’re legitimate or not. Normally this would be done over a large number of computers in order to come to a consensus of whether the values have been tampered with or not. In this example, only my computer checks it.

Then we temper with the values of 1 block and check if the blockchain is legit again which will come out as false as the data values and hash value are not equal to the value that the other blocks have.

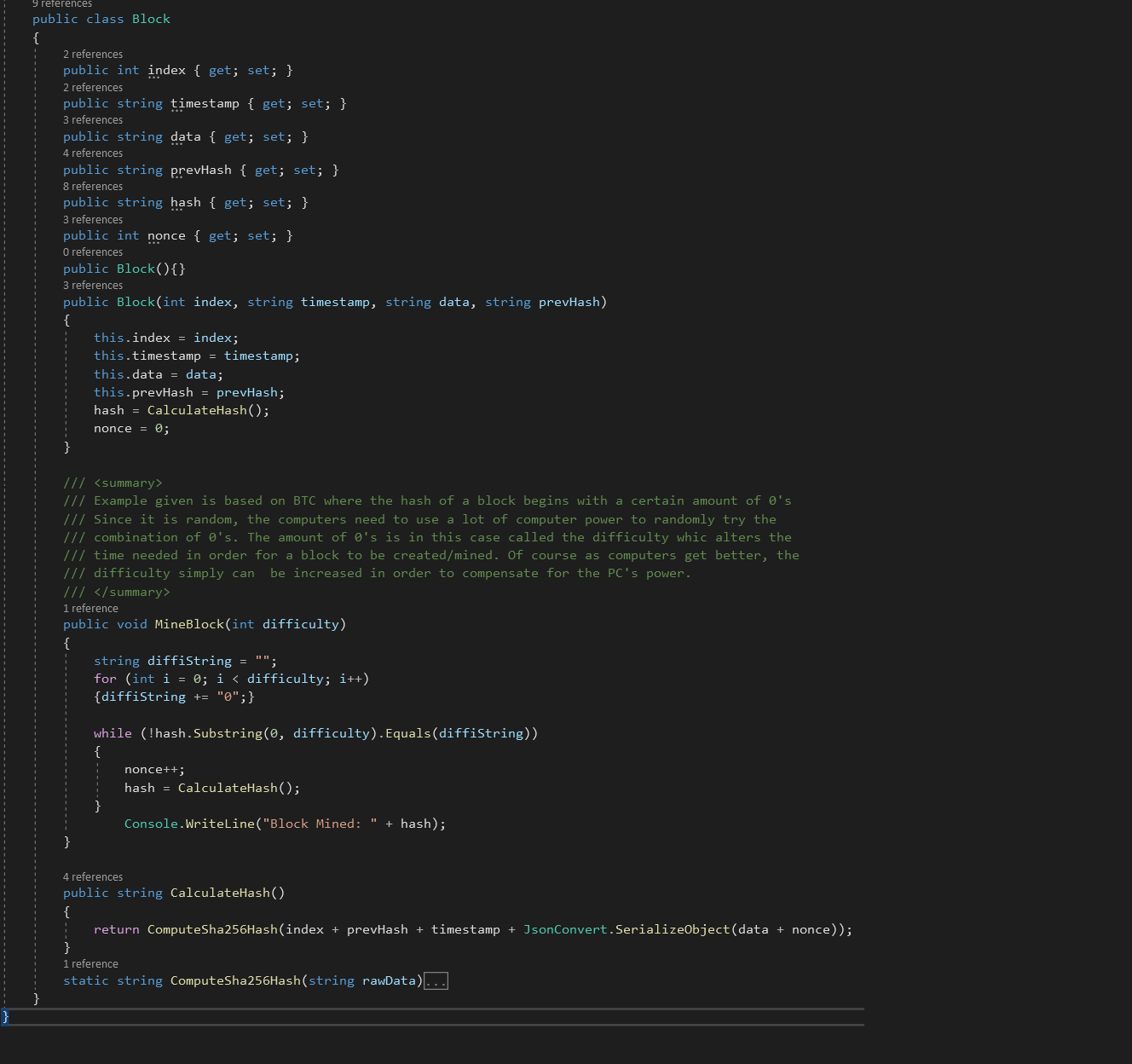
You can try out the project [here](https://drive.google.com/file/d/1jzr4PAxFbSNq4Ce7lXEsYk3-9Sr0r9XE/view?usp=sharing).

Java version: [Blockchain Java](https://drive.google.com/file/d/12QBWYnQSUF3OiOuOnoCCs1F6vZtCAjuS/view?usp=sharing)

#### Blockchain.cs



#### Block.cs



### 

#### Program.cs

#### 

# 

# Sources

|  |  |
| --- | --- |
| **Framework** | **Source** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |