

**GENERAL PURPOSE  
REPUTATION DAPP**

**MCTY MSC ELECTRONIC  
AND COMPUTING TECHNOLOGY**

**BLOCKCHAIN  
SCALABILITY: EE516**

**SEMESTER 1 2021/2022**

**ASSIGNMENT 1**

**BIROL KILIC**

**21269165**

## Declaration

<b>NAME:</b>	<b>Birol KILIC</b>
<b>STUDENT ID NUMBER</b>	<b>21269162</b>
<b>PROGRAMME</b>	<b>MSC ELECTRONIC AND COMPUTING TECHNOLOGY</b>
<b>MODULE CODE</b>	<b>EE516</b>
<b>ASSIGNMENT TITLE</b>	<b>EE516 ASSIGNMENT</b>
<b>SUBMISSION DATE</b>	<b>15/112021</b>

I understand that the University regards breaches of academic integrity and plagiarism as grave and serious.

I have read and understood the DCU Academic Integrity and Plagiarism Policy. I accept the penalties that may be imposed should I engage in practice or practices that breach this policy.

I have identified and included the source of all facts, ideas, opinions and viewpoints of others in the assignment references. Direct quotations, paraphrasing, discussion of ideas from books, journal articles, internet sources, module text, or any other source whatsoever are acknowledged and the sources cited are identified in the assignment references.

I declare that this material, which I now submit for assessment, is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

I have used the DCU library referencing guidelines (available at [https://www4.dcu.ie/library/classes\\_and\\_tutorials/citingreferencing.shtml](https://www4.dcu.ie/library/classes_and_tutorials/citingreferencing.shtml) **and/or** the appropriate referencing system recommended in the assignment guidelines and/or programme documentation.

By signing this form or by submitting material online I confirm that this assignment, or any part of it, has not been previously submitted by me or any other person for assessment on this or any other course of study.

By signing this form or by submitting material for assessment online I confirm that I have read and understood the DCU Academic Integrity and Plagiarism Policy (available at <http://www.dcu.ie/registry/examinations/index.shtml>).

Signature\_\_\_\_\_

A handwritten signature in black ink, appearing to be 'B. J. R.', enclosed within a hand-drawn circle.

15/11/2021

Date: \_\_\_\_\_

# Introduction

## Blockchain as a reputation system

Blockchain technology gained popularity by promising to solve the majority of the problems that we are facing in the current services in the internet. In simple terms blockchain is shared, globally accepted and with the help of cryptographic hash functions is an immutable database. Allowing peers to exchange data without the need of trusted third parties. Blockchain gained its popularity after bitcoin, a public distributed ledger allowing peers to exchange cryptocurrency. [0] After the invention of bitcoin, people from whole around the world tried to make variations of bitcoin for different purposes. Bitcoin utilizes distributed technology. The properties of bitcoin enable to use this technology in various services, products such as education, supply chain management, fraud detection in different services, international money transfer even allowing 3rd world countries to run their economies on a globally accepted cryptocurrencies, at least in individual bases [1]. These are all promises that blockchain technology gives us with assumption of improvements in the security and scalability domains in the underlying technology. One use case of blockchain technology could be suitable for reputation system where in the current services that provides reputation systems are untrustable in different aspects. Such as

- Service provider owns the reputation system, allowing him to do arbitrary changes without getting permission of anyone.
- Multiple identities, allowing users to cheat on their ratings and This becomes a huge problem, to understand better what is the situation, currently companies are even built on sole-purpose of cheating in reputation algorithms under the name of public relations.
- Each company has its own reputation system making impossible to achieve a uniquely identified, whole purpose reputation system (eBay, amazon).

So in this proposal our aim is to design a blockchain system with the purpose of solving the problems that we are facing with the current services and try to address the new problems that possible to occur with the blockchain technology.

## Related reputation systems and algorithms

In the recent years most of the proposals or work done to accomplish a reputation system is in the context of p2p file sharing networks, allowing users to rate the file providers, by the quality of the files that they receive.[2] TRUSTDAVIS is one possible way to achieve a fair reputation system where the basic logic behind is a third party sits between the user and the service provider called insurer, and if any conflicts occurs between the user and the service provider this third party basically decide who is right however this system is not a good way to solve the problem cause of the subjectivity of the insurer as well as the capacity. Another existing implementation of reputation model is proposed by wang [3] which is a Bayesian network-based trust model. Wang has novel approach in reputation systems by adding the trust into the reputation ecosystem by trust he means a peer's belief in another peer's capabilities, honest and reliability based on its own direct experiences. This approach allows peers reach the most suitable peer in their context However there is a one big deficiency in Wang model which assumes that the all the nodes in the ecosystem are honest. Adding trust into reputation system is good argument however wangs approach apart from assuming the nodes will be honest in their rating, he couldn't give a solution to identity management. If we consider a reputation system that will work in with real world, main concern is users providing more than one identity which any of the users can vote more than once so fairness in the system is can't be trustworthy. M.GUPTA [4] represent Another reputation system that is implemented in peer-to-peer network, system allows its participants have the option to participate in such rating algorithm for the privacy reasons but if you are actively involved in a system that involves participants more than one,

collaborative participation need the system function right. One of the most highly valuable contribution in peer-to-peer network knows as EIGENTRUST [5] algorithm in simpler terms, peers who provide authenticated files to other peers, not only trusted by means of providing file but their opinions on other peers as well assuming once a honest, will stay honest at all terms. So, this led to centralise in terms of to establish good reputation point in the system you should already be trusted in the system this led the pre-trusted peers will always be trusted. Contribution of new nodes into system becomes much harder. These are some of the worthy contributions in reputation systems based on rating the file sharing of the peers in network however their approach can be taken to implement any kind of rating mechanism from social credibility to service rating. Schaub, Alexander, [6] proposes a blockchain-based reputation model that allows customers to rate the service provider. It is more customer-centric which allows only customer to rate the transaction. Uses blind signatures for the sake of anonymity and privacy, however if we want to implement a reputation mechanism it contradicts with the nature of the reputation. Lastly In the proposal of EIGENTRUST, it defines 5 points that any p2p reputation systems should address a solution.

- system should be self-policing no central authority.
- privacy and anonymity
- not assign any profit to newcomers (can't re-establish the status)
- minimal overhead in terms of blockchain scalability issue.
- robust to malicious collectiveness

Flaws of the reputation systems are in front of the eyes of the people some real world projects are started to take place with the promise of solving the major problems that people are facing today. DREP FOUNDATION[7] is an example where they are trying to create reputation management Framework. In their proposal they claim are able to achieve

- Verify user identity, protection against duplication
- bi-directional reputation system
- general accepted reputations of individual
- Token based incentive mechanism

However, no real-world implementation currently can be tested to see what percentage they have achieved what they are claiming.

## Blockchain platforms

All the blockchains that are introduced to our life are based on the same concept of distributed ledgers and encryption. There are many types of blockchain platforms today that enables people to creating distributed and decentralized applications. Main benefits of blockchain comes from its

distributed, decentralized, time-stamped blocks are stored in a way that is secured by cryptographic hash functions making it impossible to change and verify and unverified transaction.

Each blockchain technology have their own functionality and capability, the original bitcoin in the context of functionality, only function it has is exchanging values and keeping the records in the blocks. This functionality let the bitcoin as a use case of P2P electronic cash system in an open, public, distributed, decentralized and trustless (without the need of trust between participating nodes and without a trusted intermediary) environment. Bitcoin solve the double spending problem with the proof of work scheme. In Similar fashion Byzantine generals problem is solved by assumption of hones nodes are in the majority. In order to gain majority, you will need the more than 50% of computation power in the system. On the other side bitcoin has flaws. First flaw came with the scalability issue with the POW mechanism, where its capability is only limited around 10 tx/s in later chapters we would offer some solution to this problem as the size of the network increases. Bitcoin

are powered by stack-based programming language called bitcoin script. Bitcoin script is intentionally limited. Main reason of this limitation is smart contract that implemented in bitcoin is stateless. By stateless its meant to be can only execute transfers, they cannot trigger additional contracts and this is the biggest difference between the two, there is no chaining of contracts [8]. On the other hand another blockchain platform that uses POW as an consensus mechanism is ETHEREUM with extra features of programmable smart contracts that gives anyone the opportunity write their own smart contract on specific logic and use case. Ethereum simply creates statefull contracts allows execute code to change blockchain

## System overview

### Incentive mechanism of the reputation system

Before examining the choice of architecture. I would like to give brief description about what is the logic behind the reputation algorithm that we will try to implement. The approach we take on reputation system is customer-centric what is mean by that in a normal situation, customers will rate the service provider with underlying assumption of being honest but assuming the customer will be always honest its some kind of fooling yourself. To simplify and decrease the complexity in the structure we divide users into two categories, one user can be registered as a ratees or rater cant be both(in practical they can but need to open another account). Rater are costumers, and ratees are the service providers but this doesn't mean reputation points are limited to service provider both types of users will have reputation points accordingly to their honesty in the [system](#). However if service provider is not happy with the feedback given by the user and he really believes that the feedback were given by the user is not honest. He has ability to open a publicly available ballot where both the raters and ratees can vote through the ballot. However again we stuck in situation of if the voters are going to be honest. We can solve this problem by the concept of Schelling point which is developed by game theorist Thomas Schelling[9]. He describe his solution as people as people agree on same thing in the absence of communication as an that's seems to relevant to [him](#). To make more understandable he gives an example of in the situation like this “Tomorrow you have to meet a stranger in NYC. Where and when do you meet him?”. While any place and time in the city could be a solution, the most common answer is “noon at the information booth at Grand Central Terminal”. There is nothing that makes noon at Grand Central Terminal a location with a higher payoff (any other place and time would be good, provided that both agents coordinate there), but its tradition as a meeting place makes it a natural focal point. So based on this concept with an additional incentive mechanism we can achieve a voting scheme that users are going to be honest.

[10]Vitalik buterin creator of the Ethereum show how can Schelling coin mechanism can be used as an incentive mechanism to make users stay honest. Schelling coin working can be seen below..

An example ballot can be like this;

Q-if which country has better economy?

A- America

B- Kenya

Each user that has specific amount of that coin can be in voting. As you can guest most of the majority will give their vote on America. So America is selected as right answer. Parties who voted in same as the majority will gain 10% of the coins. Parties who voted opposite side will lose 10% of their coins.

So, let's convert this ballot into a reputation scenario

Q-Alice gave 5 eths to bob for receiving Beethoven music sample, Bob gave Alice to Mozart sample. Alice rate bob as 0. (meaning not satisfied).Bob release a public ballot that he thinks he is right?

All the specified details in the ballot are publicly available on Ethereum blockchain.

A-Alice

B-Bob

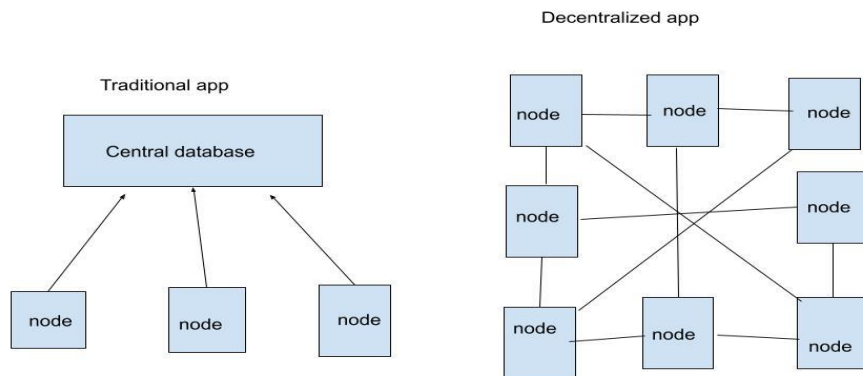
Majority of the voters will vote for Alice. So the reputation points were given by Alice is accepted as a right by the majority. However, in our system options of the ballot should be associated with the wallet address. So majority find Alice as right in this situation option a is associated with alice wallet address so alice wallet address will earn one more kleos so she is more trustable, bob will lose another kleos and majority of the voters gain kleos as well for the right answer. By this incentive mechanism every node in the reputation system is incentivised to stay honest.

## Ethereum Network

To achieve a reputation system based on underlying incentive mechanism, the proposed architecture should address number of different aspects. Decentralised app that build on top of Ethereum blockchain infrastructure will fit the demands of this related different aspects. Before going into details of the app. architecture I want to point out some general characteristics of decentralized apps have. Dapps will run on distributed peer to peer network instead of server to client model. Main benefit that come from this model as long as the nodes are participating in the network there are no down time. Another feature is the transparency, users of the dapps can see the functions and intentions about of the application that they are using, allowing a better trust relationship. Lastly Censorship resistance, on the business logic is deployed in the application, no matter what is the situation, including the developer nobody have the permission of doing changes. However likewise most of the blockchain systems a trade-off situation can be considered in dapps. Such as censorship resistance making impossible to maintenance of the app. There are some other challenges related to decentralized infrastructure in the context of scalability and security i will try to address and suggest possible solutions in later sections.

General architecture of dapps compared to traditional apps in high level can be seen in the figure below.

### Figure 1.[20]



## Components of the Architecture

### Consensus layer:

In simple terms allows nodes in the network to work together, agree on certain information and keep secure. Ethereum network choice is Proof of work. However other kind of consensus protocols are available such as Proof of stake and Ethereum has a plan to move proof of stake in order to increase scalability and decrease the waste in computational resource. However, i believe base layer should stay with POW, the implementation POS will lead to give power to the people who have higher stakes, eventually more power will lead the higher stakes, so eventually the whole security of the systems will be in hands of couple of validators moral value. Apart from their moral values such different circumstances can occur when the security and integrity of the system will rely on few validators, such as compromised private keys, this kind of attacks will all led to increase in single point of failure vectors. So to achieve scalability with ETH.2.0 will do some works POS mechanism with integration of sharding but main solutions should be achieved by layer 2 approach. I will try to give more details about scalability and security in later sections.

### Software layer

The main reason is that Ethereum platform is chosen for particular system its flexibility. Unlike bitcoin, Ethereum extends functionality of bitcoin blockchain not just transferring value but general purpose blockchain. Another reason is allowing to create token which will play major role in our incentivising the participants to stay honest with a reward mechanism. In generally software layer allow developers to generate smart contract that enable programmable transfer of value. However smart contracts implementing a large complex program will face with the problem of GAS limit. Another problem that smart contracts can have is limited storage size due to gas prices. So in order to make your application more efficient finding an external way of storing the files will be better

### EVM (Ethereum virtual machine)



ETHEREUM is a distributed state machine. ETHEREUM not only stores accounts and their related values but the states of the machine with pre-defined rules, this change happens block to block. The rule changes depending on the block are defined by EVM.[12]

## ABI (Application Binary Interface)

Smart contracts are written in high level languages like solidity, and they are executed in EVM. To EVM understand the instructions given by smart contract needs to be compiled in EVM executable bytecode. An ABI is necessary to invoke the right function as well as receive the expected data.[13]

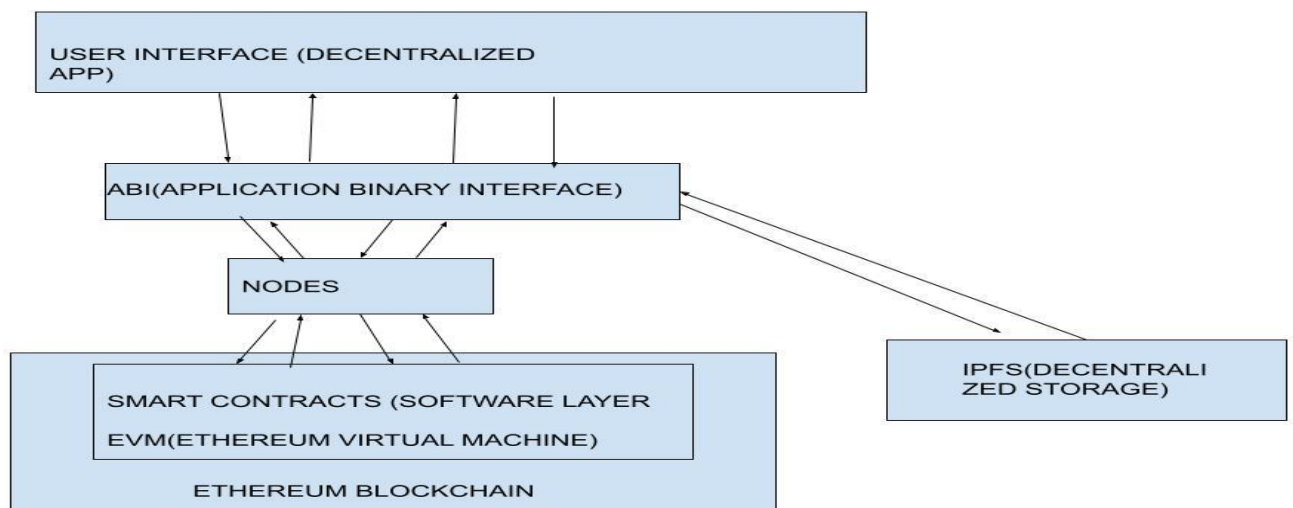
## Data storage

Due to gas cost, large files and images into smart contracts are not a good practice. We need to implement an external storage, implementing a centralised database will contradict with the whole point of decentralised architecture of blockchain so to store the files in peer-to-peer decentralised fashion IPFS is suitable option.

## Frontend layer

This layer allows developer to implement a similar technology that are used in web2 to models as a user-interface for the users. A good user interface is crucial to implement a popular app among the users cause people always be scared of unfamiliar things. Allowing users to use typical web app as in web2 model will decrease the adoption time of the users.

**Specific diagram representing the proposed Architecture can be seen in figure 2[20]**



## Architecture workflow

## ETHEREUM SMART CONTRACTS

In Ethereum platform there are two types of accounts one is externally owned accounts (EOA) and the other is contract account. EOA is related to real/legal person. Creating an EOA costs nothing Creating and executing contract account has a cost because you're using network storage and computational resources. [14]

### Architecture and workflow of smart contracts

Three different contracts will be implemented into system

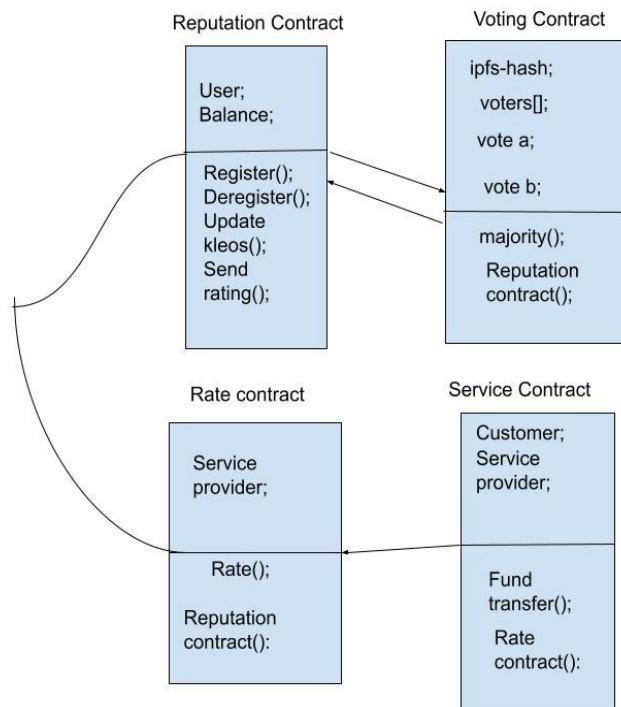
1-REPUTATION CONTRACT-keep track of users in the system and their associated kleos and eth balance. Users will be defined in two categories however they will have common properties

2-VOTING CONTRACT- Allows users to create a voting scheme in dapp.

3-SERVICE CONTRACT-Payment is done via Service contract , invokes the Rate contract

4-RATE CONTRACY-Customer rates the service via rate contract.

Figure below shows the architecture of smart contract. Figure 3



- User can register or login website with their associated EOA. Registering an account will cost some gas + 1 ether. This ether will be stored in smart contract with mapping to corresponding public key of EOA that send the ether. this prevents user to open multiple account in order to create multiple identities over the system and for incentive mechanism to nodes stay honest.

Registering will invoke reputation manager contract. Users can register as a customer or service provider.

- From application customers and service providers can agree service. For the simplicity let's assume payment going to done via cryptocurrency. To understand procedure better example can be given as Customer will agree a job of creating a flyer content from service provider. Service provider creates a service contract payment should be done via service contract. As soon as customer receives the content, he demanded payment is done by service contract. Automatically invokes the rate contract for customer to rate customer
- Customer is not satisfied with the work give rate of 0. However, service provider thinks this is unfair.
- Service provider can now publish a voting. This will cost 1 eth. So, everyone can see the details of the smart contract and necessary files can be uploaded from ipfs, which publicly available. Voting smart contract will randomly selects people from the list of reputation manager contract and allows them to vote scheme they which. Each user contributes to voting will send their votes and signed them by their private key through the app. This will let the reputation smart contract to be sure no other person than randomly selected people will vote.
- As we know in the sign-up process everybody sends 1 eth with their corresponding public key. Reputation manager will automatically discard the public keys from the list if they have eth less than 1. So, by this mechanism its making sure every participants who have selected to contribute in voting will have certain amount of eth. Some of the eth will be used as an gas price some of them will sent to majority.
- The majority of the voters who vote same thing and the winner of the election will earn kleos as well as some eth that given by Alice and minority.
- Also, another scenario to gain kleos is just by creating a voting scheme by uploading a video or photograph that you might think you are doing something valuable to the humanity by this way you can earn kleos and our reputation system can be used for ranking social credit system. The files can be uploaded through ipfs. Allowing everybody to reach the files in a secure way.

## Security Concerns

### Security breaches related to blockchain and P2P network

Currently ETHEREUM uses POW likewise BITCOIN, bitcoin mining algorithm compute SHA-256, to compute SHA-256 the most efficient way is to use ASIC (application specific integrated circuit). So in order to contribute in mining process you will need high capital, keeping small players out of the game resulting in centralized mining pools. Which in bitcoin case 51% or more processing power are in the hands of three mining pools however miners in these pools have the chance of changing their pool if the mining pool tries the 51% attack. In Ethereum case, Ethereum fetches random data from the state, so specialized CPU is not a great way to have more power in the mining process. [15]

### Eclipse attack and Sybil attack

Ethereum network is public permission less network, the attacker impossible to know beforehand who is going to validate next block so its hard to predict to locate in hardware domain. In extension to this Ethereum's node discovery protocol is similar to Kademlia. Same XOR metric with a difference of eliminating the need of identifying the target that trying to reach so nodes are selected randomly. Kademlia protocol creates a chance of vulnerability to eclipse attack. Kademlia works with the logic of each nodes knows and communicates via their neighbour nodes this is where its efficiency comes

from. The distance is based on nodes id which is every node known by the same node id through the whole network. However, in a distributed environment like Ethereum its nodes know each other by public id + keccak hash algorithm which allows to create many nodes ids as he wants. To prevent from the attack one possible solution is to add ip address to node id causing a cost for creating new node id.

## **Eclipse attack with time sorting**

Ethereum nodes stores information about other nodes in two databases one is for long term, stored in disk with the purpose of reboot and the other is short term where similar to k-buckets of Kademlia. Apart from this Ethereum has limited the number of outgoing connections. If can attacker identify the timing of this reboot process, its possible to occupy the outgoing connections. However, Ethereum implemented an update by allowing nodes to limit number of incoming connections.

## **Security breaches related to Smart Contracts**

Security related Smart contracts are so critical in general security of the application because byte-codes that are written in the smart contract are available to public, anyone can do research to the potential explosion points at the same time it's also so hard to capture errors in the code before releasing. In voting mechanism, it's so important that until the voting ends, voting's should not be disclosed. Commitment schemes with separate parts are good practice to prevent. Example: voting values can be hashed in initial phase and then sent the value in second phase [16].

## **Offline voters**

Participants who are going to involve in voting scheme. They need to be online. One possible way of is putting an economic incentive for users to be online during voting scheme. Like earning or loosing additional kleos with correlating to number of participation or misses that they are involved.

# **SCALABILITY ANALYSIS OF ARCHITECTURE**

## **Layer 1**

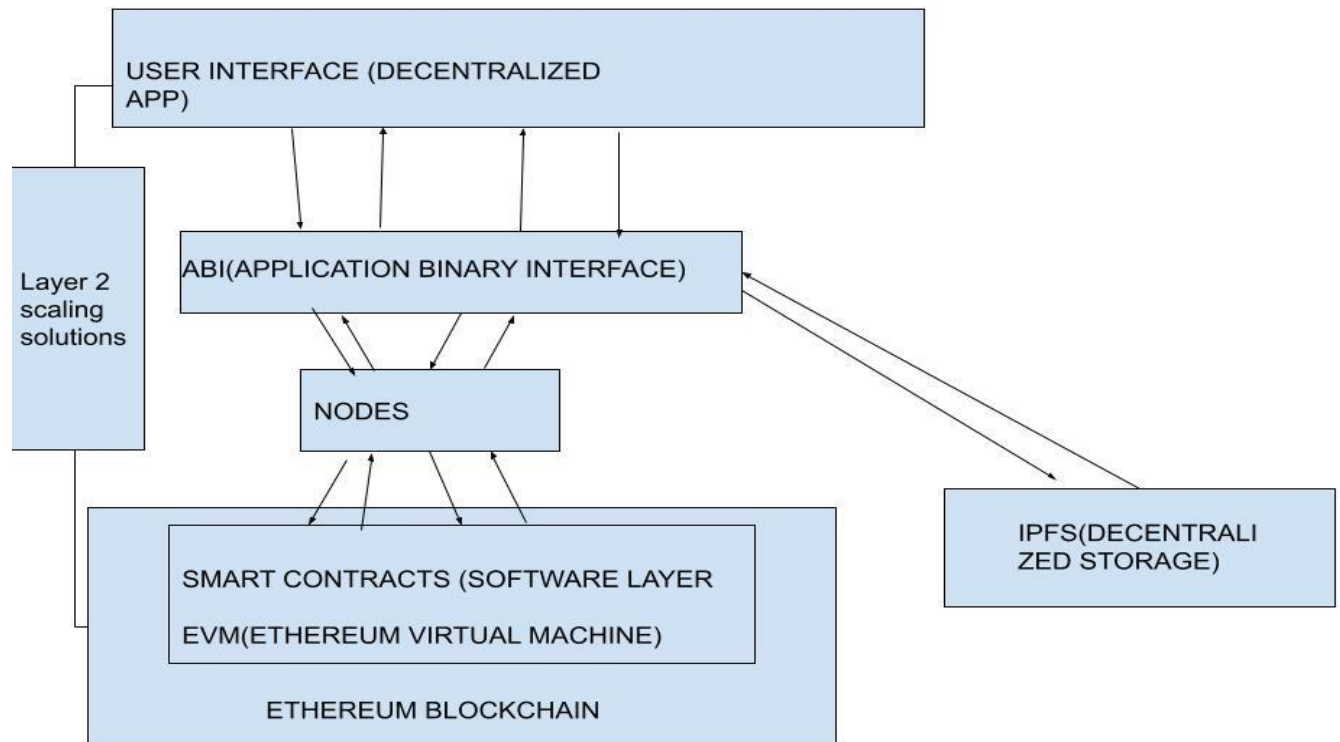
Ethereum provides a solid core blockchain infrastructure for recording necessary transactions in decentralized fashion. It's no secret that Ethereum has scalability problems as number of users grow in Ethereum, it's more likely to reach the capacity of blockchain causing a increase in the cost of transaction and decrease in the throughput. With the current condition of ETHEREUM its clear that it's impossible any decentralized app built top on Ethereum that is ready for adoption to masses. As described in blockchain platform Chapter, Ethereum uses POW mechanism that to achieve consensus on transaction history and able to process around 13 tx/s which is far low from any industry requirement. One way to achieve much higher transaction rate is to increase the frequency of the blocks or just by increasing the size of the blocks. Size of the block refers as GAS in the Ethereum. Degree of how much computational resource have been used in to execute the transactions, transactions in Ethereum are operations in the smart contracts. Also can be used as a incentive mechanism for miners to pick up your transactions instead of others. During network congestion times Gas fees reach incredible high level where making Ethereum network not worth to use. Making block size larger sounds a good idea but it comes with consequences like eliminating small players and handling the rules over from small players to big players due to increase in the cost of required

hardware devices to validate the network transactions simply Centralization, even with the current design centralization occurs in mining pools especially in bitcoin which requires specific hardware called ASICS. In the context of decreasing frequency times of block with the current POW mechanism that ethereum uses will let to miners to produce block simultaneously letting to a potential hard fork in the canonical next block on the blockchain as well as creating opportunity for various attack vectors. Even these factors are enough to understand its impossible to achieve a scalable network with POW. Apart from scalability reasons, it's not a secret that POW mechanism is not an energy efficient friendly mechanism currently miners consumes 72.5 TWh[17]equivalent to middle size country. If we assume that in next years as the community become larger, it will have the potential to consume larger energy. Ethereum has plans to shift another consensus algorithm called Proof of stake. Traditional proof of stake has several advantages over proof of work like energy efficiency, lower barriers entries. POS similar to POW have some sort of centralization in validation process. Validation process unlike pow doesn't required any work but amount of contribution in the process depends on the number of tokens that you hold unlike other cryptocurrencies Ethereum will have much lower barrier. Anyone that has more than 32.5 Ethereum can enter the consortium of validators. Main benefit of POS mechanism is the ability to tackle scalability problems that faced with POW. If our app going to reach a large community its crucial that Ethereum's switch from pow to pos should be smooth. Traditional POS eventually lead to centralisation, validators influence over the network will increase day by day. This is where sharding comes into play. In order to scale more transactions, and decrease the network congestion, one way is to splitting into vertically and the other way is splitting horizontally which is called sharding. Sharding will allow light nodes to join the network, and thus keeping the system decentralized and scalable [18].

## Layer 2

Layer 2 is term used for scalable solutions while benefiting from the secure and decentralized properties of the mainnet. Adding layer 2 solution in top of layer 1 eliminates huge amount of network load from layer 1. Mainnet should only consider to process deposits and withdraw. Checking the layer 2 transactions if it's is obeying the rules [19]

**Typical DAPP structure with a layer 2 solution can be seen below. Figure 4 [20]**



Using rollups with Ethereum 2.0 can increase transaction rate up to 100x more than the ETH 1.0. As a rollup our choice will be optimistic rollup, it will provide limited throughput compared to other rollups such as a zero-knowledge rollups, but its idea is simple, based on incentive mechanism unlike zero-knowledge-proof where its idea relays on mathematical solutions.

## Gas efficient smart contracts

Gas fees are proportional to complexity and storage size of the smart contracts and increase in the gas price, will lead higher fees, if we want to reach a large number of people, we need to keep the costs of the transactions minimal. Each functions in the smart contract use computer resource so requires gas. To make gas prices lower, we need to decrease the number of write functions in the contract.

## Future works, ethical issues and conclusion.

The progress in the computer science and networking will direct the app's future. In long term Serious advancements in computer vision and artificial intelligence may let our app's voting scheme useless. In the future it might be possible to have a reputation system with no direct involvements of the humans. Ai integrated Cameras can capture pre-described scenarios in the street and with a similar to Qr code mechanism can connect the faces with associated blockchain addresses. From my perspective

allowing governments to use blockchain technology for a control mechanism or social ranking will be a disaster for people who are favours this technology purely for privacy, protection and freedom. APP will have serious issues in the first months of the release. If app doesn't have enough voters, for malicious users it will be more easy to manipulate voting scheme. The required cost for entry will cause so much headache on people whether or not the system will promise as they work. The earned kleos should have such a great value that people need to take the risk and trust the system. To increase the values of kleos and incentivize people to enter the system a joint effort should be put with the government. On our app we didn't force users to register in the system with their real identity, so users will be pseudonymous. Another privacy issue is there are no restriction for people what they can post, in order to gain kleos ,it's possible to violate other people's privacy for your own reputation. This is not ethical. To sum up the things if we want to build a fully functional reputation system somehow we need to disclose the relations between parties into public otherwise it becomes the both parties own subjective view of the context. In our design we tried to keep minimal of disclosing information between both parties only in conditions where parties have different views about the situation. Even in worst case scenario blockchain based reputation systems will be much more trustable then the traditional ones, especially the ones that owns by the company.

## References

0-<https://bitcoin.org/bitcoin.pdf> .

1-<https://voxeu.org/article/blockchain-catalyst-change>

**2-TrustDavis: a non-exploitable online reputation system**

3- [Y.WANG](#) and jullita vasileta Trust and reputation model in peer-to-peer networks, Conference: Peer-to-Peer Computing, 2003. (P2P 2003).

4-M. Gupta, P. Judge and M. Ammar. (n.d.). A Reputation System for Peer-to-Peer Network

5-The EigenTrust Algorithm for Reputation Management in P2P Networks

6-A Trustless Privacy-Preserving Reputation System

7-<https://www.drep.org/>

8-<https://blog.bigchaindb.com/an-argument-against-smart-contracts-57f4f2a05b3d>

**9-Thomas Schelling, *The Strategy of Conflict* (1960)**

10-vitalik buterin, <https://blog.ethereum.org/2014/03/28/schellingcoin-a-minimal-trust-universal-data-feed/>

11-<https://www.shutterstock.com/image-vector/dapp-vs-app-decentralized-applications-explained-1500606653>

12-<https://ethereum.org/en/developers/docs/evm/>

13-<https://www.quicknode.com/guides/solidity/what-is-an-abi>

14-<https://ethereum.org/en/developers/docs/accounts/>

15-<https://ethereum.org/en/whitepaper/#mining-centralization>

- 16-<https://consensys.github.io/smart-contract-best-practices/recommendations/#external-calls>.
- 17-<https://ethereum.org/en/developers/docs/consensus-mechanisms/pow/>
- 18-<https://blog.ethereum.org/2020/03/27/sharding-consensus/>
- 19-<https://academy.moralis.io/blog/comparing-layer-2-ethereum-scaling-solutions>
- 20- <https://thenewstack.io/web3-architecture-and-how-it-compares-to-traditional-web-apps/>