Voting Mechanism Selection for Decentralized Autonomous Organizations

Winsome Kurniawan

Department of Information and Computing Science, Science Faculty, Utrecht University

Supervision of:

Slinger Jansen & Jan Martijn van der Werf

Abstract

With the invention of blockchain and smart contracts a new form of decentralized governance called decentralized autonomous organization emerged. By using voting mechanisms, these organizations are able to formalize the rules in how members are able to vote. However, there are many ways in how members are able to express their preferences. This research aims to improve the DAO decision context by designing a decision model of voting mechanisms in order to help organizations select a voting mechanism. By conducting a literature study and a case study, a knowledge basis and evaluation method for the decision model is established. The decision model can be categorized in how the voting weight is allocated, which are one-vote-per-participant or token-weighted voting. The one-vote-per-participant systems are further divided in the ballot types, while the token-weighted voting are divided by asking specific questions related to the use case of the voting mechanism.

Keywords: Decentralized Autonomous Organization (DAO), Voting mechanism

1 Introduction

The traditional organization has a hierarchical structure where the top of the organization has the most autonomy and the employees at the bottom the least (Sims, 2019). In this structure the top will make decisions for the organizations while the bottom carry out tasks (Diallo et al., 2018). As a result, quicker decision-making and clear designation of tasks can be made, which results in high production efficiency (Hsieh, 2018). However, due to the lack of autonomy at the bottom layer, the company does not make use of collective wisdom (Bressen, 2007). Employees possibly have more (combined) knowledge on a topic than the top decision-maker. With collective wisdom, employees are able to contribute which potentially result in better decision-making. Additionally, a centralized decision may not have the best interest of shareholders in mind (Morrison et al., 2020), in fact the decision-maker is able to choose for their own interest which allows corruption to play a role in the process (Shermin, 2017). Consequently shareholders are responsible for a decision that was not made in their interest (Beck et al., 2018).

With the invention of blockchain and smart contracts a new form of decentralized governance called decentralized autonomous organization (DAO) emerged (Murray et al., 2019). "DAOs are virtual, decentralized entities such as corporations and institutions running entirely autonomously and decentralized on a blockchain. Essential operations are automated agreeing to rules and principles defined in code, executed by smart contracts and finally, creating value for customers" (Kondova & Barba, 2019; Kypriotaki et al., 2015). In contrast to the traditional organization, a DAO has a horizontal structure where each member has an equal autonomy (Lumineau et al., 2020). Due to the decentralized nature, collective wisdom can be used. By using voting mechanisms, communities are able to formalize the rules in how members are able to vote. These votes can be used to express the interest of the shareholders. Therefore no central authority is needed, resulting in no more corruption as long as a single entity does not have the majority of the voting power.

With the emerging technology of DAOs, a growing stream of literature follows. Particularly, a unified technical and analysis framework for DAOs is already introduced by (Wang et al., 2019). However the framework did not cover the voting mechanisms of DAOs (Beck et al., 2018; Rossi et al., 2019). There are many ways in how members are able to express their preferences, and thus voting mechanisms. The aim of the thesis is to introduce a decision model to select a voting mechanism for DAOs. By introducing a decision model, two main contributions will be given. First an overview of the different voting mechanisms will be given. The second is by distinguishing the characteristics of the voting mechanisms. These contributions enable organizations who want to establish a DAO to decide on a voting mechanism for their DAO.

The thesis is structured as follows. In the next section the main research question and sub-questions are described. Further, a short explanation is given about the used methods to acquire information. Section 3 discusses blockchain, smart contracts, relevant concepts, the decision process and its voting challenges in a DAO. Section 4 establishes different decision criteria that are relevant to voting mechanisms. Section 5 introduces voting mechanisms which could be applied to DAOs. In section 6 a decision model is introduced where the voting mechanisms are distinguished from each other. Next, a case study is designed in section 7. In the next section, a discussion is presented where the limitations of this thesis are discussed. Lastly, a conclusion to the thesis is provided.

2 Research Method

To help organizations choose a voting mechanism, it is decided to create a decision model. This model will be produced by using the research approach of Design Science. Design Science research is a research approach in which a designer answers questions via the creation of artifacts, and thereby contributing to the body of knowledge (Hevner et al., 2004). By taking this approach, knowledge and understanding of a design problem and solution are acquired in building the artifact.

The Design Science framework of Hevner (2007) is applied in the context of this research in Figure 1. The first is the Relevance cycle where the application domain consists of people, organizational systems, and technical systems that work towards a goal. Problems of the application domain are extracted to design the artifact. This cycle is applied by conducting a case study to produce problems and requirements relevant to the environment. Second is the Rigor cycle which draws from the existing knowledge base to form the foundation of the artifact. By conducting a literature study, a strive is made towards contributing to the knowledge base. The last cycle is the Design cycle which iterates between the construction of the artifact, evaluation, and feedback to further refine the design. The same case study is used to assess and refine the development of the artifact.

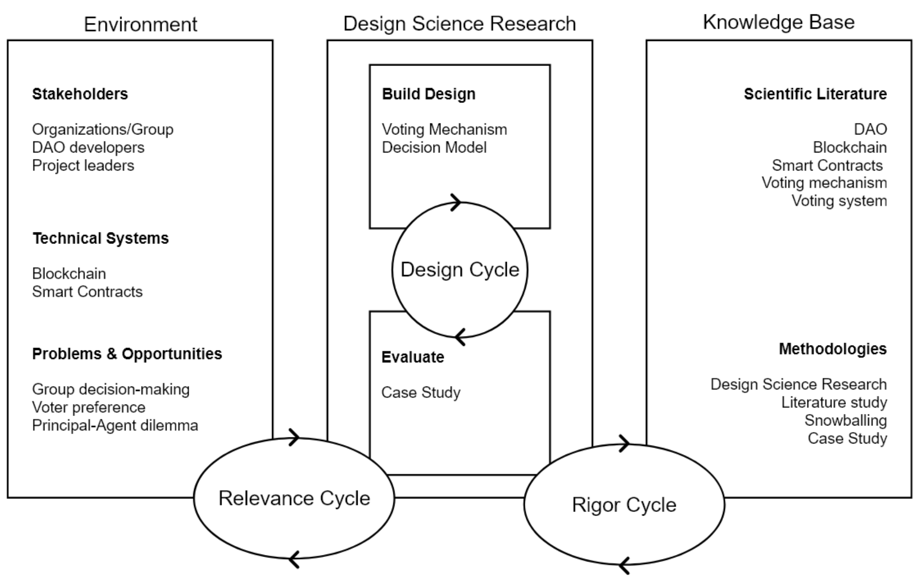

Figure 1: Design Science Research Framework applied to this research, adapted from Hevner et al. (2004)

The methodology of Hevner et al. (2004) consists of six different steps. The first step defines the research problem and justifies the value of a solution.

Second step is to determine the research objective and the scope of the research. In the third step the artifact is created. The next step demonstrates the use of the artifact to solve an instance of the problem. In the evaluation step, it is observed how well the artifact supports the solution to the problem. Last step communicates the artifact to relevant audiences. These steps are applied in Figure 2.

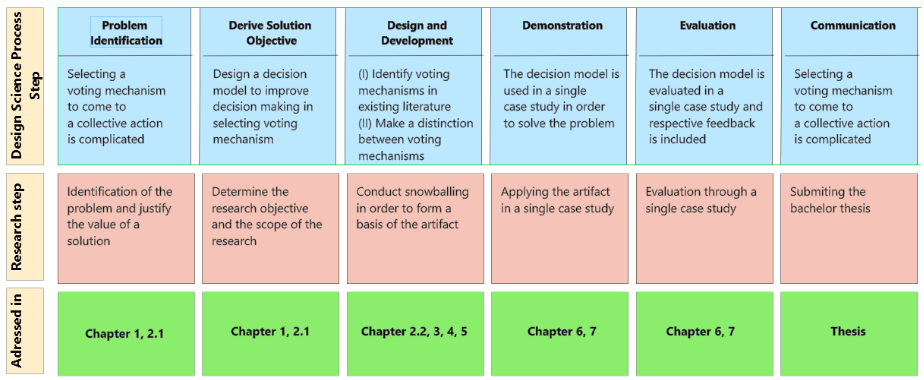

Figure 2: Design Science Process steps

2.1 Research Questions

The Design Science approach extracts the problems and requirements of the application domain to structure the research objective. Some research questions are proposed to structure the creation of the artifact and thus addressing the research objective.

Research question: To what extent can a decision model be developed that supports DAOs in selecting voting mechanisms for making decisions?

The answer to the main question will be the decision model constructed in this thesis. To reduce complexity of answering the question, the main research question is divided into multiple sub-questions.

Sub-question 1: How do virtual organizations make decisions?
The first sub-question is stated to get an understanding of the general process of decision-making in a virtual organization. Using the literature study, relevant concepts of the process will be explained.

Sub-question 2: What are the voting challenges in virtual organizations that require advanced voting mechanisms?

The previous sub-question describes the decision-making process when nothing goes wrong. However, problems always occur during decision-making. Therefore, the possibilities of what can go wrong should not be neglected. With the help of literature study, the objective of the question is to state several voting challenges that could occur in virtual organizations. It is necessary to address these challenges as the voting mechanisms attempt to reduce or mitigate the challenges.

Sub-question 3: What decision criteria need to be considered when choosing a voting mechanism?

Subsequently a list of decision criteria is established using the literature study. These decision criteria are necessary for when a voting mechanism needs to be considered. These criteria range from who has the right to vote, to how members are encouraged to participate. Here, several questions are found which are then turned into topics. Following, for each topic it is attempted to answer the found questions. These topics with the respective answers are constructed into a single framework.

Sub-question 4: What voting mechanisms are available through DAO?

Further, another list with voting mechanisms is constructed to provide an overview of the different voting mechanisms that are available in DAOs. These voting mechanisms are mainly categorized by the way the voting weights are allocated. Each voting mechanism contains how the voting process is conducted.

Sub-question 5: To what extent is the decision model useful in practice?

Besides designing and constructing a decision model, it should be evaluated to prove the validity of the decision model. This is done by conducting a single case study where a virtual organization applies the decision model to choose its own voting mechanism.

2.2 Literature study

Snowballing method is a search method that can be used as a first step to conduct literature study (Jalali & Wohlin, 2012; Wohlin, 2014). Snowballing refers to using the reference list of a paper or the citations to the paper to identify additional papers. As Jalali and Wohlin (2012) mentioned, there are no significant quality differences between Systematic Literature Review (SLR) and the snowballing method. However, the snowballing method generally has less initial papers than the SLR, which indicates less irrelevant papers are included and thus less time spent to identify relevant papers. Another reason is that the systematic literature review may not be able to capture many different wordings, only exact terms that are formulated in the search string are included. For these reasons it is decided to use the snowballing method as a search method. Thereby the guidelines of Wohlin (2014) for conducting the snowballing method are followed. First, a start set of papers is identified to conduct the snowballing method. This is done by putting in relevant keywords in Google Scholar. It is necessary to decide on the inclusion and exclusion criteria before analyzing papers. The papers should relate to at least one of the sub-questions in order to be deemed relevant. Other inclusion criteria are papers that are available and written in English. If the papers met the criteria, the topic area, titles, abstract, introduction and conclusions are reviewed. If the paper still is deemed relevant, then the paper will be included in the snowballing process.

Once the start set is decided, the backward and forward snowballing is conducted. Backward snowballing uses the reference list of the papers in the starting set to identify additional relevant papers. The papers in the reference list are analyzed as the process to identify the start set. Included papers are then put at the back of the snowballing list. After performing the backward snowballing, the paper is then used for forward snowballing. Forward snowballing finds new paper that cited the initial paper. These papers were found by the Google Scholar 'Cited by' function. Similar to the start set and backward snowballing, the same process is applied to identify relevant papers. These papers are also included at the back of the snowballing list. The last set of papers are then categorized based on the sub-questions.

2.3 Case Study

The definition of case study is "an empirical inquiry that investigates a contemporary phenomenon within its real-life context; when the boundaries between phenomenon and context are not clearly evident; and in which multiple sources of evidence are used" (Yin, 1994). A case-study approach was conducted, because the problem statement contains why and how a DAO chooses its voting mechanism, occurs in contemporary context, and variables and individual behavior, such as preference of a voting mechanism cannot be manipulated.

It is chosen to conduct a single, holistic, descriptive case study, as the goal is to apply and evaluate the acquired knowledge from the literature study and the constructed decision model to a case study. This research follows the case study research process of Runeson and Höst (2009). The following steps are:

1. Case study design: objectives are defined and the case study is planned.
2. Preparation for data collection: procedures and protocols for data collection are defined.
3. Collecting evidence: execution with data collection on the studied case.
4. Analysis of collected data.
5. Reporting.

Other than a research process, it is necessary to include a case study protocol. By following a protocol, a clear outline of the procedures can be made. Consequently, it allows for greater repeatability of the method and thus enhances validity and generalizability of the results (Sekaran & Bougie, 2016). The case study protocol of Pervan and Maimbo (2005) is followed which include the following steps:

1. Preamble: Contains information about the purpose of the protocol, guidelines for data and document storage, publication.

2. General: Provides a brief overview of the research project and the case research method

3. Procedures: Detailed description of the procedures for conducting each case, including down-to-earth details on contacts and timing.

4. Research Instrument: The selected research instruments that are used to collect data, which are either qualitative (interviews) or quantitative (surveys).

5. Data Analysis Guidelines: The strategies and techniques for analyzing the data.

6. Appendix: Sample of the participation request letter, personally sent to the case study participants.

The case study project has to meet the following requirements:

- The project must be considered as a group/organization;

- The project must be in early stage of DAO development;

- The project is in search of selecting a voting mechanism.

These steps will be implemented and more elaborated upon in chapter 7

3 Background

The objective of the literature study is to introduce key concepts of DAOs and to provide a base for the decision model and case study. These concepts are significant as it enables the creation of DAOs. Additionally, this chapter should provide an answer to sub-questions 1 and 2. The result of this is an understanding of the key concepts of a DAO, how members of a DAO are able to make collective decisions and the challenges that comes with these decisions.

3.1 Blockchain

One of the important concepts related to a Decentralized Autonomous Organization (DAO) is the blockchain. Blockchain logs and synchronizes data locally across a peer-to-peer network on a growing distributed (ledger) list of transactions. Large volumes of new transactions are bundled in blocks, which include a hash function that links the block to the previous block in the chronological sequence (Yermack, 2017). The chain of blocks forms the history of transactions since the inception of the blockchain, hence the name blockchain (Peters & Panayi, 2016). A hash function transforms data into a hashed code, which cannot be inverted to recover the original input (Yermack, 2017). If an adversary tries to change a transaction in the block, the hash function of the transaction changes along with it. This can be tracked by other nodes in the network, which consequently can decide to not update the tampered blockchain. In summary, the hash function result in the immutability of the transactions in the blockchain.

In order to include the correct block, the nodes in the peer-to-peer network need to reach consensus on which blocks to include. With consensus mechanisms, nodes are able to confirm the correct blockchain without the need to trust other nodes in the network or any trusted third party (Atzori, 2015). Consensus mechanisms incentivize nodes to correctly validate new blocks, while discouraging validating tampered blocks (Murray et al., 2019). The most common consensus mechanisms are proof-of-work and proof-of-stake (Beck et al., 2018; Tschorsch & Scheuermann, 2016). In proof-of-work, nodes that validate blocks are called miners, while validators in proof-of-stake are called stakers. Proof-of-work requires miners to solve computationally expensive cryptographic puzzles. The first miner to solve this puzzle has the ability to validate the next block. In proof-of-stake nodes are required to stake their tokens in order to participate in validating the blocks. The larger the stake, the higher the probabilities of being chosen to validate the next block. The stake can be decentivized when the staker validates an incorrect block. The task of these stakers is to reach consensus on which block is correct and ensure the prior history of the blockchain. This results in the blockchain reaching an immutable state where no prior transaction can be altered or deleted (El Faqir et al., 2020).

There exist different types of blockchain infrastructure, see Table 1, which categorizes whether authorization is required to be able to validate transactions of the network (permissioned vs permissionless), and whether reading and sub-

Table 1: Categorization of blockchain types (adapted from Peters and Panayi (2016) and visualized by Beck et al. (2018))

Access to Transactions	**Access to Transaction Validation**	
	Permissioned	Permissionless
Public	All nodes can read and submit transactions. Only authorized nodes can validate transactions.	All nodes can read, submit, and validate transactions.
Private	Only authorized nodes can read, submit, and validate transactions.	Not applicable.

mitting blockchain transactions are public or private (Peters & Panayi, 2016).

The first categorization distinguishes the right to participate in a blockchain network. The public blockchain allows anyone to participate and read all blockchain data and propose new transactions (Beck et al., 2018). On the other hand, private blockchain limits these rights to only participants within an organization or group of organizations.

The second categorization differentiate between permission to validate and thus maintain blockchain transactions. In permissionless blockchain all nodes have access to validate transactions of the blockchain, while in permissioned blockchain only nodes that are preregistered have the right to validate (Rikken et al., 2019). Generally, the owners of the permissioned blockchain are able to provide these rights to nodes. Here, two types of permissioned blockchains can be made: fully private which is owned by a single organization or consortium blockchain where the control is divided between a group of organizations (Li & Zhou, 2021).

Public blockchain often experience scalability issues. As the number of transactions increases, a higher number of validators are necessary and combining this with the low efficiency in propagating transactions and large node counts, the public blockchain is generally less efficient than its consortium and fully private counterparts (Zheng et al., 2017). Consortium and fully private blockchains are able to make quick decisions due to the low number of nodes that need to update (Rikken et al., 2019). Despite these potential benefits, permissioned blockchains receive many criticisms (Atzori, 2015), because a small number of nodes in the network can easily collaborate and alter the rules, or even revert transactions (Peters & Panayi, 2016).

In reality, most permissionless blockchains have public access (Peters & Panayi, 2016), which means that private permissionless blockchains do not exist as it requires read and submit rights to validate a transaction. As a result, there are currently no applications with permissionless private blockchain.

3.2 Smart contracts

The other important concept, which is enabled by blockchain, is smart contracts. Smart contracts are program codes that can self-execute once a certain condition has been met (Murray et al., 2019). Same as a normal written contract, the terms and penalties of the negotiations are defined in the contract, and agreed by the parties. However, smart contracts are able to monitor and self-execute when the pre-defined condition has been met, hence it is called smart (Atzori, 2015; Mendling et al., 2018; Morrison et al., 2020). Due to the self-enforcing feature of the smart contract, the program code does not need human involvement and thus no third party to monitor and execute the contract. In addition, no legal action is required when a conflict arises as the condition in the contract is pre-defined without ambiguity of words, which consequently result in the parties not having to trust each other (Wright & De Filippi, 2015). Therefore, smart contracts can be seen as more complicated transactions which then can be captured in the blockchain. This allows the smart contract to receive the same immutability and transparency features as the blockchain. Smart contracts can feature loops and have internal state (Peters & Panayi, 2016), which can be chained together to define a complex condition. These conditions can define the rules in a DAO, or in other words how actions can be performed within the organization (Sims, 2019; Virovets & Obushnyi, 2021; Wang et al., 2019). For example, a proposal is accepted if the proposal is made by a member of the community, and approved by everyone in the community.

3.3 Decentralized Autonomous Organizations

The term Decentralized Autonomous Organization (DAO) can be best explained by breaking the term down. In a traditional organization, the corporate governance consists of a hierarchical structure, where the employees work for a boss. DAOs are another form of corporate governance enabled by blockchain and smart contracts. Because DAOs are applications built on a blockchain, the need for a board of directors is eliminated (Virovets & Obushnyi, 2021). DAOs inherit the **decentralized**, immutable, and transparent features meaning that no single entity owns the resources and control of the organization, but it is shared by all participants of the organization. However, managers are there to choose a goal, while giving orders to steer the employees towards that same goal (Diallo et al., 2018). Because the organization is decentralized, participants are anonymous and thus probably will not know each other. Due to this fact, a lack of trust occurs between the participants (Morrison et al., 2020). A possibility may exist where a malicious participant attempts to reduce the value of the organization. By implementing mechanisms to incentive good behavior, DAOs tries to align the interests of participants to work towards a common goal (Hüllmann, 2018).

DAOs inherit the **autonomous** feature from smart contracts, which are used to define the rules of the organization. These rules include but are not limited to: who are able to propose a goal that the organization can work to, and voting procedures, such as which participants are able to approve or disapprove this same proposal (Virovets & Obushnyi, 2021), how certain transactions are executed, how new coins are created, at what speed the chain is updated (Hacker, 2019), changes to the protocol, business model and governance procedure itself (Dursun & Üstündağ, 2021). Consequently, smart contracts allow participants to vote on various items through the blockchain in a decentralized, democratic manner, which means that the decision-making is distributed throughout the organization (Wright & De Filippi, 2015). Ironically, DAOs are not entirely autonomous as it requires human participation to make decisions. Most commonly in public blockchains, DAOs issue native tokens to give voting rights to the shareholders (Gudkov, 2017). These tokens can be acquired during the creation phase or later buying over the token from a member who wants to sell it. By acquiring tokens, it grants the owner voting rights for the respective DAO. Tokens create an economic mechanism for shareholders to make well-thought decisions for the **organization** in the hope to create value for customers in the service a DAO provides, which in result increases the value of the token, and thus rewards the shareholder.

The use of DAOs can be diverse, ranging from a crowdfunding system where capital is pooled together. Participants of these systems are able to vote on the projects to release the required fund for the allocated project. Furthermore, DAOs can enable the cooperation of a group of developers to create and promote a product or service to bring on the market (Virovets & Obushnyi, 2021). As the concept is still in its infancy, it is hard to determine if a project is a DAO. It could be argued that any project with a decentralized organizational structure is a DAO. Usually, the projects identify themselves as a DAO.

3.4 Decision process

This section describes a general process of how decisions can be made in a group. Fama and Jensen (1983) gives four phases in the decision process. The first phase is Initiation where proposals are generated for resource utilization and structuring of smart contracts. Second is Ratification where participants are able to express their opinion regarding the proposal, different methods are described hereafter. The third phase is Implementation which is executed automatically by smart contracts. As mentioned before, smart contracts self-execute when the predefined condition in the proposal is fulfilled. Last phase is Monitoring where participants evaluate the performance of the implemented proposal.

There are two main processes to reach a group decision when more than one person has the voting power (Bressen, 2007; Karjalainen, 2020). Consensus, where all members must consent before the decision is accepted, and voting where the decision is accepted when a majority of the participants agrees. A reason to choose for voting instead of consensus is that the decision-making process is faster, and is more controllable when the scale of the community is large. The winner can be chosen when it receives a plurality or majority of the votes. Plurality means that the alternative with the most votes is selected, but this might not be the majority. More is explained in section 5. The majority spectrum ranges from 51%, to a super-majority which is typically 75-90%. It is necessary to formalize this percentage as it determines the autonomous execution of smart contracts (Zwitter & Hazenberg, 2020). The main drawback of a normal majority is that the group is split into two groups when the majority is close to 50% (Yermack, 2017) and thus increases the chance for a hard fork to occur as it indicates the community cannot reach a decision. Hard fork is a split of an organization into two branches, while each follows a different protocol from the moment it separates. The two have the same history of blockchain but it slowly drifts apart as changes are made to each one. By choosing for super-majority, the chance of a hard fork to occur is reduced.

On the other hand, trying to reach consensus does not allow a minority to occur. Instead, each idea and concerns are taken into account (Butler & Rothstein, 2007; Karjalainen, 2020). This results in the participants to be more involved and stand behind the decision of the group. In consensus, an issue is first introduced as well as providing relevant information. By doing this, it ensures that everybody has the same understanding. Subsequently, participants are able to come up with ideas and pros and cons are considered. Here emphasis is required that the community should encourage the members to speak out any hopes or concerns as this allows proposals to be more acceptable for the group. Most often, a compromise is formed where combinations of agreed ideas are brought together. An initial proposal can be formed of the combined ideas. A formal process can be conducted where each member agrees or disagrees with the proposal. There exist different ways to describe whether one is in agreement or not. Agreement expresses that the member supports the proposal and has no concerns. When someone has a Reservation one may have some minor concerns before agreeing with the proposal. Stand aside expresses that a member let a proposal be implemented despite unresolved concerns. Block disagrees with the proposal and does not want the proposal to be implemented. And the last is Abstain which does not express a preference, instead the member chooses to not participate in the decision. Reasons for this may be because the member is uninformed or not ready to participate.

When no consensus can be reached, the community can decide to make amendments to the proposal or create a new proposal. Generally consensus is preferred (Gasser et al., 2015), but the community makes use of Deliberative democracy after no consensus can be reached. In Deliberative democracy the community will fallback to voting when consensus can not be reached. However, switching from voting processes may change the behavior of the participants as it can lead to greater disagreement (Gasser et al., 2015). Hence, this method is only reliable when full acceptance is not necessary. Other methods to come to a decision is benevolent dictatorship where the community is centralized at first by appointing a leader who makes decisions for the community and slowly try to make the community more decentralized (Beck et al., 2018). Other form of benevolent dictatorship is that the leader makes the final decision only when the community is in disagreement (Karjalainen, 2020).

Blockchain voting opens the possibility to conduct decision-making on-chain or off-chain governance (Fischer & Valiente, 2021; Rikken et al., 2019). On-chain governance uses the blockchain to manage and implement changes, such as rules, proposing and voting. On the other hand, off-chain governance uses more traditional processes which are not captured into the blockchain. Examples of these actions are discussions and voting in meetings, forums or face-to-face. These off-chain methods allow for centralization of the organization where dominant actors exclude users that lack technical knowledge or financial power to make a significant impact to the organization (Dursun & Üstündağ, 2021). Due to the usage of blockchain, on-chain governance is transparent and formalizes such actions which makes it less centralized and involve mainstream users. For the moment, most DAOs use a combination of off-chain methods to collaborate and cooperate while using on-chain methods to decide and implement (Li & Zhou, 2021).

3.5 Voting challenges

This section discusses potential challenges that could occur during voting.

Unfortunately, there will always be members who disagree with each other. For this reason a protocol should determine how disagreements can be solved. Aragon plans to address this by implementing a digital jurisdiction (Ziolkowski et al., 2020). A plaintiff can open a case and submit relevant information regarding the case to a random, wider array of judges of the Aragon network, on the condition that it posts a bond to disincentive irrelevant submissions. When the plaintiff does not agree with the proposed solution, it can go to the top judges given that it places a higher bond each time.

DAOhaus has a mechanism called the ”rage quit” mechanism which makes it possible for a participant to exit their vote along with the resources if it does not agree with the result of the voting (Faqir-Rhazoui et al., 2021). There is a period where members are able to use this mechanism, and if more than 30% use this, the vote will be reverted.

Participants have also the option to exit a DAO. There are two ways to do this, namely a weak exit or a strong exit (Sims, 2019). Just as traditional organizations, participants are able to (weak) exit their organizations by selling their shares of the company, or in the case of DAOs selling their tokens. This however depends on the organizations as participants may not want to enter a private organization, especially if the participant will be a minority shareholder.

Alternatively, a hard fork is a form of a strong exit. As mentioned before, hard fork allows the organization to split in two. Hard forks are often considered as undesirable, because it can lead to significant problems, confusion, errors, economic loss, and bugs (Kaal, 2021; Rikken et al., 2019). In another perspective, hard forks allows organizations to foster experimentation and competition (Webb, 2018). Therefore forking should be seen as a feature of a DAO rather than a bug (Sims, 2019).

Another problem is the scalability of the DAO (Rossi et al., 2019). With a public blockchain everyone is able to be a member which makes the community size uncontrollable. For this reason, the larger the group the more difficult the coordination problems are. Consequently, the minimum approval of a proposal increases when it depends on the total number of users in the community (Faqir-Rhazoui et al., 2021). With a larger group comes an increase of the number of transactions. This brings many problems with it such as an overflow of transactions which result in the increase of transaction fee, increased processing time (Peters & Panayi, 2016).

The Sybil attack occurs due to the pseudonymity of the blockchain, which means that users are only known by cryptographic keys and thus cannot be recognized by name (Karjalainen, 2020). This allows the potential for users to make multiple accounts. In democracies where each account receives one vote, an attacker is able to create multiple accounts which result in the user having more voting power than it is intended. Multiple methods exist to mitigate this problem. It can be decided to make the ballot public, which means everyone is able to see what each user has voted for. This conversely brings other problems such as strategic voting which is discussed hereafter. Alternatively, the voting mechanism could include a fee when a user casts a vote, however this could increase the lack of participation in voting which is also discussed below. The last method discussed is to authenticate each voter while keeping the authentication mechanism separate from the ballot. This method is more feasible in private blockchains than public blockchains, regardless the cost to implement this is questionable.

A danger occurs when a user or a coalition decides to manipulate the voting outcomes instead of voting honestly. This problem is broadly addressed in voting systems. Some example to this is when the attacker introduces alternatives so votes for a preferred alternative are split among 2 or more similar alternatives (Burgman et al., 2014) or even bribe users to vote against their best interest. Alternatively as discussed above, if the votes of each user is public the voter is able to vote for the winner and receive the incentive that is included for winning the vote.

It is important that the members of a community participate in the decision-making as it makes the decision more legitimate (Kaal, 2021). Often participants are unresponsive which result in a lack of participation (Rikken et al., 2019). A proper incentive mechanism should be set up to mitigate this problem. So in contrast to putting a fee when casting a vote as mentioned before, members can be encouraged to participate by rewarding with additional tokens or an increase in reputation if they vote. On the other hand people also participate in political elections, because they truly care what happens in the country. This mechanism could have an opposite effect as participants could simply vote and collect the incentive while disregarding what the proposal actually is about (Sims, 2019). This results in the members being encouraged for making bad, and uninformed decisions. In relation to this, a new mechanism can be introduced to lock the tokens when the vote is set. This freezing period prevent members from selling their token for a set period, which in result may encourage them to make well-informed decisions that are good for the DAOs on the long-term as the price of the locked token will likely be affected when a bad decision is made (Kaidalov et al., 2017).

The last challenge is that people who have a high technical knowledge, such as developers will likely receive more influence in a DAO. While anyone has the ability to make proposals for updating the code, it requires a knowledge of programming to set up a smart contract (Dursun & Üstündağ, 2021). This could lead to an excessive technocracy of developers as only a few have the power to actually implement changes (Hacker, 2019; Hsieh et al., 2018; Rikken et al., 2019) and hold very technical discussion. On the other hand, one could argue that it is a good thing that the most capable makes the decisions and criticize that people without proper knowledge regarding the impact can vote as well.

3.6 Lessons from this Chapter

This chapter explained the concepts of blockchain, smart contracts, and decentralized autonomous organizations. Blockchain logs and synchronizes data locally across a peer-to-peer network on a growing distributed (ledger) list of transactions (Yermack, 2017). Smart contract is a type of transaction which consists of program codes that can self-execute once a certain condition has been met (Murray et al., 2019). These concepts are explained since they enable the creation of DAOs. As seen in Table 1, there exists multiple blockchain types. Different blockchain types are stated, because public blockchain may have different challenges than private blockchains. This relates to the main question as different types of blockchains and thus different types of DAOs are taken into account in the decision model. As a result, the distinction will return in the next chapter as a decision criteria.

The term DAO is discussed to formalize what a DAO is, it is namely a virtual organization which inherits the decentralized feature from blockchains and autonomous feature from smart contracts. However a virtual decentralized organization come with challenges since there are multiple entities that have to reach a decision. The section 'Decision process' describes how a decision can be reached in a group. Consensus and voting are the methods to reach a decision in a group, which also forms the answer on sub-question 1. The last section of this chapter addresses some of the challenges that could occur during voting and additionally answers sub-question 2. The challenges are but not limited to dispute resolution, scalability issues, Sybil attack, strategic voting, lack of participation, and excessive technocracy.

In the next chapter some decision criteria are stated. This is done to differentiate between the different voting mechanisms which is essential for creating a decision model for voting mechanism selection. This chapter follows after explaining some key concepts of DAOs because the reader now understand the decision process and voting challenges. As a result, an understanding is created for decision criteria which state some issues that can influence the voting mechanism selected.

4 Decision criteria

This section raises questions to define which topics of the decision process need to be addressed, see Table 2.

A total of 16 criteria are identified based on 6 literature references, namely Beck et al. (2018), Gasser et al. (2015), Karjalainen (2020), Pelt et al. (2021), Rossi et al. (2019), and Ziolkowski et al. (2020). These references are explored based on the following keywords: "Blockchain governance, DAO, and Decentralized governance". Criteria were included when it could affect decision-making in DAOs. The framework 2 is later used to produce questions for the case study, give answer to sub-question 3 and to differentiate between the voting mechanisms provided in chapter 5. The list of the criteria is provided in order to get an overview of the different decisions in design DAO and voting mechanisms. The criteria list is divided into two topics, namely general, and voting mechanism. It should be noted that the completeness of the list is debatable as many criteria could be seen as subjective.

4.1 General

4.1.1 Network status

Network status describes if the organization requires permission to join. Private organizations require permission to join, and thus tend to have a lower scale of members compared to public organizations (Rikken et al., 2019). As a result, consensus could be used to make collective decisions as it is more feasible with a lower scale. However, voting is not limited to public organizations (W. Zhang et al., 2018). It can also be used for consortium organizations as these organizations coordinate between several other organizations.

4.1.2 On vs off-chain voting

As mentioned before, on-chain voting formalizes decision-making processes by capturing the proposals and voting into blockchains (Pelt et al., 2021). This in turn will make the process more transparent and more audible. Off-chain voting makes use of meetings, forums, or face-to-face interaction to make a collective decision. This in result will allow dominant actors to gain more influence and exclude other members.

4.1.3 Proposing rights

One of the most important aspects to decision-making is who is eligible to participate in the decision-making. This can be divided into the eligibility to propose and to vote.

One option is to make each member in the community eligible to propose so one would be able to offer its input. Though, proposals often require technical specifications. So one may argue to only limit the right to propose to developers to guarantee the quality of the proposals. However, this might turn the organization into a technocracy where the most capable developers have the most power within the organization (Karjalainen, 2020). A solution might be not to assign a role to a member, but to put a barrier to be eligible to propose, for example a minimum amount of tokens is needed. This may guarantee that the members have a significant stake in the DAO to discourage introducing bad proposals.

4.1.4 Democracy

Democracy is a governance system where people who are affected by collective decisions of a group are able to participate in the decision (Karjalainen, 2020; Shermin, 2017). There exist three methods which differ in how individuals are able to contribute to the decision.

Direct democracy is the first form of democracy in which people make decisions directly, and thus treat each voter equally (Shermin, 2017). Traditionally, people are able to vote for issues on referendums. However, this democracy is not feasible on a large scale due to high transaction costs of the voting process, and participants that are required to inform themselves on each issue which makes it a time-consuming decision.

Because of this fact, the most common democracy currently is representative democracy, where representatives are elected to make decisions on behalf of the participants. Contrary to direct democracy, only representatives are required to delve into the issues, and subsequently does not require a high transaction cost. Despite these benefits, participants are now required to inform themselves into representatives, while these often fail to serve the best interest of the people in collective decision making (B. Zhang & Zhou, 2017).

Last is liquid democracy, which is a hybrid of direct democracy and representative democracy. In this democracy, each participant can either vote directly or delegate their vote to another participant. These delegated votes could be further delegated to others. Due to this option, the quality of the collective decision increases as participants that are motivated to inform themselves can decide for themselves, while participants who do not want to, delegate their vote to others (B. Zhang & Zhou, 2017). However, the further delegations might result in high transaction costs.

4.2 Voting mechanism

4.2.1 Voting weight

There are two main ways to distribute the voting weight (Hacker, 2019). The first is one-vote-one-user where each user only has one vote. This voting weight is used in traditional voting system. Alternatively the voting weight can be dependent on the ownership of tokens, where the voting power is in proportion to the number of tokens (Reijers et al., 2018). This distribution allow users to accumulate more votes than others (Zwitter & Hazenberg, 2020), consequently resulting in a potential of plutocracy where the more wealthy members are able to gather more votes and thus have the most power within the community (Kaal, 2021; Reijers et al., 2018). The former method treats each user equally which prevent disproportionate influence, and collusion between validators and core developers (Hacker, 2019). Conversely, the risk of Sybil attacks occurs where a user is able to create multiple accounts to gain more power.

4.2.2 Number of winners

The number of winners is important when selecting a voting system, because a voting system might only be able to choose one winner. This criteria depends on the use case of the voting mechanism. For example, if one desires to implement a proposal, it is recommended to only consider one proposal at a time. This prevents members from being overwhelmed by the amount of proposals to choose between (Sims, 2019). In contrast, a use case to choose multiple-winners is in a crowdfunding system. These proposals in this system state the required amount to fund the proposal. So members in the crowdfunding systems can decide on multiple projects to fund as long as there is funding left.

4.2.3 Alternatives

Alternative describe the option the member is able to vote for. Often there is a list of proposals to consider, which in this case the proposals are the alternatives. There are two main ways to choose which proposal the community wants to implement. First, it can be decided that members need to vote on each proposal. This means that each proposal has only two alternatives, namely 'Yes' or 'No'. Voting with only two alternatives requires only a (super)majority to agree. The drawback to this is that the voter has to consider each proposal. Thus it is likely that the voter quorum of each proposal is not met.

Another method is to select one proposal from the list of proposals. This can be done with a voting system or mechanism. The winner is then considered solely with the two alternatives 'Yes' or 'No' This means that participants are more likely to vote as there is only one proposal to focus on and more likely to support as the proposal has been the most approved between the other proposals (Butler & Rothstein, 2007).

4.2.4 Vote accumulation

As stated before, in token-weighted voting users are able to accumulate more votes. However there are two main ways to gather this. The members are able to buy tokens to gain more power within the community, or by making valuable contributions in order to earn reputation which consequently means gaining more influence within the community.

4.2.5 Ballot type

The voting systems can be further categorized in the way the voters are able to express their preference. The single ballot allows each participant to vote only once. Following, multi ballot allow participants to vote on multiple alternatives. Then is ranked ballot where the voter ranks the alternatives in an order. Last is the score ballot where the voters have to rate each alternative on a scale.

4.2.6 Majority quorum

An issue requires a majority and participation quorum, in order for the decision to be seen as legitimate (Kaal, 2021). Without these quorums, preference of the members might not represent the decision and not support the made decision. Thus, it is necessary to define the threshold for what constitutes a majority (Karjalainen, 2020). The majority quorum could be set between 51% and 100%, but 100% support is less common as the community could use consensus instead. As stated before, the majority spectrum ranges from 51%, to a super-majority which is typically 75-90%. A reason to choose for supermajority is to prevent the chance to split the community in two as a 55:45 majority indicates that the community does not agree with each other (Yermack, 2017).

4.2.7 Majority subset

The majority quorum could be specified to which proportion it is applied to (Karjalainen, 2020). This difference can make a difference, as it depends whether abstentions and absence of voting counts towards the majority (Laruelle & Valenciano, 2011). The majority can be applied to the total number of the entire community or to the present voters. When it applies to the community, the members that did not vote are counted towards disapproving. In other words, the present voters must gain the majority quorum of the total members in the community for a proposal to be implemented.

When the option 'Abstain' is made available, a distinction can be made when the proportion is applied to the present voters. The majority can be only applied to non-abstainers which means that the 'Yes' voters should outweigh the 'No' voters. Alternatively, the majority also applies to abstainers and thus the 'Yes' voters should outweigh the 'No' voters and the abstainers combined. However, Laruelle and Valenciano (2011) states that the former majority does not seem to have been used to date in any parliament.

Due to the token-weighted voting, it opened the possibility to use a token quorum, which means that a proposal should be supported by a minimum of the token supply before it can be implemented. This however may result in the more wealthy members to assert a greater influence on the decisions.

4.2.8 Participation quorum

Participation quorum guarantees that a minimum number of members of the community participated in the decision-making which prevents proposals to be implemented by a few dominant actors. However, it is difficult to find comprehensive data on voter participation in decentralised systems. Karjalainen (2020) stated that not many token holders bother to take part in on-chain ballots. There are positive outliers which lie between thirty and even fifty percent. However, these are outliers and should not be considered as common. Depending on the voting weight, more wealthy members are able to have a great influence on the participation quorum which makes this data inaccurate.

4.2.9 Participation subset

Similar to the majority subset, the participation subset can be in proportion to the entire community or total of the token supply. As mentioned before, applying to the community guarantees that more influential members are able to implement proposals, while the same members are able to influence this with the token supply.

4.2.10 Voting fee

The mechanism to pay or reward a member when voting balances the risks that comes with rewarding the members, and demotivating members to vote. The risks that come with it are in relation to Sybil attacks. By providing a reward for voting, attackers are able to create multiple accounts and thus receive the reward by voting on multiple accounts. Additionally, members potentially make uninformed decisions by disregarding what the proposal is about (Sims, 2019).

4.2.11 Participant motivation

Communities can only be sustained when members are encouraged to participate and show good behavior (Karjalainen, 2020). Without proper incentives, participants may show a lack of interest in voting or even attempt to corrupt the system (Kaal, 2021). A member could participate for self-interest, because the project seems interesting to participate in. Economic motivation could play a role as members are rewarded for their work. Furthermore, the member may be motivated to take part as a member of a like-minded community or social recognition for their contributions and influence.

4.2.12 Dispute resolution

As stated, dispute resolution is something which will be needed sooner or later as disagreement is inevitable. So the community should have an agreed process for when these disagreements need to be resolved (Karjalainen, 2020). Members are able to exit a DAO by selling their native tokens or initiating a hard fork to split the community. Other ways are to implement a mechanism to solve disputes, such as the rage-quit mechanism of DAOhaus (Faqir-Rhazoui et al., 2021). Last option is to create a jurisdiction like Aragon, or go to a physical jurisdiction. However the latter is rather limited, as DAOs are still upcoming and applies to a specific domain (Ziolkowski et al., 2020).

Table 2: Framework Decision Criteria

<table>
<tr><td rowspan="4">General</td><td>Network status</td><td colspan="2">Permission</td><td colspan="3">Permissionless</td></tr>
<tr><td>On vs Off-chain voting</td><td colspan="2">On-chain voting</td><td colspan="3">Off-chain voting</td></tr>
<tr><td>Proposing rights</td><td colspan="2">Everyone</td><td colspan="3">Core developers</td></tr>
<tr><td>Democracy</td><td>Direct</td><td colspan="2">Representative</td><td colspan="2">Liquid</td></tr>
<tr><td rowspan="12">Voting mechanism</td><td>Voting weight</td><td colspan="2">One vote</td><td colspan="3">Token-weighted</td></tr>
<tr><td>Number of winners</td><td colspan="2">Single-winner</td><td colspan="3">Multi-winner</td></tr>
<tr><td>Alternatives</td><td colspan="2">Two</td><td colspan="3">More than two</td></tr>
<tr><td>Vote accumulation</td><td colspan="2">Buying</td><td colspan="3">Earning</td></tr>
<tr><td>Ballot type</td><td>Single</td><td>Multi</td><td colspan="2">Ranked</td><td>Range</td></tr>
<tr><td>Majority quorum</td><td colspan="5">Any percentage between 51-100</td></tr>
<tr><td>Majority subset</td><td>Participants</td><td colspan="2">Community</td><td colspan="2">Token supply</td></tr>
<tr><td>Participation quorum</td><td colspan="5">Any percentage</td></tr>
<tr><td>Participation subset</td><td colspan="2">Community</td><td colspan="3">Token supply</td></tr>
<tr><td>Voting fee</td><td colspan="2">Yes</td><td colspan="3">No</td></tr>
<tr><td>Participant motivation</td><td>Self-interest</td><td colspan="2">Social incentive</td><td colspan="2">Economic incentive</td></tr>
<tr><td>Dispute Resolution</td><td>Mechanism</td><td colspan="2">Jurisdiction</td><td>Sell</td><td>Hard fork</td></tr>
</table>

5 Voting mechanism

The goal of this chapter is to produce a list of voting mechanisms and to provide an answer to sub-question 4. However what is a voting mechanism? A voting mechanism allows a mapping of the voters' preferences of the alternatives that represents the collective preferences (Elkind & Lipmaa, 2005). Voting mechanisms differ in the degree to which individuals are able to express their preferences for the alternatives, in how the votes are counted, and how alternatives are eliminated (Gavish & Gerdes, 1997). However they all share a common goal, namely to represent as many voters as possible with the election results.

The voting mechanisms can be categorized in the way the voting weights are allocated, namely voting systems which are used in traditional voting systems such as elections and token-weighted voting which are (theoretical) voting systems for DAOs. These different voting weights are described in the subsections down below.

5.1 Voting systems

First is the voting systems, which as mentioned before is typically used in traditional voting systems. The voting systems weight their vote by the one-vote-one-user rule. In other words, each participant has only one vote which is equally weighted for all members (Van Erp et al., 2002). This section included the voting systems of Burgman et al. (2014), Cheng and Deek (2007), Gavish and Gerdes (1997), and Pacuit (2011).

The voting systems can be further categorized in the way the voters are able to express their preference, or in other words the ballot type. Note that there is no single consensus in the categorization in the ballot types. While some researchers categorize the four given categorizations (Gavish & Gerdes, 1997; Kaidalov et al., 2017; Laslier, 2012), others combine the single and multi ballot into one category (Brams & Fishburn, 2002; Burgman et al., 2014; Pacuit, 2011). The given ballot categorizations are focused on single-winner systems as most use cases allow only one single winner (Sims, 2019). However, one use case exists where multi-winner systems can be used. The crowdfunding system where members are able to select multiple projects depending on the funds is able to use multi-winner systems. It is decided to focus on the most important single-winner systems (Rivest & Smith, 2007) which means that the list of voting systems is not complete.

5.1.1 Single ballot

The single and multi ballot does not allow voters to express the degree of preference of one candidate over another (Gavish & Gerdes, 1997). Although it limits the voters to express their preferences, it also makes it a less complex method.

The most simplest voting system is plurality with only two alternatives (Brams & Fishburn, 2002). Plurality chooses the alternative which received most of the votes. However, plurality is different from majority, because the alternative with the most votes may not always have more than 50 percent of the votes (Alfurhood & Silaghi, 2018). So in plurality the alternative with the most votes wins regardless if it has the majority of the votes. This comparison does not matter when there are only two alternatives, as each voter casts their vote and the alternative with the majority of the votes or has the most votes is declared the winner. However, in plurality voting with multiple alternatives, more problems are introduced (Kaidalov et al., 2017) including the dispersion of votes across similar alternatives. This may result in an alternative winning the voting, while it would not be selected when the votes of the similar alternatives were combined.

The run-off voting is able to prevent this by including two rounds of voting where the top two alternatives move on to the second round. In the second round, the participants vote again but this time only between the remaining alternatives. Compared to plurality voting, the run-off does not have the voter dispersion problem as the voters are combined in the second round.

Negative voting allows voters to choose one alternative to either vote for or against (Pacuit, 2011). When a vote for an alternative is given, it receives 1 point, voting against means the alternative receives -1 point. The winner is the alternative with the highest total number of points. A drawback to this voting system is that each voter could potentially vote against one alternative, which means that the winner could potentially have 0 votes but still win the election. In other words, the winner is then selected based on the least liked alternative instead of the most preferred one.

5.1.2 Multi Ballot

Approval voting is a voting system which allows voters to vote on multiple alternatives (Rivest & Smith, 2007). The voter considers every alternative and votes for the alternatives he or she approves. Unintentionally, users create its own approval threshold and consider each alternative according to that threshold. Each vote will count as 1 vote and the alternative with the most votes wins. However this voting system tends to select alternatives that may be satisfactory but not ideal from any one perspective as the users are not able to give their degree of preferences (Burgman et al., 2014).

5.1.3 Ranked Ballot

In contrast to the single and multi ballot, the ranked ballot allows voters to provide any information on how a voter would rank the alternatives if their first choice is eliminated. However, ranking all of the alternatives can be demanding, which will likely increase with the addition of alternatives (Gavish & Gerdes, 1997). It can be difficult for voters to make distinctions between the alternatives.

The instant run-off has a similar name to the run-off, as the alternatives with the fewest votes are eliminated. However, instant run-off uses a ranking ballot where voters rank the alternatives in order of preference. Following, the alternative with the fewest votes is eliminated and all votes are recounted. This process will be repeated till a majority is reached. Similar to this method is the Coombs rule, where instead of eliminating the alternative with the fewest first-place votes, the alternative with the most last place vote is eliminated (Gavish & Gerdes, 1997; Pacuit, 2011).

Other voting system that uses the ranking ballot is the Borda Count protocol (Burgman et al., 2014). Each alternative receives a score based on the rank. For n alternatives, first place gets n-1 points, second gets n-2, and the last place receives no points. These points are summed up together and the alternative with the most points wins. When alternatives have an even number of points than the one with the most first-place votes wins. Here other variations exist by assigning different weights to different ranks (Cheng & Deek, 2007).

5.1.4 Score Ballot

The ranked ballot only allows voters to express their preference in a linear degree. Conversely, score ballot allows voters to express their level of preference (Gavish & Gerdes, 1997).

In range voting, the voters have to rate each alternative within a predetermined, fixed, numerical scale, such as a scale from zero to ten. Here there are two variants to determine the winner (Baujard et al., 2018; Pacuit, 2011). The alternative with the highest sum or the highest average wins.

5.2 Token-weighted voting

Second way to allocate the voting weight is token-weighted voting where the votes are weighted by the financial stake that users have in the respective system. The voting power one-token-one-vote system is often indicated by the amount of the native tokens owned by a user. As a result, the more tokens one has, the more power within the organization, which prevents disproportionate influence of those that are less affected by a decision (Hacker, 2019). The list of token-weighted voting systems is extracted from Kaal, 2021, snowballing method is further performed on each system to gather information from the source. This means that this list at the time of writing is complete.

5.2.1 Quadratic voting

In quadratic voting, members are able to buy any number of votes (Lalley & Weyl, 2018). However, the costs are increased in a quadratic function which means that buying one vote equals one credit, while buying two votes equals four credits, etc. This token-weighted voting system is associated with the tyranny of the majority problem. In this problem, the proposal is won by the majority of the voters who had a weak preference, while a sizable minority had a strong preference (Pacuit, 2011). Quadratic voting allows the minority to override the majority by buying more votes.

5.2.2 Conviction voting

The conviction voting of Aragon takes all proposals in the community in consideration (Faqir-Rhazoui et al., 2021). Users are able to vote a portion of its token in several proposals and at any time the user is able to remove their vote. So a user is able to vote all tokens on proposal A, but if the user also wants to vote on proposal B, the user has to remove some tokens of proposal A, hence it is called conviction. The longer the vote is kept in the decision, the more the conviction adds up and thus give more weight to the decision. This token-weighted voting system incentive long-term members to stick to their decision for the long-term. However, if the vote is removed from the decision, the conviction decays over a period. The token-weighted voting system comes with a dynamic threshold which is tied to the treasury funds. The more funds in the treasury, the lower the threshold for a proposal to be accepted.

5.2.3 Futarchy

The idea of futarchy was originally proposed by Hanson et al. (2003) with the slogan: 'vote values, but bet beliefs'. In futarchy a proposal is created with a related metric, for example a proposal is made to fire a member when a company does not reach a yearly revenue of 40.000 for this year. In a binary voting with yes or no, two prediction markets would emerge with each their own token. After a predetermined period, both markets will be closed to see which prediction market has the higher value. If the 'no' market is higher it means that the company did not reach the yearly revenue goal and consequently result in the member to be fired. The trades in the 'no' market are all reverted while the winning market are paid some amount per token.

5.2.4 Token Curated Registry voting

Token Curated Registry is a token-weighted voting system which aims to create and maintain high-quality lists (Asgaonkar & Krishnamachari, 2018; Kosmarski & Gordiychuk, 2020). Members are able to introduce an item in the list with a required token deposit backing the introduced item. Following other members are able to challenge with an equal amount of tokens if he thinks that the item is false or does not belong in the high-quality list. Other members are then able to vote on the challenge and the tokens are redistributed to the winner, while the item is added or rejected depending on the winning side.

5.2.5 Reputation-based voting

In reputation-based voting members are only able to gain voting power by contributing and thus earning reputation within the organization (Kaal, 2021). This reputation is tied to a non-fungible token which means that this token cannot be bought or sold to others. A member of the community is able to stake their reputation to vote on the decisions and the voting power depends on the number of reputation a user has. Furthermore, members receive economic incentives which are paid in proportion to their reputation. This should further encourage members to increase their own reputation by contributing to the DAO. This token-weighted voting system has the advantage that it uses non-fungible tokens which avoids corruptive elements such as Sybil attacks. Additionally, it makes it harder for the members and non-members to manipulate the decision-making in the community. Even if an attacker somehow is able to take over the non-fungible token, he is not likely to maintain the reputation and thus the economic incentive over time, as the reputation can depreciate or even be burned if the user misbehaves or is inactive.

Table 3: Voting Mechanisms Overview

Voting mechanism	Description
Plurality voting	Plurality chooses the alternative which received most of the votes. Even when an alternative does not receive the majority of the votes.
Run-off voting	Run-off includes two rounds of voting where the top two alternatives move on to the second round. In the second round the participants vote again but only between the remaining alternatives.
Negative voting	Negative voting allows voters to choose one alternative to either vote for or against.
Approval voting	Approval voting is a voting system which allows voters to vote on multiple approved alternatives.
Instant run-off	The process of Instant run-off eliminates the alternative with the fewest votes. Then all votes are recounted. This process will be repeated till a majority is reached.
Borda count	Each alternative receives a score based on the rank and the points are summed up together.
Range voting	In range voting, the voters have to rate each alternative within a predetermined, fixed, numerical scale.
Quadratic voting	In quadratic voting, members are able to buy any number of votes with the costs increasing in a quadratic function.
Conviction voting	Decisions made in conviction voting gain more weight the longer the decision is kept.
Futarchy	Decisions in Futarchy are made based on market speculation. For each outcome a prediction markets along with respective tokens are created.
Token Curated Registry voting	Token Curated Registry is a system which aims to create and maintain high-quality lists.
Reputation-based voting	In reputation-based voting, members are only able to gain voting power by contributing and thus earning reputation within the organization. This reputation can be staked to vote.

6 Decision model

This section describes the decision model to select a voting mechanism for DAOs that are in the early stage of development and strive to allow members to express an opinion on an issue.

The decision model is developed by combining the decision criteria with the voting mechanisms. The first five topics of the voting mechanism section are used for the decision model (see Table 4). These topics are: voting weight, number of winners, alternatives, vote accumulation, and ballot type. These decision criteria are reflected in the questions that makes a distinction between the voting mechanisms, such as the first question: "How is the voting weight allocated?" relates to the decision criteria "voting weight". Vote accumulation provides different ways to acquire votes and is a part of distinguishing token-weighted voting systems with the question: "How are participants able to accumulate votes?". The other topics are part of categorizing one-vote-one-user systems which is described down below. Thus by combining the criteria from chapter 4 and voting systems from chapter 5 a decision model is created in Figure 3.

The model makes a main distinction in the allocation of the voting power. These are one-vote-one-user and token-weighted voting. The left branch makes a distinction in the different voting systems. These are then further categorized by the number of alternatives the participants have to choose from, specifically "How many alternatives are available?". If there are only two alternatives then a majority of the votes is sufficient to come to a decision. With more than two alternatives, the voting systems depend on how many alternatives are able to win the voting. This is done by asking the following question: "How many alternatives can be considered the winner?" Not included is multi-winner systems, hence the outcome is generalized into multi-winner systems. The last makes a distinction in the ballot type with the question "Which ballot type is preferred?".

The other branch distinguishes token-weighted voting systems by asking more specific questions. The distinction between token-weighted voting systems are made by asking a specific question which is unique to the token-weighted voting system. For example, reputation-based voting differs from other voting mechanisms in the way the participants are able to accumulate votes (Kaal, 2021). Participants that contribute to the DAO receive reputation in return, and more reputation means more voting power. In other voting mechanisms, participants are only able to accumulate votes by buying. Token Curated Registry requires more specific categorization as this voting mechanism can only be applied to maintaining a list (Asgaonkar & Krishnamachari, 2018; Kosmarski & Gordiychuk, 2020). This voting mechanism can thus be distinguished by asking the following question: "Does the use case involve maintaining a list?" The penultimate voting mechanism that can be distinguished is Futarchy. In contrast to other voting mechanisms, Futarchy leaves the decision to market speculation and thus makes a decision based on supply and demand of the markets (Hanson et al., 2003). The last two voting mechanisms are quadratic and conviction voting. In conviction voting, participants are encouraged to vote for the long-term as the weight of the vote grows with time (Faqir-Rhazoui et al., 2021). This mechanism differ from quadratic voting as quadratic voting does not provide this mechanism.

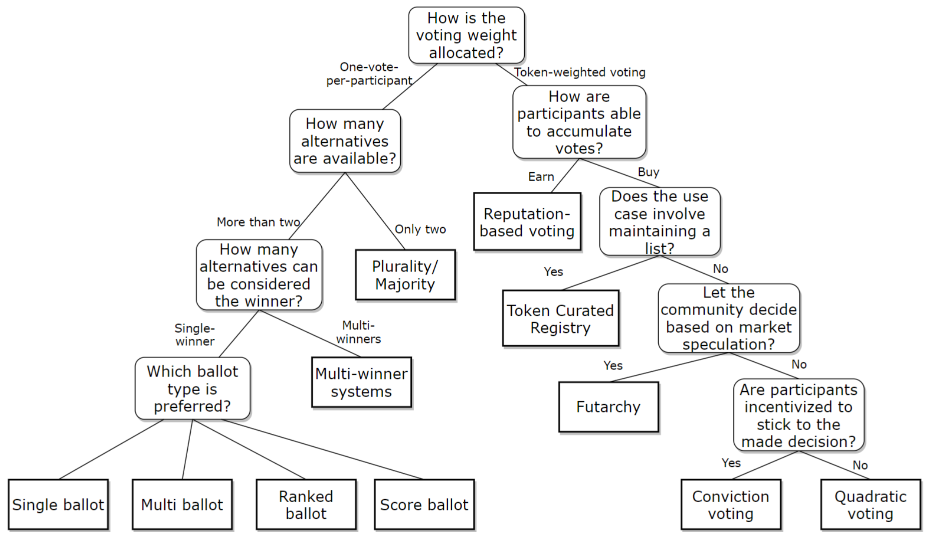

Figure 3: Decision Model Voting Mechanism

7 SearchSECO Case Study

In this chapter, each step of the case study research process of Runeson and Höst (2009) is applied to the respective case study. The execution of the case study can be seen in the last section of the chapter. Additionally the questions posed along with the answers are provided in Appendix A.

7.1 Case study design

The objective of the case study is to apply and evaluate the acquired knowledge from the literature study and the constructed decision model to a case study. To enhance the validity of the framework, it is desired to select subjects that are intended to use the framework (Wohlin et al., 2012). The case study is conducted on the SecureSECO project as they meet the requirements of the case study. As stated in their website, the goal of the SecureSECO initiative is to secure and increase trust in the software ecosystem, through the use of distributed ledger technology and empirical software engineering research. The case study is in relation with the last research question: ”How can the decision model be evaluated?”

7.2 Preparation for data collection

A structured interview is used to collect data during the case study. Beforehand, questions and answers regarding the decision criteria aspects are stated based on the literature study, see Table 2. This framework is sent before the interview to allow the case study subjects to prepare for the interview. At the same time, the artifact is then assessed by the subjects to improve the quality of the framework. By defining questions before the interview is conducted, a structured interview is made. However, at the end of the interview an opportunity occurs where suggestions can be made regarding the framework. Based on both of these suggestions, the framework is corrected and updated accordingly.

7.3 Collecting evidence

Before the start of the case study, permission was asked to the subjects to record the interview for further analysis. Following, each question in Appendix A is asked in sequential order to the subjects. Predefined answers are available in the framework, however the subjects are able to give their own answer that is not included in the framework. The subjects discuss the questions and are required to give a collective answer. Thereafter, the subjects are asked to give an explanation to why the certain answer is chosen. This allows the researcher to get a full understanding of the given answer. Afterwards, the answers of the subjects are summarized to explicitly state the chosen aspects of their voting mechanisms. As stated previously, at the end of the interview there is an opportunity where the subjects are able to provide suggestions regarding the questions, or answers to the questions.

7.4 Analysis of collected data

Qualitative data is gathered through conducting the interview. The questions to the interview are based on the literature study. The results of the interview are used to evaluate the constructed framework and decision model. Since the questions on the decision model is based on decision criteria, the answers given on the interview is applied both on the decision criteria framework and the decision model. This is done by making a transcription of the recorded interview. This allows the researcher to analyze the interview in a textual form. A coding process is conducted on the transcription, which means that relevant remarks are highlighted and used for analyzing the case study. The emphasis is on the answers provided and the reasoning behind the answers. The results can be seen in section 7.6.

7.5 Reporting

This section describes the main findings of the case study. These main findings are reported according to the structure of Robson (2002), which include:

- What the study is about;
- A clear sense of the studied case;
- The history of inquiry;
- Provide basic data in a focused form;
- The conclusion and put into the context it affects.

The case study is performed to apply and evaluate the decision model in order to enable DAOs to select a voting mechanism to allow members of a DAO to express their opinions. The case study is conducted through a structured interview with six participants of the SecureSECO group. The group consists of the project leader of SecureSECO, project leaders of subprojects SearchSECO, and TrustSECO and three other members. These questions are stated such that the participants were encouraged to think about the different aspects of voting mechanisms. The result of this case study is a voting mechanism selected by SecureSECO itself that was based on the answers given to the questions. The sense for SecureSECO to participate in this case study is that there is no artifact to allow DAO or organizations to select a voting mechanism, thus the project is able to by participating in the case study. For the history of inquiry, the subject is selected based on the requirements that were met. The project leaders were contacted to participate and a meeting was made when agreed. Several days before the meeting, the framework included the questions and answers provided to the subjects in order to prepare for the meeting. During the interview, the set of questions were followed as stated in Appendix A. After the interview, a transcription including the coding is conducted. The gathered data were relevant for choosing a voting mechanism for the organization. The conclusion can be seen in the section Conclusion.

7.6 Case study results

For this case study, the project SecureSECO is chosen based on convenience sampling. The goal of the SecureSECO initiative is to secure and increase trust in the software ecosystem, through the use of distributed ledger technology and empirical software engineering research (Jansen et al., 2020).

To achieve this the project leader created SecureSECO and split the project into two, namely SearchSECO and TrustSECO with each their own project leader.

SearchSECO proposes a hash based index for code fragments that enables searching source code at the method level in the worldwide software ecosystem. The plan is to create a parser that extracts the methods from the code files. By making the methods findable, the methods can be analyzed for vulnerabilities which subsequently improve the trust in the software ecosystem.

TrustSECO uses a distributed ledger that stores trust facts about software packages to support the trust that customers of the package managers have (Hou et al., 2021). Such trust data can be whether the package contains known vulnerabilities, whether the package stems from a reproducible build, whether the package is maintained frequently, whether its developers are reputable, etc.

SecureSECO is still in the early stages of development. The project leader wants the community to decide what happens with SecureSECO so different expertise will be combined and improve the decisions made, such as selecting a DAO platform (Baninemeh et al., 2021). This initiative can be enabled by creating a DAO for the project, however SecureSECO does not know what the best voting mechanism is for the respective project.

To join the SecureSECO project, users are able to join via GitHub. GitHub is an open collaboration platform for software developers, thus no permission is required to join SecureSECO. Subsequently, SecureSECO allows everyone in the community to propose as these proposals do not have to be accepted. However, users need to meet a certain threshold to be able to propose, the reason is so the users are not overwhelmed by the proposals while maintaining the quality of the proposals. Further research needs to be conducted to determine the threshold. Thereafter, it is chosen for a direct democracy which allows participants to vote directly on the issue. For this reason, no one will be able to gather the majority of the voting power.

The community will make use of on-chain voting as off-chain voting would obscure transparency and involvement of the community. On-chain voting allows the community to use token-weighted voting. Members are then able to use their weights to vote on an issue. This means that each user may have more voting power, depending on the number of tokens. However, members are only able to accumulate votes by earning as the members of SecureSECO prioritize contributing to the project, which means that the voting mechanism reputation-based voting is chosen. Consequently, members that contribute the most will be the biggest decision-makers in the community.

The topics Alternatives, Number of winners, and Ballot type are provisional as SecureSECO is still in its early stages. This means that the community does not know yet what decisions need to be made. Possibly, these topics can differ depending on the use case of the decision. So for now it is chosen for more than two alternatives, and for single-winner. The preferred ballot type is left open as the decision may depend on the use case. These decisions could change in the near future.

For the majority quorum and subset, it is chosen for a majority of 51% of the participants, because the community wants it to be as open and flexible as possible. This allows the community to react quickly to emergencies, because only a majority of the active participants needs to support the decision for it to be approved. Further a participation quorum of 30% of the community is set as the tokens in reputation-based voting can reduce after a certain time. This means that the community will be more active as the reputation of inactive members will decay over time.

At first instance, a voting fee may seem unnecessary because it is chosen for a token-weighted voting. So there is no benefit for an attacker to create multiple accounts as the voting power depends on the number of tokens. However, the community is planning to set a limit of 10% the amount of voting power one user may have to prevent a user from gaining the majority of the voting power. Subsequently, this again may encourage the attacker to create multiple accounts to gain power. Yet it is chosen to not set a voting fee, because it takes a lot of time and contribution to get to the 10% of the voting power.

A member could join the community for all three of the motivations. Members may be interested in the project or join for the community. Even if there is a reputation-based voting, there is an economic incentive as members will be economically awarded depending on the reputation.

Members are able to sell or force a hard fork to solve disputes in the community, however same as the previous topics this decision is provisional as the community did not experience a dispute yet.

8 Analysis of the Results

In conclusion, SecureSECO decided on token-weighted voting, which brings the path of the decision model (Figure 3) to the right. Additionally, SecureSECO prioritise members to accumulate votes by contributing to the project. Therefore the decision model suggests SecureSECO to select reputation-based voting as the community prioritizes contributing to the project. Thus, it can be concluded that the decision tree works for this case study as a voting mechanism is chosen.

In reputation-based voting each member will receive its own address containing the reputation. Members are able to accumulate reputation by contributing to the DAO. More valuable contribution will receive more reputation. However how valuable the contribution is, is decided by the community. More foundational and long-term contribution might be deemed more valuable. For example, the decision for a community to select a consensus mechanism is deemed a core decision because all transactions of the community is involved in the consensus mechanism.

The received reputation is not-transferable which means that reputation can not flow from one member to another, and thus cannot be bought or sold. In result members with more reputation has more voting power within the community as the high number of reputation indicates that the member made significant contributions. Additionally, funds can be distributed to contributors based on the amount of reputation. However this reputation could also be reduced, for example due to inactivity. An algorithm should be set how much the reduction is during this inactivity period. These incentive and disincentive mechanisms encourage users to acquire more reputation and thus contributing to SecureSECO, which is in alignment with the prioritization of the community.

Despite being able to select a voting mechanism supported by the decision model, it is not perfect. For example in the case study it was not clear what issues the DAO will be voting on. However a main issue was how trust scores are calculated. A decision criteria could be added in which different types of issues are presented in order to prevent this problem. The different types of issues can be further characterized in the trade-off between speed and quality of the decision-making process, and the impact of the decision-making on the community. Additionally, the answers to decision criteria were not leading since notes could be added to the answers. For example, the answer to ”Who are eligible to propose?” is everyone or core developers. However SecureSECO desired to add a condition that a certain threshold should be met before anyone can propose. Therefore answers to the decision criteria should be revised.

9 Discussion

This research has developed a decision model of voting mechanism for DAOs by combining the literature study and case study. This contribution involves disciplines of Information Science in how users are able to seek agreement through information systems.

9.1 Threats to validity

For the case study the validity threats of Wohlin et al. (2012) and Yin (1994) were followed, which distinguishes four aspects of validity: construct validity, internal validity, external validity, and conclusion validity.

Construct validity refers to whether an accurate measure has been used for the concepts being studied. Decision-making is typically defined as a process or a set of activities concerning stages of problem identifying, data collection, defining alternatives, and selecting a shortlist of alternatives as feasible solutions (Kaufmann et al., 2012). To reduce the threats to the construct validity multiple literature resources are used to construct the decision criteria (Table 4) and the voting mechanisms overview (Table 3).

Internal validity determines whether the study is sound by trying to verify claims about the cause-effect relation within the context of the study. To mitigate the threats of internal validity, the design science process steps are followed to construct the decision model for the voting mechanism selection. Moreover a snowballing method is performed to include variables related and thus maximize the context of the study.

The external validity is concerned to which the research findings can be generalized. The term external validity is often used interchangeably with generalizability. The decision model is supported by conducting a case study. This is done to evaluate the validity of the decision model, however a threat to this validity is that the results are based on only a single case, which could indicate that the findings can not be generalized. Conversely, the case study is conducted on a real-life, existing setting rather than a theoretical, laboratory setting which strengthen the representativeness of the chosen sample.

Conclusion validity verifies whether the methods of the study, for example data collection method, can be reproduced with similar results. To mitigate threats to the conclusion validity, a protocol is constructed and reviewed where the process of the case study are defined and the research question and research objective are stated during the preparation. Additionally, the questions asked during the interview are based on multiple sources of literature study, which minimizes the introduction of incorrect facts and results. By preparing a protocol beforehand, the research allows for a great repeatability (Sekaran & Bougie, 2016). Before the interview, the research objective is stated to avoid misunderstandings between interviewer and interviewees.

9.2 Limitations

During the case study, there were several points that can be addressed. Arguably, the topic On vs Off-chain voting in Table 2 may be unnecessary as the goal when creating a DAO is to use on-chain voting, else the same traditional organization can be used when using off-chain voting. Instead the topic inflation mechanism could be added which describes how the value of a certain token can be manipulated, Potential reasons for this is to drive the price of the tokens up, maintain the inflation rate and thus the value of the token, or reduce reputation tokens as a mechanism for inactive members. Alternatively, voting power limitation could be added which describes if there is a limit to the voting power for a single user. This in turn prevents a single user from gaining the majority of the voting power. However, the feasibility of this mechanism should be researched further.

This research experienced limitations in time and resources. Without these limitations, there could possibly be room for multiple case studies. With multiple case studies more projects could be considered to apply the constructed decision model. This means that more requirements and use cases can be taken into account, and thus improve the applicability and generalizability of the model. Alternatively, another method such as expert interviews could be used to evaluate the artifact. These experts could improve the decision criteria by providing remarks or adding and removing topics. Subsequently this might increase the quality of the interview questions and thus the collected data.

The chosen subject was in its early stages of development which means that some answers provided were unsure. However, the participants were introduced to the topics of voting criteria. As a result the participants were acquainted with the possibilities and forced them to make decisions depending on its use case. Still some answers were left open as the implementation of voting mechanism were still absent.

Besides these points, the DAO ecosystem keeps evolving which potentially result in more voting mechanisms that could be considered. Other future work may include multi-winner systems where voting systems allow multiple alternatives to be selected.

9.3 Community governance

In order to put this thesis in a broader perspective we have to take a step back. Instead we reflect back on why voting mechanism selection for a DAO is different than other communities.

The difference between them could be derived from the organizational structure as DAOs tend to have more horizontal structure. Traditional organizations often have a hierarchical structure with an employer on the top of the hierarchy and employees at the bottom. As a result, the consequences of DAOs do not depend on a single person (Morrison et al., 2020; Shermin, 2017). Furthermore, DAOs are able to make use of collective knowledge by involving members in the decisions (Bressen, 2007). Another difference is that DAOs make use of blockchain and smart contracts which results in the rules and transactions of the DAO being transparent, public, and immutable (Murray et al., 2019). The last characteristic is essential since a fault in the code could only be fixed until a voting process is held. Until then malicious attackers could exploit the fault which is detrimental for the DAO.

Therefore, voting mechanism selection for DAOs are different than other communities since multiple entities are involved in the decision-making, and the choice of a voting mechanism is hard to reverse.

10 Conclusion

The aim of this research is to create a decision model to select a voting mechanism for decentralized autonomous organizations. A literature study and a case study is conducted to answer the research question, namely:

Research question: To what extent can a decision model be developed that supports DAOs in selecting voting mechanisms for making decisions?

The first four sub-questions were answered by the literature study and the last sub-question by the case study.

Sub-question 1: How do virtual organizations make decisions?

There are two main processes to reach a group decision, which are consensus and voting. In voting the decision is accepted when a majority support the proposal, while in consensus all members must give their consent before the decision is accepted.

Sub-question 2: What are the voting challenges in virtual organizations that require advanced voting mechanisms?

There are many issues that could go wrong in decision making. The voting challenges that could occur are dispute resolution, scalability issues, Sybil attack, strategic voting, lack of participation, and excessive technocracy.

Sub-question 3: What decision criteria need to be considered when choosing a voting mechanism?

Decision criteria are stated in Table 2, which are categorized into two sections. The section General ranges from who are able to join the network, to who are eligible to vote. The voting mechanism section addresses topics such as whether to introduce voting fee, and the motivation to participate in the network.

Sub-question 4: What voting mechanisms are available through DAO?

The voting mechanisms can be categorized in how the voting weight is allocated. First the voting can be weighted by one-vote-per-participant, which is typically used in traditional voting systems. These voting systems can be further divided in the way the voters are able to express their preferences, which are the four types of ballot.

The single ballot allows each participant to vote only once, and includes the voting systems plurality, run-off, and negative voting. Multi ballot includes approval voting which allows participants to vote on multiple alternatives the participants approve. Ranked ballot where the voter ranks the alternatives in an order. Instant run-off and borda count belong to the ranked ballot. Last, the score ballot where the voters have to rate each alternative on a scale.

In token-weighted voting the voting is measured by the amount of tokens a user has. Examples of token-weighted voting are quadratic voting, conviction voting, futarchy, token curated registry voting, and reputation-based voting.

Sub-question 5: To what extent is the decision model useful in practice?

By using the decision model in this study, decision-makers are able to select a voting mechanism of the shortlist provided. However, it is not complete since the list decision criteria related to the decision model is debateable due to its subjectivity. A reason for this is that few researches on decision criteria in decentralized autonomous organizations are done since the concept of DAO is still novel.

By answering these sub-questions, an answer to the main question can be formulated. A decision model to select a voting mechanism for decentralized autonomous organizations can be developed by providing insight in how virtual organizations are able to make group decisions, considering the voting challenges that comes with it, ponder about the decision criteria that comes with selecting a mechanism, and give an overview of the voting mechanisms that are available for DAOs.

11 Appendix A: Interview

For each criteria, a question was generated. The following questions were used during the interview:

1. Is permission required to join the network?
2. Are the votes conducted on-chain or off-chain?
3. Who are eligible to propose?
4. What kind of democracy exist in the DAO?
5. How is the voting weight allocated?
6. How many alternatives are available?
7. How many alternatives can be considered the winner?
8. Are participants allowed to accumulate more votes (buying or earning)?
9. What ballot type is preferred?
10. What percentage constitutes as a majority?
11. Of which subset must a majority be obtained?
12. What participation quorum should be obtained?
13. Of which subset must the participation quorum be applied to?
14. Are fees included when a vote is casted?
15. What motivates members to participate?
16. How are disputes resolved?

Network status	Permission	**Permissionless**		
On vs Off-chain voting	**On-chain voting**	Off-chain voting		
Proposing rights	**Everyone**	Core developers		
Democracy	**Direct**	Representative	Liquid	
Voting weight	One vote	**Token-weighted**		
Alternatives	Two	**More than two**		
Number of winners	**Single-winner**	Multi-winner		
Vote accumulation	Buying	**Earning**		
Ballot type	**Single**	**Multi**	**Ranked**	**Range**
Majority quorum	51%			
Majority subset	**Participants**	Community	Token supply	
Participation quorum	30%			
Participation subset	**Community**	Token supply		
Voting fee	Yes	**No**		
Participant motivation	**Self-interest**	**Social incentive**	**Economic incentive**	
Dispute Resolution	Mechanism	Jurisdiction	**Sell**	**Hard fork**

Figure 4: Questions asked for the interview along with the given answers

References

Alfurhood, B. S., & Silaghi, M. C. (2018). A survey of group decision making methods and evaluation techniques. *The Thirty-First International Flairs Conference*.

Asgaonkar, A., & Krishnamachari, B. (2018). Token curated registries-a game theoretic approach. *arXiv preprint arXiv:1809.01756*.

Atzori, M. (2015). Blockchain technology and decentralized governance: Is the state still necessary? *Available at SSRN 2709713*.

Baninemeh, E., Farshidi, S., & Jansen, S. (2021). A decision model for decentralized autonomous organization platform selection: Three industry case studies. *arXiv preprint arXiv:2107.14093*.

Baujard, A., Gavrel, F., Igersheim, H., Laslier, J.-F., & Lebon, I. (2018). How voters use grade scales in evaluative voting. *European Journal of Political Economy*, *55*, 14–28.

Beck, R., Müller-Bloch, C., & King, J. L. (2018). Governance in the blockchain economy: A framework and research agenda. *Journal of the Association for Information Systems*, *19*(10), 1.

Brams, S. J., & Fishburn, P. C. (2002). Voting procedures. *Handbook of social choice and welfare*, *1*, 173–236.

Bressen, T. (2007). Consensus decision making. *The change handbook: The definitive resource on today's best methods for engaging whole systems*, 212–217.

Burgman, M. A., Regan, H. M., Maguire, L. A., Colyvan, M., Justus, J., Martin, T. G., & Rothley, K. (2014). Voting systems for environmental decisions. *Conservation biology*, *28*(2), 322–332.

Butler, C. L., & Rothstein, A. (2007). *On conflict and consensus: A handbook on formal consensus decisionmaking*. Citeseer.

Cheng, K.-E., & Deek, F. P. (2007). A framework for studying voting in group support systems. *2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07)*, 16–16.

Diallo, N., Shi, W., Xu, L., Gao, Z., Chen, L., Lu, Y., Shah, N., Carranco, L., Le, T.-C., Surez, A. B., et al. (2018). Egov-dao: A better government using blockchain based decentralized autonomous organization. *2018 International Conference on eDemocracy & eGovernment (ICEDEG)*, 166–171.

Dursun, T., & Üstündağ, B. B. (2021). A novel framework for policy based on-chain governance of blockchain networks. *Information Processing & Management*, *58*(4), 102556.

El Faqir, Y., Arroyo, J., & Hassan, S. (2020). An overview of decentralized autonomous organizations on the blockchain. *Proceedings of the 16th International Symposium on Open Collaboration*, 1–8.

Elkind, E., & Lipmaa, H. (2005). Hybrid voting protocols and hardness of manipulation. *International Symposium on Algorithms and Computation*, 206–215.

Fama, E. F., & Jensen, M. C. (1983). Separation of ownership and control. *The journal of law and Economics*, *26*(2), 301–325.

Faqir-Rhazoui, Y., Gallardo, J. A., & Hassan, S. (2021). A comparative analysis of the adoption of decentralized governance in the blockchain through daos. *Research Square*.

Fischer, A., & Valiente, M.-C. (2021). Blockchain governance. *Internet Policy Review*, *10*(2), 1–10.

Gasser, U., Budish, R., & West, S. (2015). Multistakeholder as governance groups: Observations from case studies. *Berkman Center Research Publication*, (2015-1).

Gavish, B., & Gerdes, J. H. (1997). Voting mechanisms and their implications in a gdss environment. *Annals of Operations Research*, *71*, 41–74.

Gudkov, A. (2017). Legal aspects and distributed character of the decentralized network organization. *Available at SSRN 2911498*.

Hacker, P. (2019). Corporate governance for complex cryptocurrencies? a framework for stability and decision making in blockchain-based organizations.

Hanson, R. et al. (2003). Shall we vote on values, but bet on beliefs? *Journal of Political Philosophy*.

Hevner, A. R. (2007). A three cycle view of design science research. *Scandinavian journal of information systems*, *19*(2), 4.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, 75–105.

Hou, F., Farshidi, S., & Jansen, S. (2021). Trustseco: A distributed infrastructure for providing trust in the software ecosystem. *International Conference on Advanced Information Systems Engineering*, 121–133.

Hsieh, Y.-Y. (2018). The rise of decentralized autonomous organizations: Coordination and growth within cryptocurrencie.

Hsieh, Y.-Y., Vergne, J.-P., Anderson, P., Lakhani, K., & Reitzig, M. (2018). Bitcoin and the rise of decentralized autonomous organizations. *Journal of Organization Design*, *7*(1), 1–16.

Hüllmann, T. A. (2018). Are decentralized autonomous organizations the future of corporate governance? *Bsc, Unpublished*.

Jalali, S., & Wohlin, C. (2012). Systematic literature studies: Database searches vs. backward snowballing. *Proceedings of the 2012 ACM-IEEE international symposium on empirical software engineering and measurement*, 29–38.

Jansen, S., Farshidi, S., Gousios, G., van der Storm, T., Visser, J., & Bruntink, M. (2020). Searchseco: A worldwide index of the open source software ecosystem.

Kaal, W. A. (2021). Blockchain-based corporate governance. *Stanford Journal of Blockchain Law & Policy*, *4*(1), 19–10.

Kaidalov, D., Kovalchuk, L., Nastenko, A., Rodinko, M., Shevtsov, O., & Oliynykov, R. (2017). Ethereum classic treasury system proposal. *IOHK RESEARCH REPORT*.

Karjalainen, R. (2020). Governance in decentralised networks. *Available at SSRN 3551099*.

Kaufmann, L., Kreft, S., Ehrgott, M., & Reimann, F. (2012). Rationality in supplier selection decisions: The effect of the buyer's national task environment. *Journal of Purchasing and Supply Management*, *18*(2), 76–91.

Kondova, G., & Barba, R. (2019). Governance of decentralized autonomous organizations. *Journal of Modern Accounting and Auditing*, *15*(8), 406–411.

Kosmarski, A., & Gordiychuk, N. (2020). Token-curated registry in a scholarly journal: Can blockchain support journal communities? *Learned Publishing*, *33*(3), 333–339.

Kypriotaki, K., Zamani, E., & Giaglis, G. (2015). From bitcoin to decentralized autonomous corporations-extending the application scope of decentralized peer-to-peer networks and blockchains. *International conference on enterprise information systems*, *2*, 284–290.

Lalley, S. P., & Weyl, E. G. (2018). Quadratic voting: How mechanism design can radicalize democracy. *AEA Papers and Proceedings*, *108*, 33–37.

Laruelle, A., & Valenciano, F. (2011). Majorities with a quorum. *Journal of Theoretical Politics*, *23*(2), 241–259.

Laslier, J.-F. (2012). And the loser is… plurality voting. *Electoral systems* (pp. 327–351). Springer.

Li, Y., & Zhou, Y. (2021). Research on the reciprocal mechanism of hybrid governance in blockchain. *Journal of Economics & Management Research. SRC/JESMR/127*, 3.

Lumineau, F., Wang, W., & Schilke, O. (2020). Blockchain governance—a new way of organizing collaborations? *Organization Science*.

Mendling, J., Weber, I., Aalst, W. V. D., Brocke, J. V., Cabanillas, C., Daniel, F., Debois, S., Ciccio, C. D., Dumas, M., Dustdar, S., et al. (2018). Blockchains for business process management-challenges and opportunities. *ACM Transactions on Management Information Systems (TMIS)*, *9*(1), 1–16.

Morrison, R., Mazey, N., & Wingreen, S. (2020). The dao controversy: The case for a new species of corporate governance. *Front. Blockchain 3: 25.*

Murray, A., Kuban, S., Josefy, M., & Anderson, J. (2019). Contracting in the smart era: The implications of blockchain and decentralized autonomous organizations for contracting and corporate governance. *Academy of Management Perspectives*, (ja).

Pacuit, E. (2011). Voting methods.

Pelt, R. v., Jansen, S., Baars, D., & Overbeek, S. (2021). Defining blockchain governance: A framework for analysis and comparison. *Information Systems Management*, *38*(1), 21–41.

Pervan, G., & Maimbo, M. (2005). Designing a case study protocol for application in is research. *Proceedings of the Ninth Pacific Asia Conference on Information Systems*, 1281–1292.

Peters, G. W., & Panayi, E. (2016). Understanding modern banking ledgers through blockchain technologies: Future of transaction processing and smart contracts on the internet of money. *Banking beyond banks and money* (pp. 239–278). Springer.

Reijers, W., Wuisman, I., Mannan, M., De Filippi, P., Wray, C., Rae-Looi, V., Vélez, A. C., & Orgad, L. (2018). Now the code runs itself: On-chain and off-chain governance of blockchain technologies. *Topoi*, 1–11.

Rikken, O., Janssen, M., & Kwee, Z. (2019). Governance challenges of blockchain and decentralized autonomous organizations. *Information Polity*, *24*(4), 397–417.

Rivest, R. L., & Smith, W. D. (2007). Three voting protocols: Threeballot, vav, and twin. *USENIX/ACCURATE Electronic Voting Technology (EVT 2007)*.

Robson, C. (2002). *Real world research: A resource for social scientists and practitioner-researchers*. Wiley-Blackwell.

Rossi, M., Mueller-Bloch, C., Thatcher, J. B., & Beck, R. (2019). Blockchain research in information systems: Current trends and an inclusive future research agenda. *Journal of the Association for Information Systems*, *20*(9), 14.

Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering*, *14*(2), 131–164.

Sekaran, U., & Bougie, R. (2016). *Research methods for business: A skill building approach*. John Wiley & Sons.

Shermin, V. (2017). Disrupting governance with blockchains and smart contracts. *Strategic Change*, *26*(5), 499–509.

Sims, A. (2019). Blockchain and decentralised autonomous organisations (daos): The evolution of companies? *Available at SSRN*.

Tschorsch, F., & Scheuermann, B. (2016). Bitcoin and beyond: A technical survey on decentralized digital currencies. *IEEE Communications Surveys & Tutorials*, *18*(3), 2084–2123.

Van Erp, M., Vuurpijl, L., & Schomaker, L. (2002). An overview and comparison of voting methods for pattern recognition. *Proceedings Eighth International Workshop on Frontiers in Handwriting Recognition*, 195–200.

Virovets, D., & Obushnyi, S. (2021). Decentralized autonomous organizations as the new form of economic cooperation in digital world. *The USV Annals of Economics and Public Administration*, *20*(2 (32)), 41–52.

Wang, S., Ding, W., Li, J., Yuan, Y., Ouyang, L., & Wang, F.-Y. (2019). Decentralized autonomous organizations: Concept, model, and applications. *IEEE Transactions on Computational Social Systems*, *6*(5), 870–878.

Webb, N. (2018). A fork in the blockchain: Income tax and the bitcoin/bitcoin cash hard fork. *North Carolina Journal of Law & Technology*, *19*(4), 283.

Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. *Proceedings of the 18th in-*

ternational conference on evaluation and assessment in software engineering, 1–10.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., & Wesslén, A. (2012). *Experimentation in software engineering*. Springer Science & Business Media.

Wright, A., & De Filippi, P. (2015). Decentralized blockchain technology and the rise of lex cryptographia. *Available at SSRN 2580664*.

Yermack, D. (2017). Corporate governance and blockchains. *Review of Finance*, *21*(1), 7–31.

Yin, R. K. (1994). Case study research: Design and methods, applied social research. *Methods series*, *5*.

Zhang, B., & Zhou, H.-S. (2017). Brief announcement: Statement voting and liquid democracy. *Proceedings of the ACM Symposium on Principles of Distributed Computing*, 359–361.

Zhang, W., Yuan, Y., Hu, Y., Huang, S., Cao, S., Chopra, A., & Huang, S. (2018). A privacy-preserving voting protocol on blockchain. *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, 401–408.

Zheng, Z., Xie, S., Dai, H., Chen, X., & Wang, H. (2017). An overview of blockchain technology: Architecture, consensus, and future trends. *2017 IEEE international congress on big data (BigData congress)*, 557–564.

Ziolkowski, R., Miscione, G., & Schwabe, G. (2020). Exploring decentralized autonomous organizations: Towards shared interests and 'code is constitution'.

Zwitter, A., & Hazenberg, J. (2020). Decentralized network governance: Blockchain technology and the future of regulation. *Frontiers in Blockchain-Blockchain for Good, Forthcoming*.