Genesis DAO
Ecosystem Integration

Utilizing Stylograph to build an extendable Ecosystem

Stylograph is a framework aimed at enhancing the functionality of substrate-based chains in the Polkadot ecosystem with plugin-like functionality. It's built as part of a Grants Program.

Genesis DAO is the first use case and implementation of Stylograph. We will provide hookpoint functionality to extend the basic functionality of Genesis DAO with configurable ink! contracts.

The ask for this proposal is $365,000, which will be split over a 6-month period to fund the development and implementation of the framework.

This equals 77,000 DOTs on the current strike price of $4.74.

Another 500 DOTs is requested as a payback from the deposit bound of the Governance V1, where we already initiated the proposal.

The total ask is therefore 77,500 DOTs.

Introduction

GenesisDAO is the first use case and the first implementation of *Stylograph* as *Genesis DAO* will have a marketplace, an ecosystem and an open source community to build complex DAO tooling on top of the core infrastructure.

Within this treasury proposal we are introducing the first components for this.

They will be loosely coupled, meaning that other ink! protocols can utilize this functionality even outside of DAO contexts (e.g. the vesting wallet can be used for token issuance to investors) and will serve - alongside with friendly documentation - as a starting point for others to explore the potential of Stylograph.

The contracts are as well easily extendable to allow for a growing ecosystem of DAO extension on the Genesis DAO chain.

Genesis DAO is already operating on testnet:

- Node Explorer
- Backend Service Documentation
- Frontend (Under Development)

The respective code is open source:

- Node
- Backend
- Frontend

Genesis DAO is aiming to be a system parachain, which is why it will use DOT as its token for fees and won't issue its own token.

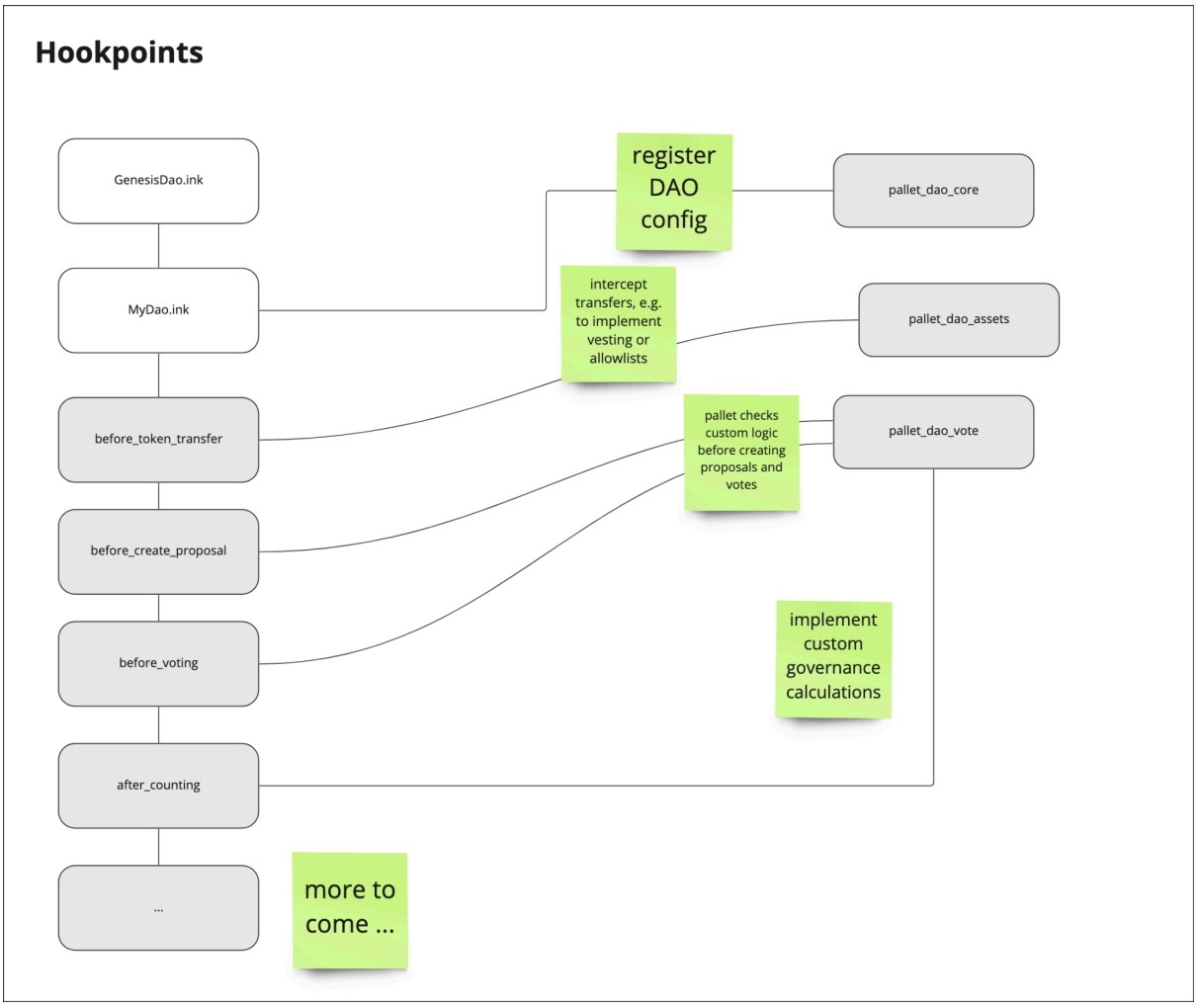
The project will be led by Deep Ink Ventures, a venture studio based in Berlin.

Ecosystem Integration

The implementation of Stylograph alongside with *pallets_contracts* will serve as the next roadmap iteration of *Genesis DAO*. Our substrate chain will be morphed from a pure pallet-extrinsic oriented infrastructure to an ink! powered ecosystem for DAOs.

Genesis DAO Stylograph Callbacks

Genesis DAO will implement callbacks for ultimate flexibility and is utilizing the *ink! Hookpoint Facade* from stylograph to serve as a base contract to extend the entire *Genesis DAO* core functionality.
This allows *ink!* extension developers to wire deeply into the core of Genesis DAO with the option to alter the logic of the entire DAO, Token, Proposal and Voting lifecycle.



This serves as a basis point for the modules described in the next section but can be used by developers to easily roll out infrastructure, extensions and plugins for the DAO ecosystem built with ink! on top the Genesis DAO substrate chain.

The initial examples are based on the voting mechanisms, so we are illustrating the details here, to illustrate how the implementation may take place.

Genesis DAO ships with a simple mechanism to configure the governance type for DAO.

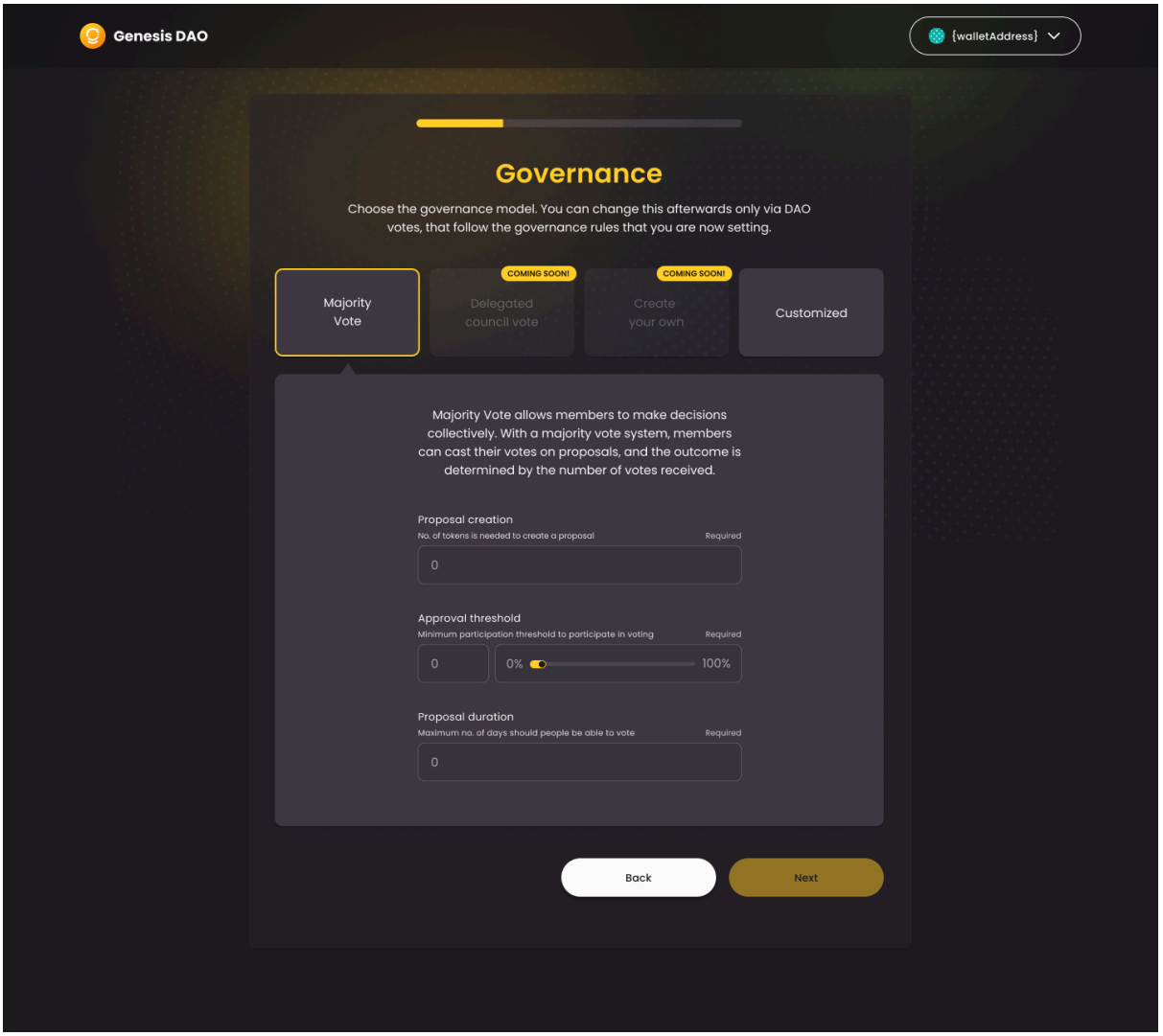
It initially ships with majority voting - the basic mechanism.

```
#[derive(Clone, Encode, Decode, Eq, PartialEq, RuntimeDebug, MaxEncodedLen, TypeInfo)]
pub enum Voting {
    Majority {
        // how many more ayes than nays there must be for proposal acceptance
        // thus proposal acceptance requires: ayes >= nays + token_supply / 1024 *
            minimum_majority_per_1024
        minimum_majority_per_1024: u8,
    },
}
```

When it comes to calculating the votes we currently have a *match* block with only one option:

```
    match governance.voting {
        Voting::Majority { minimum_majority_per_1024 } => {
            if votes_for > votes_against && {
                let token_supply = Assets::<T>::total_historical_supply(
                    asset_id.into(),
                    proposal.birth_block,
                )
                .expect("History exists (horizon checked above)");
                let required_majority = token_supply /
                    Into::<AssetBalanceOf<T>>::into(1024_u32) *
                    minimum_majority_per_1024.into();
                // check for the required majority
                votes_for - votes_against >= required_majority
            } {
                proposal.status = ProposalStatus::Accepted;
            }
        },
    }
```

The *match* block is already a preparation for the ecosystem extension, and we've even highlighted a few of them in our current designs to excite users of the things to come:



We are now adding a new governance type - *CustomVoting* - that is delegating the calculations to the *ink!* contracts configured within the stylograph pallet.

Ink! Integrations

ink! Vote Escrow Extension

Vote Escrow tokens are a popular concept in decentralized finance (DeFi) and have been popularized by Curve, a decentralized exchange platform.

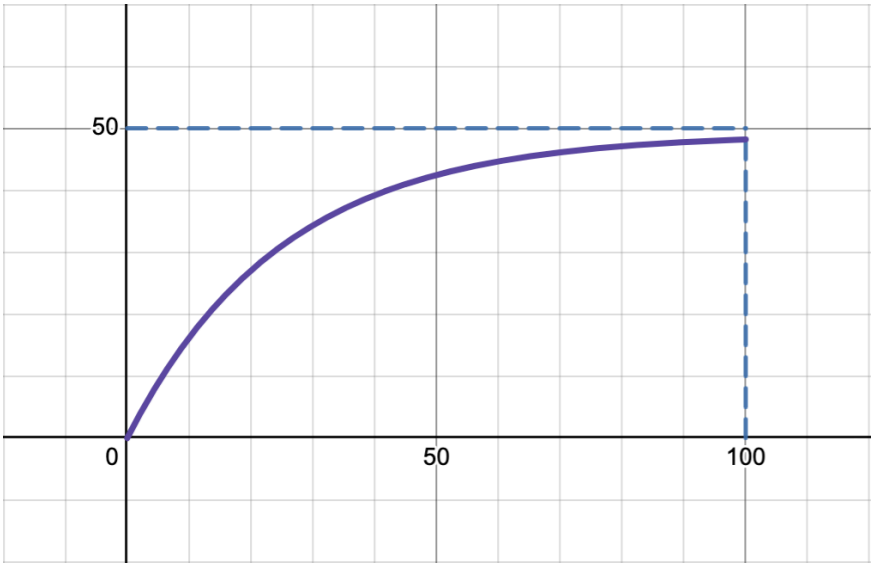
The primary purpose of Vote Escrow tokens is to incentivize long-term commitment to a platform or ecosystem by allowing users to lock their tokens for a specified period of time. In return, users receive voting power and, often, additional rewards in the form of boosted rewards. This mechanism helps ensure that participants who are actively engaged and committed to the long-term success of the platform have greater influence over its governance decisions.

We will make the boost function easily extendable by just overwriting the boost function and provide a basic function with an exponential growth rate bounded by a maximum lock up period:

$$V_e(t) = \frac{V_l(t)(1-e^{-\frac{t}{T}})}{1-e^{-\frac{T_m}{T}}}$$

Where:

- $V_e(t)$ represents the number of Vote Escrow tokens at time t,
- $V_l(t)$ represents the number of locked tokens at time t,
- t is the lock duration in time units,
- T is a time constant for the growth of the voting power,
- T_m is the maximum lock duration.



This plot was generated with 50 locked up tokens and a maximum lock up time of 100 days. The violet line represents the voting power returned by the custom voting extension of the ink contract.

ink! Vesting Wallet

A common primitive for use cases like DAO tooling, token issuance, initial token offerings and many more are vested wallets.

Vesting is a mechanism that gradually releases funds or tokens to individuals or entities over a predetermined period. For DAOs (Decentralized Autonomous Organizations), vesting makes sense for several reasons:

1. Incentivizing long-term commitment: Vesting schedules encourage participants to remain engaged with the DAO for an extended period. By gradually releasing tokens over time, participants are motivated to contribute to the DAO's growth and success.
2. Aligning interests: Vesting ensures that the interests of founders, team members, and investors are aligned with the long-term success of the DAO. Since tokens are released over time, stakeholders are incentivized to work towards the project's success to maximize the value of their tokens.
3. Reducing the risk of token dumping: Vesting schedules can reduce the risk of participants selling a large number of tokens immediately after receiving them, which could negatively impact the token price and overall market stability.
4. Retaining talent: By offering vesting rewards to team members and contributors, a DAO can attract and retain skilled individuals who are willing to commit their time and expertise to the project.

A vesting contract serves as a custodian, holding tokens on behalf of the user until they become eligible for release. The contract includes a release function that defines the terms under which the tokens can be unlocked and released to the user.

Initially, the contract will implement a linear release function, which gradually releases tokens at a constant rate over the vesting period. However, this release function can be easily overridden or extended by other contracts to implement custom vesting schedules, such as those with varying release rates or milestone-based releases.

$$U(t) = V_t - \frac{t-t_0}{T} V_t U(t) = V_t - \frac{t-t_0}{T} V_t$$

Where:

- $U(t)$ is the number of unvested tokens at time t,
- t is the time elapsed since the beginning of the vesting period,
- t_0 is the starting time of the vesting period,
- T is the total duration of the vesting period,
- V_t is the total number of tokens to be vested.

The function gives a linear release of five Tokens over ten days.

Users can withdraw their vested tokens according to the release function defined in the contract. The contract keeps track of the vested and unvested tokens for each user, and releases tokens as they become eligible for withdrawal.

The vesting contract can integrate with the voting mechanism of Genesis DAO using Stylograph. This approach ensures that the vested tokens still contribute to a user's influence in governance decisions whilst not available for trading - promoting long-term commitment to the DAO.

The vesting contract can also be extended to incorporate additional features, such as compatibility with the aforementioned Vote Escrow tokens.

ink! Delegated Council Voting

Delegated Council Voting is a governance model that combines elements of direct democracy with representative democracy in a DAO context. In this model, DAO members elect a council of representatives who are entrusted with decision-making powers on behalf of the community. The idea is to ensure that decisions are made by individuals who have demonstrated expertise, dedication, and alignment with the community's values and goals.

Delegated Council Voting can offer several advantages for DAO governance:

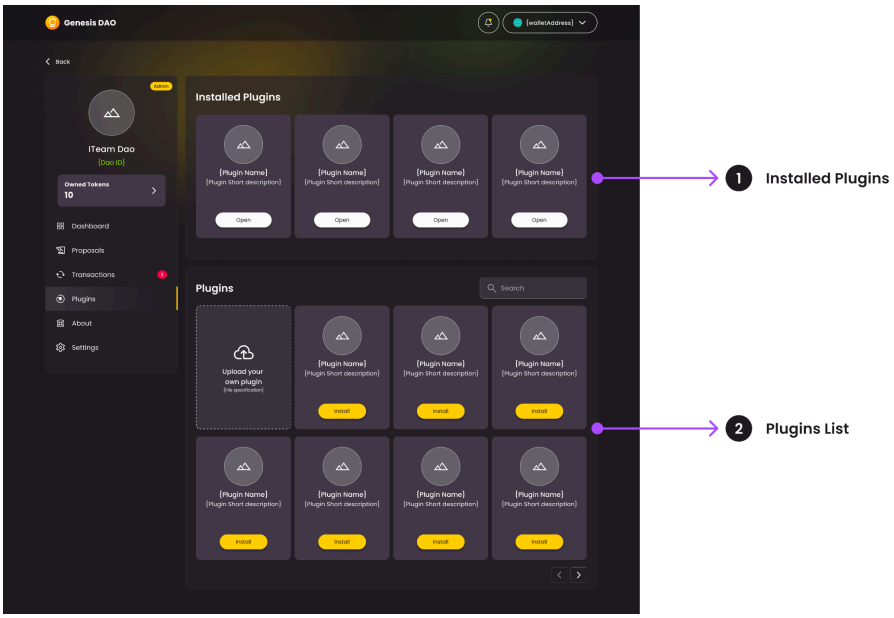
- It streamlines decision-making by entrusting a smaller, dedicated group with decision-making powers, reducing the time and effort required for consensus.
- Council members, who are often experts or highly engaged community members, can bring specialized knowledge and experience to the decision-making process.
- It encourages participation from a broader range of community members, who may be more likely to delegate their voting power to a trusted representative than to participate in every decision themselves.

The ink! contracts require a (configurable) amount of token holders to delegate their voting power for a limited number of candidates that are stored in the contract as well. While the threshold is met, the ink contract intercepts the voting mechanism and only counts votes of the delegates - proportional to the respective underlying voting power for each delegate. The latter can be fine tuned by min/max values and exponential easing to prevent concentration on single delegates.

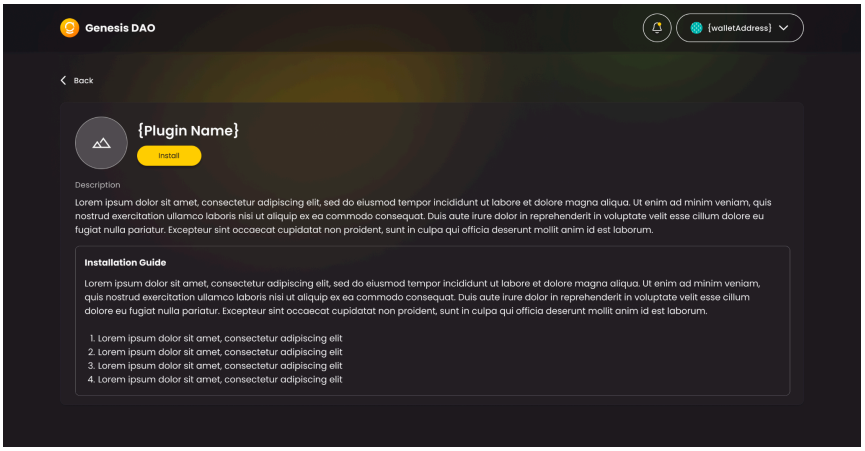
Plugin and Extension Store

While the plugin and extension system is quite complex, we are committed to a no-code experience for DAO communities.

Therefore we as well want to provide a plugin and extension store that allows users to select extensions from a list of extensions that have been published by the community and *install* them to their dao by deploying the respective smart contract and register the specific hook point to their DAO.



Overview Page Mockup



Detail Page Mockup

Budget Plan

Measurement is given in Days (1 Day ~= 8 hours of FTE).

Project Management and Administration - $35,000

- Coordination of team members and resources (~10 Days)
- Regular progress reports and updates (~10 Days)
- Budget oversight and management (~10 Days)

Ecosystem Setup - $70,000

- project and infrastructure setup (~20 Days)
- setup of the pallet_contracts and ink! ecosystem (~20 Days)
- stylograph integration (~20 Days)

Component Development - $150,000

- ink! Staking Boost Extension (~25 Days)
- ink! Vesting Wallet (~25 Days)
- ink! Delegated Council Voting (~25 Days)
- Wireframes, Designs and Frontend Integration for all Components (~50 Days)

Plugin and Extension Store - $35,000

- Design Setup and core integration (~10 Days)
- Backend Setup and extension management (~10 Days)
- ink! deployment integration (~10 Days)

Testing and Documentation - $40,000

- Testing of the developed framework and DAO components (~15 Days)
- Ensuring compatibility with Polkadot and Substrate ecosystem (~5 Days)
- Developing written user guides and technical documentation for the project (~5 Days)

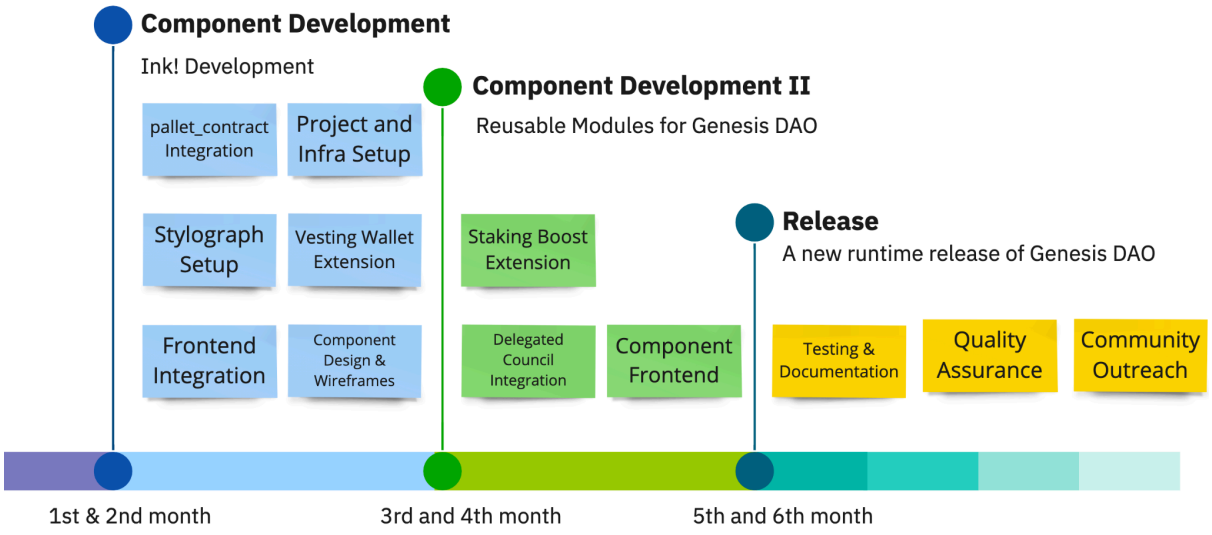
Marketing, Partnerships and tech support for clients $35,000

- Engaging with the community to encourage adoption and collaboration (~5 Days)
- Providing tech support to developers and users of the framework (~5 Days)
- stakeholder management, feedback collection, partnerships (~5 Days)
- partnership Integration management (~5 Days)
- Organizing a hackathon for Genesis DAO ecosystem (~10 Days)

Roadmap

This plan is divided into three distinct phases:

- Component Development
- Component Development II
- Release



Over the first two months, the project team will focus on setting up the ink! ecosystem and integrating the stylograph pallet and will start with the components and it's wireframes. It'll finish the Frontend Integration part by then

Months three and four are dedicated to the development of the components alongside with their frontend interface.

Finally, in the last two months, the project will be released to a new runtime of Genesis DAO, giving extension developers and communities the flexibility and tools to customize their DAOs with stylograph.

All phases will be accompanied by our business and marketing team to align the needs of the project with the community and to help extension / integration developers and teams to onboard onto Genesis DAO fast.