Trustworthy Crowdfunding

Crowdfunding is currently a popular way for creators to raise money to finance projects. There is a trust issue with crowdfunding though. The majority of crowdfunding projects do not deliver on time and some never deliver at all.

A decentralised crowdfunding application which implements a blockchain network and smart contracts would be a good solution for this problem due to how traceable and transparent each donation and transfer of funds would be. It would add an accountability trail and allow users to see if the funds are being used to fulfill the explicit goals of the project.

Every crowdfunding platform also charges fees, between 3 and 5% of all raised funds, and they act as a middle man. You transfer the money to them and then trust them to pay the creator/project developer if appropriate or to refund you. With a blockchain network no middleman is needed.

Scope:

Have to decide on the initial scope of the project because there are different types of crowdfunding:

- Kickstarter: Creators set goals and a time period in which to complete them. Can also set milestones and sometimes rewards for donations of a certain size or reaching milestones. Kickstarter only pays out after the goal of the project is reached. With smart contracts we could release funds after certain milestones are reached. Has a website where projects can be browsed.

- Patreon: Subscription model where patrons regularly contribute a set amount of money, either weekly, monthly or per creation to digital creators and in return, often but not always, they will receive exclusive content. Does not have a website where projects can be browsed. Smart contracts could check though transaction history to see if a particular person has paid the subscription. If they have the contract could release the content to them in the form of another transaction.

- Gofundme: Emergency and charitable causes. No goals but there is an issue with trust as there is no accountability in how this money gets spent. It's rife with scams.

Note: As per the meeting the initial focus of the project will be to implement a kickstarter type platform. If time permits the scope of the project could be expanded to include the other models.

Picking a blockchain:

The important considerations when choosing which blockchain to use are:

- Scalability: Blockchains suffer from what is called “the scalability trilemma”. This means in order for a project to be scalable one of three considerations must be sacrificed in order to prioritise the other two. The three considerations are speed, security and decentralisation. I think for this project speed is the least important. This is because these are long term projects being financed. There is no need for transactions to be instantaneous like they might need to be for trading stocks or gambling on sports or similar.

- Adoption rate, Functionality and community support: We definitely need smart contract functionality. We want a platform with a high adoption rate just so it is more likely to be updated over time and it will be supported in the future. Community support is important for identifying bugs and security issues.

- Security: We want a platform with a good track record of maintaining security.

- Public or private: We want a public network so that anyone can donate and view where the funds are going.

Based on these requirements I think the main choice is between Ethereum and EOS.

EOS has faster transaction processing speeds at the cost of having more centralisation and supports more complicated smart contracts.

Ethereum has a slower transaction per second rate. It has more decentralisation. It uses the gas model for transactions- this means the user bears the burden of costs. This just makes sense to use, the gas prices will be a very small addition to or subtraction from each donation. It uses Solidity or Vyper to code and compile smart contracts.

For this project I think Ethereum best meets all the requirements.

How the platform will work / how users interact with it:

In a blockchain network every client browser communicates with its own instance of the application, there is no main server. The user will need to download geth which is the official client software provided by Ethereum which will download a copy of the blockchain and make them a node on the network.

There is a javascript library called web3.js which can interact with geth nodes. I can use this to build a web based front end that users can interact with. They can use this to create projects, set goals and donate to projects with the complexity being abstracted from them.

It could also be possible to have metrics for users to view the status of projects, donations or how funds are being spent. To get this info we can call read only functions coded within smart contracts to return values that then can be formatted in a visually interesting and informative way.

Smart contracts will be written and compiled into bytecode using Solidity or vyper.