The Relay Project

Community Source Principles

What is Community Source?

A type of software model defined by community ownership of Source as well as service towards a community of users. In this model, it is suggested that no one or more entities make stake a claim to the source code and related intellectual property once it's considered Contributed Source. At the same time, all Source is considered community-owned by the active and contributing users of that community.

Community-maintained

No administration

A CS network has no privileged accounts, no hidden administration section, no backdoor access. A user is a user is a user. Typically a user account may assume one or more roles temporarily or may be granted them in order to perform maintenance, dispute resolution, and other such tasks. It doesn't matter who you are, just what you can do. Your task history is the evidence.

Community-oriented dispute resolution system

It can be thought of as a cyber jury system. Users submit complaints and disputes to the community which are handled in the context of blind justice. However, punitive actions by the community towards users are discouraged. Instead practices of Harm Reduction should be applied in such scenarios as involving damage or abuse of users

Self-sustaining

Cryptocurrency Wallets are used to keep track of site budgets for such things as server costs, and real-world maintenance. Users may maintain their own private budgets but all public budgets must remain publicly visible at all times. Additionally, members of a CS community should attempt to anticipate and prepare for future scenarios that may jeopardize the integrity of the system.

Resilience and data protection

A Community-Sourced application should be resilient to data loss, and should never allow the integrity of user data to be compromised.
No user data should ever be accessible by the server or anyone else without first having access granted by that user. This is accomplished by the user re-encrypting their data with the public key of the entity they wish to authorize.

No profit model

A community-sourced project might encounter problems if it were to attempt to follow a classic profit model because the financial interests would be liable to re-prioritize or even compromise the project's main service goals. That isn't to say that profit itself is incompatible with a community-sourced project, but it does mean that profit (non-profit styled or the other kind) would have to come from an endeavor by a user of the system, rather than the system itself, like a charity set up by users, or a merchandise section.

Revenue Streams

A community-sourced application typically must have some kind of income stream to meet its goal of being self-sustaining. Additionally, users may seek their own revenue streams and conduct business as long as it meets the standards and regulations of that community

No ads. No fees

Application interfaces build and maintained for public purposes should remain free of advertisement at all times. Advertisement may only exist in designated areas.
No service fees should be charged for the usage of any software under a Community Source license. Donations are the prefered way of lobbying users for funding towards software services.

Donation streams

The primary source of revenue is donations. With a financial framework at the disposal of the user, donations should be quick, simple, and effortless. It is understood that donations to public Donation Streams are always publicly tracked, so each user can trace exactly where their donation is going and what it's used for. A Donation Stream typically has a stated purpose like 'Server Costs', 'R&D', or even 'Help the homeless'. Donations can come in the form of service tips and must meet community standards. Advertising is never permissible.

Contribution streams

A Contribution Stream is an revenue source whereby a software developer will submit a site contribution and attempt to derive a source of revenue from it. This is achieved by optioning the users of that contribution to donate to the revenue source that helped create it. Contributions will be peer-reviewed and must meet all community standards as well as task requirements. Multiple contributors may split a pool within a Contribution Stream and negotiate payment schedule priorities if necessary. For example, a gaming-oriented community-sourced network may have developers that create online games or features for that community in exchange for donations as a source of income. Donation requests must meet community standards. Advertising is never permissible.

Investment streams

An Investment Stream is a type of financial contribution towards a community priority, like a new piece of software. An investment differs from a donation in that it may have a repayment schedule and may attempt to derive profit from the investment. The terms of repayment are set between the investor and contributors and must meet community standards.

Investment Streams may be used in combination with Contribution Streams to provide funding towards a stated purpose. Other users may apply to complete tasks in exchange for that funding. No site fees should ever be applied if users wish to engage in their own business transactions, but donations may be requested

Ownership and Licensing

Open source

A community-sourced application cannot have it's source hidden from the community and is therefore required to remain open-source, although security precautions are still taken

A Community Source license may not be compatible with an open source licenses that allows ownership over intellectual property in conflict with Community Source core concepts. Changes to a more traditional license that allows IP ownership should also be resisted because such a move would be liable to affect the integrity of the system.

Users are allowed to view local source code if they wish as well as submit requests, feedback and submit Contributions.

Ownership licenses

Source code ownership by community -
The software contributed under a Community Source agreement cannot be owned by anyone, only the community it was created for.

Source ownership by individual

Ownership Licenses will not be compatible with Community Source if the license attempts to claim ownership over any part of the code. Community Source does not apply to software that is not officially accepted as a contribution by the community. Any intellectual property created by users will not be affected by this agreement unless it is officially submitted as a 'contribution' at the creator's discretion.

Registration and Invites

Invite System

As a necessary step of maintaining the integrity of the user base, the only way to register is to be invited. This provides a semi-public invite tree that can be used to trace, reduce and repair harm from malicious registrations. This is also a critical step for gauging and responding to community growth and maintaining healthy revenue streams.

Invite Tree

Users are allowed to invite new users to the network within the limits set by the community. As users choose to invite someone they are notified that the actions of that new users may reflect on them in the future. In an investigation of a malicious attack by one account may trace back to a source of bad

registrations. Harm Reduction measures may be determined by arbitration or automatically to reduce the harm potential of an entire tree section at once. Similarly, granted roles may be curtailed for an entire group using the invite tree in the event of malicious infiltration.

Invite Network

As a user invites or is invited to the network, this creates a small network of relationships that can be invoked in a community setting. For example, a user may option to leave a backup copy of their protected private key with their invite sender, incase they ever lose it. A user might invite everyone in their close invite network to a game of Real-Time "Texas Hold'em".

Privacy and Finance

Privacy is an illusion. Encryption is not

A core concept of a community-sourced application is that all matters of privacy, role assignment, and permissions are handled by encryption. That is to say if you can decrypt it, you can use it. If a user created a file (or wallet) that user will be the only person that can decrypt it. If the entire server was stolen off the rack, the data would be useless to anyone without each user's private keys and passphrase. However, a backup, even if public, would theoretically restore everyone's data without compromising privacy or integrity of the data.

Financial framework - cryptocurrency

A critical core feature of a community-sourced project in this context is a solid underlying financial framework involving, naturally, cryptocurrency. Each user has complete control over their finances and may create transactions with other users, and trade for items and services. User wallets are private and encrypted in such a way that the files are useless without the user's private key to unlock it. This measure of user control is necessary to maintain the user's trust and should never be constrained in the future as to not compromise that trust.

Secured Wallets

When created, a crypto-currency's private information will be encrypted with the user's private key and then immediately destroyed. Any money then sent to the public address will only be accessible by someone with the private key. This also allows other users to be granted limited or full access to a secured wallet

Distinction with modern bitcoin wallet technologies

Secured Wallets in a community-sourced application differ from other cryptocurrency storage methodologies in that there's no central authority that can access the user wallets, and no wallet can be accessed even if stolen. While some banks offer offline cryptocurrency storage to protect from hackers, they themselves still remain a central authority. This leaves unsecured banks still vulnerable to mass theft scenarios either by hackers or malicious employees. Secured Wallets, by distinction, cannot be used if stolen, regardless of how they are stolen. They still require, at some point in the timeline, the authorization of the original user's private key to decrypt the secured wallet.

Public Secured Wallets

This is a type of Secured Wallet where limited access is given to community members to perform

transactions on the behalf of a public interest. For example, a public budget might be created to manage server costs and allow users limited access to make real-world server payments out of the public wallet.

Contribution

Task system

The task system is how work will eventually be done in this system, and how work gets done in a community-sourced application. Essentially, task entries will define R&D tasks and act as receipts for the users who performed them. Each task value will be qualitatively determined either before or after it is performed with all work accounted for. A fledgling community-source project (like this one) may need time for an income stream to be realized, so the task system provides a guarantee that their work will eventually be accounted for (and paid out).

Example: Coding tasks

Any tasks involving programming will be source-controlled and peer-reviewed. There will be ample opportunity for new code to be approved before it makes it into the master repository. A commit that meets the approval of the community will be paid out according to it's determined task value and schedule

Path concept

The path concept in this context defines a core concept of a community-sourced application which helps the user get a feel for 'where' they are on a website. They understand they can always go up, down, left, right according to the site content. It's not a 404, just a new path. Users may create new nodes on the site tree, essentially 'building up' a website after it launches. This can be anything like helpful advice, organized files, etc. Up to the users.

Law and compliance

The concept of a community-source project has not been vetted or researched for the purposes of this or any project, and is only an idea at the time of this writing. When determining our license we will make every effort to comply with existing laws and regulations and do not anticipate any conflicts

Currency disclaimer

This project deals only with cryptocurrencies and does not utilize world currencies like USD. There are no currency conversions or financial operations that deal in world currencies. This is not a casino, it's a gaming platform. We comply with all laws, are not a bank, and are not a bitcoin exchange, for example.