

Clarys.AI - OpenGov Data Backbone & Accountability Explorer (Beta v1)

Proponent: 146ZZqm2cMHLf3ju7oc8M9JnPaAktuADAKThagKnXqzjPJbZ

Date: August 19th 2025

Phase: 1 of 2 (Phase Two will follow after Beta v1 delivery and adoption)

Amount Requested: USD 130,00

Previous treasury proposals: W3F DF grant: tinyurl.com/clarysintro ;
<https://polkadot.polkassembly.io/referenda/1679>

Category: Public Goods / Governance Infrastructure

Licensing: Code (MIT or Apache-2.0), Datasets (CC-BY-4.0; CCO considered upon community request)

Executive Summary

Polkadot's OpenGov has matured into one of the most capable collective decision systems in crypto. Yet the information environment around it is fragmented. A typical voter's journey crosses Polkassembly and SubSquare, dives into forum threads and bounty pages, peeks at GitHub activity, skims OGTracker and scans social channels like Reddit. Each step adds context. Each hop adds doubt. Where did a claim come from? Is a "new" proposal actually the same request across a different track? Did we fund something similar six months ago?

Clarys.AI is the governance intelligence layer designed to reduce that cognitive burden. We are building a provenance-first data backbone and a **usable** product, the **Accountability Explorer**, that unifies on-chain and off-chain governance signals, resolves entities across mechanisms, classifies duplication with clear definitions and reveals relationships visually. The data and interfaces are **open by default**: public GraphQL/REST APIs, daily CSV/Parquet exports and permissive licenses so any researcher or builder can reuse and extend the work without lock-in.

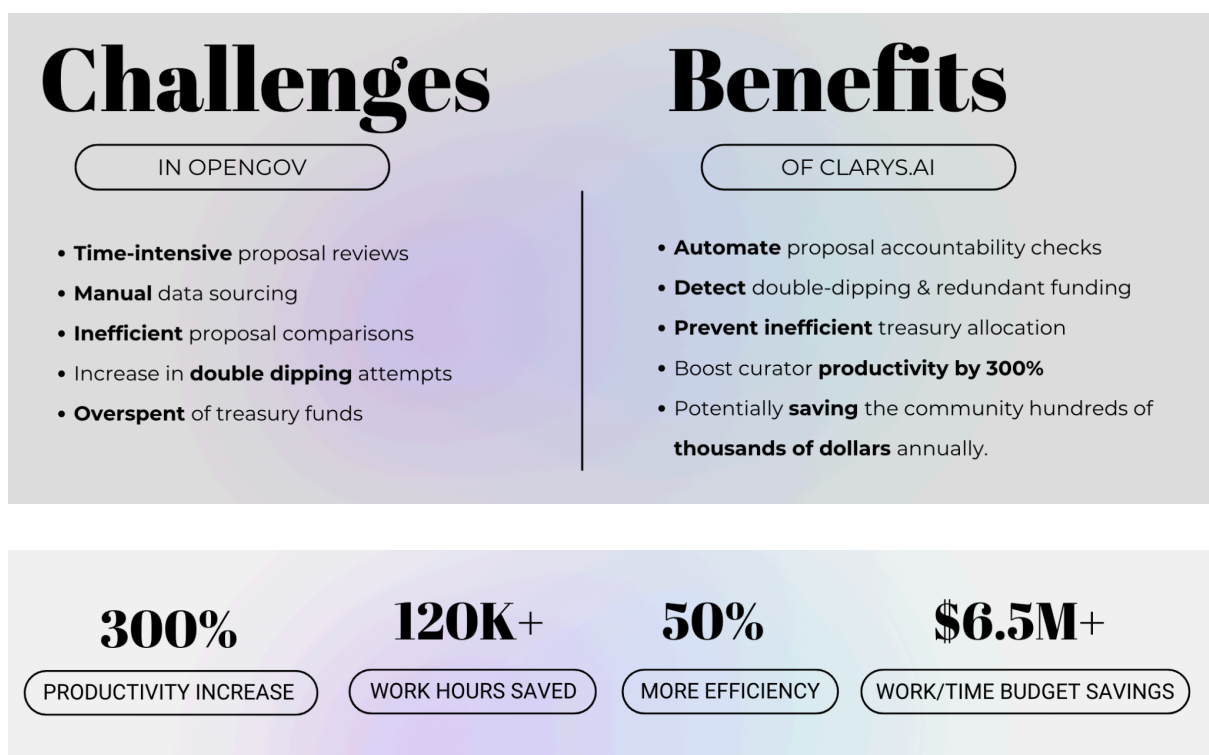
We listened to the community's feedback on our prior referendum: *Phase One must deliver something people can use*. In response, this proposal centers not just on backend improvements but on a tangible, everyday tool for the ecosystem. **Beta v1** ships an interactive explorer with related-proposal graph views, duplication flags, provenance popovers, filters and sortable tables, alongside open datasets and integration hooks for governance frontends. To make quality measurable and model-agnostic, we will publish a transparent **evaluation harness**: 150 canonical governance questions run nightly with accuracy, citation coverage, latency and cost dashboards, comparing open-source and commercial models under schema constraints.

This document details the product, the data and architecture behind it, the implementation plan, and a fully calculated budget for **USD 130,000** to deliver **Beta v1** in approximately twelve weeks.

The first proof of concept version of Clarys.AI was developed as an MVP and its development was funded by Web3 Foundations Decentralized Future grant, you can read the [intro article here](#).

MVP Access - Receive public access to the MVP here:

<https://forms.gle/xSLw2sUTnG1dTioXA>



1. Context & Rationale

Clarys.AI began with a simple observation: people make better decisions when the underlying data is verifiable and the connections between facts are easy to see. The MVP we shared validated interest in a knowledge-graph-based approach to OpenGov, but feedback made two priorities clear. First, Phase One needed to **prioritize a functional product** that complements existing frontends rather than ask the community to fund plumbing they cannot touch. Second, the governance knowledge graph must **blend on-chain and off-chain** signals (comments, discussions, repositories, community trackers) for the explorer to feel complete.

Beta v1 is our response: a focused, public-good release that transforms disparate threads into a unified, queryable and verifiable picture of what's happening and why.

2. Product: Accountability Explorer v1

What it is. A web application that lets voters, delegates, DVs, bounty curators and researchers search proposals, navigate categories, discover related or potentially duplicate requests, inspect provenance with one click and export structured results. It is opinion-light and evidence-heavy: tooltips and popovers point back to sources; duplication badges are explained; graph views help users “see” neighborhoods and histories.

How it feels. Start at any proposal. See its category, mechanism, proposer and beneficiary, and the discussions that shaped it. Click **Related** to open a graph showing adjacent proposals (SPM: same project across mechanisms) and potential duplicates (SPx: same proposal across mechanisms), with reason codes. Hover signals reveal provenance, block height, forum URL, GitHub issue, or comment link, so trust is never blind. Filter, sort, copy a shareable link, or download the slice as CSV.

How it differs from the MVP. The MVP proved the direction; Beta v1 makes it daily-use. The explorer gains the related-proposal graph, duplication badges, provenance popovers, better filters and tables, public APIs, and broad off-chain coverage so context lands in one place. It is also **integration-ready**: an embeddable widget and API recipes help Polkassemblies and SubSquare surface related/duplicate context inline.

3. Data & Sources (Phase One Coverage)

A provenance-first backbone underpins the explorer. Every record carries source metadata (type, URL/block hash, timestamp, retrieval method, checksum) and passes consistency checks in CI. Coverage in Phase One includes:

- **On-chain governance events** and core entities (referenda, bounties, tips, categories, proposers, beneficiaries, mechanisms), with schema versioning and migration scripts.
- **Polkassemblies**: proposal pages and **comments**, as well as off-chain governance discussions tied back to proposals.
- **SubSquare**: proposal pages and **comments**.
- **Polkadot Forum**: threads relevant to governance (e.g., runtime releases, tech-talk).
- **GitHub**: repos and issues/PRs in Fellows/Parity orgs that link to funded work.

- **Bounty portals/pages** where available and relevant to Treasury flows.
- **Reddit:** r/Polkadot and r/Kusama discussions referencing specific proposals or categories.
- **OGTracker:** ecosystem metrics that contextualize initiatives, linked as off-chain references.
- **Kusama DAO** and **OpenDAO** discussion feeds, where accessible.

The objective is not merely breadth but **traceability**. Every ingestion pathway logs transforms and enables replay to ensure reproducibility.

4. Entity Resolution & Duplication Taxonomy

Entity resolution links proposers/beneficiaries across mechanisms and identities, enabling coherent histories and funding views. Duplication is classified in two distinct ways:

- **SPM — Same Project across Mechanisms.** The project is the same but the scope or mechanism differs (e.g., research phase vs. implementation bounty). SPM is **allowed** and marked **Related** with reason codes and context, not flagged as problematic.
- **SPx — Same Proposal across Mechanisms.** The same deliverables are requested in multiple venues, concurrently or serially, without transparent linkage or explanation. SPx is **flagged** with an explanation field, a confidence score and reviewer overrides that leave an audit trail.

The explorer visualizes both via badges and a graph view, so users can distinguish legitimate multi-track efforts from potential double-funding risks.

5. Architecture & Methodology

The system follows a modular, **Phase-Two-ready** architecture while keeping Phase One pragmatic:

- **Ingestion & ETL.** Substrate RPC/WS and scheduled batch/stream jobs (Kafka/Redpanda where needed) feed a dbt-style transform layer. Deterministic tests verify referential integrity and schema invariants. Daily snapshots are produced to

object storage (S3/R2) alongside signed artifacts.

- **Storage & Graph.** Postgres holds OLTP rows and materialized views for APIs. A **Neo4j (lite)** knowledge graph stores relationships across proposals, entities, categories, mechanisms, and off-chain references. A FAISS vector index with Sentence-BERT embeddings supports semantic retrieval; fusion ranking blends semantic scores and graph centrality.
- **APIs.** A typed **GraphQL** API exposes entities and relations; a **REST** API serves snapshots and bulk queries. Keys and rate limits protect stability while keeping access open.
- **Frontend.** The explorer is built in React/Tailwind with server-rendered pages for fast initial paint. The UI emphasizes provenance visibility and low-friction navigation, keeping opinion to a minimum and traceability to a maximum.
- **Evaluation Harness.** A nightly runner executes **150 canonical governance questions** curated from real voter/delegate needs. Outputs are schema-constrained and citation-checked. We measure **accuracy**, **citation coverage**, **latency**, and **cost**, compare **open-source models** with commercial providers, and publish the dashboards and raw CSV/JSON.
- **MCP Pathway (forward-looking).** The codebase will support **Model Context Protocol** patterns for extraction and integration. In Phase Two we plan to evaluate **Phala's MCP service** for privacy-preserving, auditable context brokering once Beta v1 adoption anchors the data backbone.

6. APIs, Datasets & Integrations

Public access is central to the project's value. Every day, the system publishes CSV/Parquet snapshots of key tables and relations. The GraphQL API provides structured reads for frontends and research; the REST API enables bulk export and historical slices. We will ship an **embeddable "Related Proposals" widget** that governance frontends can drop into detail pages to show SPM/SPx context with provenance and reason codes. Integration recipes (queries, payloads, eventing/webhooks or polling) will be documented thoroughly so teams can move quickly without a custom integration cycle.

7. Implementation Plan & Timeline (12 Weeks)

Phase One is organized into three milestones. Each concludes with a public changelog, a dataset published and a short written report summarizing what changed, what shipped and what's next.

Milestone 1 - Schema & ETL (Weeks 0–4).

Finalize Schema v1 with provenance fields and migration scripts. Stabilize the on-chain ingestion path and publish daily snapshots. Stand up the first off-chain connectors: **Polkasassembly/SubSquare comments** and **Reddit**. Release **API alpha** and begin seeding the vector/graph indices. CI starts enforcing integration tests and schema invariants.

Milestone 2 - Graph & Explorer (Weeks 5–8).

Enable the **Neo4j** graph with core relations (PROPOSED_BY, FUNDS_TO, RELATED_TO, DUPLICATE_OF, DISCUSSED_IN). Add fusion ranking (semantic + graph). Roll out **Explorer v1 beta** with search, filters, provenance popovers, duplication badges, and the related-proposal graph view. Expand off-chain with **GitHub**, **bounty portals**, **Polkadot Forum**, and **OGTracker** connectors.

Milestone 3 - Evaluation & Integrations (Weeks 9–12).

Publish **Evaluation Harness v1** with 150 canonical questions and nightly dashboards (accuracy, citation coverage, latency, cost). Harden APIs with keys and rate limits. Ship the **embeddable widget** and **integration recipes** for governance frontends. Close with a public delivery report and a complete dataset dump.

8. Success Criteria

We will consider Phase One successful when:

- The explorer is **available and responsive** (median < 2 s per query; P95 < 5 s) and used by voters/delegates in day-to-day evaluation.
- At least **95%** of indexed proposals are linked to one or more entities (proposer/beneficiary), improving traceability.
- Duplication detection reaches **≥ 70%** correctness on top-10 related/duplicate suggestions in audited samples.
- The evaluation harness shows **≥ 90%** citation completeness across the question set, with dashboards and raw outputs published.
- At least one governance frontend embeds our **Related Proposals** widget or directly uses the APIs soon after delivery, and a handful of builders or researchers adopt the datasets in the first three months.

9. Budget (USD 130,000)

The budget mirrors the working structure of the project and includes explicit resource calculations (hours × rate), fixed operational costs, and modest contingency. All line items are for **Phase One (Beta v1)** only.

9.1 Resource Calculations

Workstream	Hours	Rate (USD/hr)	Subtotal (USD)
Data Engineering & ETL	200	100	20,000
Backend Services & Knowledge Graph	200	110	22,000
Frontend - Accountability Explorer v1	160	100	16,000
Off-chain Connectors (PA/SS/Reddit/GH/Forum/OGT)	120	100	12,000
MLOps & Evaluation Harness v1	120	100	12,000
DevOps & Observability	80	100	8,000
QA & Automation	80	75	6,000
Product & Program Management	167	120	20,000
Community & Documentation	—	—	4,000

Infrastructure (compute, storage, observability; 3 months)	—	—	4,000
Contingency (~6%)	—	—	6,000
Total	—	—	130,000

Notes on calculations.

The resource lines reflect the practical effort to ship Schema v1, multi-source ETL, vector/graph retrieval, a responsive explorer, evaluation harness and integration hooks. Rates are blended per discipline and aligned with prior scoping; the **Product & PM** figure reflects coordination, community updates, and delivery management across three milestones. Infrastructure covers conservative usage for the Beta period (databases, object storage, dashboards). Contingency addresses unforeseen connector variance and minor scope risks without requiring change requests.

10. Governance, Licensing & Community Process

The project is designed to **strengthen** the OpenGov ecosystem rather than centralize it. Core artifacts are open-licensed (MIT/Apache-2.0 code; CC-BY-4.0 data). We will publish a public roadmap and weekly changelogs, run bi-weekly office hours during implementation, and hold a midpoint review to collect feedback. Everything critical to reuse—schemas, APIs, datasets, and the embeddable widget—will be documented with examples and versioned releases.

11. Roadmap (Beyond Beta v1)

Once the Beta is in use, the next wave of work will focus on deeper reasoning and integrations. This includes the **Reasoning Assistant** with multi-agent orchestration (Retriever, Reasoner, Citation-Checker, Policy-Guard), expanded analytics, confidence scoring and uncertainty surfaces in the UI, **MCP-based** extraction/integration pathways, and evaluation of partner-hosted MCP services (including Phala) for privacy-preserving, auditable context brokering. We also plan to bring online additional infrastructure modules referenced in our internal plan (e.g., **Pana** and **Chibi server**) as part of hardening and scaling. These items build on the same provenance-first backbone delivered in Phase One and will be proposed after Beta v1 adoption and community input.

12. The Team

Clarys.AI was founded and is led by Zoé Meckbach, a long-standing and respected contributor to the Polkadot ecosystem since 2019. Zoé has held influential roles including Global Head Ambassador, Bounty curator, Parachain Council Member and Advisor, positions that reflect deep expertise in governance, ecosystem growth and community engagement. Her leadership anchors Clarys.AI in the realities and priorities of Polkadot's governance landscape.

Under Zoé's direction, Clarys.AI's vision is being shaped in close collaboration with a specialized development team comprising experts in both Polkadot OpenGov and advanced AI technologies. Following the successful delivery and approval of the MVP by the Web3 Foundation's Decentralized Futures program, the team was strategically restructured to strengthen its focus on deep technical proficiency in the Polkadot blockchain and robust AI capabilities.

For the beta development phase, expanding the team was a deliberate step to ensure Clarys.AI meets the highest standards of functionality, transparency and user experience for the Polkadot OpenGov community. The enhanced team combines governance-savvy developers, AI engineers experienced in deploying cutting-edge machine learning tools and blockchain specialists with hands-on knowledge of OpenGov mechanisms.

Operating across global locations, the Clarys.AI team is fully dedicated to delivering a comprehensive, production-ready platform with seamless integrations into existing OpenGov infrastructure. **Our team's mission is to empower all Polkadot stakeholders with efficient, secure and intelligent tools for decentralized governance participation.**

The Team



Zoé McFox
FOUNDER

- Polkadot native since 2019
- Head Ambassador & public speaker
- Parachain council member / advisor
- Founder & Curator of Bounty #17
- Founder Clarys.AI
- M.Sc. Cybersecurity & Business

Kris
AI LEAD

- 5+ years AI engineering Lead in industry & research
- Built products in multi-agent systems and computer vision
- Research focuses: Federated Learning and agent memory
- Published in top ML conferences (AAAI, INFOCOM)

Kaus Tamuly
AI ENGINEER

- Ex-Microsoft ML

Ahan

AI ENGINEER

- Ex-Amazon AI

**+ FE, BE, DevOps, UI/UX Design, tech content
Product & Project management**

13. Appendices - MVP Use Cases

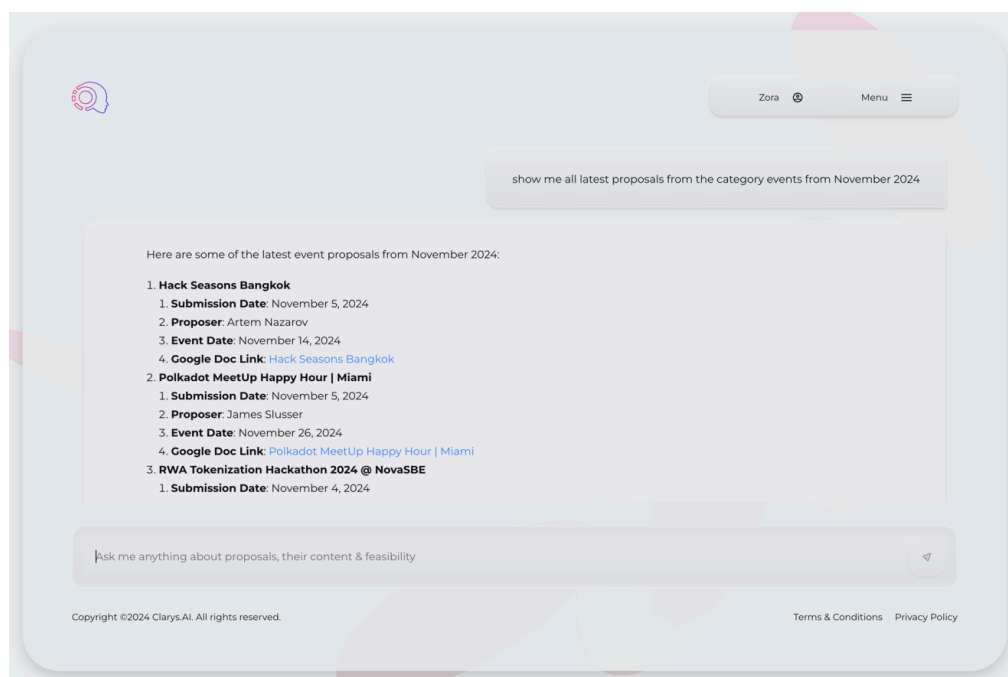
[Clarys.AI](#) is being developed for OpenGov stakeholders, such as DOT holders, bounty curators, DVs, DAOs, etc; but more importantly it is being developed to help onboard a new era of builders and participants into the ecosystem, by solving the information overload and anxiety of these new DOT contributors.

Before making final voting decisions on OpenGov or bounty submission proposals, stakeholders often share common questions and concerns. Typical questions include:

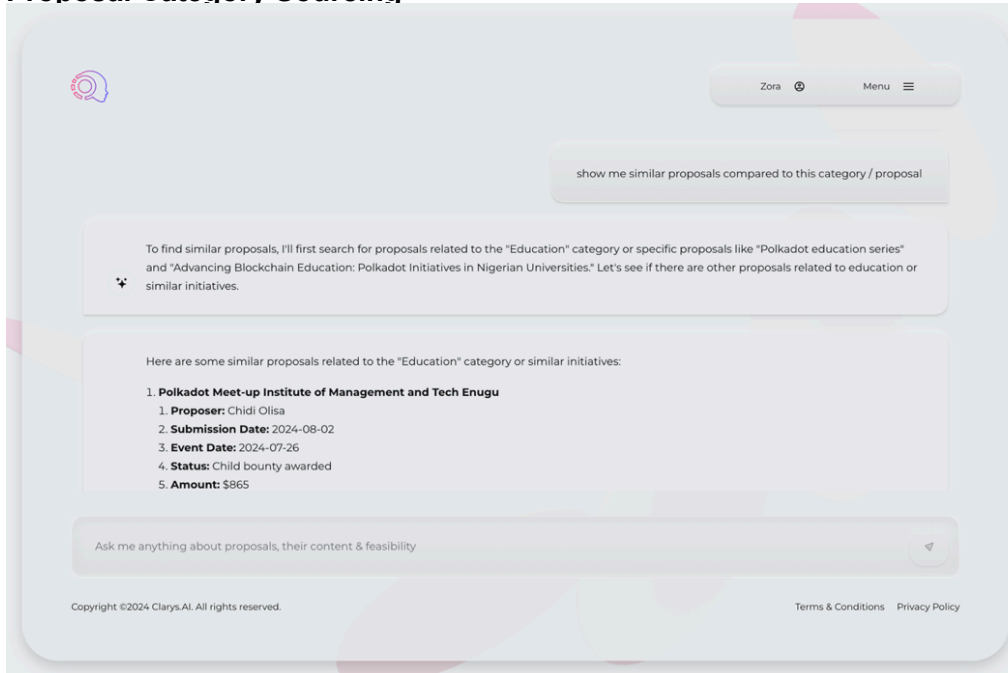
- Is the following proposal accountable and trustworthy?
- Are there similar proposals to this one?
- What proposals exist in a specific category?
- Are there connections between proposals, e.g., via wallet addresses?

One of the use cases covered by [Clarys.AI](#) helps sourcing information as well as supporting the end users' decision-making process such as (on example of an Events Bounty curator):

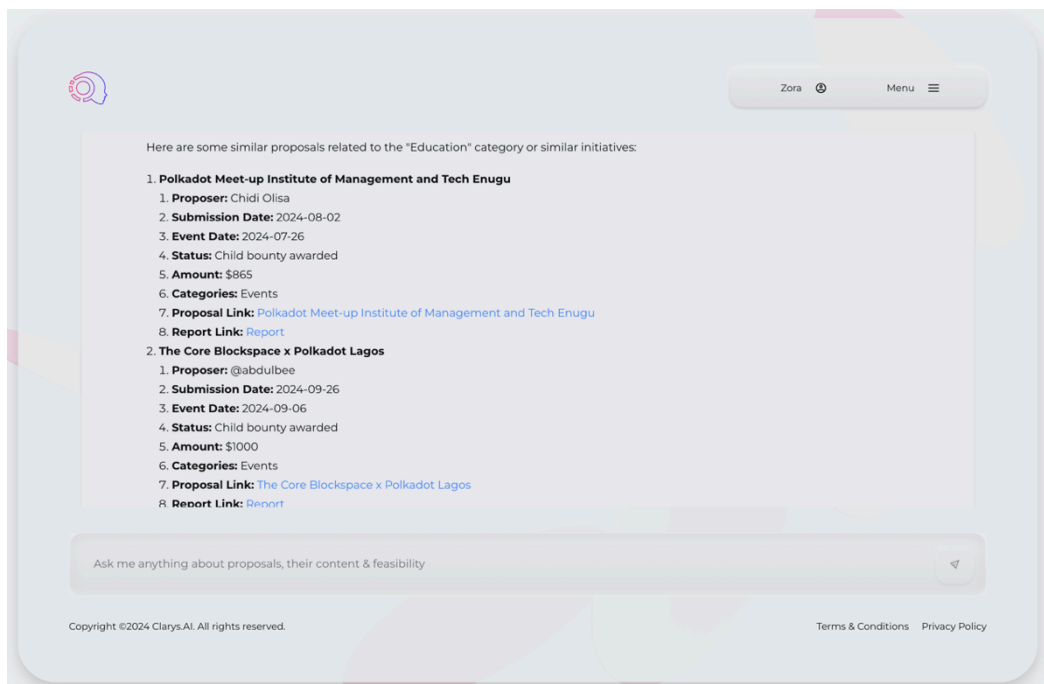
- **Increased transparency** of off-chain Bounty proposal submissions via a user-friendly NLP search function.



- **Proposal Category Sourcing**



- Example prompt to **source on-chain and off-chain** submitted proposals within the category of events proposals to provide the user a better overview of proposals submitted to the events bounty (off-chain process) and the treasury (on-chain process):



- **Double Spending:**

The following example prompts are designed to prevent double-dipping and double-spending within Treasury and separate Bounty structures for specific proposal topics or categories. These prompts enhance data insights, helping to reduce duplication of effort and minimize the risk of double-spending.

