

Proposal for Aztec Block Explorer Grant

Title: AztecScan

Contact Details:

- Email: wietze@kryha.io
- Telegram: [@wietzes](https://t.me/wietzes)

Summary

Following the RFP from Aztec, this document outlines the timeline, team expertise, in/out of scope features, information design, high level system architecture, design look and feel, project milestones, future roadmap and budget requested to build an initial version of a user-friendly Aztec block explorer. The proposed solution is constructed in such a way that it respects Aztec user's privacy, while at the same time providing valuable insights to users.

As a reliable data services platform for privacy focussed networks, with a track record of building high quality and user friendly applications, Obscura is well positioned to build and maintain the Aztec (testnet) block explorer.

Estimated Start and End Date:

- Start: 5th of August 2024
- End: 8th of November 2024

Details

Team Expertise

The team behind this proposal is [Kryha Labs](#), operator of [Obscura network](#). Our team comprises web3 developers and UI/UX designers with vast experience in building solutions and infrastructure for privacy focussed networks like Aleo & Mina. Most notably, our team developed the first [Aleo block explorer](#) in 2019, an [indexer](#) on Aleo with a wide range of API endpoints (see API section) and reliable RPC services for both the Aleo and Mina networks. Additionally, some of the team members were awarded "Best Noir Developer Tool" & "Best Proof-of-Concept of Aztec Sandbox" during the ETH London hackathon [[Github](#)]. Other notable projects include the ZK Liar's dice game Boloney [[Github](#)], as reference implementation of the ZK gaming toolkit [[Github](#)] built on Aleo, and KREAd [[Github](#)] a dApp on Agoric utilising composable NFTs. Visual examples of the dApps can be found below:

[

1600×621 141 KB

](<https://europe1.discourse-cdn.com/business20/uploads/aztec/original/2X/7/7cf369316bd9dd1f8463579a3cc1eb88e96c0b1a.jpeg>)

Kryha Labs - consisting of 19 people - is currently building Obscura Network, a data services platform for web3 builders to access privacy oriented networks like Aztec, Aleo & Mina. Kryha Labs has been working on web3 solutions for the past 8 years. We have a long term strategy to develop the necessary building blocks for the next generation of privacy dApp builders and are committed to do so on Aztec as well.

Detailed team credentials can be found in Appendix 1.

Block explorer structure and features

Homepage

- Latest 20 blocks
- Latest 20 transactions
- Display and visualise key metrics like transactions over Time, latest (safe) block, gas fee estimation, etc.
- Main navigation and search for block number or hash, transaction hash, contract address, contract class id.

Search result page or modal (in case of multiple hits)

- Block(s), transaction(s), or contract(s) based on query

All blocks list page

- List of all blocks with their number, hash, status and timestamp
- Key metrics
- Pagination

Block details page

- The block details according to Aztec Block Explorer RFP Appendix 1, among others fee details and a list of block related transaction effects.

All transactions list page

- List of all transactions with their hash, status, timestamp
- Key metrics
- Pagination

Transaction details page

- The transaction details according to Aztec Block Explorer RFP Appendix 1, among others transaction effects, status and fee details.

All contracts list page

- List of all contracts with their address and class id
- Key metrics
- Pagination

Contract details page

- The contract details according to Aztec Block Explorer RFP Appendix 1, among others public and private code, logs, related public transactions, class and version details.

System Architecture

A web-based application that allows users to browse and query Aztec Network data. It provides a comprehensive interface for tracking transactions, viewing addresses and exploring blocks.

[

1196×1257 72.1 KB

](https://europe1.discourse-cdn.com/business20/uploads/aztec/original/2X/4/4c0e91e61a053e0f34d8df8d5f636b94058e19c4.png)

Back-End

The back-end is involved in listening to new data coming from Aztec & Ethereum nodes, parsing the data accordingly and storing it for later consumption by the front-end client. We consider the components below as the groundwork for a future Indexer.

Components:

Data Ingestion Layer (Aztec.js, Ethers.js):

- Listeners

: connect to Aztec & Ethereum nodes to actively listen for new blocks and transaction status.

- Parsers

: encode block & transaction data into the corresponding schema.

Storage Layer (NoSQL, Redis):

- Archive Data

: block and transaction data for long term storage.

- Logs

: operational logs for monitoring and alerting services.

API (API Gateway, Express, OpenAPI):

- REST API

: mapping resources to endpoints for clear and easy querying.

- WebSocket

: to establish a persistent connection between the client and the server, allowing to push updates to the client as soon as new information becomes available.

Infrastructure:

- Docker

: containerizing applications.

- Kubernetes

: orchestrating containers.

- Nginx

: serving the web application and acting as a reverse proxy.

- AWS

: Hosting of deployed versions

Front-End Client

A Progressive Web Application (PWA) designed to be highly responsive and user-friendly. It will offer an intuitive interface that is accessible on various devices, including desktops, tablets, and mobile phones. The design will closely align with Aztec's branding, ensuring a cohesive and recognizable user experience.

Components

- Framework

: React with Next.js for server-side rendering.

- Styling

: Tailwind CSS, rapid and efficient styling.

- State Management

: Zustand, managing global application state.

- Data Fetching

: React Query, simple caching and background updates.

- API Integration

: Fetch for base communication with the back-end.

- Anonymous Metrics

: base analytic tracking to monitor application health (request count and status, up-times, node sync, etc.). User IP will not be tracked and user data will not be collected by the application.

Look & Feel teaser

Design vision

Since block explorers operate so close to the core of their blockchains, we believe they should be recognisable as such. That's why we will stay very close to the Aztec brand and overall look & feel. We'll be using the same font, colours, and - where possible - a similar breathable design. But since this is a tool (as opposed to the Aztec website for example) it will have a higher information density. We'll follow existing explorer patterns where possible, but also emphasise Aztec's uniqueness with great UX and UI around the public versus private content.

Disclaimer: the visualisation below is just an example of what the explorer interface could look like. A more comprehensive design will be made during the first phase of the project.

[

1600×943 199 KB

](https://europe1.discourse-cdn.com/business20/uploads/aztec/original/2X/8/8aa18bda68b0ce36f24184b1a22428f2f7d7e288.jpeg)

Grant milestones and roadmap

Milestone 1: Design and Initial Setup (Week 32-34)

- Design: comprehensive UI/UX design
- Front-end: main layouts and base components
- Back-end:
- Complete architecture design
- Data definitions (DB schemas)
- Initial infrastructure setup
- Complete architecture design
- Data definitions (DB schemas)
- Initial infrastructure setup

Milestone 2: Core Functionality Development (Week 35-38)

- Back-end:
- Ingestion layer (listeners & parsers)
- Data storage layer (Archive Data)
- Ingestion layer (listeners & parsers)
- Data storage layer (Archive Data)
- Front-end:
- Block components
- Transaction components
- Block components
- Transaction components

Milestone 3: Core Functionality Development (Week 39-41)

- Back-end:
- REST API
- Websocket integration
- Contract parsing
- REST API

- Websocket integration
- Contract parsing
- Front-end:
- Contract components
- Webworker integration
- Contract components
- Webworker integration

Milestone 4: Advanced Features, Testing & Release (Week 42-45)

- Back-end:
- Monitoring services
- Search optimization
- Monitoring services
- Search optimization
- Front-end:
- Search integration
- Search integration
- QA:
- UAT
- Stress test
- UAT
- Stress test

Longer term roadmap

As a data services provider to privacy focused networks, it is strongly our interest to further develop and maintain the Aztec explorer, both for the current and future testnet(s) and mainnet. We are therefore committed to continue improving the explorer as described below.

Below is a full view of what we envision the Aztec Explorer could become, including an Indexer for extended reliability, and integration of AztecJS on the client side for local decryption of private data.

[

1600×1343 149 KB

](<https://europe1.discourse-cdn.com/business20/uploads/aztec/original/2X/9/9f4a7218818bbc58a120281c968a4aa880d28bcf.png>)

Roadmap components (to be prioritised):

- Wallet integration: giving the user the ability to locally decrypt private information associated with the account using their view key, keeping private data accessible to the user/owner only.
- API Gateway: with the Indexer in place, we can directly serve other third-party applications through gRPC endpoints and even provide a GraphQL interface for multiple different front-end clients.
- Aztec Indexer: facilitating efficient data retrieval, organisation, and presentation, the indexer is responsible for processing and organising chain data to ensure fast and accurate querying.
- Auto-scalable Ingestion Layer: using Kafka to further decouple the ingestion and serving of data, services can easily scale independently.

- Search Engine: Elasticsearch, advanced search capabilities.
- Cache: Redis to optimise search experience, speeding up repeated queries and reducing database load.
- Independent Monitoring Service: with increased complexity in the overall system the need for a central logging service becomes necessary. Also the inclusion of an alerting system that can notify the relevant parties of any degradation of the system.
- Private Data: we want to offer our users the ability to connect their wallets and access their private data through the client. For this we'll leverage Aztec.JS to access the users view keys locally and retrieve the relevant data.
- CDN: serving the front-end code from a content delivery network with nodes across the globe to ensure minimum round trip delays and protection against most common attacks.
- Network switch: switch between different Aztec networks to view block and transaction data for a specific network.
- Sorting and filtering: sort and filter list pages and search results by tag, type, date, etc.
- Note Discovery Schemes: Implement support for various note discovery schemes to help users find their notes efficiently.

It is in both Aztec's and Obscura's interest to maintain and build out the explorer for the long term and Obscura is committed to do so. Currently, Obscura provides data services like RPCs and APIs as a SaaS offering with a Freemium model. We aim to add additional data services for Aztec to our current offering to be able to keep the block explorer free to use for all users.

Grant Amount Requested: \$42,000

14 weeks in total * \$3000 per week

Grant budget rationale

The full requested budget will be allocated to cover the costs of the three team members working on design, FE/BE development and testing of the block explorer. A breakdown of the costs by milestone and time effort can be found in the table below.

Milestone

Time (weeks)

Budget

1

3

\$9,000.00

2

4

\$12,000.00

3

3

\$9,000.00

4

4

\$12,000.00

Total

\$42,000.00

The infrastructure and maintenance costs will be covered by Kryha Labs/Obscura.

Questions:

1. It would be great to meet the Aztec team in person if we're selected to build the block explorer. Can we meet you at your office or at an event (e.g. EthCC?).
 2. Are there any specific chain data visualisation requirements or preferences from the Aztec community?
 3. Can the development team get early access to any Aztec testnet resources to better align with network updates?
 4. Does the indexer need to be open-source? Apache 2.0?
-

Appendix I - Team credentials

Wietze

- Technology Director

[\[Github Kryha\]](#)[\[Github Wietze\]](#)

Previous experience

Wietze is a seasoned blockchain engineer with extensive experience leading enterprise projects at Kryha. He has showcased web3 technology's value on Ethereum-based chains. His expertise includes developing the block explorer and chain indexer on Aleo's network, and leading projects such as the ZK game [Boloney](#) and the composable NFT dApp KREAd on Agoric.

Currently, Wietze oversees all engineering activities for Obscura, leading the Core Engineering team. He is responsible for software design, architecture, development, and deployment across ZK chains, including Aleo and Mina, focusing on enhancing data privacy and security.

Competences:

- 8 years professional programming experience working on apps, dApps and SaaS solutions
- 7+ years Web3, software architecture, JavaScript/TypeScript experience
- 6+ years containerisation experience (Docker, K8s)
- 5+ years technical (project) management skills and microservice solutions, focussed on event driven architectures
- 4+ years CI/CD / devops experience
- 3+ year working with ZK chains (focus on Aleo and Leo)
- Novice level of Rust experience, mostly focussed on review and edits to Aleo VM/OS
- Study background in Artificial intelligence
- 2+ years experience as a Teaching assistant

Wouter Middendorf

- Design Director

[\[LinkedIn\]](#)

Wouter has 13+ years of experience working in product design, development, and innovation at various agencies and startups of which over 7 years at director/management level. In recent years, Wouter has designed (enterprise) web3 products from the ground up. Most notably, in relation to this RFP, Wouter has made the UI/UX designs of the Aleo block explorer. This block explorer is then built by our development team. Next to this, he has designed the initial version of a [Battery Passport](#) for the Global Battery Alliance, demonstrated at the World Economic Forum. Furthermore, he designed [Re|Source](#), a digital platform for traceability of minerals in the Tesla supply chain, based on decentralised and ZK technology.

Competences and education:

- 13+ years in User Experience (UX)/ User Interface (UI) design
- 7+ years of Product Management experience

- MSc. Design for interaction (cum laude)
- MSc. Design research for interaction

Filip Harald

- Customer Engineering Lead

[\[Github\]](#)[\[LinkedIn\]](#)

With 8+ years of experience building SaaS with micro-service architecture, Filip has gained expertise in creating robust platforms and managing customer expectations. Within web3, Filip has been quickly catching up with the relevant tech the past 2 years. The thing he's most proud of is his contribution to BuildGuild, where he built the educational demo-app that teaches the basics of Zero-Knowledge: [\[Github\]](#). The latest achievement he made, together with his colleagues, is winning the prizes "Best Noir Developer Tool" & "Best Proof-of-Concept of Aztec Sandbox" during ETHLondon hackathon with a ZK-voting: [\[Github\]](#).

Competences:

- 8+ years professional programming experience on apps, distributed systems, SaaS solutions, IoT, JavaScript/TypeScript, Docker and K8s
- 6+ years software architecture, CI/CD/devops, developing microservice solutions, focused on event driven architectures and for distributed systems (edge computing).
- 4+ years technical (project) management skills
- 1+ year experience working with ZK-tech
- 0.5 year in Rust (hobby-projects)
- Study background in Computer Science
- 2+ years as a Teaching assistant

Privilege Mendes

- Frontend engineer

[\[Github\]](#)[\[LinkedIn\]](#)

Privilege, a former Electronic Design Engineer now Frontend Developer, has 6+ years of professional programming experience. He leads frontend architecture and solution development for both end-user and admin interfaces.

Currently, as the lead frontend developer at Kryha, he is implementing the Obscura Platform and developed the KREAd frontend [\[KREAd Github\]](#), a dApp utilising composable NFTs on Agoric, in four months. His preferred tech stack includes Typescript, React, NextJS, and Tailwind CSS. Outside of work, he creates engines for WeRoad, PWAs, and React Native mobile apps.

Competences:

- 6+ years of professional programming experience on web app UI, cross-platform mobile-first app UI, embedded systems, React, JavaScript/TypeScript, Python, Tailwind, HTML, and CSS.
- 2+ years of Docker experience, including managing containers in a development environment.
- 1+ year of Web3 and Dart experience, as well as Vue (hobby projects).
- Bachelor's Degree in Computer/Electronic Engineering.