COMMUNITY DEVELOPMENT AND EDUCATIONAL IMPROVEMENT SCHEME

CDEIS Minerals

White Paper

An implementation-led overview of the CDEIS platform, its financial and learning services, system architecture, governance, operating limits, and community purpose.

SYSTEM	TOKEN REFERENCE	CULTURE
Learning, commerce, research, governance	**Gold-price benchmark; no redemption claim**	**Inclusion, transparency, learning, support**

Document Purpose

This white paper explains what CDEIS does, how the system works, and the values that guide the platform. It is intended for users, contributors, sellers, teachers, administrators, partners, and community stakeholders.

Generated 2026-11-20 | CDEIS Lab

WHITE PAPER STRUCTURE

Table of Contents

The document is organized from purpose and daily operation to governance, culture, and future direction.

01 Executive Summary

02 Vision and Purpose

03 Token Model

04 User Identity, Wallets, and Account Recovery

05 Dashboard and User Experience

06 Education System

07 Commerce, Sellers, and Contributors

08 Community and Communication

09 Governance and Administration

10 Advanced System Architecture

11 Financial Core and Risk Controls

12 CDEISLAB, Research, and Knowledge Workflows

13 Live Classroom and Peer Media

14 Operations, Data Protection, and Service Limits

15 How the System Works

16 Security and Reliability

17 Culture of the System

18 Roadmap and Future Direction

19 Conclusion

01

Executive Summary

CDEIS (Community Development and Educational Improvement Scheme) is an integrated web platform spanning accounts, an application-level token ledger, education, research, community communication, commerce, and administration. Its current architecture connects browser applications to Node.js and PHP APIs backed by SQLite, with selected services implemented in both runtimes and advanced modules requiring Node.js.

The platform has grown beyond a basic wallet and portal. It now includes server-enforced financial simulation with risk controls, source-attributed research records, curriculum-driven tutoring, protected learning documents, teacher-led live classrooms, community and commerce workflows, operational audit records, and responsive user interfaces. These capabilities remain distinct services with explicit permissions and operational limits.

This paper describes implemented application behavior and identifies important boundaries. A gold-price reference used for token pricing does not establish physical gold reserves or a right of redemption. Token records are stored in the application database; they are not settled by an independent public blockchain network. Product, financial, education, and AI capabilities must be understood within those limits.

02

Vision and Purpose

CDEIS exists to support community development and educational improvement through practical digital infrastructure. The purpose is not only to run a token ledger, but to connect that ledger to learning, commerce, communication, proof of account, and user empowerment.

The platform supports online participation through shared identity, learning, communication, and record-keeping workflows. It is intended to help users learn, collaborate, access community services, and inspect their activity through a single set of connected applications.

KEY POINTS

- Keep application records of accounts, token movements, learning activity, and privileged operations.
- Provide accessible registration, login, wallet, and dashboard tools.
- Support education through a student portal, teacher applications, learning items, submissions, and classroom communication.

- Support commerce through product listings, seller applications, checkout, token pricing, and reward splits.

- Support trust through audit events, account documents, transaction records, explicit service boundaries, and operational health checks.

03 Token Model

The application models a stated supply of 1,000,000,000,000 CDEIS units and records balances and transfers in its SQLite-backed service state. Reserve and system wallets distribute units through application workflows. This is a centralized application ledger with blockchain-inspired records; it does not currently provide public-chain consensus, independent validators, public-chain finality, or proof of token ownership outside the service.

A configurable product-pricing reference maps one CDEIS unit to one gram of gold and converts a gold quote into local-currency estimates using foreign-exchange data. This is a pricing convention only. The software does not establish that each unit is backed by allocated physical gold, guarantee redemption, or provide a legally enforceable claim on a commodity. Live quotes depend on external providers and may be unavailable or stale.

KEY POINTS

- Configured supply: application workflows do not provide ordinary users with a mint operation.

- Service-controlled records: balances and transaction history are maintained by the application database and its APIs.

- Wallet identifiers: wallet addresses identify accounts in the platform; they are not proof of an external chain account.

- Price reference: gold and foreign-exchange quotes support estimates; they do not imply physical reserves or redemption.

- Operational dependency: backups, access control, database integrity, and service availability govern the records.

04 User Identity, Wallets, and Account Recovery

Users register with profile details and authenticate with email and password. Password verification supports a legacy salted hash format and a scrypt format for upgraded records; operators must protect the password salt and keep HTTPS enabled. Authenticated sessions are used by server-backed account and learning services.

A platform wallet identifier and wallet-operation credentials are associated with account workflows. These application records should not be confused with a hardware-backed or externally audited custody system. Users and operators should protect account access and treat any self-key material as sensitive.

Account recovery and profile updates use the fields and checks implemented by the deployed version. Recovery information is personal data and must be collected only when necessary, protected in storage, and not treated as a substitute for stronger identity verification.

KEY POINTS

- Registration page: creates the account and wallet details.
- Login page: authenticates and opens the dashboard.
- Forgot password page: resets access only when recovery details match.
- Dashboard profile: lets users maintain phone, father name, date of birth, country, gender, and profile picture.
- Session controls: password changes and recovery workflows must invalidate sessions where the active implementation supports it.
- Privacy boundary: profile and recovery data require access controls, retention rules, and secure hosting.

05 Dashboard and User Experience

The user dashboard brings together profile and account records, wallet views, token transfer and purchase interfaces, commerce, statements, support, chat, research, and learning entry points. Its functions are backed by server APIs and SQLite records where available; some pages remain standalone or browser-oriented tools.

The application is responsive and provides dedicated mobile layouts. A shared navigation system connects the student portal, CDEISLAB, library, markets, classroom, community and commerce pages. Availability still depends on the deployment serving the required API routes, database and static assets together.

KEY POINTS

- Profile and avatar management.
- Wallet balance lookup and wallet security key retrieval.
- Token buy and peer transfer flows.
- Account statement, proof of account, and transaction history PDF downloads.
- Marketplace checkout and order status visibility.
- Student learning submissions and classroom participation.
- Private messaging and chat notifications.

06

Education System

The education workspace combines learner accounts, course access, enrollment workflows, assignments, teacher applications, structured curriculum, classroom communication and progress records. Current curriculum modules include authored ZIMSEC study support, minerals and aviation foundations, and a teaching library with source-aware CDEISLAB handoffs.

Learning records are account-scoped where implemented. Quiz scores and study tracks indicate activity or mastery within the application; they are not external qualifications or professional licences. Teachers require administrator review and course assignment before leading a live period.

KEY POINTS

- Student portal and course overview.
- Enrollment requests and application review.
- Assignments, exercises, tests, exams, and project submissions.

- Classroom creation, membership, and classroom chat.
- Teacher application and learning content contribution.
- Account-level learning progress, study schedules and review workflows.
- Authenticated access to supplied study PDFs and extracted page text.
- Clear distinction between course participation, assessment results and recognized qualifications.

07

Commerce, Sellers, and Contributors

The commerce layer provides product discovery, seller applications, wallet-priced checkout, order status, fulfillment records, and administrative review. Price calculations may use the configured gold and currency quote service; that quote remains an estimate and does not establish token redemption or physical backing.

Contributor records can represent teachers, sellers, affiliates and partners. Where configured, order settlement records apply platform-defined reward shares. External payment providers, tax handling, legal seller obligations and physical delivery remain operational responsibilities outside the software ledger.

KEY POINTS

- Public store and product listing pages.
- Seller application flow.
- Product checkout with token conversion.
- Reward splits for sellers, affiliates, and platform operations.
- Order review, fulfillment, cancellation, and admin notes.
- Contributor profiles for teachers, sellers, creators, and partners.

08 Community and Communication

The community layer includes private messaging, community reports and status interactions, classroom chat, and browser-based peer communication features. Text records are handled through application APIs. Audio/video media uses browser WebRTC and requires HTTPS or localhost, microphone/camera permission, compatible browsers and reachable ICE servers.

Peer media does not pass through a central recording service in the live classroom design. Direct connectivity varies by network; restrictive NAT and firewalls require a correctly configured TURN relay. The deployment must verify calls across its intended networks before presenting calling as reliably available.

KEY POINTS

- Private direct messaging.
- Classroom chat and learning conversation.
- Community posts, comments, reactions, ratings, shares, and gifts.
- WebRTC call setup with explicit browser capture permissions and network-dependent connectivity.
- Support channels for payment proof and account help.

09 Governance and Administration

The admin area is the control center. It supports user review, account updates, platform configuration, integrity reporting, audit visibility, product management, order review, student portal administration, APK release publishing, and operational analytics.

Governance is built around visibility and traceability. Important actions create audit records, and admin access is protected so platform controls are only available to authorized operators.

KEY POINTS

- User listing, creation, editing, and deletion.

- Admin configuration and system policy controls.
- Audit events for authentication and privileged operations.
- Application-ledger integrity diagnostics.
- Product and order administration.
- Student portal and classroom administration.
- APK release registry and official download management.
- Metrics and SLO-style operational visibility.

10 Advanced System Architecture

CDEIS is delivered as a multi-page web application with a shared Node.js HTTP service, an API client that handles root and subdirectory deployments, and a PHP/PDO compatibility layer for selected workflows. Persistent application records use SQLite. The browser provides responsive interfaces; authorization, validation and mutations are enforced by server routes for protected capabilities.

The codebase is organized into route modules and domain libraries for identity, community, commerce, financial simulation, academic services, teaching and live classrooms. Static assets and APIs must be deployed from a compatible release. A PHP-only host cannot provide every Node-only capability, and deployments must not serve SQLite files, private configuration, source code, backups or protected curriculum documents publicly.

KEY POINTS

- Node.js 22.13 or newer is the documented runtime for the full service.
- SQLite provides durable service state; multi-instance shared writes are not a supported scaling pattern.
- Selected core operations have Node and PHP implementations; advanced classroom and study services require Node.

- The API client supports same-origin and subdirectory routing with a PHP fallback where implemented.

- HTTPS, persistent storage, protected file rules and runtime-compatible assets are deployment requirements.

11 Financial Core and Risk Controls

The Financial Core is a separate signed-in paper-trading module. It creates a virtual USD account, accepts manually entered long and short positions, applies server-side risk checks, records closes and reviews, and stores balanced integer-cent demo postings. Idempotency receipts prevent an accepted browser retry from posting the same operation twice. Account data is isolated by authenticated user.

Its controls are product rules for a virtual simulation, not financial advice or live brokerage controls. It has no broker execution, external deposits or withdrawals, custody, live mark-to-market, margin engine, spread/slippage simulation, or independent strategy validation. The existing CDEIS token wallet is not used as the paper account.

KEY POINTS

- Maximum planned risk per position: 1% of realized demo balance, including entered costs.

- Maximum total planned open risk: 3% of realized demo balance.

- Minimum net planned reward/risk: 1.5.

- Daily, weekly, consecutive-loss and historical drawdown rules can block new entries; closes remain available.

- Research observations retain source, timestamp and user-recorded provenance; they are not automatically verified or used to train a model.

- The ledger is a demo equity record, not custody or customer-liability accounting.

12 CDEISLAB, Research, and Knowledge Workflows

CDEISLAB combines authored teaching content, account-scoped learning sessions, approved shared knowledge, document and curriculum context, and research-oriented responses. When an external AI provider is configured, selected teaching requests may use it; authored curriculum and deterministic fallbacks remain separate paths. The system does not claim to fine-tune model weights from learner activity.

Research and market-analysis interfaces preserve source and data-status distinctions. Market prices can be stale, unavailable or user-supplied. Signal analysis is not order execution. AI-generated explanations and proposed setups are informational outputs that require user review; deterministic server rules govern supported financial-core actions.

KEY POINTS

- Knowledge records can be searched, reviewed and curated by authorized administrators.
- Curriculum handoffs retain lesson and source context for the tutor.
- Source-attributed observations are marked user-recorded unless separately verified.
- Paper-trading and research workflows do not submit orders to a broker.
- External model keys and data-provider credentials belong in protected server configuration.

13 Live Classroom and Peer Media

The live course classroom coordinates room membership, teacher assignment, session state, presentation floor, exercises, feedback and short-lived signaling through server APIs and SQLite tables. Students receive media by default; a teacher grants a student the floor before their browser sends a presentation. Ending a period or losing the teacher heartbeat closes the active session.

Audio and video are browser-to-browser WebRTC streams. Each presenter sends a stream to each listener, so network upload capacity limits practical class size. The implementation is not a recording archive or a large-scale media server; larger lectures require an SFU and load testing. A configured STUN/TURN service is necessary for reliable use across restrictive networks.

KEY POINTS

- Only approved and course-assigned teachers can lead the configured course.
- Room state, membership, floor changes and exercise visibility are checked by the server.
- Signaling data expires and the classroom does not retain audio/video recordings.
- Screen sharing and media capture require user permission and a secure browser context.
- Real-network reliability and screen-picker behavior must be checked on the target deployment.

14 Operations, Data Protection, and Service Limits

The service exposes health and operational diagnostics, records selected security and administrative events, and includes SQLite backup and restore procedures. Production operation depends on maintaining the Node runtime, database volume, HTTPS proxy, secrets, backups, update process and external provider configuration. A successful local check does not certify a production deployment.

The platform stores personal profiles, account activity, learning records and operational logs. Operators should define lawful purpose, access roles, retention, incident response, backup protection and deletion handling for each deployment. Feature availability and legal obligations vary by jurisdiction and by the real services offered.

KEY POINTS

- Use one SQLite-writing application instance and keep the database on persistent storage.
- Keep environment files, database files, backup archives and protected source material out of public web paths.
- Set unique production secrets, HTTPS-only cookies and the exact database-compatible password salt.
- Configure and monitor external email, AI, gold-price, foreign-exchange and TURN providers only where used.

- Run release checks against isolated data and validate restore procedures before relying on backups.

15 How the System Works

A browser sends requests to an application API. The API authenticates the user, checks route-specific permissions, validates the request and applies a domain operation. Successful changes are persisted to SQLite and returned to the relevant dashboard. Some financial writes group account changes, ledger postings, idempotency receipts and audit events into one database transaction.

The same pattern connects identity, wallet operations, learning, commerce and administration, but these domains have different records and authorization rules. A dashboard combines their views; it does not turn them into one financial account or one external blockchain.

KEY POINTS

- Identity record: establishes the authenticated account used by protected APIs.
- Token ledger: records application-level balances and movement subject to service controls.
- Learning record: stores course access, progress, submissions and classroom work where implemented.
- Commerce record: stores product orders, configured reward splits and fulfillment status.
- Operational record: captures selected audit events, health information and release controls.

16 Security and Reliability

The server includes password-hash verification, authenticated sessions, privileged-route checks, request and payload limits, input validation, same-origin checks for selected writes, rate limiting, audit events and integrity diagnostics. Security depends on production configuration, HTTPS, secret protection, access controls, safe deployment rules and timely updates; no software feature alone guarantees security.

Operational tooling includes health endpoints, request identifiers, metrics, graceful shutdown, backup and restore procedures, contract checks, unit tests and browser flows on isolated data. SQLite currently requires a single active writer service instance and reliable persistent storage.

KEY POINTS

- Password salts, session secrets and provider keys must remain private and deployment-specific.
- Authentication and authorization are enforced by API handlers for protected workflows.
- Sensitive account, wallet and recovery data require HTTPS and least-privilege operational access.
- Rate limiting protects API endpoints from abuse.
- Audit events preserve sensitive operational history.
- Backups and restore procedures support recovery of platform records.
- OpenAPI contract and isolated integration checks help validate supported API behavior.

17

Culture of the System

The culture of CDEIS is community first. Technology is treated as a bridge, not a barrier. The system is built for people who may need education, digital identity, commerce, wallet access, and support in one place.

The culture values transparency because balances, transactions, supply, account documents, and audit events are visible in structured ways. It values education because learning workflows sit beside wallet and commerce tools. It values inclusion because the app is web-based, installable, and prepared for Android APK distribution. It values accountability because privileged actions and wallet movement leave records.

CDEIS also values practical support. The site provides phone, email, and WhatsApp channels because a community system must serve real people, not only software processes.

KEY POINTS

- Inclusion: rural and urban users should be able to participate.

- Education: learning is part of the economy, not separate from it.
- Transparency: token movement and account records should be explainable.
- Accountability: admin and wallet actions should be traceable.
- Practical innovation: technology should solve everyday community problems.
- Human support: users need real support channels, not only automated screens.

18 Roadmap and Future Direction

The platform is a substantial application system, while production readiness remains deployment-specific. The next engineering phase should prioritize reliable account recovery, independent security review, privacy and retention controls, repeatable releases, production monitoring, provider failover, and realistic capacity validation before adding higher-risk financial or identity services.

Longer-term options include a separately audited ledger architecture, stronger financial-data precision and reconciliation, scalable media delivery, formal learning assessment governance, and integrations with approved external providers. Each expansion requires its own threat model, operating owner, data policy and evidence of readiness.

KEY POINTS

- Independent security assessment and documented incident response.
- Privacy, retention and account-recovery controls aligned with each deployment.
- Capacity-tested database and media architecture before multi-instance growth.
- Reconciliation and evidence controls for any external payment or settlement service.
- Clear source, licensing and review workflows for curriculum and research.
- Mobile distribution and accessibility improvements with maintained release evidence.

19 Conclusion

CDEIS has developed into an integrated platform for application-led token records, education, research, simulated finance, commerce, community communication, administration and operational support. Its advanced capabilities are grounded in modular APIs, account-scoped workflows, SQLite persistence, risk controls, audit records and authored curriculum.

The system must be presented with honest boundaries: a gold-price reference is not proof of gold backing; application-ledger records are not public-chain settlement; simulated finance is not brokerage; learning progress is not professional certification; and peer media depends on network and deployment configuration. Responsible operation means making those distinctions clear while continuing to improve reliability, security, accessibility and community value.