

Y8 Platform

A Holographic Wallet, VR/XR Space, and Premium Financial Command Center

Technical White Paper & Pitch

Jason Sprouse · June 2026

Imagine your wallet is a room.

You step into *The Beach* — a holographic XR space. Your balances float beside you. Your AI agents greet you. You sign a bank payment with a gesture. You mint a proof that your body was here, at this moment, in this space. Your on-chain settings remember your preferences across every surface: phone, browser, headset.

This is the Y8 Platform: the first full-stack **holographic financial OS** built on a wallet-native, agent-driven, XR-ready architecture.

Abstract

The Y8 Platform is a four-application suite that converges holographic spatial computing, Web3 wallet identity, and premium financial services into a single coherent experience. Its four pillars are: *y8-app*, a holographic wallet and agent marketplace deployed as a Progressive Web App; *the-beach*, a Babylon.js XR/VR immersive space with real-time multiplayer and a spatial financial interface; *y8-premier*, a premium financial command center integrating live bank data (Plaid), stablecoin-to-crypto swaps, Rafiki Interledger global payments, and crypto income tracking; and *y8-server*, a NestJS agentic API powering intelligent agent routing, SIWE authentication, and real-time WebSocket coordination. Across all four surfaces, the platform introduces two novel on-chain primitives deployed on Base Sepolia: *Y8Settings*, a per-wallet on-chain preference store that persists user state across 2D, XR, and mobile surfaces; and *Y8ProofOfEntropy*, an ERC-721 NFT protocol that mints device-liveness proofs from multi-sensor telemetry (GPS, accelerometer, network, battery), creating a cryptographic bridge between the physical world and the holographic space. All applications are installable as Progressive Web Apps and deployed on Fly.io. The platform is designed as the financial and identity infrastructure layer for the next generation of XR-native commerce, PPP corridor operations, and AI-agent-driven service delivery.

1 Platform architecture overview

The platform exposes a single wallet identity—a Base-chain address authenticated via Sign-In with Ethereum (SIWE)—across all four surfaces. Whether you are viewing your accounts on a desktop browser in *y8-premier*, walking through *the-beach* in a VR headset, or glancing at your *y8-app* PWA on a phone, your settings, balances, and agent interactions are all anchored to the same on-chain identity [1, 2, 3].

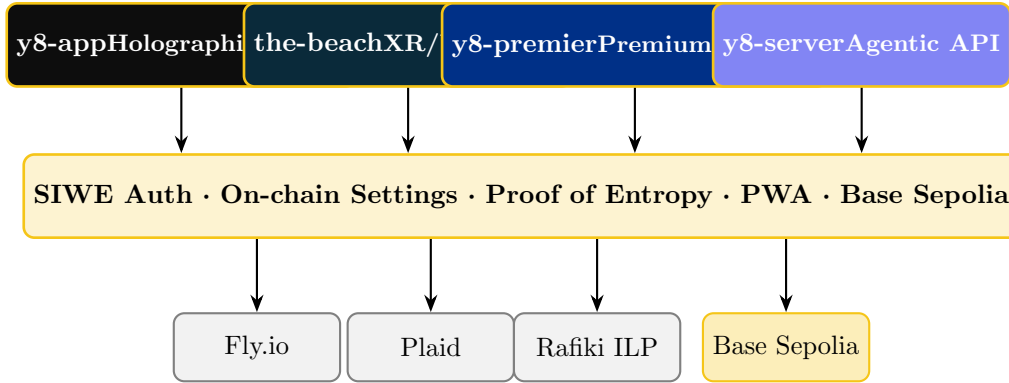


Figure 1: Y8 Platform architecture: four applications sharing a common identity, on-chain, and financial infrastructure layer.

2 y8-app: the holographic wallet

y8-app	
Live: y8-web-jason-sprouse.fly.dev	Stack: Next.js 14, wagmi, Base Sepolia, PWA

y8-app is the **primary identity surface** of the Y8 Platform. It is a wallet-native application: every page, every toggle, and every action is anchored to a connected wallet address. It is also the entry point to the holographic layer—the set of on-chain states, NFTs, and agent relationships that persist regardless of which surface the user accesses.

2.1 The Space: a wallet-native dashboard

The `/space` section of y8-app is the platform’s *holographic headquarters*: a dashboard that displays the user’s agent network, entropy proofs, and on-chain settings. Think of it as a personal mission control that happens to live on a blockchain.

Route	Surface	Description
<code>/space</code>	Dashboard	Agent network, activity feed
<code>/space/agents</code>	Marketplace	Browse and deploy AI agents
<code>/space/settings</code>	Control panel	On-chain settings (6 toggles)
<code>/space/entropy</code>	Lab	Mint Proof-of-Entropy NFTs
<code>/space/sunshade</code>	Lifestyle	SunShade utility page

2.2 On-chain settings: your preferences live on Base

The `Y8Settings` smart contract (Base Sepolia) stores six boolean preferences per wallet address. A single **Deploy to Base Sepolia** button deploys the contract from the browser via MetaMask—no

CLI, no private key management. Once deployed, each toggle change writes a transaction to Base, creating a permanent, timestamped record of the user’s preferences [2, 3].

Setting	State	Platform effect
GPS Location	on/off	Enables rideshare dispatch; contributes to entropy score
Active Driver	on/off	Appears in y8-server agent routing pool
Online Status	on/off	Visible to other Y8 users in the-beach and agent network
Notifications	on/off	Push notifications via service worker Push API
Privacy Mode	on/off	Hidden from public discovery across all surfaces
Dark Mode	on/off	Theme preference synced across 2D and XR surfaces

These settings form the **cross-surface control plane**: when a user turns on Active Driver in y8-app’s settings page, that state is readable by y8-server’s agent router, by the-beach’s dispatch visualization, and by y8-premier’s financial dashboard—all without a central database, because the source of truth is the blockchain.

2.3 Progressive Web App: the wallet in your pocket

y8-app is fully PWA-compliant. On mobile Chrome, the browser presents an “Add to Home Screen” prompt. On desktop, the address bar shows an install button. Once installed, the app launches from the home screen, fills the full screen, and operates offline (serving cached routes from the service worker). The theme color (#f5c518) tints the Android status bar gold, reinforcing the brand at every touch point.

3 the-beach: the holographic XR space

the-beach

Live: y8-beach-jason-sprouse.fly.dev **Stack:** NestJS, Babylon.js, WebXR, Socket.IO, PWA

the-beach is the **immersive layer** of the Y8 Platform. It is a Babylon.js XR/VR environment with a NestJS back-end, real-time multiplayer via Socket.IO, and a WebXR-compliant interface that works in a standard browser (desktop 3D) or a VR headset (full spatial computing mode) [7, 8, 9, 10].

3.1 What holographic means here

In the context of the Y8 Platform, *holographic* refers to the convergence of three properties:

1. **Spatial:** information is presented in three dimensions and responds to the user's position and gaze (WebXR head tracking, hand tracking, or controller input).
2. **Persistent:** the space is anchored to on-chain state. Entering the-beach with a wallet that has Y8Settings deployed means the space already knows your preferences, your online status, and whether you are an active driver.
3. **Multi-user:** the WebSocket multiplayer layer (Socket.IO) broadcasts avatar positions, agent statuses, and financial events in real time, so multiple users inhabit and share the same space simultaneously [9].

3.2 Spatial financial interface (roadmap)

The near-term roadmap for the-beach extends the platform's financial layer into the spatial computing interface. Planned holographic panels include:

- A **floating balance panel** that displays the user's Plaid-linked account balances and net worth alongside their avatar.
- A **spatial swap console** where the user picks up a token object (ETH, SOL, BTC) and drops it into a swap zone, with real-time price feeds rendered as holographic ticker tape.
- An **Interledger payment portal:** a gateway in the XR space that accepts an ILP address (spoken or typed) and dispatches a Rafiki transfer, with a visual beam indicating the cross-border payment route.
- An **Entropy altar:** a physical-world interaction zone where the user's device sensors are collected, the entropy score is visualized as particle density, and the Mint button triggers the Y8ProofOfEntropy contract.

4 y8-premier: the premium financial command center

y8-premier

Live: y8-premier-jason-sprouse.fly.dev **Stack:** Vite, React, Express, Plaid, Recharts, PWA

y8-premier is the **premium financial companion** to the holographic wallet. It is the surface where traditional finance meets on-chain. It is where you see your Chase checking balance alongside your ETH holdings, and where you can send a dollar to someone in Lagos via Interledger in the same UI as a swap from USDC to Bitcoin.

4.1 Premium access model

Access to y8-premier is gated by the Y8 identity layer (SIWE authentication) and a \$50 deposit commitment—a design that positions premium financial access as a real value exchange rather than a free tier [1]. This deposit mechanism is the foundation of the platform’s monetization model and maps directly to the “linear bike share” and PPP corridor models described in Section 8, where a rider or operator deposits a small amount to unlock premium service tiers.

4.2 Plaid bank connectivity

The Plaid integration connects the user’s bank accounts (checking, savings, investment, credit) to the y8-premier dashboard. Six server-side endpoints handle the full lifecycle of a bank connection:

Endpoint	Action
POST /api/plaid/create_link.token	Open Plaid Link modal
POST /api/plaid/set_access.token	Exchange token after linking
GET /api/plaid/status	Check session link state
GET /api/plaid/accounts	Fetch all account balances
GET /api/plaid/transactions	Last 30 days of transactions
POST /api/plaid/disconnect	Unlink bank account

The Plaid credentials (PLAID_CLIENT_ID, PLAID_SECRET, PLAID_ENV) are stored as Fly.io secrets—the first and most critical environment variable check the server performs on startup, with an explicit console log confirming their presence:

```
console.log(  
  '[Y8 Premier] Plaid: ${  
    process.env.PLAID_CLIENT_ID ? ' CLIENT_ID set' : ' MISSING'  
  } | ${  
    process.env.PLAID_SECRET ? ' SECRET set' : ' MISSING'  
  } | ENV=${process.env.PLAID_ENV || 'sandbox'}'  
);
```

After linking, the dashboard renders:

- **Net Worth Banner:** total assets minus liabilities in a gradient card at the top of the Financial page.
- **Account Cards:** per-account balance, type, last four digits, and available funds.
- **Transaction Feed:** scrollable list of the last 30 days of transactions with category tagging, pending indicators, and debit/credit color coding.
- **Spending Chart:** interactive Recharts pie chart showing spending by category, built from the transaction feed in real time.

4.3 USDC/USDT-to-crypto swap panel

The swap panel provides direct exchange from USDC or USDT to SOL, ETH, or BTC using live CoinGecko pricing (no API key required). The conversion formula is simple: since stablecoins are 1:1 with USD, the output is $Q_{\text{crypto}} = Q_{\text{stable}}/P_{\text{crypto}}$, where P_{crypto} is the live spot price in USD. The panel refreshes prices on every token selection change and displays the indicative rate below the output field.

The swap execution is designed to route through the 1inch or Uniswap V3 router on Base Mainnet in production [11, 2]. The current sandbox returns a synthetic transaction hash, allowing the full UI flow (amount entry, price quote, confirmation, result display) to be tested without mainnet gas.

4.4 Rafiki Interledger: global payments from the dashboard

The Rafiki panel enables the user to send money to any ILP address in any currency. The user enters a recipient ILP address (e.g., `$wallet.interledger.org/alice`), selects a currency (USD, EUR, GBP, BTC, ETH, XRP), enters an amount, and clicks Send. The server routes the transfer through the Rafiki ILP stack and returns a `ILP-{transactionId}` confirmation [5, 6].

In the holographic vision, this panel is the *planetary payment gateway*: a single interface that can settle value anywhere in the world, whether the recipient is a bank account (via Plaid + ACH), an on-chain address (via Base + swap), or an ILP wallet (via Rafiki). No per-country payment rail configuration is required.

4.5 RSIncomePanel: crypto income tracking

The RSIncomePanel, positioned at the bottom of the Financial page, displays running income statistics for BTC, ETH, and SOL alongside a simulated yield feed. This panel completes the unified financial picture: traditional bank accounts at the top, stablecoin swap in the middle, Interledger payments alongside, and crypto income at the bottom—all on one page, all anchored to the same wallet identity.

5 Proof of Entropy: the physical-digital bridge

Proof of Entropy (Y8POE)

“Your device is a witness. Every sensor reading, every GPS ping, every accelerometer sample is a signed statement that you were here, alive, in this moment.”

The **Proof-of-Entropy** protocol is the Y8 Platform’s most distinctive technical contribution. It creates a cryptographic bridge between the physical world and the holographic space by transforming multi-sensor device telemetry into a unique, non-replayable, on-chain record.

5.1 Protocol flow

1. **Collect.** The `useEntropy` hook gathers readings from up to six sensor classes: 32 random bytes from `crypto.getRandomValues()`, a sub-millisecond timestamp from the Performance API, screen geometry and pixel density, GPS coordinates (if permitted), device accelerometer and gyroscope readings (if mobile), network connection type and RTT, and battery level and charging state.
2. **Hash.** All collected values are serialized to JSON and hashed with SHA-256 using the browser’s SubtleCrypto API, producing a `bytes32` value.
3. **Score.** An entropy quality score (0–100) is computed from the fraction of optional sensor sources that successfully contributed. A device with GPS, motion, and network active scores 83; a desktop browser without any optional sensors scores 33.
4. **Mint.** The user calls `mintProof()` on the Y8ProofOfEntropy contract. The contract checks the `usedHashes` mapping (replay protection), increments the token counter, and emits a `ProofMinted` event. The ERC-721 token is permanently associated with the entropy hash, the device type, the sensor profile, and the quality score.

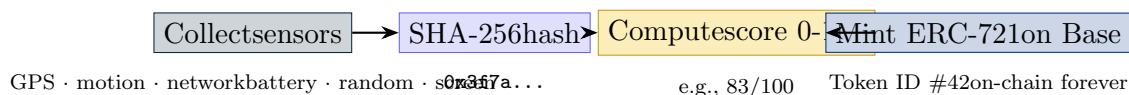


Figure 2: Proof-of-Entropy protocol: from device sensors to on-chain NFT.

5.2 Applications of Proof of Entropy

Holographic liveness. In an XR session, the Entropy altar in the-beach collects sensor data from the headset and controller. The resulting Proof-of-Entropy token is the XR session’s *liveness proof*: a verifiable record that a real human device was present in the holographic space at a specific time.

Insurance binding. A coverage binder signed at rental activation includes the Proof-of-Entropy token ID as a `deviceProofNFT` field, cryptographically binding the policy to a specific device liveness event at a specific location and time [12].

PPP field audit. In infrastructure operations, a field inspector mints a Proof-of-Entropy token on-site (GPS required, score ≥ 60). The token ID is embedded in the inspection report, providing an on-chain timestamp and location reference that cannot be backdated.

6 y8-server: the agentic routing layer

y8-server

Live: y8-server-jason-sprouse.fly.dev

Stack: NestJS, Socket.IO, JWT, SIWE

y8-server is the intelligence layer of the Y8 Platform. It maintains an in-memory registry of active agents, each with a wallet address, capability set (**sales**, **support**, **consulting**), load score, and geographic service area. Service requests are dispatched using pluggable routing strategies [10, 9]:

- **Least-load:** send to the agent with the lowest `currentLoad` / `maxLoad` ratio.
- **Nearest-location:** send to the geographically closest agent (haversine distance to the request’s GPS coordinates).
- **Highest-rating:** send to the agent with the highest `performanceScore`.
- **Round-robin:** distribute requests evenly across all eligible agents.

In the holographic context, y8-server is the *dispatch grid*: the system that decides which AI agent avatar walks toward you in the-beach when you request a service. The Active Driver toggle in Y8Settings is a live feed into this routing layer—when a user’s on-chain settings show `activeDriver: true`, their wallet identity enters the dispatch pool as a human-agent endpoint.

7 Premium access model and monetization

Tier	Features	Access
Free (y8-app)	Holographic wallet, space dashboard, agent marketplace, Proof-of-Entropy minting, PWA, on-chain settings	SIWE wallet login
Premium (y8-premier)	Bank connectivity (Plaid), net-worth dashboard, transaction feed, USDC/USDT swap, Rafiki ILP payments, RSIncomePanel	\$50 deposit
XR (the-beach)	Holographic spatial interface, multi-player, spatial financial panels, Entropy altar, agent dispatch visualization	Wallet + (Premium recommended)

The \$50 deposit design is intentional: it is low enough to be accessible but high enough to signal intent. In the PPP corridor model (Section 8), the same deposit can unlock a rental activation, an insurance binder, or an operator seat in the agentic dispatch grid.

8 PPP corridor application

The Y8 Platform is the technology substrate for a North Texas PPP mobility and energy corridor (Denton/Parker Station → Farmersville → Beavers Bend) with a hydrogen + data ecosystem anchor. The platform’s four layers map directly to corridor operational needs:

Platform Layer	Corridor Application
Holographic Wallet (y8-app)	Rider identity; operator seat credentials; NPE wallets for drones, swap stations, and sensor gateways
XR Space (the-beach)	Corridor digital twin; field operations control room; training simulations for dispatch and maintenance
Premium Finance (y8-premier)	Concession fee collection; maintenance payment rails; operator net-worth and cash-flow dashboard
Agentic API (y8-server)	Driver dispatch; fleet routing; incident response coordination; vendor and contractor seat management

8.1 Proof of Entropy in the field

For a PPP infrastructure operator, Proof of Entropy resolves a persistent audit problem: how do you prove a field inspection happened, at the right location, by the right person, without relying on a central database that can be altered? The Y8 answer: require the inspector to mint a Proof-of-Entropy token (GPS required) on-site. The token’s GPS contribution to the entropy hash creates a location-binding that is stored on Base forever. The inspection report embeds the token ID. The auditor verifies on BaseScan. No trust required.

8.2 Pro forma summary (updated)

Category	CapEx	Monthly net
E-bike build (hyper, 20 kW unit)	\$6,000–\$7,500	—
Rental fleet (10 bikes + trailer)	\$68,100	\$4,500
Autonomous mowing (RaaS per crew)	—	\$3,800
Linear bike share (100-bike corridor)	~\$700K	\$35,000
PPP hydrogen + data hub	~\$3.86M	Grant-funded
Y8 Platform (SaaS layer)	\$0 hosting	\$50/premium user

9 Roadmap: toward a fully holographic financial OS

1. **Phase 1 (Complete):** PWA installability, SIWE auth, on-chain settings (Y8Settings), Proof of Entropy (Y8ProofOfEntropy), Plaid bank connectivity, stablecoin swap, Rafiki ILP, RSIncomePanel. Four apps live on Fly.io.
2. **Phase 2 (In progress):** Active Driver GPS broadcast to y8-server; persistent session storage (Fly.io Postgres / Redis); rental + insurance binder signing with `deviceProofNFT` extension; the-beach spatial financial panels.
3. **Phase 3 (Planned):** Holographic XR control room in the-beach (live incident visualization, digital twin of the PPP corridor); ERC-4337 account abstraction for sponsored contract deployments (remove gas barrier for new users); 1inch/Uniswap V3 live swap execution on Base Mainnet; Rafiki ILP production node [13].
4. **Phase 4 (Vision):** Full WebXR hand-tracking interaction for all financial panels; XR-native insurance binder signing via gesture; entropy collection from XR headset sensors (6DoF motion data as high-entropy source); NPE wallets for every infrastructure node in the PPP corridor.

10 Conclusion

The Y8 Platform is not a wallet that happens to have some features. It is a **holographic financial operating system**: an identity layer (SIWE + on-chain settings), a liveness layer (Proof of Entropy), a financial layer (Plaid + swap + ILP), an XR layer (the-beach + Babylon.js + WebXR), and an intelligence layer (y8-server agentic routing)—all unified under a single wallet address and all installable on any device as a Progressive Web App.

The \$50 deposit is the handshake between the traditional financial world and the holographic one. Plaid connects your bank. The swap panel moves your value into crypto. Rafiki sends it anywhere. The Entropy altar proves you were there. The XR space makes it feel like the future—because it is.

References

- [1] Spruce Systems, *Sign-In with Ethereum (EIP-4361)*. <https://docs.login.xyz/>.
- [2] Coinbase, *Base: an Ethereum L2*. <https://base.org>.
- [3] Wagmi, *React hooks for Ethereum*. <https://wagmi.sh>.
- [4] Plaid Technologies, *Plaid API documentation*. <https://plaid.com/docs/>.
- [5] Interledger Foundation, *Rafiki open-source ILP stack*. <https://rafiki.dev>.
- [6] Interledger Protocol, *Send money across payment networks*. <https://interledger.org>.

- [7] Babylon.js, *Babylon.js 3D engine*. <https://www.babylonjs.com/>.
- [8] MDN Web Docs, *WebXR Device API*. https://developer.mozilla.org/en-US/docs/Web/API/WebXR_Device_API.
- [9] Socket.IO, *Real-time, bidirectional event-based communication*. <https://socket.io/>.
- [10] NestJS, *A progressive Node.js framework*. <https://nestjs.com/>.
- [11] Uniswap Labs, *Uniswap V3*. <https://docs.uniswap.org/>.
- [12] Ethereum Improvement Proposals, *EIP-712: Typed structured data hashing and signing*. <https://eips.ethereum.org/EIPS/eip-712>.
- [13] Ethereum Improvement Proposals, *EIP-4337: Account Abstraction*. <https://eips.ethereum.org/EIPS/eip-4337>.
- [14] Fly.io, *Run full-stack apps close to your users*. <https://fly.io>.
- [15] World Bank PPP Resource Center, *About Public-Private Partnerships*. <https://ppp.worldbank.org/about-public-private-partnerships>.