



HOUSE OF JOSHI SWAP

TECHNICAL WHITEPAPER

Version 1.0 - August 2026

swap.thehouseofjoshi.com



Abstract

House of Joshi Swap is a non-custodial multichain swap and bridge interface designed to make on-chain asset exchange understandable, verifiable, and accessible from a single application. The system combines wallet-native transaction authorization, chain-aware token discovery, third-party liquidity routing, transaction simulation, minimum-output protection, and transparent fee handling.

The application supports 21 configured EVM networks, a native Solana swap experience, and a distinct native XRP Ledger experience. House of Joshi Swap does not custody user funds, operate the underlying liquidity pools, or promise execution at a fixed price. Users retain control of their wallets and approve every transaction.

This whitepaper explains the product architecture, deployed contracts, fee model, security controls, supported ecosystems, current limitations, and development roadmap. It is a technical product document, not an offer of securities or financial advice.

1. Problem and vision

On-chain trading remains fragmented. Users must choose the correct network, verify token identifiers, compare routes, understand approvals, manage gas, and distinguish a wallet request from a completed transaction. Multichain users face additional complexity because EVM chains, Solana, and the native XRP Ledger use different transaction models, wallet standards, and asset representations.

House of Joshi Swap aims to provide a clear interface across these environments while preserving the core property of self-custody. Its design goals are:

- keep users in control of keys and signatures;
- make the selected network, asset, fees, and expected output visible;
- use executable third-party liquidity rather than simulated prices;
- protect swaps with minimum-output checks and transaction validation;
- identify supported contracts and XRPL issuers explicitly;
- separate discovery-only assets from executable routes; and
- expose protocol activity through public blockchain data.

2. Product overview

The application is organized around four user-facing functions.

2.1 Swap

The swap interface lets a user select a network, sell asset, buy asset, amount, and wallet. A quote is requested from configured routing providers. When a supported executable route is available, the interface presents expected output, price impact, slippage, network costs, and the House fee before requesting a wallet signature.

2.2 Cross-chain routing

Cross-chain routes are obtained from third-party bridge and liquidity infrastructure. House of Joshi Swap does not operate the bridges. Route availability, settlement time, destination gas, and final output depend on the selected provider and public networks.

2.3 Token discovery

The application maintains chain-aware token catalogs and can display trending or profile-associated assets from external sources. Displaying a token, logo, price, or trend does not mean that a safe or executable route exists. The interface distinguishes catalog and discovery experiences from enabled swap execution.

2.4 Transaction history and verification

Public transaction identifiers are used to show transaction status. The application can simulate supported EVM transactions before submission and checks transaction receipts after execution. Native XRPL transactions are prepared for wallet authorization and validated against XRPL results.

3. Supported networks

House of Joshi Swap is designed for a multichain environment. The complete current application configuration is listed below.

Network	Chain ID	Native asset	Current role
Ethereum mainnet	1	ETH	Swap network
Optimism	10	ETH	Swap network
Cronos	25	CRO	Integration and token catalog
BNB Smart Chain	56	BNB	Swap network and deployed House router
Unichain	130	ETH	Swap network
Polygon	137	POL	Swap network
Monad	143	MON	Swap network and deployed House router
Sonic	146	S	Swap network and deployed House router
World Chain	480	ETH	Swap network and deployed House router
HyperEVM	999	HYPE	Swap network and deployed House router
Robinhood Chain	4663	ETH	Swap network
Mantle	5000	MNT	Swap network and deployed House router
Base	8453	ETH	Swap network and deployed House router
Plasma	9745	XPL	Swap network and deployed House router
Arbitrum	42161	ETH	Swap network
Avalanche C-Chain	43114	AVAX	Swap network
Ink	57073	ETH	Swap network and deployed House router
Linea	59144	ETH	Swap network and deployed House router
Berachain	80094	BERA	Swap network and deployed House router
Scroll	534352	ETH	Swap network and deployed House router

Network	Chain ID	Native asset	Current role
Zora	7777777	ETH	Token discovery and catalog
Solana	Native SVM - no EVM chain ID	SOL	Native Solana swap experience
XRP Ledger	Native ledger - no EVM chain ID	XRP	Native XRPL swap experience

"Swap network" means the application exposes that network for route requests. Execution still depends on live provider support, liquidity, deployed router configuration, token compatibility, and operational RPC endpoints. A network appearing in the interface does not guarantee that every asset pair is executable. Catalog and discovery networks provide token information but do not imply an executable route.

4. EVM router deployments

House of Joshi Swap executes supported EVM swaps through `HojswapRouterV2`. BNB Smart Chain is shown below as a representative production deployment; the additional deployment table records the newly enabled networks.

Property	Value
Network	BNB Smart Chain mainnet
Chain ID	56
Native asset	BNB
Explorer	https://bscscan.com
Router	0x2C5F372746330465C3f4084CE6C6aBce22a48B4d
Router implementation	HojswapRouterV2
House fee	1% (100 basis points)

The repository includes BNB RPC configuration, BscScan transaction and address links, BNB-specific router configuration, BNB token metadata, and Solidity contract source. The router address contains deployed bytecode on BNB Smart Chain mainnet.

4.1 Ethereum mainnet profile

Property	Value
Network	Ethereum mainnet
Chain ID	1
Native asset	ETH
Explorer	https://etherscan.io
House router	0x2C5F372746330465C3f4084CE6C6aBce22a48B4d application fallback
Current role	Swap network; execution depends on live provider and router availability

4.2 Optimism profile

Property	Value
Network	Optimism mainnet
Chain ID	10
Native asset	ETH
Explorer	https://optimistic.etherscan.io
House router	0x2C5F372746330465C3f4084CE6C6aBce22a48B4d configured
Current role	Swap network

4.3 Cronos profile

Property	Value
Network	Cronos mainnet
Chain ID	25
Native asset	CRO
Explorer	https://cronoscan.com
House router	0x2C5F372746330465C3f4084CE6C6aBce22a48B4d configured
Current role	Integration and token catalog; swap UI currently disabled

4.4 Unichain profile

Property	Value
Network	Unichain mainnet
Chain ID	130
Native asset	ETH
Explorer	https://uniscan.xyz
House router	0x2C5F372746330465C3f4084CE6C6aBce22a48B4d configured
Current role	Swap network

4.5 Polygon profile

Property	Value
Network	Polygon mainnet
Chain ID	137
Native asset	POL
Explorer	https://polygonscan.com
House router	0x2C5F372746330465C3f4084CE6C6aBce22a48B4d configured
Current role	Swap network

4.6 Robinhood Chain profile

Property	Value
Network	Robinhood Chain mainnet
Chain ID	4663
Native asset	ETH
Explorer	https://robinhoodchain.blockscout.com
House router	0x2C5F372746330465C3f4084CE6C6aBce22a48B4d configured
Current role	Swap network

4.7 Base profile

Property	Value
Network	Base mainnet
Chain ID	8453
Native asset	ETH
Explorer	https://basescan.org
House router	0x6aCaf964bCf4551CC55AfaF12d6e6a8ef7138875 configured
Current role	Swap network and deployed House router

4.8 Arbitrum profile

Property	Value
Network	Arbitrum One
Chain ID	42161
Native asset	ETH
Explorer	https://arbiscan.io
House router	0x2C5F372746330465C3f4084CE6C6aBce22a48B4d configured
Current role	Swap network

4.9 Avalanche profile

Property	Value
Network	Avalanche C-Chain mainnet
Chain ID	43114
Native asset	AVAX
Explorer	https://snowtrace.io
House router	0x2C5F372746330465C3f4084CE6C6aBce22a48B4d configured
Current role	Swap network

4.10 Additional EVM router deployments

The following production networks use the shared deployed House router address 0x2C5F372746330465C3f4084CE6C6aBce22a48B4d.

Network	Chain ID	Native asset	Default configured pair
Linea	59144	ETH	ETH / USDC

Network	Chain ID	Native asset	Default configured pair
Scroll	534352	ETH	ETH / USDC
Mantle	5000	MNT	MNT / USDC
World Chain	480	ETH	ETH / USDC.e
Sonic	146	S	S / USDC
Berachain	80094	BERA	BERA / USDC.e
Ink	57073	ETH	ETH / USDC
Monad	143	MON	MON / USDC
HyperEVM	999	HYPE	HYPE / USDC
Plasma	9745	XPL	XPL / USDT0

Each deployment uses owner-managed approval pairs for the 0x entry point and allowance holder. A deployed router does not guarantee liquidity for every pair; execution remains dependent on a valid live quote, approved router/spender pair, RPC availability, and sufficient liquidity.

4.11 Zora profile

Property	Value
Network	Zora mainnet
Chain ID	7777777
Native asset	ETH
Explorer	https://explorer.zora.energy
House router	No House router configured
Current role	Token discovery and catalog; swap execution disabled

4.12 Native Solana profile

Property	Value
Network	Solana mainnet
Chain ID	Native SVM; no EVM chain ID
Native asset	SOL
Explorer	https://solscan.io
House contract	Not applicable; swaps use Solana instructions rather than an EVM router
Current role	Native Solana swap experience through Jupiter

The Solana interface supports compatible installed and mobile wallets, token discovery, disconnected quote previews, wallet balances after connection, and Jupiter-routed swap execution.

4.13 Native XRP Ledger profile

Property	Value
Network	XRP Ledger mainnet
Chain ID	Native ledger; no EVM chain ID
Native asset	XRP
Explorer	https://livenet.xrpl.org
House contract	Not applicable; XRPL uses native transactions rather than an EVM router
House fee destination	rUG7tHZ5sGCVxuhkAiL9fUqVFhki2Z6bVU
Current role	Native XRPL swap experience

The word "configured" identifies the address selected by the application configuration. It does not by itself guarantee that a contract remains available, verified, approved for every liquidity provider, or executable for every asset pair. Users and directory reviewers should confirm current bytecode and verification independently on the relevant explorer.

5. System architecture

5.1 Client application

The production interface is a Next.js application. It manages network and token selection, wallet connection, balances, quote presentation, transaction preparation, and user-facing status. Browser-exposed configuration is limited to values intentionally prefixed as public environment variables.

5.2 Server routes

Server-side API routes protect private provider credentials and normalize quote, token, bridge, price, and transaction data. Quote responses are treated as untrusted input and are validated before they are presented or converted into wallet requests.

5.3 EVM routing contract

Hoj swapRouterV2 is the protocol's EVM execution layer. It supports native-asset and ERC-20 sells, sends the configured fee to the House wallet, executes only owner-approved router and spender pairs, measures received output by balance delta, and enforces the user's minimum output.

Core safeguards include:

- reentrancy protection;
- emergency pause control;
- router and spender pair allowlisting;
- exact transaction parameters supplied by the user-approved route;
- minimum-output enforcement;
- fee-on-transfer input accounting; and

- events for swap and bridge-route metadata.

The router does not discover liquidity. Routing providers produce calldata for third-party liquidity venues; the House router enforces fee and settlement rules around that route.

5.4 Native Solana flow

Native Solana swaps use Solana wallet-standard connections and Jupiter routing. The application retrieves chain-aware token metadata and quote output independently of wallet connection, while a compatible connected wallet is required to sign and submit the final transaction. Mobile wallet options may open an installed application or its download page when the application is not installed.

5.5 Native XRP Ledger flow

Native XRPL uses classic addresses, trust lines, issued-currency identifiers, and XRPL-native transactions. The application supports wallet flows including Xaman, Crossmark, GemWallet, and compatible WalletConnect wallets when those providers are available in the user's environment.

For issued assets, the currency code and issuer together identify the asset. The application currently includes carefully configured pairs involving XRP and selected issued assets such as RLUSD, native USDC, SOLO, CasinoCoin, and XRdoge. Users must still verify issuer, trust line, liquidity, and wallet request details.

6. Fee model

The House fee is 1% of the gross sell amount unless a future interface version clearly discloses a different fee before authorization.

For supported EVM router swaps, the fee is enforced in the same router transaction as execution. The contract emits the gross sell amount, fee amount, and resulting buy amount. Network gas, liquidity-provider fees, bridge fees, token taxes, and wallet-provider costs are separate from the House fee.

For native XRP Ledger swaps, the current flow requests the disclosed 1% House fee in XRP after the swap has validated. The wallet displays the fee transaction for independent authorization. The native XRPL House destination currently configured by the application is `rUG7tHZ5sGCVxuhKAiL9fUqVFhki2Z6bVU`.

The EVM fee destination currently configured by the application is `0x6736d2eA9807297F0e56967361B9410854B86a5f`.

House of Joshi Swap has no protocol token, token sale, mining program, yield promise, or token-based revenue entitlement described by this whitepaper.

7. Wallets and custody

House of Joshi Swap is non-custodial. It does not request or store private keys or recovery phrases. Wallet providers handle account access and signatures. The application prepares requests, but the user decides whether to authorize them.

Users are responsible for:

- confirming the selected network;
- verifying token contracts or XRPL issuers;
- reviewing approvals and spend amounts;
- checking recipients, minimum output, fees, and slippage;
- maintaining sufficient native currency for network fees; and

- protecting their wallet and recovery phrase.

8. Security model

Security is layered across the interface, server routes, wallet, router contract, and public networks.

The application uses input validation, server-side credential handling, chain-aware token identifiers, restricted router/spender pairs, minimum-output checks, transaction simulation where supported, receipt verification, and emergency pause capability. Contract and application activity remains publicly inspectable.

Important limitations remain. House of Joshi Swap depends on third-party RPC nodes, routing providers, bridges, wallets, token issuers, and public blockchains. Smart-contract defects, malicious tokens, depegging, issuer controls, price manipulation, liquidity loss, bridge failures, MEV, network outages, or user error can cause loss.

There is currently no external security audit. Internal review and automated testing do not replace an independent audit. Users should transact cautiously and verify every wallet request.

9. Data, privacy, and transparency

Public wallet addresses and blockchain transactions are visible on public ledgers. Server, hosting, analytics, wallet, RPC, and routing providers may process technical and usage information needed to operate their services. House of Joshi Swap does not treat public blockchain activity as private.

The application publishes Terms of Service and a Privacy Policy. Protocol contracts, configuration, and application source are available through the official public repository. EVM swap volume and fee accounting has been submitted to the DefiLlama dimension-adapters repository for independent indexing from emitted contract events.

10. Governance and operations

House of Joshi Swap is currently operated by the House of Joshi project team. Router administration is owner-controlled and includes changing the House wallet, managing approved router/spender pairs, managing supported bridge destination identifiers, and activating an emergency pause.

These privileges create operational trust assumptions. Administrative actions are visible on-chain, but the present architecture is not governed by a DAO or token vote. Future decentralization should be introduced only with documented controls, tested migrations, and clear user communication.

11. Roadmap

Development priorities include:

1. expand independent contract review and security testing;
2. improve route availability and fallback behavior without weakening validation;
3. add clearer per-network execution and discovery status;
4. improve native XRPL liquidity and trust-line diagnostics;
5. publish richer on-chain volume, fee, and route analytics;
6. increase contract verification and deployment documentation across supported networks; and
7. evaluate multisignature or time-delayed administration for sensitive EVM controls.

Roadmap items are goals, not guarantees. Features may change based on security, provider availability, network conditions, and regulatory requirements.

12. Conclusion

House of Joshi Swap provides a single non-custodial interface for navigating swaps across 21 configured EVM networks, Solana, and the native XRP Ledger. Its architecture combines third-party liquidity discovery with chain-specific wallet authorization, transparent fees, minimum-output protection, and public transaction verification.

The project prioritizes clear network identity, user-controlled signatures, and verifiable configuration. It does not eliminate the risks of digital assets or third-party protocols. Users should independently evaluate every token, route, contract, issuer, and wallet request before transacting.

Legal notice

This whitepaper is provided for technical and informational purposes only. It is not financial, investment, legal, tax, or accounting advice; an offer or solicitation; or a promise of availability, performance, profit, or future development. Digital assets and blockchain transactions involve substantial risk, including total loss. Use of House of Joshi Swap is subject to the published Terms of Service and Privacy Policy.