

FOS: The Digital Collectibles and Allocation Protocol

Robinhood Chain / Base VRF and CCIP / Three-Plan Referral Edition

FOS Project Documentation

October 12, 2026

Contents

1 Executive Summary	2
2 Market Context and Comparison Limits	2
3 Platform Infrastructure	3
4 Collection & Protocol Mechanics	4
5 Referral Program	6
6 Smart Contracts	7
7 Verify Contract	9
8 Main Application Integration	10
9 Security & Validation	11
10 Community & Growth	12
11 Campaign & Leaderboard Design	12
12 Public Roadmap	13
13 Team and Governance Scope	16
14 Risks, Eligibility, and Legal Notice	16
15 Future Vision and Expandability	17
16 Appendix: Fee Funding and Failure Cases	17
17 Appendix: Technical Boundaries and Glossary	17
18 References and Revision Provenance	18

1 Executive Summary

FOS offers Fos Digital Collectibles as NFT consumer products with protocol-defined promotional distributions, an independent single-level referral program, and authenticity certificates. This edition updates the December 2025 whitepaper using the supplied Persian project contract specification. User-facing contracts move to Robinhood Chain. Base remains the specified randomness-service chain through Chainlink VRF and CCIP.

The product vocabulary follows the original FOS documentation: Fos Digital Collectible, promotional distribution, recipient, allocation, loyalty reward, and cycle. Exact contract identifiers remain unchanged in technical references. This whitepaper explains the reported design. It is not an independent contract audit, proof of a live bridge deployment, legal determination, or promise of individual allocations. The documentation repository does not contain the final Solidity sources or deployment ABI bundle. Unverified operational contract addresses are not published here.

1.1 Vision and mission

The project aims to make collectible participation and allocation accounting visible through on-chain records rather than opaque operator databases. Its practical goals are clear contract responsibilities, explicit fee and reserve accounting, verifiable randomness, and accessible user information. Owner configuration permissions and cross-chain delivery dependencies are part of that design, not exceptions to it.

1.2 What changed in this edition

- Robinhood Chain is the user transaction network; Base is retained for the specified VRF integration, connected through CCIP.
- Referral has nominal \$2/\$5/\$10 plans; the direct referrer's active plan sets 50%/60%/82% commission on registrations and upgrades.
- Upgrades pay the current configured price difference and retain the original referrer.
- Buddy/Samurai/Noble use 5%/3%/2% admin fees and funded cycle pools of nominal \$1,425/\$6,305/\$12,250.
- Free positions are funded from FOSNFT reserves; selection uniqueness is recorded per collectible mint, not per wallet.
- Selection completion is asynchronous; failed direct ETH payments become deferred claims.
- Verify issues separate ERC-721 certificates and distinguishes existence from current validity.

2 Market Context and Comparison Limits

The earlier whitepaper discussed digital collectibles, loyalty systems, and trust in centralized promotions. Those motivations remain relevant context, but historical Base adoption figures do not describe Robinhood adoption. The existing 2025 research is historical and has not been independently refreshed here.

Current economic statements in this edition follow the supplied collection and referral arithmetic. Per-collectible pool fractions are 95%, 97%, and 98% by collection, not a universal project-wide return rate. Gas, oracle, bridge, audit, development, and compliance costs are not assumed to vanish. Neither automation nor a terminology choice determines regulatory classification.

The original references are retained in the unchanged historical PDF. Current operational rules should be read from this edition and verified against the actual deployment.

3 Platform Infrastructure

3.1 Networks

User-facing FOSNFT, Processor, Referral, and Verify interactions belong on **Robinhood Chain**, not Base. Base serves the specified VRF randomness integration, with CCIP transporting cycle requests and results.

Network	Chain ID	Native currency	Role
Robinhood Chain	4663	ETH	User-facing project network
Robinhood Chain Testnet	46630	Test ETH	Separate testing network
Base	Environment-specific	ETH	VRF service side of the specified cross-chain design

The Robinhood IDs, currency, and URLs below are corroborated by the public [ethereum-lists network registry](#), with a [separate testnet entry](#). This corroborates network metadata; it does not verify FOS deployments or CCIP support.

- Mainnet RPC listed in the registry: <https://rpc.mainnet.chain.robinhood.com>
- Mainnet explorer listed in the registry: [Robinhood Chain Blockscout](#)
- Testnet RPC: <https://rpc.testnet.chain.robinhood.com/rpc>
- Testnet explorer: [Robinhood testnet explorer](#)
- Official documentation linked by the registry: [Robinhood Chain documentation](#)

Verify the selected environment and live RPC chain ID before signing. Never reuse Base or Base Sepolia addresses for a Robinhood transaction. The Base environment used by the VRF integration must be confirmed separately.

3.2 ETH pricing and gas

The displayed dollar values are nominal product labels. On-chain ETH prices and cycle settings are the transaction source of truth. Ordinary mints require the exact configured price; free positions require zero purchase value. Network gas still applies. ETH remains volatile, and neither fixed gas costs nor USD-equivalent awards are guaranteed.

3.3 Wallet access

Use an EVM wallet configured for the correct Robinhood network. MetaMask and Coinbase Wallet are documented integration targets; connector behavior must be verified in the deployed main application. This Next.js repository hosts documentation and an AI assistant, not the minting application itself.

The interface does not need a user's private key. Contracts nevertheless hold cycle funds, reserved free-position contributions, and deferred payments according to their rules. An interface described as non-custodial does not mean that funds never enter a contract.

3.4 Cross-chain visibility

Show Robinhood cycle status and transaction receipts separately from Base VRF and CCIP delivery status. A successful last-mint receipt is not evidence that the remote random result has arrived or that the allocation transaction has completed.

3.5 Network and promotional distribution architecture

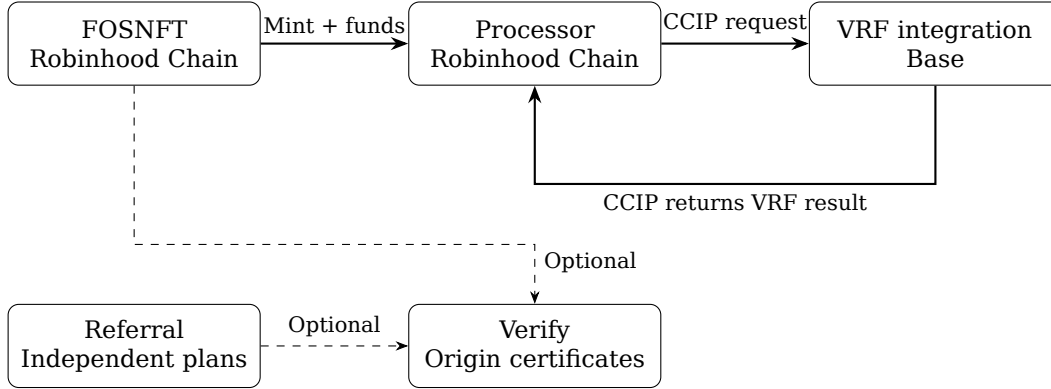


Figure 1: Specified component boundaries. CCIP/VRF deployment and callback details require final implementation verification.

4 Collection & Protocol Mechanics

4.1 Minting

Call `buyAndMint(collectionType, tokenURI)` on FOSNFT on Robinhood Chain. The documented type IDs are 3 (Buddy), 13 (Samurai), and 25 (Noble). A successful transaction mints a Fos Digital Collectible, registers its mint record with Processor, and sends the cycle's configured allocation-pool contribution to Processor. Only FOSNFT may register those records there. If Verify is connected and FOSNFT is approved, a separate authenticity certificate is also issued.

Throughout this reference, numbered collectibles refer to the mint records used for protocol-defined promotional distributions. Each collectible has one mint record; selection uniqueness applies to that record, while a wallet can have several recorded mints. Product wording does not change the selection mechanism or the number of allocations.

4.2 Prices and accounting

Collection	Type ID	Nominal price	Admin fee	Admin per paid mint	Pool contribution per collectible	Full 500-collectible pool
Buddy	3	\$3	5%	\$0.15	\$2.85	\$1,425
Samurai	13	\$13	3%	\$0.39	\$12.61	\$6,305
Noble	25	\$25	2%	\$0.50	\$24.50	\$12,250

Dollar amounts describe the product model, not a dollar-denominated contract balance or guaranteed allocation. Payments and allocations use native ETH. Read the configured on-chain price before sending a transaction; the production USD-to-ETH conversion policy is not supplied in the contract specification.

For each paid collectible:

```
adminCut = price * adminBps / 10000
prizeShare = price - adminCut
```

The admin share remains in FOSNFT. There is no additional admin deduction from recipients' allocations. The amount sent to Processor must equal the contribution stored for that cycle.

4.3 Five gas-only positions

Positions **10, 100, 200, 300, and 400** are free in every cycle of every collection. The buyer pays network gas but sends zero purchase value. FOSNFT supplies the full pool contribution for each free collectible. Admin withdrawals must preserve the reserve for remaining free positions.

For a full cycle there are 495 paid positions and 5 funded free positions. In nominal dollars:

Collection	Buyer purchase inflow	Gross admin share from 495 paid positions	Free-position funding	Remaining admin amount	Cycle pool
Buddy	\$1,485	\$74.25	\$14.25	\$60.00	\$1,425
Samurai	\$6,435	\$193.05	\$63.05	\$130.00	\$6,305
Noble	\$12,375	\$247.50	\$122.50	\$125.00	\$12,250

These calculations exclude gas, oracle/bridge costs, and any other operating costs. The pool plus remaining admin amount equals the 495 buyers' total purchase payments.

4.4 Weights and uniqueness

Collectibles 1–100 have two entries each in the selection pool; 101–500 have one. Positions 10 and 100 are both free and double-weighted. The initial full-cycle selection pool contains 600 entries.

After a collectible's mint record is selected, all of its weighted entries are removed. That record cannot be selected again in the cycle. A wallet with multiple recorded mints may receive more than one allocation through distinct records. Double weight is a relative selection weight, not a guarantee of an allocation or a claim that every final inclusion probability is exactly doubled.

4.5 Green Eye and asynchronous selection

The 500th collectible closes the cycle and is known as the Green Eye.

1. FOSNFT registers collectible 500 with Processor on Robinhood Chain. That cycle closes to further collectibles.
2. A cycle-specific randomness request is sent through Chainlink CCIP to the Base-side integration.
3. Chainlink VRF on Base produces verifiable randomness.
4. CCIP returns the result to the Robinhood-side integration.
5. Processor uses the returned randomness and collectible weights to select 17 distinct collectibles and process allocations.
6. After the selection completes, the next cycle of that collection starts automatically.

The last mint starts this asynchronous sequence. Selection, transfers, and the next cycle do **not** all finish in the last buyer's transaction. Bridge contract names, callbacks, fee funding, supported CCIP lane, and deployed bridge addresses must be checked against the final implementation; this repository contains documentation, not those Solidity contracts.

Do not assume a fixed 18× gas charge or an automatic fee reimbursement. Each transaction has its own gas cost, and oracle/CCIP fees depend on the final integration.

4.6 Distribution and claims

Group	Selected collectibles	Share of pool	Per-collectible share
Primary	1	50%	50% plus integer-rounding residue

Group	Selected collectibles	Share of pool	Per-collectible share
Secondary	6	30%	5%
Tertiary	10	20%	2%

Processor attempts direct payment. A rejected payment is recorded in `pendingWithdrawals` for a later `claim()` rather than described as reverting every recipient's payment.

4.7 Cycle settings

Capacity, contribution per collectible, group counts/splits, and the number of double-weighted positions are recorded at cycle start. Changes apply to subsequent cycles; an empty active cycle may be updated where the contract permits. The target capacity is 500; old 20-collectible settings were tests.

The new specification does not establish the previously described 365-day `liquidatePool()` mechanism. Do not rely on that recovery path unless it is confirmed in the final deployed contract.

4.8 Asynchronous completion

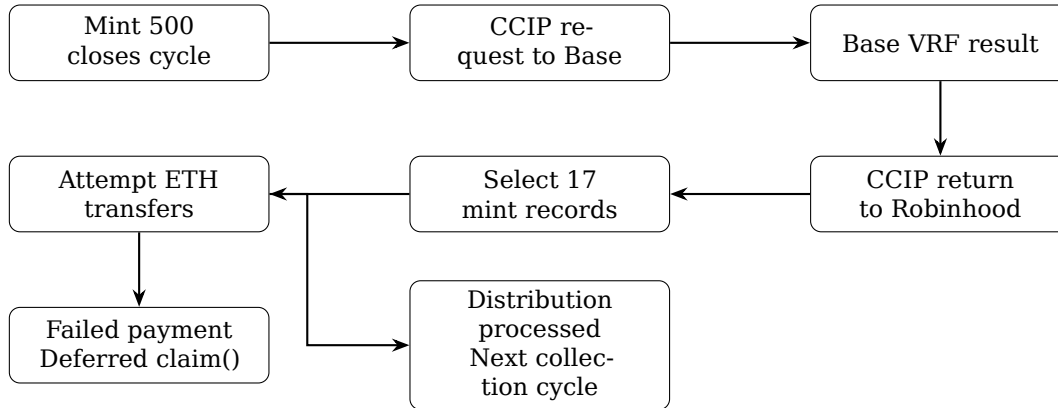


Figure 2: Selection processing is separate from the last buyer's mint. Deferred claims remain owed after the selection is completed.

5 Referral Program

Referral is independent of NFT purchases and cycle allocations. Each wallet registers once with a plan and, optionally, an existing registered direct referrer.

5.1 Plans and payment splits

Referrer's active plan	Share of a direct referral's payment	Admin share
\$2 plan	50%	50%
\$5 plan	60%	40%
\$10 plan	82%	18%

The **referrer's own active plan at payment time** determines the split, for both registration and upgrades. The buyer's plan determines the payment amount, not the commission percentage. Without a referrer, 100% goes to the admin. Dollar labels are nominal plan names; exact payment is the configured ETH price.

5.2 Register and share

Call `register(address referrer, uint256 tier)` with tier **2**, **5**, or **10** and exact configured payment. The referrer must already be registered and cannot be the registering wallet. Use the zero address to register without a referrer. The wallet address identifies the referral link; `referredBy` records the direct referrer permanently. A registration with a referrer increases that referrer's `referralCount` by one.

5.3 Upgrade by paying the difference

Registered users may call `upgrade(uint256 newTier)` to move 2→5, 2→10, or 5→10. Downgrades and registering the same wallet again are prohibited.

`cost = tierPrice[newTier] - tierPrice[oldTier]`

Read `getUpgradeCost` and the current prices before submitting. Nominal differences are \$3, \$8, and \$5 respectively; actual payment uses the current configured ETH difference. The upgrade is split with the **same original direct referrer**. Without one, the full upgrade payment goes to the admin.

5.4 Example

1. A has the \$2 plan. B registers through A with the \$2 plan. A receives 50% of that payment, nominally \$1.
2. A upgrades to the \$10 plan.
3. B later upgrades from \$2 to \$5 and pays the nominal \$3 difference.
4. A now receives 82% of that difference, nominally \$2.46; the admin receives \$0.54.

If A had stayed on the \$2 plan, even a \$10 registration by B would use A's 50% rate. A's upgrade changes only subsequent payments; previous payments are never recalculated.

5.5 Single-level accounting

If A refers B and B refers C, A can receive a share of B's registration and upgrades. C's registration and upgrades can pay B, not A. Plan levels are price/commission plans, not additional generations of referral compensation.

5.6 Direct payment and deferred claims

The contract attempts direct ETH payment to the referrer. If that transfer fails, it records the amount in `pendingWithdrawals`; the referrer uses `claim()` to collect it. `withdraw()` is for the admin's available balance, not the referrer's claim function.

5.7 Configuration and risk

The owner can change plan prices, commission basis points, the admin wallet, and the Verify connection. Price ordering must remain `price(2) < price(5) < price(10)`. Initial documented rates are 5000, 6000, and 8200 basis points out of 10000. Read current settings rather than assuming those values are globally immutable.

Reported test prices of 0.0000002, 0.0000005, and 0.000001 ETH are test settings, not production USD conversion rates. Referral participation and any tax obligations remain the user's responsibility. Payments depend on actual direct-referral activity and do not guarantee income.

See [Referral Contract](#) and [Verify](#).

6 Smart Contracts

6.1 Contract suite

Contract	Responsibility	Specified network
FOSNFT	NFT mint, purchase payment, admin balance, free-collectible reserve	Robinhood
Processor	Authorized collectible registration, cycle state, selection, allocations	Robinhood
Referral	Registration, plans, upgrades, and direct-referral payments	Robinhood
Verify	Separate ERC-721 certificates, origin registry, validity status	Robinhood
VRF/CCIP integration	Carry cycle requests to Base and return VRF results	Robinhood and Base

Only FOSNFT may register collectible mint records in Processor. Referral is independent of both contracts. Verify is optional and may be connected to FOSNFT and Referral; Processor has no direct Verify connection in the supplied structure.

6.2 Three independent collection types

Collection	Type ID	Nominal price	Admin fee	Admin per paid mint	Pool contribution per collectible	Full 500-collectible pool
Buddy	3	\$3	5%	\$0.15	\$2.85	\$1,425
Samurai	13	\$13	3%	\$0.39	\$12.61	\$6,305
Noble	25	\$25	2%	\$0.50	\$24.50	\$12,250

Dollar amounts describe the product model, not a dollar-denominated contract balance or guaranteed allocation. Payments and allocations use native ETH. Read the configured on-chain price before sending a transaction; the production USD-to-ETH conversion policy is not supplied in the contract specification.

All three target 500 collectibles, with 17 selected collectibles and 50/30/20 group allocation. There is no specified 100-cycle lifetime cap. Only the initial weights and price labels are described here; per-cycle stored settings and current on-chain prices control execution.

6.3 Standards and authority

NFT/certificate documentation uses ERC-721. Do not infer ERC-721A, ERC-1155, UUPS upgradeability, timelocks, or audit status from older drafts. Referral owner controls include prices, split basis points, admin wallet, and Verify connection. Verify owner controls include approved minters, origin types, URI configuration, admin wallet, and revocation.

6.4 Deployment status

Robinhood Chain ID **4663** is reported by the supplied specification and corroborated by the public network registry. The specification provides incomplete addresses for some components and does not supply the final Base-side bridge deployment. Accordingly this reference does not publish operational FOS contract addresses, ABIs, RPC credentials, or callback selectors. The named functions in individual pages are specification references, not a replacement for a verified ABI.

Genesis allowlist and growth campaign contracts remain design material in the [roadmap](#), not confirmed members of this deployed four-contract suite.

7 Verify Contract

Verify issues a separate ERC-721 certificate for supported official events. It does not hold allocation pools, run referrals, pay allocations, or perform recipient selection.

7.1 Origins

ID	Constant	Use
1	ORIGIN_FREE_MINT	Free mint/campaign
2	ORIGIN_LOTTERY	Official FOSNFT cycle mint
3	ORIGIN_STORY	Story/content
4	ORIGIN_ADMIN	Manual admin issuance
5	ORIGIN_REFERRAL	Initial referral registration

Owner may define or enable/disable origin types using `setOriginType`.

7.2 Issuance and authorization

- `adminMint(to, originTypeId, originId, uri)` is limited to owner or `adminWallet`.
- `mintFromContract(to, originTypeId, originId, uri)` requires `approvedMinters[msg.sender] == true`.
- Owner authorizes a source with `setApprovedMinter`; the source sets its Verify connection with `setVerifyContract` under its own access controls.
- Ordinary wallets cannot directly mint certificates.

FOSNFT's connected mint path issues origin type 2, `originId = fosTokenId`, and the supplied URI. Referral's connected registration path issues origin type 5, origin ID 0, and empty URI. The specification does not issue a new certificate on referral upgrade. Processor has no direct Verify integration in the supplied design.

7.3 Records

Token IDs begin at 1 and increase. `getNftInfo` exposes the record; `totalMinted` gives the issuance count.

Field	Meaning
<code>exists</code>	Record is registered
<code>active</code>	Current validity flag
<code>originType</code>	Origin category
<code>originContract</code>	Source contract or issuing caller
<code>originId</code>	Source event/asset identifier
<code>minter</code>	Initial recipient of the Verify NFT
<code>createdAt</code>	Issuance timestamp

`minter` is historical provenance, not current ownership. Read ERC-721 ownership for the current holder. A certificate and its linked FOSNFT are separate tokens.

7.4 Check current status

Code	Status	Meaning
0	NOT_REGISTERED	No record

Code	Status	Meaning
1	OFFICIAL_ACTIVE	Official and currently active
2	OFFICIAL_REVOKED	Officially issued but revoked

isOfficial(tokenId) checks existence only. It may return true for a revoked token. Interfaces must use getStatus to display current validity.

Only owner may revoke(tokenId, reasonCode). Revocation sets active = false; it does not burn the NFT, transfer ownership, or erase the record/history. Owner also manages approved minters, origins, admin wallet, and URI configuration. adminWallet can issue manual certificates but is not described as having owner-only revocation authority.

8 Main Application Integration

This Next.js repository is the documentation website. Older PHP paths and wallet functions described in previous drafts are main-application design references; their implementation is not present here. The supplied contract update defines the following integration requirements.

8.1 Wallet and network

Use Robinhood Chain for user transactions. Confirm eth_chainId matches the selected environment: mainnet 4663, testnet 46630. A Base wallet balance or Base Sepolia chain ID is not a substitute. Obtain RPC/explorer settings from the [network reference](#), keeping environments separate.

8.2 Mint interface

1. Read collection type, cycle state, next collectible position, and exact configured ETH price.
2. Show nominal dollar labels separately from the required ETH transaction value.
3. For positions 10, 100, 200, 300, and 400, send zero purchase value and show the wallet's network gas estimate.
4. Submit buyAndMint(lotteryType, tokenURI) using the verified FOSNFT ABI and address.
5. Wait for a successful receipt, then refresh NFT/collectible data. Handle next-position races by refreshing and asking the user to sign a new transaction.
6. After collectible 500, display pending randomness/delivery until the cross-chain result and selection finalization are confirmed. Do not show a receipt as completed allocation settlement.

8.3 Referral interface

Display current plan prices and commission rates from the contract. Registration selects 2/5/10 and a valid direct referrer or zero address. Upgrade options are only higher plans. Read getUpgradeCost immediately before signing. Explain that the referrer's current plan sets the split, and upgrades retain the original referredBy.

Show direct payments separately from pendingWithdrawals. Referrers claim deferred ETH with claim(); admin withdraw() is a different permissioned operation. Refresh balances and plan status after confirmed receipts.

8.4 Certificate interface

Use Verify getStatus for current validity, getNftInfo for origin details, and ERC-721 ownership for the current holder. Do not call a revoked certificate active just because isOfficial is true. Show original recipient separately from current owner.

8.5 Leaderboard and campaigns

The retained scoring design is referral +1, Buddy +2, Samurai +4, Noble +6. Upgrade points, free-mint points, time-period ranking, and campaign snapshots require the final app/indexer policy; the supplied contract summary does not define all of them. `referralCount` and `purchaseCount` are evidence for indexing, not a complete specified leaderboard ABI.

8.6 Operational views

Show collection/cycle identifiers, Robinhood transaction hashes, pending randomness, Base VRF request correlation, CCIP delivery state, selected collectible IDs, attempted payments, and deferred claims. Filter logs by chain and contract version; validate server responses and never ask users for wallet private keys.

9 Security & Validation

The supplied contract summary describes intended behavior. This repository does not include Solidity sources, final ABIs, independent audit reports, or evidence of a deployed VRF/CCIP lane. An audit, timelock, multisig, or public retry function must not be claimed as completed solely because it appeared in an older draft.

9.1 Authority and funds

Referral owner can change prices, percentages, admin wallet, and Verify connection. Verify owner controls authorized minters, origin types, URI configuration, and revocation. Cycle settings are stored per cycle; updates generally apply to future cycles, with permitted empty-cycle updates. Publish real deployment roles and settings after verification.

The interface does not hold private keys, but contracts hold pool funds, free-position reserves, and deferred payment liabilities. Native ETH is volatile; automation does not remove smart-contract, network, or regulatory risks.

9.2 Required contract checks

- Registration: once per wallet, registered non-self referrer, zero-address path, exact payment, fixed `referredBy`.
- Plans: strict price ordering, higher-plan upgrades only, current difference payment, referrer's current basis-point rate, no retroactive payment changes.
- Funds: admin withdrawable balance excludes pending claims and FOSNFT's remaining free-collectible reserve.
- Mint: only FOSNFT registers in Processor; exact contribution matching; independent collection cycles; free positions send zero purchase value.
- Selection: double-weighted entries are removed together after selection; 17 distinct collectibles; repeated wallet recipients allowed; rounding residue assigned to primary.
- Claims: failed pushes create correct liabilities; successful claims cannot repeat; transfers and claims resist reentrancy.
- Verify: only approved issuers; origin linking; initial recipient versus current owner; revoked records remain official but inactive.

9.3 Cross-chain checks

Verify the actual CCIP-supported lane and configured routers/senders, Base VRF environment and authorization, cycle/request correlation, replay/stale response rejection, and exactly-once finalization. Test delayed messages, failed request delivery, failed return delivery, unavailable fee balances, and rejected recipient transfers.

Keep a pending cycle visible when randomness is unavailable. Do not substitute block timestamps, prevrandao, or an operator-selected result for the specified VRF design. A supported retry/recovery path and the fee-funding policy must come from the final implementation; the specification does not supply their callable ABI.

9.4 Operational readiness

Use verified full addresses and ABIs per environment, verify deployed bytecode and role owners, check RPC chain ID, and monitor both chains. Multisig/timelock protection, unit/invariant/fuzz tests, independent audit, and privacy/access controls are recommended checks, not verified features of this documentation repository.

No implemented 365-day liquidation guarantee, fixed 18× gas multiplier, or no-risk claim is supported by the new contract summary.

10 Community & Growth

10.1 Recognition points

Action	Retained product score
Valid direct referral registration	+1
Buddy purchase	+2
Samurai purchase	+4
Noble purchase	+6

Referral referralCount records direct registrations; purchaseCount[user][type] records collection activity. The final app/indexer must define the ranking period, free-position treatment, and tie-breaking. Referral upgrades change the plan and commission rate, not the number of direct registrations; no new upgrade-points rule is supplied.

Points are recognition metrics and are distinct from collectible selection weight and referral commissions. Changing a referral plan does not change the 1–100 double-weight rule in a selection.

10.2 Genesis access

The retained roadmap proposes social verification through X, Discord, YouTube, and Telegram, followed by a gas-only Genesis campaign for eligible wallets. Actual launch dates, quotas, proof format, and early-adopter benefits require campaign configuration. Genesis eligibility must not be confused with the five ordinary cycle positions that are free for the buyer.

10.3 Growth campaign

The planned separate campaign targets 5,000 users and a 30-day window, with milestones and a top-three 50/30/20 split. Activation, funds, deployment, and snapshots are not verified in the new contract summary. Any funding must use available admin receipts after free-position reserves and deferred user liabilities, never Processor’s cycle pool.

See [Leaderboard](#), [Campaigns](#), [campaign design](#), and the [16-phase roadmap](#).

11 Campaign & Leaderboard Design

The existing roadmap describes a separate growth campaign targeting 5,000 users, a 30-day window, milestones \$20k/\$50k/\$100k/\$200k/\$500k/\$1M/\$2M, and 50/30/20 allocations to the top three leaderboard accounts. The new contract specification does not supply a campaign ABI, deployment,

automatic activation proof, or funded milestone state. Those remain design targets rather than live guarantees.

11.1 Funding boundary

Only available admin funds may be committed to a campaign. Referral admin receipts vary with the direct referrer's current \$2/\$5/\$10 plan (50%/40%/18%), or 100% without a referrer. FOSNFT retains collection-specific admin fees (5%/3%/2%) but must reserve funding for remaining free positions. Neither Processor pools nor deferred user payments may be redirected as campaign revenue.

The new summary does not establish automatic routing to a campaign vault. Publish that adapter/funding implementation before treating it as active.

11.2 Required campaign specification

Confirm the user-count source, activation authority/date, funding window, incremental versus cumulative milestone funding, unique snapshots, tie-break rules, rounding, claim eligibility, and recovery behavior. A dollar milestone table is arithmetic, not evidence that funds exist. Do not allocate the same funded balance again at multiple milestones.

11.3 Leaderboard

Retained product scoring is direct referral +1, Buddy purchase +2, Samurai +4, Noble +6. The supplied summary provides referralCount and purchaseCount; it does not establish an on-chain global ranking or new points for referral upgrades. Verify the app/indexer scoring and snapshots before making allocation settlement claims.

The campaign is separate from the 500-collectible cycle's 17-recipient selection. Its planned three-account 50/30/20 split must not be confused with the cycle's 1/6/10 collectible groups. See [Growth Campaign](#).

12 Public Roadmap

This is a development roadmap, not evidence that every feature has launched. The current core specification places user-facing contracts on Robinhood Chain, uses Base VRF through CCIP, and provides three referral plans. Genesis/campaign contracts, launch status, and future features need separate deployment confirmation.

12.1 Phase 1 — Community Introduction & Whitelist Access Program

This phase focuses on onboarding the first members of the ecosystem and establishing a foundation of verified early supporters.

- Launch of the official website and ecosystem overview
- Introduction of the FOS Protocol across the community channels
- Opening of the Whitelist Access Program for early adopters
- Participation-based tasks designed to help users learn about the platform and engage with the community
- Whitelisted users receive priority access to the next participation stage

The Whitelist is a contribution-based access layer, not a financial feature. It is designed to recognize genuine community participation rather than speculation.

12.2 Phase 2 — Genesis Free-Mint Access (Whitelisted Users Only)

Following the whitelist period, a dedicated Free-Mint Campaign is opened exclusively to verified members.

- Whitelisted users may mint selected FOS Digital Collectibles
- Participation requires network gas only (no protocol fee during this window)
- Genesis assets and any Processor cycle eligibility must follow the separately published campaign configuration
- The objective of this phase is to reward early contributors and establish the first active user base

This phase reinforces our principle of accessibility and loyalty recognition, while remaining fully aligned with the protocol's product-first architecture.

12.3 Phase 3 — Limited-Access Application Release (Stability Validation Stage)

Before enabling full public access, the platform enters a restricted availability period.

- The application is opened to a limited number of users
- System performance, UX behavior, and contract execution are monitored
- Feedback from real-world usage is incorporated
- Technical and operational stability are validated

Only after the protocol demonstrates reliability in this controlled environment does the platform proceed to full availability.

12.4 Phase 4 — Public Launch & Open Access

After stability validation, the protocol becomes publicly accessible.

- Full public access to the application
- Ongoing onboarding of new participants
- Continued growth through community-driven initiatives
- Expansion of ecosystem transparency and engagement tools

Cycles follow their stored settings and asynchronous randomness lifecycle. Authorized owner controls and oracle/bridge operations remain part of the documented architecture.

12.5 Phase 5 — Growth Campaign (Activated at 5,000 Users)

Once the ecosystem reaches 5,000 verified participants, the protocol activates a time-limited Growth Campaign Layer as defined in the technical and economic documentation.

- The campaign operates for a predefined time window
- A separately configured campaign may receive available admin funds after free-position reserves and deferred-user liabilities
- Distributions occur based on milestone thresholds
- Allocation follows a published structure and is tied to leaderboard performance and contribution ranking

Participation in the campaign is optional, does not affect product purchase consideration, and remains strictly promotional in nature.

12.6 Phase 6 — FOS Lab: Innovation Playground

FOS Lab is where new ideas are born, tested, and refined together with the community.

This is a controlled experimental environment where we introduce and test new participation mechanics, new pool structures, and new interaction models — without affecting the stability of the live system.

What users get: - Early access to experimental features - A safe testing space separate from the main platform - Opportunities to share feedback and influence feature development - A community-driven innovation loop

FOS Lab is where ideas evolve into real features.

12.7 Phase 7 — Desktop Experience Upgrade

FOS expands beyond mobile with a fully-optimized desktop experience designed for active users and long session interaction.

What users get: - A redesigned interface for large screens - Enhanced navigation and workspace layout - Better visibility of cycles, history, and activity - A smoother, more efficient user workflow

Whether you're on mobile or desktop, FOS delivers a consistent, high-quality experience.

12.8 Phase 8 — Expanded Wallet Support

FOS becomes even more accessible by supporting a wider range of Web3 wallets.

What users get: - Support for more browser and mobile wallets - WalletConnect integration for universal access - Smoother onboarding for new users - Future-ready design for smart-wallet standards

Our goal is simple: use the wallet you like — and connect instantly.

12.9 Phase 9 — Multi-Chain Expansion

FOS expands to compatible EVM networks through a security-first rollout strategy.

What users get: - Access to FOS across multiple chains - Faster transaction options depending on network - Broader ecosystem participation - Seamless experience across supported networks

Expansion follows a careful, audit-guided deployment path to ensure stability and trust.

12.10 Phase 10 — Dashboard V2: Transparency Reinvented

Dashboard V2 introduces a new level of clarity, insight, and system visibility.

What users get: - Deeper participation and engagement analytics - Clear contribution and cycle activity views - Historical timeline of ecosystem behavior - Real-time system and transaction insights

Dashboard V2 strengthens our commitment to openness, fairness, and transparency.

12.11 Phase 11 — Advanced User Profile

Built on the foundation of Dashboard V2, the Advanced User Profile transforms the wallet into a living participation identity.

What users get: - Profile identity with optional avatar - Personal activity timeline - Participation footprint and engagement history - Non-financial reputation indicators

Profiles reflect how users grow inside the ecosystem, not just what they hold.

12.12 Phase 12 — Priority Access Mode

Active and committed users gain access to exclusive early-entry opportunities in selected cycles.

What users get: - Early participation windows for top contributors - Access opportunities based on engagement and consistency - Recognition through participation effort — not entitlement - Occasional access to special pools and events

This feature celebrates loyal members who help grow and support the ecosystem.

12.13 Phase 13 — Loyalty Framework

A structured loyalty layer that acknowledges long-term participation and positive ecosystem behavior.

What users get: - Progression based on engagement consistency - Recognition for holding collectibles - Participation-based milestones - Community-value alignment

The loyalty framework is built on trust, contribution, and continuity.

12.14 Phase 14 — Time-Based Digital Collectibles

Collectibles that evolve over time through progressive engagement and responsible holding behavior.

What users get: - Unlockable attributes over time - Holding-based collectible progression - Optional transformation mechanics - A visual reflection of participation history

Time becomes part of the collectible's story and identity.

12.15 Phase 15 — Missions & Group Participation Quests

Participation becomes collaborative and goal-driven through personal and group missions.

Individual Missions: - Engage across cycles - Complete activity milestones - Unlock progression achievements

Group (Squad) Missions: - Team-based participation journeys - Shared targets and milestones - Cooperative progress experiences

This transforms ecosystem participation into an interactive community adventure.

12.16 Phase 16 — Community Voting & Participation Governance

A gradual introduction of responsible user-driven decision participation.

What users get: - Voting on selected feature proposals - Participation-aligned voting conditions - Optional gated voting groups - Feedback-guided evolution of the ecosystem

Governance is introduced thoughtfully — step-by-step, with stability first.

13 Team and Governance Scope

The earlier edition described founder, Solidity engineering, and prospective advisory roles. This revision does not invent individual identities, audit sign-offs, or implemented multisig/timelock deployments. Document actual owners and admin roles from the final contracts. Referral configuration authority and Verify revocation authority are described above; they must not be replaced by a blanket claim that no rules can change.

14 Risks, Eligibility, and Legal Notice

NFT acquisition does not guarantee an allocation, income, liquidity, resale value, or capital appreciation. Selection is weighted and probabilistic. The nominal dollar examples are explanatory; actual configured ETH prices, integer rounding, balances, and completed transactions control on-chain results.

Users face smart-contract, key-management, network, oracle, bridge-delivery, market-volatility, and regulatory risks. Cycle pools, free-position reserves, and deferred payments are held and accounted for in contracts. A non-custodial interface does not mean funds never enter a contract or that loss is impossible.

The existing project access policy requires adults and restricts the United States, Canada, United Kingdom, EEA, China, and sanctioned jurisdictions. Do not bypass those restrictions. Project terminology does not independently establish legal status. Obtain qualified jurisdiction-specific advice; the technical specification is not a legal opinion.

Read the project's current Terms of Service and Privacy Policy. Final blockchain transactions cannot be reversed by this documentation interface. Network fees are spent even for reverted transactions. Completed sales and any statutory consumer rights depend on applicable terms and law; this document does not create or waive legal rights.

15 Future Vision and Expandability

The retained sixteen-phase roadmap includes Genesis access, gradual rollout, growth incentives, experiments, desktop improvements, expanded wallets, additional networks, dashboards, profiles, priority access, loyalty, evolving collectibles, missions, and prospective governance. These are plans rather than proof of launch. A future governance or chain deployment requires its own specification, security review, roles, and rollout evidence.

The specified Base randomness leg is infrastructure for Robinhood selection processes; it is not evidence that the user's minting transaction occurs on Base or that arbitrary multi-chain pool deployments already exist. New collection tiers and campaign adapters must preserve pool isolation, free-position funding, and user payment liabilities.

16 Appendix: Fee Funding and Failure Cases

Event	Specified behavior or validation requirement
Incorrect purchase value	Ordinary mints require exact configured price; free positions require zero purchase value
Next position changes before confirmation	Refresh the next position and amount; rely on actual contract checks
Insufficient free-position reserve	Funding must cover the configured contribution; admin withdrawal must preserve remaining reserves
VRF or CCIP delay	Cycle awaits a valid correlated result; no immediate completion promise
Repeated or stale result	Final implementation must validate sender, cycle correlation, and exactly-once finalization
Selected wallet rejects ETH	Record a deferred amount in <code>pendingWithdrawals</code> for <code>claim()</code>
Referrer rejects ETH	Record a deferred amount for that referrer to <code>claim()</code>
No referrer	Full registration or upgrade payment is allocated to admin
Verify certificate revoked	Token and history remain; current status becomes <code>OFFICIAL_REVOKED</code>

No fixed 18-times gas multiplier, unconditional fee reimbursement, deployed public retry ABI, or implemented 365-day liquidation guarantee is established by the supplied update. Bridge/oracle fee sources, retry authority, and recovery behavior need final implementation validation.

17 Appendix: Technical Boundaries and Glossary

Term	Meaning
Robinhood Chain	User-facing project network; mainnet registry ID 4663
Base VRF side	Specified service-chain integration where Chainlink VRF produces randomness
CCIP	Transport for requests/results across the specified chains
Cycle	Collection-scoped set of up to the stored capacity, targeted at 500 collectibles
Green Eye	Final target-position collectible that closes the cycle and initiates randomness processing
Collectible	Registered selection entry; distinct from wallet-level uniqueness
Weight	Two selection entries for positions 1–100, one for later positions
Deferred claim	Recorded payment liability after a direct transfer fails
Basis points	Rate units out of 10000; 8200 bps is 82 percent
Verify certificate	Separate ERC-721 origin NFT with official existence and active/revoked status

The specification supports ERC-721 certificates and native ETH payments. It does not establish compiler-version pins, ERC-721A, UUPS proxies, an Ownable2Step/timelock deployment, or an independent audit. Function names are specification references, not an ABI. Exact deployed return types and callback signatures must come from verified code.

18 References and Revision Provenance

1. Supplied project contract specification, preserved in docs/specifications/project-contracts-fa.md.
2. Current canonical protocol and contract pages in content/docs; consolidated references in Fos doc.
3. Ethereum network registry: [Robinhood mainnet](#) and [testnet](#). These corroborate network meta-data, not project deployment.
4. Official documentation linked by that registry: [Robinhood Chain](#).
5. [Chainlink VRF documentation](#) and [CCIP documentation](#). Service documentation is not proof of a deployed FOS lane.
6. Historical December 2025 whitepaper retained byte-for-byte in public/archive/whitepaper-2025.pdf, including its original research references.

The original PDF was produced by LaTeX, but its exact original editable source is not recoverable from the PDF alone. `original-facsimile.tex` preserves its pages visually by embedding the unchanged PDF; it is not presented as the recovered original typesetting source. This edition's `whitepaper.tex` is a complete editable source for the revised PDF.