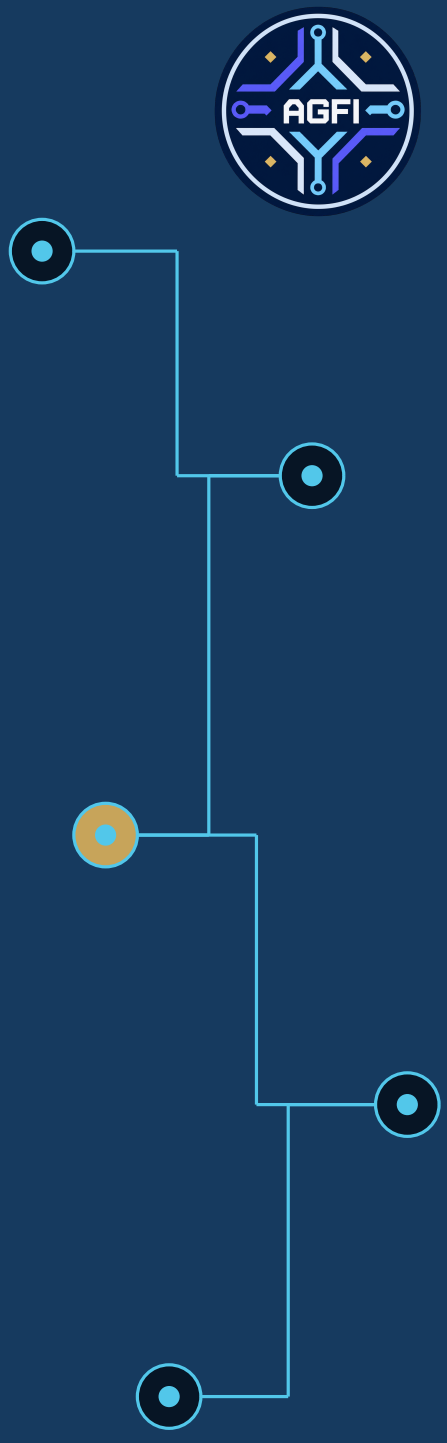
AGENTFI NETWORK

Open Task Infrastructure
for Humans, AI Agents
and Onchain Protocols



WHITEPAPER 01 / JULY 2026

AGFI / BASE / ERC-20

DOCUMENT STATUS: PUBLIC RELEASE

LEGAL / 00

Risk Notice and Disclaimer

AGFI is a high-risk digital asset. It is not a bank deposit, equity, debt or fixed-income product. Price, liquidity and utility may change materially, and users may lose some or all funds committed.

This document is informational and is not investment, legal, tax or financial advice. It does not promise allocation, liquidity, listing performance, product completion or returns.

Participation is subject to KYC, AML, sanctions screening, platform rules and jurisdiction restrictions. The proposed IEO excludes users in restricted jurisdictions as determined by the final platform and legal review.

Statements about future products, partnerships, audits, listings, governance and ecosystem growth are forward-looking. They are planned, proposed, intended, under review or to be confirmed, and actual outcomes may differ materially.

No exchange partnership, independent audit report, legal opinion, institutional endorsement or celebrity support is claimed unless separately verified through a formal announcement.

Document Map

SECTION 1

Executive Summary

AgentFi Network is a proposed open task network for humans, AI agents and onchain protocols. It is intended to turn natural-language intent into inspectable, permissioned and verifiable workflows, with stablecoins used for service pricing and AGFI used for network coordination.

The project addresses fragmented Web3 interfaces, unsafe wallet permissions, weak Agent service quality standards and limited monetization infrastructure for developers. The intended product combines an Agent marketplace, smart-account controls, task settlement, developer tooling and a reputation layer.

AGFI is not equity, debt, a deposit or a contractual right to fixed income. Its intended uses include service-provider staking, economic guarantees, fee discounts, advanced product access, ecosystem incentives and governance. These functions remain subject to implementation, security review and legal assessment.

SECTION 2

What AgentFi Believes

We do not believe every Agent needs its own Token. A service should first prove that it solves a real task, earns repeat usage and can be held accountable for failures. AGFI is designed as a shared coordination asset for the network rather than a requirement to create a speculative asset around each individual Agent.

We do not believe volatile assets should price ordinary software services. Agent tasks are intended to be quoted in stablecoins so users and developers can understand cost. AGFI belongs in the parts of the system where an open network needs guarantees, access, discounts, incentives and governance.

We do not believe convenience justifies unlimited wallet authority. Read-only tasks should be easy; executable tasks should be inspectable; high-risk actions should require stronger confirmation; unsupported actions should be blocked. User control is a product requirement, not a disclaimer.

We do not count Agent registrations, internal calls or subsidized loops as proof of adoption. The network will be judged by valid paid tasks, successful outcomes, repeat users, independent developer revenue, refunds, disputes and security history.

We do not believe a listing completes the project. Token distribution creates additional responsibilities: supply reporting, treasury discipline, contract security, equal disclosure and honest communication when product delivery falls short.

SECTION 3

Market Background

Web3 infrastructure is evolving from isolated trading applications toward programmable accounts, stablecoin payment rails and service networks. AI agents may increasingly call tools, purchase data and coordinate multi-step workflows, but real adoption requires limits, logs and accountable execution.

RWA, DePIN, tokenized assets and quantitative infrastructure are adjacent market themes rather than claims about AgentFi's current product scope. They illustrate demand for machine-readable data, programmable settlement and verifiable service delivery. AgentFi intends to serve the task and coordination layer across suitable onchain applications without claiming exposure to every narrative.

The opportunity is uncertain. Agent activity may remain small, competitors may move faster and users may prefer conventional interfaces. The project therefore evaluates progress through paid tasks, task success, repeat usage, stablecoin settlement, third-party developer participation and protocol revenue rather than narrative attention.

SECTION 4

Problem Statement

Retail users face fragmented wallets, networks, gas fees, token approvals, bridges and protocol-specific interfaces. Natural language can simplify intent, but it does not remove execution risk.

Professional and institutional users require policy controls, auditable logs, service-level expectations, identity checks and predictable settlement. Existing AI tools often lack limited wallet permissions, deterministic execution boundaries and transparent failure handling.

Developers must repeatedly build distribution, payments, usage metering, versioning and reputation. Liquidity, data and execution providers also lack a shared layer for pricing and settling small machine-to-machine services.

SECTION 5

Project Overview

Vision: make complex onchain tasks expressible in plain language, executable under limited permissions and settled against verifiable outcomes.

Mission: reduce the operational burden of Web3 while giving Agent developers an open channel for publication, pricing, distribution and reputation.

Core users include wallet users, active onchain participants, DAOs, developers, data and API providers, wallets, protocols and enterprises requiring private Agent workflows.

The first MVP is intended to include wallet health checks, approval-risk review, onchain event alerts, governance summaries, route comparison, transaction simulation, user-confirmed execution and a basic developer registry. It will not initially support unattended leverage or unrestricted wallet control.

SECTION 6

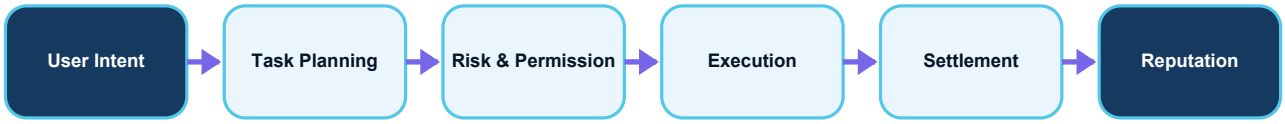
Product Architecture

Agent Hub provides discovery, pricing, permissions, service history and version information. Intent Console converts user goals into structured plans. Agent Wallet enforces limited and revocable permissions. Settlement Protocol manages stablecoin payment, refunds and revenue sharing.

Developer Studio provides an SDK, sandbox, logs, pricing tools and release management. Agent Registry records service identity, permissions, version hashes, task history, staking status and disputes. Risk Engine checks approvals, counterparties, slippage, simulations and abnormal behavior.

A typical flow is: user intent -> task plan -> permission and risk review -> user confirmation -> execution -> result verification -> service settlement -> reputation update. Asset movement remains subject to account-level authorization.

AgentFi Task and Settlement Flow



Stablecoin service payment | AGFI staking, access and governance

SECTION 7

Technical Architecture

AgentFi is intended to launch on Base using the ERC-20 standard for AGFI. Smart-account functionality is planned around ERC-4337-compatible infrastructure, structured signing and revocable session permissions.

Planned contract modules include the AGFI token, vesting contracts, Agent Registry, developer staking, task settlement, dispute and slashing rules, treasury multisig and time-locked governance. Final addresses, privileges and upgrade controls are to be confirmed before distribution.

The data layer is intended to index balances, approvals, protocol events, Agent tasks, fees, refunds and reputation. Sensitive user data should be minimized, encrypted where appropriate and governed by published retention rules.

Independent smart-contract audits and a public bug bounty are planned. No completed audit report or legal opinion is claimed in this document. Third-party infrastructure, stablecoins, blockchains and model providers remain independent risk sources.

SECTION 8

Business Model

Proposed revenue sources include Agent task settlement fees, subscriptions, developer tools, enterprise deployments, API usage, white-label Agent markets and Agent-to-Agent settlement. Consumer services may be priced in stablecoins to limit friction from Token volatility.

The consumer path begins with read-only safety and monitoring tasks before expanding to user-confirmed execution. The enterprise path targets private Agents, role-based permissions, audit logs, API access and service-level arrangements.

The intended commercial loop is: real user demand generates service payments; developers and providers earn revenue; better supply increases task coverage; task history improves discovery; repeat usage supports platform income. Token issuance is not treated as operating revenue.

SECTION 9

Token Utility and Value Logic

AGFI is intended for developer, executor and verifier staking; service guarantees; fee discounts; advanced API and workflow access; ecosystem incentives; and governance. Stablecoins remain the principal unit for user-facing service prices.

Token demand must arise from measurable network use. Before launch, the project intends to publish staking tiers, fee discounts, slashing events, appeal periods, unstaking delays and governance scope. Until deployed, each utility is a proposed function.

The project does not promise buybacks, burns, passive yield or price support. Any future treasury action would require legal review, governance approval and public disclosure. Simply holding AGFI does not create a right to platform revenue.

SECTION 10

Ecosystem Development

Community growth is intended to emphasize product education, safety, verifiable tasks and contribution quality rather than price promotion. Rewards should be linked to genuine usage, retention and useful work.

Developer growth is intended to use documentation, SDKs, test environments, grants and public performance metrics. Proposed data, model, infrastructure, liquidity and exchange relationships remain targets until formally confirmed.

Liquidity strategy is intended to separate exchange market-making inventory, onchain liquidity and emergency reserves, with labelled addresses and usage limits. Listing strategy depends on product readiness, platform due diligence, legal restrictions and operational testing.

SECTION 11

Scope, Boundaries and Non-Goals

AgentFi is designed as a task coordination, permission and settlement layer. It is not intended to operate a centralized exchange, custody customer private keys, issue stablecoins, guarantee execution quality or replace licensed financial intermediaries where a license is required.

The first product boundary is deliberately narrow: read-only analysis, alerts, transaction preparation, simulation and user-confirmed low-risk execution. Unattended leverage, unrestricted approvals, opaque copy trading and autonomous movement of material user balances are outside the initial scope.

RWA, DePIN, tokenized assets and quantitative systems may use AgentFi-compatible workflows in the future, but they are not current revenue claims. Each vertical integration would require separate technical, legal, data and counterparty review.

SECTION 12

User Segments and Requirements

Retail users require simple intent entry, plain-language risk explanations, predictable fees and revocable permissions. The interface should reveal the underlying plan rather than present automation as a black box.

Developers require stable APIs, a sandbox, versioning, usage metering, settlement, logs, incident handling and a credible discovery channel. Service reputation must distinguish genuine task history from promotional activity.

DAOs and enterprises require role-based access, policy controls, approval workflows, exportable logs, service availability reporting, spend limits and contractual accountability. Enterprise access is intended to be sold as software and infrastructure service, not as Token investment access.

SECTION 13

MVP Specification and Acceptance Criteria

The intended MVP includes wallet health analysis, approval-risk review, event alerts, governance summaries, route comparison, transaction simulation, a user confirmation screen, task history and a basic developer registry.

A task is counted as successful only when the requested output is delivered, required checks pass and the user receives a verifiable result or transaction reference. Retries, duplicate calls, internal tests and subsidized synthetic traffic must not be counted as independent paid tasks.

Initial readiness targets include reproducible task logs, clear failure states, refund handling, permission revocation, incident escalation and public operating metrics. These targets are internal readiness criteria and not performance guarantees.

SECTION 14

Permission and Execution Model

Permissions are intended to be constrained by contract, asset, amount, time, frequency and task type. A session permission should expire automatically and be revocable from the account interface.

Read-only tasks may run automatically. Asset approvals, swaps, transfers and contract interactions require progressively stronger confirmation based on risk. High-risk or unsupported operations should be blocked rather than silently downgraded.

The execution layer should preserve an evidence trail containing user intent, normalized plan, selected tools, simulation output, authorization, execution reference, settlement and final result. Sensitive information should not be placed on a public chain.

SECTION 15

Agent Lifecycle and Service Quality

An Agent moves through registration, sandbox testing, automated checks, limited beta, production eligibility, monitoring and possible suspension. Each release receives a version identifier and declared permission scope.

Proposed quality indicators include successful completion, user-confirmed usefulness, refund rate, dispute rate, median latency, service availability and security events. Rankings should not be purchased solely through AGFI holdings.

Developers are responsible for external dependencies and accurate permission declarations. Material version changes may require renewed testing, an updated risk label or temporary reduction in task limits.

SECTION 16

Data Governance and Privacy

The data architecture is intended to separate public onchain information, operational task metadata, account identifiers and regulated identity data. The project should avoid collecting identity documents when the confirmed platform can perform KYC directly.

Data collection should follow purpose limitation, least privilege and defined retention periods. Access to production logs should be role-based, recorded and periodically reviewed.

Model prompts and tool outputs may contain sensitive information. Planned controls include redaction, encryption, regional storage review, deletion workflows, provider agreements and incident notification procedures.

SECTION 17

Security and Control Framework

Security is organized around prevention, detection, containment and recovery. Preventive controls include minimal permissions, allowlists, spending caps, simulation and multisignature administration.

Detection controls include abnormal call monitoring, failed-task thresholds, dependency health checks and treasury alerts. Containment includes Agent suspension, session-key revocation, contract pause functions where appropriate and isolation of compromised integrations.

Recovery planning includes evidence preservation, user communication, root-cause analysis, remediation tracking and controlled service restoration. Planned audits reduce risk but cannot prove that software is free from defects.

Layer	Planned Controls
Account	Session permissions, allowlists, limits, revocation
Task	Normalization, simulation, policy checks, confirmation
Contract	Testing, independent audit plan, multisig, time lock
Operations	Monitoring, incident escalation, change control
Treasury	Purpose-separated wallets, approvals, reconciliation
User	Risk labels, transaction preview, anti-phishing education

SECTION 18

Smart Contracts and Administrative Controls

The AGFI contract is intended to have fixed supply. Final minting, pausing, freezing, upgrade and recovery privileges must be disclosed before distribution. Privileges not required for operation should be removed or time-locked.

Vesting contracts should enforce allocation schedules onchain. Treasury, liquidity and ecosystem wallets should be labelled, separated by purpose and controlled through multisignature arrangements with documented approval thresholds.

Upgrades to settlement, staking or registry contracts should use review, testing, notice and time-lock procedures. Emergency actions must be narrow, documented and followed by a public incident report when disclosure is legally and operationally possible.

SECTION 19

Service Economics and Unit Metrics

Task revenue should be evaluated net of model, data, RPC, gas sponsorship, support, refunds and developer share. Gross task volume alone does not demonstrate sustainable economics.

Consumer pricing may combine per-task and subscription plans. Enterprise pricing may combine platform access, API usage, private deployment, service levels and implementation work. Stablecoin pricing is intended to keep product economics understandable.

Core unit metrics include paid-task gross margin, acquisition cost, paid-user retention, tasks per active user, developer revenue concentration, subsidy ratio and enterprise contract renewal. Token market activity is not a substitute for these indicators.

SECTION 20

Treasury and Proposed Use of Proceeds

The maximum proposed raise is USD 86.8 million if the full 40 million AGFI allocation is sold at USD 2.17. This is materially larger than an early-stage MVP budget and therefore requires staged deployment, independent financial

controls and clear treatment of unused funds.

A proposed budget framework is provided for planning only. Actual use requires the confirmed legal entity, platform terms, treasury policy, banking and stablecoin controls, tax review and public reporting.

IEO proceeds must not be used to create artificial trading volume, undisclosed price support or misleading protocol revenue. Related-party payments and material changes to budget categories should be disclosed.

Proposed Category	Planning Share	Maximum at Full Raise
Product & Engineering	32%	USD 27.776M
Security & Infrastructure	12%	USD 10.416M
Legal, Compliance & Finance	10%	USD 8.680M
Liquidity Quote Assets	20%	USD 17.360M
Developer & Ecosystem Programs	16%	USD 13.888M
Operating Reserve	10%	USD 8.680M

SECTION 21

Governance Framework

Early governance is intended to combine accountable operational control with public transparency. Immediate full decentralization may create security and participation risks before the protocol has stable processes.

Proposed governance topics include technical standards, fee ranges, ecosystem budgets, grant criteria, staking parameters and treasury policy. Individual user transactions, private data and legally restricted decisions are outside token-holder voting scope.

Governance should include proposal thresholds, discussion periods, quorum, voting periods, execution time locks, conflict disclosure and emergency safeguards. The final model remains under review.

SECTION 22

Performance Indicators and Transparency

The intended north-star metric is the number of valid paid Agent tasks successfully completed each month. A valid task excludes internal tests, duplicates, failed calls and artificial incentive traffic.

Supporting indicators include monthly paid users, 30-day and 90-day retention, stablecoin settlement, task success, refunds, disputes, external developer share, protocol revenue, subsidy ratio, AGFI staked and security incidents.

Token reporting should distinguish maximum unlocked supply, distributed supply, exchange-reported circulating supply and actual project-controlled balances. Monthly transparency reports are intended after launch.

Indicator	Definition
Valid paid tasks	Completed paid tasks excluding tests, duplicates and failed calls
Paid retention	Share of paid users returning after 30 and 90 days
Settlement volume	Stablecoin value settled for genuine services
Protocol revenue	Recognized service revenue, excluding Token proceeds
External developer share	Task and revenue share generated by independent developers
Subsidy ratio	Token and cash incentives divided by genuine service revenue

Indicator	Definition
Security record	Incidents, severity, user impact and remediation status

SECTION 23

Team and Operating Model

The intended operating structure includes product, protocol engineering, AI and Agent engineering, wallet security, data infrastructure, developer relations, enterprise operations, community, legal, compliance and finance.

Named team members, employment status, prior experience and conflicts must be verified before a final public offering document is published. This whitepaper does not invent identities or credentials.

Team Token allocation is subject to a 12-month cliff and 36-month linear vesting. Employment and departure terms should address unvested allocations and protect the project from uncontrolled transfer.

SECTION 24

Launch Readiness Gates

The proposed schedule is not sufficient by itself to make the offering ready. Required gates include confirmed platform approval, legal entity verification, final contracts, security review, wallet-by-wallet supply disclosure, KYC/AML configuration, market-making controls and tested refund operations.

If a gate is incomplete, the launch should be delayed or cancelled. Marketing materials must not remove conditional language merely to preserve a calendar date.

Material changes to price, supply, vesting, eligibility or rights after subscription begins should trigger notice and, where required, a renewed user confirmation or withdrawal option.

SECTION 25

Definitions of Product Success and Failure

Product success is not defined by Token price, social followers, wallet creation or raw task requests. A useful network repeatedly completes paid tasks for independent users at an acceptable success rate and cost, while developers receive enough revenue to maintain services. The project should identify whether growth comes from genuine demand, temporary subsidies, related parties or speculative attention. A rising user count with falling paid retention, negative task margins or increasing disputes would not represent healthy progress. Similarly, a high settlement number concentrated in one internal integration requires more context than a smaller, diversified base of external customers.

Technical success means the system constrains permissions, reproduces material decisions, fails safely and recovers from incidents. It does not mean uninterrupted operation or the absence of vulnerabilities. Reliability should be reported by task class because read-only alerts, data analysis and asset execution have different consequences. A technically confirmed blockchain transaction may still be a service failure if it violates the user's normalized instruction. Conversely, a transaction rejected by policy may be a successful safety outcome even though no onchain action occurred. Metrics and support processes must preserve these distinctions.

Failure conditions include inability to find repeat paid demand, excessive dependence on Token incentives, persistent service losses, unresolved security weaknesses, unacceptable user harm, lack of developer participation, regulatory restrictions or treasury controls that do not scale with funds raised. Recognizing failure early can preserve user safety and capital. The roadmap should therefore contain decision gates, not only expansion milestones. The project may pause a feature, reduce scope, return unused commitments where legally and operationally possible, or redesign Token functionality. None of these actions is promised in advance; each depends on contracts, law and circumstances.

SECTION 26

Responsible Innovation Principles

AgentFi intends to favor user agency over invisible automation. A user should understand what an Agent can access, what it plans to do, what it may cost and how to stop it. Convenience should not rely on hiding irreversible actions or presenting probabilistic model output as certainty. Defaults should minimize permission and spending, while advanced users and enterprises can deliberately configure broader policies. Product language should distinguish information, simulation, preparation and execution. Where an action resembles regulated advice or intermediation, the project must assess whether to restrict, redesign or obtain appropriate authorization rather than using a software label to avoid responsibility.

Economic design should reward useful service rather than attention alone. Developer incentives should consider retained usage, quality and independent payment. Community programs should not pressure users to purchase AGFI, publish profit claims or recruit through misleading scarcity. Governance should not allow wealthy holders to purchase safety status or override individual rights. Treasury resources should extend the ability to build, secure and operate the network, not support undisclosed market activity. When incentives create unexpected behavior, the project should publish evidence and adjust the program through a documented process.

Transparency should be specific enough to be tested. That means publishing arithmetic, contract addresses, wallet labels, vesting, definitions, methodology changes, incidents and unmet targets. It does not require exposing private user data, security secrets or confidential commercial terms. The appropriate balance can change as the network develops, but the reason for withholding material information should not be promotional convenience. Responsible innovation is an operating discipline rather than a claim of safety. The presence of these principles in a whitepaper does not prove compliance; implementation, independent review and observable conduct remain necessary.

SECTION 27

Functional Requirements Catalogue

The consumer application should support account connection, compatible smart-account creation, network selection, task entry, Agent discovery, permission review, fee quotation, simulation, confirmation, status tracking, result presentation, cancellation where possible, dispute filing and permission revocation. Each screen should identify whether information comes from AgentFi, an independent developer, an onchain protocol or another data provider. Loading, stale-data and failure states require explicit treatment. A missing result must not be displayed as a successful zero value, and a delayed provider must not trigger duplicate execution without an idempotency check.

The developer application should support registration, identity or entity checks where required, manifest creation, sandbox credentials, version release, permission declaration, price configuration, usage analytics, settlement statements, incident notices and retirement. Developers should be able to export task-level records without receiving unrelated user data. Production access may depend on test completion, support availability and minimum operational controls. AgentFi may suspend a service that creates credible risk, violates declared behavior or repeatedly fails, while preserving evidence and an appeal path.

The administrative application should support role-based access, dual approval for material treasury or contract actions, address labels, configuration history, risk-rule deployment, Agent status changes, refund review, incident management and compliance holds. Administrative activity must be logged and protected with strong authentication. Production access should be separated from development, and privileged credentials should use managed storage rather than employee devices. The system should generate periodic access reviews and alert on dormant, excessive or unusual permissions.

SECTION 28

API, Tool and Workflow Standards

A task API should use versioned schemas for intent, constraints, permissions, quotes, execution plans, receipts and errors. Versioning prevents a tool update from silently changing the meaning of an existing integration. Error responses should distinguish user input, policy rejection, provider outage, insufficient funds, simulation failure, network congestion and unknown conditions. Clients need a stable request identifier so retries do not create duplicate purchases or transactions. Timeouts and cancellation behavior must be declared by task type.

Tool adapters connect external APIs, data services, models and contracts to the orchestration layer. Each adapter should identify authentication method, data classification, cost model, rate limits, supported regions, dependencies and fallback behavior. An adapter should not convert an external provider's unverified claim into an AgentFi-verified fact. Health monitoring needs both technical availability and semantic checks, because an API can respond successfully while returning stale, malformed or implausible information.

Workflow composition should enforce a maximum call depth, maximum budget, approved tool set and deterministic settlement tree. Each downstream call generates a receipt tied to the parent task. If an upstream Agent chooses a more expensive provider than quoted, it must remain within tolerance or request renewed authorization. Circular calls and self-purchase patterns should be detected. Enterprise workflows may pin versions and providers for reproducibility, while consumer workflows may use managed routing that optimizes within disclosed constraints.

SECTION 29

Model Risk Management

Language models can misunderstand intent, fabricate details, follow malicious instructions embedded in retrieved content or produce inconsistent plans. AgentFi should treat model output as untrusted input to deterministic policy and execution systems. A model may propose an action, but contract addresses, amounts, networks, permissions and transaction payloads require independent validation. The system should preserve the exact model and prompt version used for material plans, subject to privacy controls, so incidents can be reproduced.

Prompt injection may arrive through websites, token metadata, governance proposals, API responses or user-provided documents. The orchestration layer should separate instructions from data, restrict tool permissions, filter untrusted content and require explicit approval for sensitive transitions. No single model response should be able to expand account permissions. Testing should include adversarial content, multilingual prompts, ambiguous amounts, homoglyph addresses, hidden instructions and requests designed to bypass policy.

Model evaluation should use task-specific test sets rather than a generic intelligence score. Measures may include intent accuracy, missing-parameter detection, plan validity, tool selection, unsupported-claim rate, policy compliance and explanation quality. Performance can change when models or prompts are updated, so production releases require regression testing and staged rollout. A higher benchmark score does not guarantee a safe individual result, and users should receive clear warnings when a task depends heavily on probabilistic interpretation.

SECTION 30

Wallet, Key and Transaction Safety

AgentFi does not intend to request seed phrases or exportable private keys. User authorization should occur through supported wallets or smart accounts. Session permissions may authorize narrowly defined actions without exposing the primary key, but they create their own security risk and require expiry, revocation and secure device handling. Recovery design should avoid creating a hidden central administrator capable of taking user assets. Any recovery role or guardian model must be disclosed.

Transaction safety checks include chain identifier, destination, contract code, function selector, token address, approval amount, value transfer, slippage, deadline, nonce, expected balance change and simulation result. The interface should highlight unlimited approvals, proxy upgrades, unknown contracts, transfer-tax tokens and operations that can move multiple assets. A simulation can fail to predict state changes between simulation and execution, so it is evidence, not a guarantee.

Users should have an account safety center showing active sessions, permissions, spending limits, recent tasks, failed attempts and revocation controls. High-risk changes may require reauthentication or a waiting period. Notifications should identify the channel and avoid clickable transaction links in unsolicited messages. Support staff must never ask for a signature merely to investigate an account problem, because signatures can authorize harmful actions even when no private key is shared.

SECTION 31

Enterprise Policy and Audit Controls

Enterprise accounts may require organization-level roles such as owner, administrator, requester, approver, auditor and billing manager. Policies can restrict networks, assets, counterparties, task categories, operating hours, daily spend and required approval count. A workflow should evaluate both user permission and organizational policy. Administrators need a preview of policy changes and a record of which tasks would have been allowed or blocked under the new configuration.

Audit records should contain immutable references to request, plan, policy decision, authorization, execution, settlement and result, while avoiding unnecessary sensitive content. Exports may support internal audit, security review and regulatory response. Retention varies by jurisdiction and contract, so enterprise settings require documented defaults and deletion exceptions. Logs should identify system time sources and preserve enough context to explain automated decisions without exposing proprietary security rules.

Enterprise service-level arrangements may define availability, support response, recovery objectives, maintenance windows and reporting. They should exclude guarantees for public blockchains, stablecoins or independent providers beyond AgentFi's control. Credits or remedies, if offered, are contractual service remedies and not Token returns. A private deployment still depends on software updates, model providers and security processes; private infrastructure does not automatically make a workflow compliant or safe.

SECTION 32

Treasury Reporting and Financial Controls

A full sale would create a treasury materially larger than the immediate product requirements of an MVP. The project should therefore use an annual budget, quarterly allocation limits and staged release from treasury custody. Material payments require documented purpose, approver separation and counterparty checks. Stablecoin and fiat exposures should be diversified within a board- or governance-approved risk policy, taking account of issuer, custody, banking, chain and concentration risk.

Financial reports should distinguish IEO proceeds, service revenue, grants, investment income if any, operating expenses, developer payments, liquidity deployment, refunds and Token transactions. Token transfers should not be booked as service revenue merely because they have a market price. Related-party transactions, founder-controlled vendors and compensation should be disclosed under a defined policy. Treasury performance should not be marketed as a return available to AGFI holders.

Reconciliation should connect bank, stablecoin, custody, exchange and onchain balances to the general ledger and labelled-wallet report. Exceptions need investigation and documented resolution. Independent financial review may be appropriate as the treasury and user base grow, but this document does not claim an audit. Public reporting can aggregate commercially sensitive details while still showing category spending, runway, material commitments and changes from the proposed use-of-proceeds framework.

SECTION 33

Valuation, Concentration and Sensitivity

At USD 2.17 per AGFI, the one billion fixed supply produces a USD 2.17 billion FDV. FDV is a simple price-times-supply measure and does not represent cash in the treasury, enterprise value, discounted future cash flow or a guarantee that all Tokens can trade at that price. With 94 million maximum TGE-unlocked AGFI, the corresponding maximum unlocked market capitalization is USD 203.98 million. Actual exchange-reported circulation may be lower depending on treatment of project-controlled balances.

The valuation creates execution expectations. If product usage, developer participation and revenue remain early, the market may apply a lower price regardless of the IEO terms. Conversely, a low reported circulating supply can create a high short-term price without demonstrating that the entire supply supports that valuation. Analysts should review both FDV and circulation, as well as future monthly unlocks, wallet concentration, liquidity depth and the price paid by earlier investors.

Sensitivity is linear for headline valuation: each USD 0.10 change in AGFI price changes FDV by USD 100 million and changes the value of 94 million maximum TGE-unlocked Tokens by USD 9.4 million. This arithmetic is not a forecast. It illustrates why small percentage price moves can correspond to large changes in implied value. The project should avoid using temporary market price as proof that the product or treasury has achieved equivalent economic value.

SECTION 34

IEO Operational Playbook

Before subscription opens, the platform should publish the verified project page, sale terms, supported assets, eligibility, contract status, allocation formula, precision, refund policy, vesting and official channels. Internal teams should freeze material configuration unless a controlled change is approved. Customer support needs tested responses for KYC, account funding, unsupported regions, duplicate accounts, stablecoin transfers and phishing. The project must not collect commitments through community staff or personal wallets.

During subscription, monitoring should cover platform availability, identity checks, duplicate behavior, stablecoin status, abnormal concentration, public misinformation and support volume. Because the method is pro rata, submission speed should not change allocation. Any outage that affects meaningful access may require extension under a published policy. Demand figures should not be selectively disclosed to influence participation, and unverified social-media claims should be corrected through official channels.

After closing, invalid accounts are removed before the allocation ratio is computed. The platform applies precision rules, deducts final purchase amounts, releases unused funds and records the vesting balance. Reconciliation must confirm that aggregate allocation does not exceed 40 million AGFI. Before listing, the platform and project verify initial distribution, project-controlled balances, market-making inventory, deposit and withdrawal settings, trading pairs and incident contacts. Every step should produce an auditable sign-off.

SECTION 35

Post-Listing Disclosure Calendar

During the first week, the project should publish confirmed contract addresses, distribution completion, known operational issues and the next unlock basis. It should avoid daily price interpretation. A status page should report product and network availability, while the exchange remains responsible for its own trading and account systems. Material incidents or changes to deposits, withdrawals or contract controls require prompt notice through verified channels.

Monthly transparency reporting is intended to cover maximum unlocks, actual distributions, labelled-wallet balances, treasury category spending, product tasks, paid users, stablecoin settlement, protocol revenue, refunds, disputes, external developer participation and security events. Metrics require definitions and comparative periods. If a metric changes because methodology changed, both the old and new basis should be explained rather than presenting the series as directly comparable.

Quarterly updates should assess roadmap delivery, financial runway, ecosystem grants, enterprise pilots, contract changes, governance proposals and risk outlook. Missed targets should be disclosed with cause and revised action. Announcements about integrations should specify whether they are live, in testing, under contract or merely proposed. This cadence is intended to create a record that can be evaluated independently of Token price.

SECTION 36

Testing, Verification and Release Assurance

Contract testing should include unit tests, integration tests, invariant tests, access-control tests, vesting arithmetic, precision, failure recovery and adversarial scenarios. Deployment scripts and configuration require review because a correct contract can be deployed with incorrect parameters. Final contract bytecode and source verification should match the reviewed version. Administrative addresses, multisig thresholds and time locks must be checked on the intended network.

Application testing should cover supported browsers, mobile layouts, wallets, networks, localization, accessibility, transaction previews, signing, retries, refunds, notification and account recovery. Financial values require consistent decimal handling. Time displays should show UTC and local conversion without changing the legal deadline. Load testing should include demand spikes around subscription close and listing, but synthetic performance tests must not be represented as real users or tasks.

Release assurance uses staged environments, change approval, rollback plans and production observation. Some blockchain actions cannot be rolled back, so deployment decisions need additional care. A final readiness review should record open risks and responsible owners rather than declaring a system absolutely secure. Independent review, bug bounties and monitoring provide layers of assurance, but none removes the possibility of defects, misuse or third-party failure.

SECTION 37

Token Launch Design Context

Recent exchange-hosted token distributions commonly publish a compact set of terms before participation: sale supply, total supply, accepted assets, opening and closing times, minimum commitment, allocation method, release schedule and refund mechanics. AgentFi follows that disclosure pattern because a participant should be able to reproduce the basic economics without relying on promotional commentary. The fixed-price proposal therefore states the USD 2.17 price, 40 million public allocation, USD 86.8 million maximum raise and 50 percent initial public release in the same section. These terms remain conditional on a confirmed platform and completed legal and operational review.

Distribution design varies across the market. Some offerings use fixed-price subscriptions, some use proportional allocation, and others use auctions or asset-locking reward pools. AgentFi currently proposes a fixed-price, pro-rata subscription rather than an auction or first-come-first-served process. This choice is intended to reduce network-speed competition, but it does not ensure broad ownership. A USD 5 million account cap can still create concentration, particularly if total demand is modest. The final platform may require lower caps, tiered limits, regional limits or additional allocation controls.

Industry launch materials increasingly separate token utility from ownership rights and disclose investor and contributor vesting. AgentFi adopts that distinction: AGFI is intended to coordinate access, staking, service guarantees and governance, not to represent equity or a contractual share of operating profit. The project must substantiate utility through deployed products, contracts and operating records. A description of future utility is not equivalent to current network demand, and the listing event does not complete the product roadmap.

SECTION 38

Competitive and Adjacent Landscape

The AI-agent and Web3 market includes consumer agent applications, developer frameworks, decentralized service marketplaces, model and compute networks, smart-account providers, intent solvers and payment protocols. AgentFi does not need to replace each category. Its intended position is the coordination layer that makes services discoverable, permission-aware, payable and auditable. This position succeeds only if it integrates credible external tools while maintaining a consistent user experience and security policy. Dependence on external services creates cost and availability risk that must be measured rather than hidden.

Consumer agent marketplaces demonstrate demand for easier discovery, while decentralized agent-service networks demonstrate that software can purchase specialized work from other software. Smart-account ecosystems demonstrate configurable authorization, and stablecoin payment protocols demonstrate machine-readable settlement. AgentFi combines these patterns into a task lifecycle, but combination alone is not differentiation. Defensible value would need to come from task data, developer distribution, security controls, enterprise policies, repeat usage and a reputation history that can be independently inspected.

Adjacent narratives such as tokenized assets, DePIN, RWA infrastructure and quantitative systems can generate demand for reliable data and automated workflows. AgentFi may support those workflows as customers or integrations, but it should not market itself as owning the underlying assets, hardware networks or investment strategies. Each claim must identify whether AgentFi provides software, executes a user-approved action, routes a third-party service or merely displays external information. This boundary helps users understand operational responsibility and helps the project avoid attributing third-party activity to its own protocol.

SECTION 39

End-to-End User Journey

A user journey begins with account access and an eligibility check. A user may connect an existing wallet or create a compatible smart account. The interface should explain whether a task is read-only, preparatory or executable, and whether any stablecoin or AGFI payment is required. Before a task is submitted, the user sees the Agent identity, developer, current version, requested data access, estimated fee and known dependencies. The product should not use a generic approval button for materially different permission levels.

The Intent Console converts natural language into a normalized task containing objective, assets, network, amount, limits, deadline and acceptable actions. If required parameters are missing, the system asks the user rather than inferring high-risk values. The plan is routed to one or more Agents and tools, followed by policy checks and simulation. The user receives a human-readable summary and, where relevant, machine-readable transaction data. Approval should cover the final plan, not an earlier draft that may change during routing.

After execution, the system records completion status, transaction references, service charges, tool costs, developer share and any refund. The user can report a problem without disclosing credentials in a public channel. A completed transaction does not automatically mean the task met the requested objective, so the system distinguishes technical execution from service success. Reputation updates should consider confirmed completion, user feedback, refunds, disputes and evidence quality. This lifecycle creates a basis for operational analytics and fairer service discovery.

SECTION 40

Developer Integration Framework

Developers are expected to integrate through a published SDK and service manifest. The manifest should declare the Agent name, version, operator, supported tasks, input and output schema, external dependencies, networks, required permissions, pricing, expected latency, failure conditions and support policy. Any use of user data or third-party model providers should be disclosed. A developer may update code, but material changes should create a new version rather than silently changing the service behind an existing reputation record.

The sandbox should provide simulated balances, deterministic fixtures, test contracts, mocked data sources and failure injection. Developers need to test timeouts, malformed input, unavailable providers, unexpected token behavior, price movement, gas changes and permission rejection. Production eligibility should require a minimum test suite and

successful review of declared permissions. Passing automated checks is not an endorsement and does not remove the developer's responsibility for service behavior.

Commercial tooling should show billable tasks, gross service revenue, external costs, platform fees, refunds, disputes and net developer proceeds. Developers should be able to set per-task prices, subscription eligibility and enterprise terms within platform constraints. Pricing must not hide mandatory third-party costs. The platform may limit pricing or require additional disclosure for services that resemble regulated financial activity. Developer incentives should reward retained usage and useful work, not merely Agent deployment or self-generated transactions.

SECTION 41

Agent Manifest and Registry Specification

The proposed Agent manifest is a signed document that gives applications a stable way to understand a service before calling it. Required fields may include a unique identifier, semantic version, operator address, code or package reference, endpoint, supported chain identifiers, task taxonomy, permission schema, settlement assets, fee formula, privacy notice, geographic limits and emergency contact. The manifest hash may be anchored onchain while larger documentation remains in an appropriate content store. A hash proves that a document has not changed; it does not prove that its claims are true.

Registry status is intended to include draft, sandbox, limited beta, production, restricted, suspended and retired. Movement between states depends on objective conditions and platform review. A suspended Agent should not accept new executable tasks, while users retain access to historical records and dispute channels. Retirement should state whether support continues for previously completed work. Registry decisions may be appealable, but security containment should not wait for a lengthy governance vote when credible immediate harm is identified.

The Registry may record staking and slashing data, but stake size must not be presented as a security certification. A well-funded malicious service can post collateral, while a useful small developer may have limited capital. Discovery therefore needs a composite view of declared permissions, operating history, external dependencies, task outcomes, reviews and security events. Any scoring formula should disclose its inputs and resist manipulation through self-dealing, duplicated users or paid reviews.

SECTION 42

Reputation, Ranking and Anti-Manipulation

Reputation is useful only if the underlying observations represent real service. AgentFi intends to distinguish verified paid tasks, subsidized tasks, internal tests, free trials and disputed tasks. The interface may show each category separately instead of collapsing them into one activity count. Revenue concentration, repeat users and task complexity provide additional context. An Agent completing thousands of trivial self-funded tasks should not outrank one completing a smaller number of high-value tasks for independent users.

Ranking manipulation may include wash usage, related-party reviews, incentive farming, Sybil accounts, misleading names, copied code and temporary service quality around evaluation periods. Controls can combine identity signals, payment-source analysis, behavioral patterns, task diversity, refund behavior and manual review. These controls will produce false positives and should support appeal. Anti-manipulation methods themselves should not expose enough detail to make evasion trivial.

A ranking system affects developer income and user safety, so changes require testing and notice. Sponsored placement, if ever offered, must be visibly labelled and should not override safety restrictions. Token holdings should not directly purchase an unqualified safety badge. Governance may define broad ranking principles, while sensitive abuse detection remains an operational function. Periodic reports should show how many Agents were approved, restricted, suspended and restored, without exposing private security details.

SECTION 43

Task Settlement and Payment Lifecycle

A task payment can include a user-facing service price, third-party data or model costs, gas sponsorship, developer revenue and platform fee. The quote should identify which components are estimates and which are fixed. For multi-step workflows, the system may reserve a maximum budget and settle actual usage after completion. Unused budget should return according to the disclosed method. Stablecoin fees, network fees and conversion costs can change between quote and execution, so tolerance rules are required.

Settlement states may include quoted, authorized, reserved, processing, completed, partially completed, failed, disputed, refunded and cancelled. State transitions should be logged and idempotent, preventing duplicate charges when a client retries. A technical provider outage should not be mislabelled as user cancellation. Where a task uses multiple providers, the system needs a policy for partial work and whether the user receives a partial result, a partial refund or a full refund.

Agent-to-Agent payments introduce additional complexity because an upstream Agent may purchase several downstream services before the final result is known. The initiating user must see the maximum budget and relevant service chain. The network should cap recursive calls, prevent circular purchasing and require each service to expose a machine-readable receipt. AGFI may support staking, access or discounts around settlement, but user-facing service value should remain understandable in stablecoin terms.

SECTION 44

Disputes, Refunds and Slashing

A dispute process should separate objective failures from subjective dissatisfaction. Objective examples include duplicate charging, execution outside permission, missing deliverables, invalid transaction references or a service unavailable during its stated window. Subjective cases may involve usefulness, interpretation or expected quality. The platform can automate clear cases while reserving complex cases for evidence review. Filing a dispute should not require the user to reveal a private key, seed phrase or unnecessary identity data.

Refund rules should be defined by task class before purchase. Read-only information tasks may be refundable if no result is delivered; executable tasks may not be reversible once a valid onchain transaction is confirmed, even if market conditions subsequently change. A route comparison is not a guarantee that the final execution price will remain available. Users must understand the difference between refunding a service charge and reversing an irreversible blockchain transaction.

Slashing is intended for provable violations, not ordinary business underperformance. Potential events include deliberate misrepresentation, unauthorized execution, falsified evidence, repeated refusal to deliver after accepting payment or collusion in verification. Each event needs evidence standards, maximum penalties, notice and appeal. Slashed AGFI may be directed to user remediation, security reserves or destruction only under published rules. The existence of collateral does not guarantee full compensation.

SECTION 45

Liquidity and Market Structure

The Token allocation reserves 60 million AGFI for liquidity, with 35 million included in the TGE maximum unlocked calculation and the balance scheduled over 24 months. This allocation must be separated into exchange inventory, onchain liquidity and unused contingency balances before listing. Tokens held in a labelled project wallet but available for use may be treated differently from tokens already deposited with a market maker. The project should report each category rather than relying on a single circulating-supply headline.

Liquidity provision requires both AGFI inventory and quote assets. The proposed use-of-proceeds framework allocates up to 20 percent of a full raise to quote-asset liquidity, but actual deployment depends on platform agreements, market conditions, legal review and treasury risk limits. Market-making agreements should define inventory ownership, permitted venues, transfer rights, reporting, prohibited self-trading, termination and return of unused tokens. A market

maker does not guarantee price stability.

A USD 2.17 offering price and 94 million maximum TGE-unlocked supply imply a USD 203.98 million maximum unlocked market capitalization. This is a material valuation for an early-stage product. Thin free float, concentrated ownership or limited quote depth can produce rapid moves in either direction. Limit orders, venue concentration, withdrawal status and stablecoin conditions all affect realized liquidity. The project must not use proceeds to fabricate volume or conceal project-controlled trading.

SECTION 46

IEO Allocation Mechanics and Examples

The proposed sale accepts valid commitments during the announced window and applies pro-rata allocation after ineligible, duplicate or invalid accounts are removed. If valid demand is equal to or below USD 86.8 million, eligible commitments may receive full allocation subject to precision and account controls. If valid demand exceeds that amount, each user's final purchase value equals the user's valid commitment divided by total valid commitments, multiplied by the sale cap. Token quantity equals final purchase value divided by USD 2.17.

For illustration, if valid demand is USD 434 million, the sale is five times subscribed. A valid USD 100,000 commitment would receive approximately USD 20,000 of final allocation, or approximately 9,216.59 AGFI before platform rounding. The unused USD 80,000 would be released or refunded under platform rules. This arithmetic example explains the mechanism only; it does not predict demand, allocation or market value.

The USD 5 million account cap is unusually large relative to the USD 86.8 million sale cap. One maximum commitment represents approximately 5.76 percent of the raise and approximately 2,304,147 AGFI before pro-rata scaling. Multiple large accounts could materially concentrate distribution if demand is weak. The platform should apply beneficial-owner checks, related-account controls and concentration reporting, and may lower the cap if its risk or fairness standards require.

SECTION 47

Token Supply Reporting

Four supply concepts should be kept separate. Total supply is the fixed one billion AGFI created under the final contract. Maximum unlocked supply is the amount no longer contractually locked at a given time. Distributed supply is the amount transferred to recipients, liquidity arrangements or operational addresses. Circulating supply is determined under the methodology of the relevant platform or data provider and may exclude certain project-controlled balances. These quantities are related but not interchangeable.

At TGE, the maximum unlocked amount is 94 million AGFI under the current design. It includes 20 million public-sale tokens, 8 million ecosystem, 12 million community, 10 million marketing, 35 million liquidity and 9 million reserve tokens. An unlocked ecosystem or reserve balance may remain in a project multisig and may not be considered circulating. Before listing, AgentFi should provide a wallet-level schedule identifying owner, purpose, lock, controller and whether each balance is expected to trade.

Monthly reporting should reconcile beginning balances, contractual unlocks, actual transfers, distributions, staking, burns if any, and ending balances. Reports should identify material deviations from schedule and explain operational transfers between labelled wallets. Changes to vesting should not occur through informal announcements. Contract-enforced schedules and public addresses give stronger assurance, although smart-contract controls and administrative privileges must also be reviewed.

SECTION 48

Token Utility Implementation Plan

Developer staking may be tiered by permission risk and service volume. A read-only information Agent may require little or no stake, while a service capable of preparing or executing asset transactions may require a larger guarantee and stronger review. Stake should not create an unrestricted right to publish. Eligibility also depends on identity, technical checks, declarations and operating history. Unstaking may include a delay so that newly discovered disputes can be resolved.

Fee benefits should be simple enough to audit. A user or enterprise may receive a published discount or higher API quota based on AGFI holdings or stake, but the discount should not exceed sustainable service economics. The product must remain usable through stablecoin payment where appropriate. Forcing every user to acquire a volatile Token before testing the product could reduce adoption and make utility appear artificial.

Governance utility is intended to mature gradually. Early proposals may be advisory or limited to transparent ecosystem budgets and technical parameters. Security response, private data, employment, legal compliance and individual disputes should not be decided by open token vote. The project should disclose which decisions are binding, who can propose, who can execute and what emergency controls remain with accountable operators.

SECTION 49

Economic Sustainability and Scenario Analysis

A base operating scenario should model revenue from paid tasks, subscriptions, APIs and enterprise services against model, data, infrastructure, support, refunds and developer payouts. The network becomes more sustainable when a growing share of costs is covered by genuine service revenue rather than Token incentives. A high settlement volume can still be unprofitable if third-party costs, subsidies or refunds are larger than platform fees.

A slower-adoption scenario assumes fewer paid users, greater price sensitivity, higher support cost and developer concentration. Under this scenario, the project should reduce incentive emissions, delay geographic or chain expansion and preserve treasury runway. It should not attempt to replace missing product demand with larger Token rewards or trading campaigns. Material underperformance against public targets should be reported with revised operating assumptions.

A higher-adoption scenario introduces different risks: dependency bottlenecks, increased fraud attempts, greater stablecoin exposure, privacy obligations and service concentration. Scaling requires redundant providers, rate limits, stronger developer review and enterprise-grade incident response. Positive product metrics do not eliminate Token valuation risk. The market may price AGFI independently of operating fundamentals, and holders have no contractual right to a particular valuation multiple.

SECTION 50

Go-to-Market Strategy

The initial consumer strategy focuses on problems with visible value and limited permissions: approval review, wallet health, event alerts, governance summaries and transaction preparation. These tasks let users evaluate reliability without granting broad execution authority. Educational onboarding should explain wallets, networks, stablecoins, signatures and irreversible transactions. Growth should be measured through repeat task use rather than account registrations alone.

Developer acquisition is intended to combine technical documentation, reference Agents, office hours, sandbox credits, targeted grants and transparent marketplace analytics. Grants should use milestones and clawback or cancellation terms for non-delivery. The project should avoid measuring ecosystem strength by the number of registered Agents because duplicate, abandoned or trivial services can inflate that metric without improving user outcomes.

Enterprise distribution may involve wallets, DAOs, data platforms and Web3 applications that need embedded Agent capabilities. Proposed integrations should begin with technical pilots and explicit success criteria. A pilot, memorandum, discussion or API test must not be marketed as a revenue-generating partnership. Commercial contracts, implementation readiness, data obligations and support responsibility must be confirmed before a relationship is described as active.

SECTION 51

Community Operations and Communications Integrity

Community content should help users understand product status, Token rights, allocation, vesting, account security and risks. Price targets, coordinated trading, selective disclosure and fabricated urgency are inconsistent with the intended operating standard. Moderators and assistants must not receive user funds, credentials, private keys or one-time codes, and must not operate accounts on behalf of participants.

The community may serve users familiar with traditional equities, but IEO participation differs materially from an IPO. AGFI does not represent corporate ownership, voting rights over the operating company, dividends or a residual claim. Educational comparisons should highlight both procedural similarities and legal differences. Registration for a document pack or announcement reminder does not create an allocation right and should not collect unnecessary financial information.

Incident communication should use verified channels, timestamps and a consistent status page. Suspected contract, wallet or account incidents should be acknowledged quickly without speculation. The project should explain affected functions, containment, user actions and the next update time. Marketing activity should pause during a material unresolved security event. Post-incident reports should distinguish confirmed facts, open questions and remediation.

SECTION 52

Compliance Operations

KYC and AML are operational processes, not a sentence in a disclaimer. The confirmed platform is expected to verify identity, screen sanctions, detect duplicate accounts and apply geographic restrictions. Higher-risk or unusually large commitments may require source-of-funds review. The project should avoid receiving identity documents directly when the platform can perform the regulated function, thereby reducing unnecessary privacy and security exposure.

Jurisdiction rules can change and may differ for initial distribution, secondary trading, staking, governance and Agent services. Eligibility at account registration does not guarantee eligibility at allocation. A second check may be required before settlement or withdrawal. Users must not use false information, nominees or technical circumvention. Frozen or rejected funds are handled under platform and legal requirements, not informal community promises.

Marketing review should cover websites, whitepapers, social channels, translations, influencer arrangements and support scripts. Statements about utility must reflect deployed functionality; statements about partnerships require authorization; statements about audits require final reports; and statements about legal status require actual professional analysis. Recordkeeping should preserve approved versions, publication times and material corrections.

SECTION 53

Listing Readiness and Post-Listing Operations

Listing readiness includes contract verification, token precision, deposit and withdrawal testing, supported network confirmation, wallet monitoring, market-making controls, supply reconciliation, incident contacts and public contract addresses. A listing date does not override incomplete security or legal gates. If operational conditions are not met, delay is safer than launching with unresolved controls.

At listing, the project should monitor exchange and onchain liquidity, failed deposits, withdrawal status, abnormal transfers from labelled wallets, phishing contracts and impersonation. Monitoring is intended to detect operational issues, not to manage or target the market price. Project personnel should follow information-handling and personal-trading policies to reduce conflicts and misuse of non-public information.

Post-listing reporting should prioritize product delivery, task metrics, treasury use, unlocks and security rather than price commentary. If price rises sharply, the project should not imply that fundamentals guarantee continuation. If price falls, it should not promise recovery or undisclosed support. Material information must be released through official channels with equal access, and corrections should be clear and timestamped.

SECTION 54

Incident Response and Business Continuity

Potential incidents include contract exploits, compromised administrative keys, malicious Agents, data leakage, model-provider outages, chain congestion, stablecoin disruption, incorrect supply reporting and platform unavailability. Each incident type requires an owner, severity level, escalation route, containment playbook and communication template. Contacts and backups must be tested before launch rather than created during an emergency.

Business continuity should identify critical services and acceptable recovery objectives. Read-only data services may fail differently from executable wallet functions. During uncertainty, the platform should fail safely: pause affected execution, preserve user ability to revoke permissions, prevent duplicate settlement and maintain access to status information. A degraded service should be clearly labelled rather than silently returning incomplete results.

Recovery ends only after control effectiveness is verified. The project should document root cause, affected users, financial impact, remediation and remaining risk. Where compensation is considered, it depends on evidence, available resources, legal constraints and published policy; it is not guaranteed by the existence of a reserve allocation or developer stake. Lessons should update threat models, tests and developer requirements.

SECTION 55

Research Method and Reference Framework

This whitepaper uses public market materials to understand current disclosure patterns, including official launch-platform explanations of fixed sales, proportional reward pools and auction mechanisms, and official documentation for AI-agent service networks. These references inform structure only. AgentFi does not claim affiliation with the platforms or projects referenced, and their mechanisms, performance and regulatory status should not be attributed to AgentFi.

The design favors explicit supply arithmetic, vesting, eligibility, refunds, product boundaries, security controls and measurable operating indicators. It avoids using market narratives as proof of demand. Quantitative statements about AgentFi are limited to the proposed Token and IEO parameters or clearly labelled internal targets. External market claims should be supported by dated sources in the final publication process.

Reference materials for further diligence include the official ERC-4337 specification and Ethereum account-abstraction documentation, public launch-platform materials, and official Olas documentation on agent-service marketplaces and coordination. These references do not constitute endorsements. URLs and access dates should be checked before public release because documentation may change.

SECTION 56

Glossary and Interpretation

Agent means a software service capable of interpreting a task and calling approved tools. It does not imply human-level intelligence or guaranteed autonomy. Intent means a user's stated objective and constraints. Task means a normalized unit of requested work. Tool means an API, contract, model, data source or execution component called by an Agent. Solver or executor means a service that identifies or carries out an approved path.

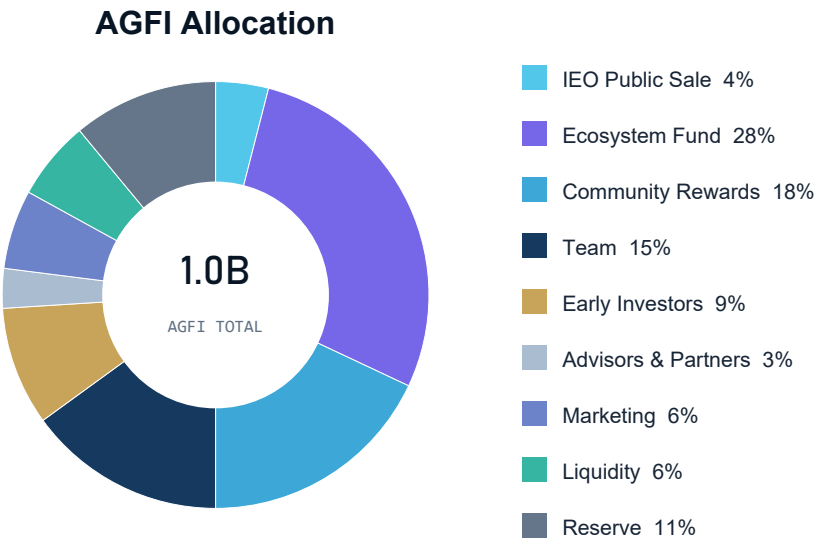
TGE means Token Generation Event. Total supply means all AGFI created under the final contract. Unlocked supply means Token no longer subject to contractual vesting. Distributed supply means Token transferred to recipients or operational arrangements. Circulating supply follows the methodology of the reporting platform. FDV means total supply multiplied by the stated price and does not represent cash raised or current company value.

IEO means an exchange-hosted initial Token offering. KYC and AML refer to identity verification and anti-money-laundering controls. Pro-rata allocation gives each valid participant the same fraction of a valid commitment when demand exceeds supply, subject to platform rules. Listing means opening a trading market.

Tokenomics

$2.17	40M	94M	$2.17B
IEO PRICE	PUBLIC SALE	TGE UNLOCK	FDV

Metric	Value
Token Name	AgentFi Network Token
Symbol	AGFI
Network / Standard	Base / ERC-20
Total Supply	1,000,000,000 AGFI
IEO Price	USD 2.17 per AGFI
IEO Allocation	40,000,000 AGFI (4%)
Maximum Raise	USD 86,800,000
TGE Maximum Unlocked Supply	94,000,000 AGFI (9.40%)
TGE Maximum Unlocked Market Cap	USD 203,980,000
Fully Diluted Valuation	USD 2,170,000,000



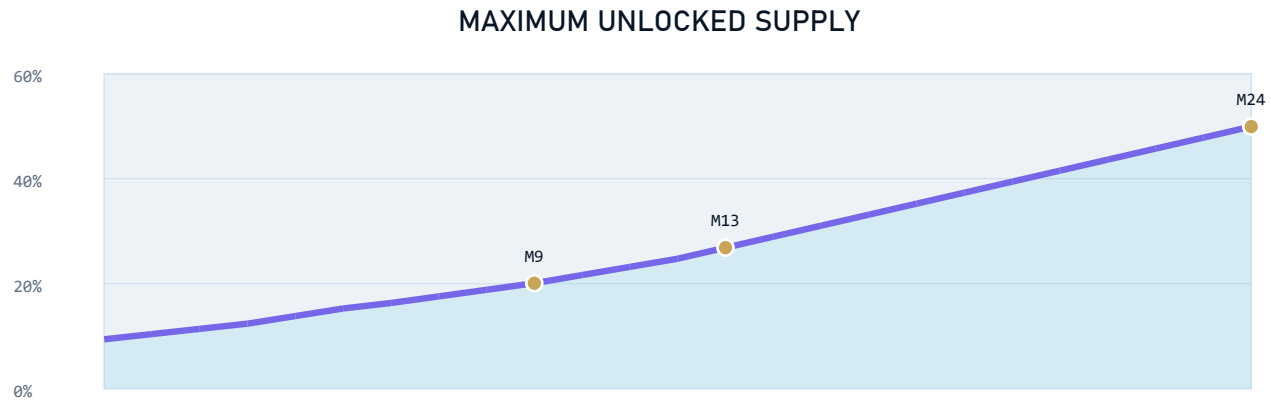
Allocation	Share	Tokens
IEO Public Sale	4%	40,000,000
Ecosystem Fund	28%	280,000,000
Community Rewards	18%	180,000,000

Allocation	Share	Tokens
Team	15%	150,000,000
Early Investors	9%	90,000,000
Advisors & Partners	3%	30,000,000
Marketing	6%	60,000,000
Liquidity	6%	60,000,000
Reserve	11%	110,000,000

Vesting Schedule

Allocation	Release Schedule
IEO Public Sale	50% at TGE; 10% per month for M1-M5
Ecosystem Fund	8M at TGE; 3-month cliff; balance over 60 months
Community Rewards	12M at TGE; balance over 48 months
Team	12-month cliff; 36-month linear vesting
Early Investors	9-month cliff; 30-month linear vesting
Advisors & Partners	12-month cliff; 24-month linear vesting
Marketing	10M at TGE; balance over 36 months
Liquidity	35M at TGE; balance over 24 months
Reserve	9M at TGE; 6-month cliff; balance over 48 months

TGE maximum unlocked supply is 94M AGFI because the public-sale TGE release is 50%. Unlocked treasury allocations are not necessarily in actual circulation; exchange reporting must distinguish unlocked, distributed and tradable supply.



24-Month Maximum Unlock Schedule

Month	New Unlock	Cumulative Maximum	Supply Share	Month	New Unlock	Cumulative Maximum	Supply Share
TGE	94.0000M	94.0000M	9.40%	M13	20.9847M	268.5764M	26.86%
M1	9.9306M	103.9306M	10.39%	M14	20.9847M	289.5611M	28.96%
M2	9.9306M	113.8611M	11.39%	M15	20.9847M	310.5458M	31.05%
M3	9.9306M	123.7917M	12.38%	M16	20.9847M	331.5306M	33.15%
M4	14.4639M	138.2556M	13.83%	M17	20.9847M	352.5153M	35.25%
M5	14.4639M	152.7194M	15.27%	M18	20.9847M	373.5000M	37.35%
M6	10.4639M	163.1833M	16.32%	M19	20.9847M	394.4847M	39.45%
M7	12.5681M	175.7514M	17.58%	M20	20.9847M	415.4694M	41.55%
M8	12.5681M	188.3194M	18.83%	M21	20.9847M	436.4542M	43.65%
M9	12.5681M	200.8875M	20.09%	M22	20.9847M	457.4389M	45.74%
M10	15.5681M	216.4556M	21.65%	M23	20.9847M	478.4236M	47.84%
M11	15.5681M	232.0236M	23.20%	M24	20.9847M	499.4083M	49.94%
M12	15.5681M	247.5917M	24.76%				

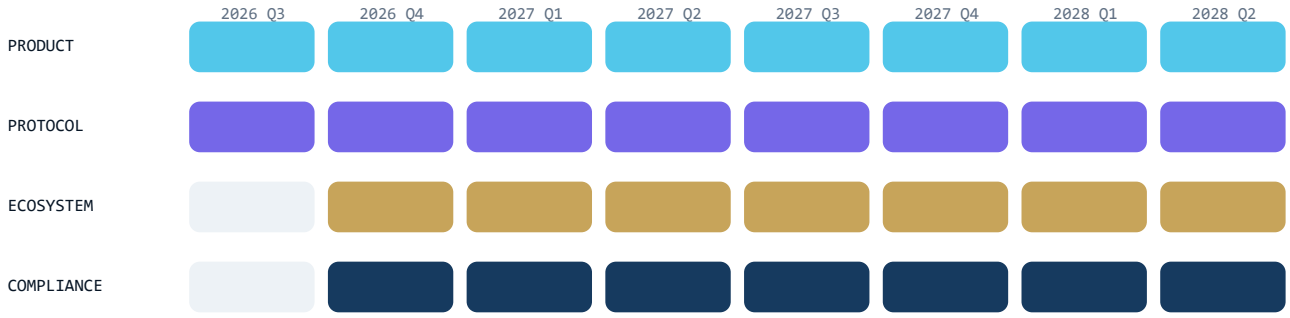
IEO Details

Item	Proposed Rule
IEO Opens	July 13, 2026 at 10:00 UTC
IEO Ends	July 20, 2026 at 10:00 UTC
Pricing Confirmation / Allocation Result	July 21, 2026 at 14:00 UTC
Initial Token Distribution	July 22, 2026 at 12:00 UTC
Official Listing	July 22, 2026 at 14:00 UTC
Supported Assets	USDT and USDC, subject to final platform rules
Minimum Subscription	USD 100 equivalent
Maximum per Account	USD 5,000,000 equivalent
KYC	Required; jurisdiction and sanctions screening apply
Allocation Method	Pro rata after invalid accounts are removed; not first-come-first-served
Refunds	Unused frozen funds are expected to return after final allocation, subject to compliance holds
Public Release	50% at TGE; remaining 50% in five monthly tranches of 10%

The schedule is conditional on exchange confirmation, contract readiness, security review, KYC/AML configuration and legal checks. If required conditions are incomplete, the offering must be delayed or cancelled. Subscription does not guarantee allocation, liquidity or returns.

A USD 5,000,000 commitment equals approximately 5.76% of the maximum raise and approximately 2,304,147 AGFI before pro-rata scaling. This high account cap may increase concentration and fairness concerns. Final allocation remains subject to the pro-rata method and platform controls.

24-Month Roadmap

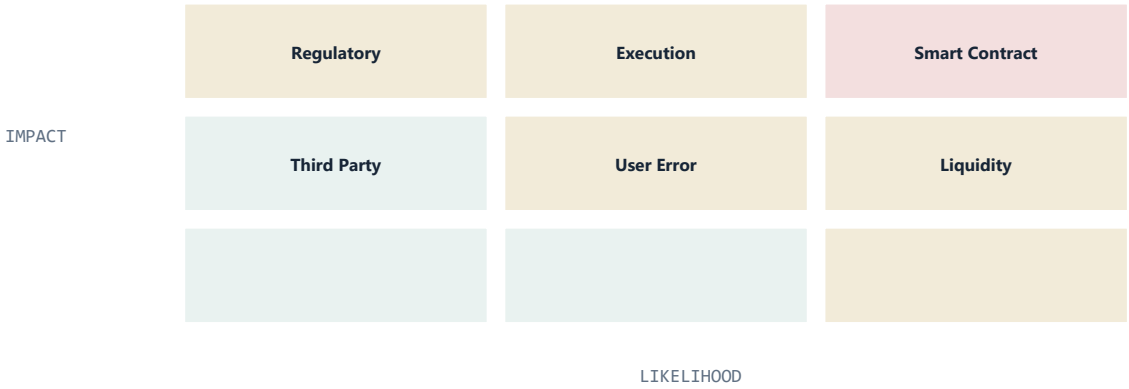


Targets are conditional on readiness gates.

Period	Product, Technology, Users, Ecosystem, Compliance and Market
2026 Q3	Launch readiness: public product verification, account-abstraction controls, final contracts, platform review, KYC/AML configuration, listing operations.
2026 Q4	Agent marketplace beta, stablecoin task settlement, SDK alpha, security testing and public operating metrics.
2027 Q1	Paid task validation, developer revenue dashboard, refund and dispute workflows, third-party Agent onboarding.
2027 Q2	Mainnet V1, Agent Registry, permission center, bug bounty and expanded enterprise pilots.
2027 Q3	Staking test environment, solver and verifier design, multi-provider routing and developer grants.
2027 Q4	Enterprise console, Agent-to-Agent service procurement and broader API distribution.
2028 Q1	Second EVM network evaluation, open executor testing and private Agent environments.
2028 Q2	Protocol V2, time-locked governance, open curation and annual network transparency report.

Roadmap items are targets, not guaranteed delivery dates. Listing activity must not replace product, security or compliance readiness.

Risk Factors



Risk	Description
Digital asset and market risk	AGFI may experience material price volatility or loss of value. The IEO price is not a floor or a prediction.
Liquidity risk	Order-book or onchain depth may be limited, causing spreads, slippage or inability to trade at an expected price.
Smart-contract risk	Token, vesting, settlement, account and third-party contracts may contain defects or be exploited.
Regulatory risk	Issuance, trading, staking, payments and Agent services may be restricted or reclassified in certain jurisdictions.
Execution risk	The project may not deliver the roadmap, user adoption, revenue, integrations or security objectives.
Unlock risk	Public, ecosystem, investor, team and other allocations release over time and may increase available supply.
AI and user risk	Models may produce incorrect outputs. Users may misconfigure permissions, select wrong networks or approve unsafe transactions.
Third-party risk	Stablecoins, chains, wallets, models, data providers and trading platforms may fail, suspend service or change policies.

Legal and Compliance Notes

KYC, AML, sanctions screening and jurisdiction restrictions are required. Final eligibility is determined by the confirmed platform and applicable law.

This document is not a securities offering statement and does not determine AGFI's legal classification. Classification requires jurisdiction-specific professional review.

Users are responsible for confirming eligibility, understanding wallet and network risks, protecting credentials and using only funds they can afford to lose.

Platform due diligence, listing or technical integration does not remove project, market, smart-contract, liquidity or regulatory risk.

Contact and Official Links

Item	Link
Website	www.agfinw.com
Whitepaper	https://www.agfinw.com/whitepaper.pdf
Support Email	suppot@agfinw.com