

Kitchen Together Blockchain Marketplace - Whitepaper v1 Public Preview

Public preview document for external readers, based on the internal working draft and updated for a testnet-only release scope.

>

Status: `Whitepaper v1 Public Preview (Public)`

Release Date: `2026-02-27`

Language Scope: `EN only (VN follow-up release note)`

1. Executive Summary

Kitchen Together is a cooking game platform with the potential to expand into a creator-driven ecosystem (UGC). This whitepaper proposes a **hybrid on-chain/off-chain** model to:

- Enable creators to earn income from maps they build.
- Enable players to buy and own maps through a web marketplace (primary sale).
- Record ownership transparently on blockchain.
- Sync entitlements back to the game for stable map unlock behavior.
- Avoid mobile policy conflict by keeping all crypto actions on web (`web-only crypto`).

The immediate goal is to launch a **web-only marketplace MVP** under controlled scope and compliance guardrails in a **testnet-only public preview** mode. Mainnet remains a future milestone with **no committed launch date** in this release.

Current go-live strategy follows a **points-first rollout**:

- First validate retention with an off-chain points system.
- No token/airdrop commitment until data is strong enough.
- Move to fuller on-chain mode only when KPI and compliance conditions are met.

KITCH token follows a **Hybrid (Treasury Release + Burn)** approach:

- Maximum total supply (hard cap): `1,000,000,000 KITCH`
- No minting beyond hard cap
- Part of supply is locked in treasury and released by phase
- Marketplace burn mechanism supports long-term balancing

Public Preview note: Detailed fee split, vesting cadence, unlock schedule, and issuance timeline are currently `Subject to Final Spec` and are not a mainnet issuance commitment.

2. Market Problem and Opportunity

2.1 Limitations of the Current Model (Traditional Mobile Game)

In a traditional mobile game economy:

- Players buy content but cannot resell ownership.
- Creators struggle to get fair and transparent rewards.

- In-game digital assets stay trapped in a closed system.
- In-app platform fees (for example Apple/Google) compress margins and push user prices higher.

2.2 Why Kitchen Together Has an Opportunity

Kitchen Together has a strong foundation for a creator economy because:

- Core gameplay naturally supports UGC (maps, challenges, social experiences).
- User-created content can be rated, ranked, and transacted.
- Gameplay data (off-chain) and ownership layer (on-chain) can be separated cleanly.
- A web-first rollout lowers policy risk on mobile.

2.3 Core Thesis

Blockchain is not meant to replace core gameplay. It adds:

- **A transparent ownership layer**
- **A marketplace transaction layer**
- **A creator incentive layer**
- **A long-term economy expansion layer**

3. Product Vision

3.1 Long-Term Vision

Turn Kitchen Together into an open cooking-game economy where:

- Players can be content consumers and optional collectors/traders
- Creators are true producers with real earning potential
- Studio runs a sustainable marketplace with fee-based revenue
- Digital ownership is transparent and verifiable

3.2 Non-Negotiable Design Principles

1. Web-only crypto

- All wallet/token/blockchain marketplace actions run on web.
- No crypto trading UI in the mobile app.

2. Ownership on-chain, data off-chain

- Ownership, transactions, and key events are on-chain.
- Heavy map/metadata and gameplay state remain off-chain.

3. Soft launch before scale

- Testnet beta -> readiness evaluation -> future mainnet phase (no committed date) -> broader scale.

4. Build in parallel with compliance

- Product work does not wait for 100% legal completion, but red-line boundaries must be clear before launch.

4. Proposed Solution: Hybrid On-chain / Off-chain

4.1 High-Level Architecture

Kitchen Together runs on two layers:

Game layer (off-chain)

- Core cooking gameplay
- Map creation and play sessions
- Game state, user profile, map metadata
- No crypto UI

Web marketplace layer (on-chain + backend index)

- Wallet connection
- Primary map listing/purchase in KITCH
- Ownership and transaction recording
- Entitlement sync to backend/game

4.2 Why Hybrid Instead of Full On-chain

Full on-chain is not practical for Kitchen Together because:

- Map/game state data is large and frequently updated
- On-chain storage costs are high
- Gameplay UX needs real-time responsiveness beyond transaction confirmation latency
- Many gameplay actions do not require blockchain-grade guarantees

Hybrid separates responsibilities correctly:

- Blockchain handles ownership and value transfer
- Backend/game handles speed, flexibility, and UX quality

4.3 Source of Truth by Component

| Component | Source of Truth | Notes |

| --- | --- | --- |

| Map ownership for marketplace | On-chain | Technical/legal ownership reference |

| Fast inventory rendering | DB snapshot / indexer | UX-first, with reconciliation |

| Map metadata / asset data | Off-chain DB/object storage | Not stored directly on chain |

| In-game entitlement state | Backend snapshot synced from on-chain | Includes fallback and manual re-sync |

5. Ecosystem Roles

5.1 Player (Buyer)

- Buys maps with KITCH on web marketplace
- Receives entitlement and map unlock in game
- MVP model is primary sale only (no secondary market)

5.2 Creator

- Builds maps in game/tooling
- Lists maps on web marketplace
- Earns revenue from map sales

5.3 Trader/Collector (Optional Persona)

- Participates for collection/trading intent
- Can increase marketplace liquidity
- Not a required persona for MVP success

5.4 Studio / Platform Operator

- Operates marketplace, moderation, compliance, support
- Collects platform fee
- Manages treasury release and staking reward policy (if enabled)
- Ensures stable entitlement sync operations

5.5 Admin / Support

- Moderates maps/listings
- Handles pending tx / delayed sync / entitlement mismatch incidents
- Uses re-sync tooling and operational runbooks

6. MVP Scope (v1)

6.1 In Scope (MVP)

- Creator map listing on web marketplace
- Buyer purchase flow in KITCH on web
- Primary sale only (no secondary sale)
- Fee split including platform fee / burn / staking reward per tokenomics spec
- On-chain ownership record
- Entitlement sync from blockchain -> backend/game
- In-game unlock behavior based on ownership
- Basic admin moderation (list/delist/temp hide)

6.2 Out of Scope (MVP)

- DAO governance
- Subscription model
- Advanced creator analytics
- Cross-game partnership layer
- Mobile app wallet/token UI integration
- Large-scale public launch

6.3 MVP Definition of Done

MVP is complete only when all are true:

1. Creator can list maps on web.
2. Buyer can purchase maps with KITCH on web.
3. Fee split / burn behaves according to spec.
4. Ownership is recorded on-chain and indexed to backend.
5. Game unlock follows ownership correctly.
6. Support has manual re-sync tooling.
7. Closed beta completes with no unresolved P0 incidents.
8. Launch checklist and operations runbook are ready.

6.4 Rollout Policy (*Points-first, No Early Token Promise*)

To avoid shifting user behavior into pure token farming, MVP applies:

1. Phase 1: points-only off-chain (primary sale + user/creator actions + anti-bot checks).
2. Phase 2: conditional reward campaigns; no fixed token allocation promise for all accounts.
3. Phase 3: enable broader blockchain/token campaigns only when:
 - retention/conversion KPI passes internal thresholds
 - bot/farm rates stay within acceptable bounds
 - legal/compliance checklist is complete

7. Core User Flows

7.1 First-Time Buyer Flow

1. User opens web marketplace (possibly through valid deep link from game).
2. Connects wallet.
3. Holds or tops up KITCH.
4. Selects map and confirms purchase transaction.
5. Transaction is confirmed on-chain.
6. Indexer/backend updates ownership snapshot.
7. Game reads entitlement and unlocks map.

Critical UX requirements

- Clear states: signature requested -> pending -> confirmed -> failed
- No dead-end screens
- Delayed sync notice plus retry option

7.2 Creator Listing Flow

1. Creator builds/publishes map in game/workshop.
2. Creator opens web marketplace.
3. Selects an eligible map for listing.
4. Inputs price/terms (if any), confirms listing.
5. Listing appears after backend/index validation.

7.3 Primary Sale Settlement

1. Creator creates primary listing.
2. Buyer purchases with KITCH.
3. Contract distributes values by tokenomics split:
 - seller proceeds
 - platform fee
 - burn
 - staking rewards (if enabled)

7.4 Error Handling and Support

System must support at minimum:

- Long pending transaction
- Failed transaction
- On-chain confirmed but game not unlocked
- Indexer lag / missed event
- Ownership snapshot mismatch vs on-chain

Principles:

- Provide manual re-sync endpoint/tool
- Include logs and traces for support investigations
- Maintain a clear runbook by incident type

8. KITCH Token Economy (Token Utility & Tokenomics)

8.1 KITCH Utility Role

KITCH is the utility token for Kitchen Together marketplace and broader ecosystem use cases:

- Payment token in web marketplace primary sales
- Marketplace value distribution through fee split
- Incentive/reward programs in ecosystem (subject to policy)
- Growth activation and creator incentive operations
- (Future) staking/governance/ecosystem participation if enabled

8.2 Tokenomics Design Philosophy

KITCH follows **Hybrid (Treasury Release + Burn)** to balance:

- **Network bootstrap** (incentive budget needed)
- **Long-term sustainability** (no unlimited minting)
- **Hard-cap narrative** (clear maximum supply)
- **Phase flexibility** (gradual treasury release)

8.3 Total Supply and High-Level Allocation

Hard Cap: 1,000,000,000 KITCH

Principles:

- No mint beyond hard cap
- Part of supply circulates early; part remains locked (team, private sale, treasury, etc.)
- Locked tranches follow vesting/release schedule

High-level allocation (draft structure):

Group	Ratio	Estimated Amount	Status
Public Sale / IDO	10%	100,000,000	Circulating
Liquidity Pool	20%	200,000,000	Circulating
Ecosystem Fund (active tranche)	20%	200,000,000	Circulating
Team & Advisors	15%	150,000,000	Locked + vesting
Private Sale	10%	100,000,000	Locked + vesting
Treasury	25%	250,000,000	Locked + phased release

Editorial note

- Table above reflects current draft structure.
- Public final whitepaper should include detailed monthly/quarterly unlock schedule and cliff.

8.4 Treasury Release Schedule (Draft)

Release comes from locked treasury, not new minting:

Stage	Year	Draft Release	Purpose
Bootstrap	1	50,000,000	Airdrop/creator rewards/ecosystem activation
Growth	2	40,000,000	Creator and user program expansion
Maturity	3	25,000,000	Sustain higher-quality growth
Balance	4	15,000,000	End of acceleration cycle balancing
Stability	5+	0 (default)	Strategic reserve, subject to governance/policy

8.5 Burn Mechanism

Marketplace transactions include a burn component in fee split.

Example draft split for 1000 KITCH purchase:

- Seller: 87.5%
- Platform fee: 2.5%
- Burn: 5%
- Staking rewards: 5%

Burn mechanism goals:

- Reduce circulating supply pressure over time
- Link marketplace activity with relative deflation pressure
- Balance treasury release effect

8.6 Net Supply Dynamics (Conceptual)

- Early phase: release > burn (bootstrap priority)
- Mid phase: release ~ burn (balancing)
- Later phase: burn may exceed release if volume is high

Important communication rule

- Do not market this as a guaranteed “deflationary profit” promise.
- Describe as a mechanism that can produce net deflation only under certain activity conditions.

8.7 Anti-Abuse Requirements (Mandatory)

Tokenomics spec and business logic must include controls against:

- Self-trading / wash volume
- Sybil farming
- Listing spam
- Creator reward claim abuse
- Bot-driven manipulation (policy + moderation + analytics)

9. Fee Model and Unit Economics

9.1 Revenue Model

Initial platform revenue sources:

- Platform fee on marketplace transactions
- (Future, if enabled) premium services/creator tooling/featured placement

9.2 Why Web Marketplace Is Economically Advantageous

- Avoids/reduces dependence on in-app platform fee paths for crypto/web transactions
- Increases margin for creators and platform
- Can pass part of efficiency to users through better pricing

9.3 Three-Sided Balance

Fee/tokenomics goals must balance:

- **Buyer:** fair price, stable unlock, simple UX
- **Creator:** direct revenue path from maps
- **Platform:** sufficient fee to operate, support, and grow ecosystem

Public final whitepaper should add:

- 3-5 unit economics examples by map price tier
- Low/medium/high volume scenarios
- Burn/release impact under each scenario

10. Technical Architecture

10.1 Minimum Components

- Smart contracts
- Web frontend
- Backend API
- Blockchain indexer / event listener
- Database
- Object storage (map data/metadata)
- Entitlement service for game
- Admin/moderation tooling

10.2 Responsibility Separation

Smart Contracts

- Ownership and marketplace execution
- Fee split / burn logic
- Treasury release / vesting (if implemented on-chain)
- Events for indexer consumption

Backend + Indexer

- Sync chain events to DB snapshots
- Serve APIs to web/game
- Reconciliation when events are missed or reorg happens
- Logging, tracing, alerting

Game Integration Layer

- Read ownership/inventory snapshot
- Retry and cache logic
- User-visible delayed sync states

10.3 Entitlement Sync (Technical Core)

Standard flow:

1. On-chain transaction confirmed
2. Event listener receives event
3. Indexer writes snapshot
4. Entitlement service updates user rights
5. Game fetches inventory/ownership
6. Map unlock is applied

Mandatory requirements:

- Idempotent processing
- Reconciliation fallback job
- Manual re-sync support tool
- SLA tracking for `time-to-unlock`

10.4 Reorg / Missed Event / Indexer Outage Handling

Whitepaper does not need code-level detail, but should state:

- Event-driven path is primary
- Block-range reconciliation is fallback
- Snapshot must be re-validated against on-chain under suspicion
- Support can trigger re-sync by user/listing/tx reference

10.5 Map Data and Metadata

Principles:

- Blockchain does not store full map data
- Off-chain systems store metadata, version, asset references
- Public technical docs should clearly define:
 - map_id identity model
- Versioning policy
- Map edit policy post-list/post-sale
- Effects on ownership and entitlement

11. Smart Contracts (Design Framework)

11.1 Proposed Modules

- Marketplace Contract
- Fee Distribution
- Treasury Release / Vesting
- Token Contract (if not separately deployed)
- Optional admin/pause/permission modules

11.2 Governance and Operational Permissions

Spec must clearly define:

- Owner
- Multisig
- Pauser
- Treasury release / parameter update permissions (if any)

Public whitepaper should explain:

- Purpose of each permission class
- Which permissions are temporary in beta
- Plan to reduce privileged control over time (if applicable)

11.3 Emergency Controls

To protect users and systems in early phases:

- Trading pause
- Listing disable switch
- Non-arbitrary incident response procedure
- Incident disclosure and recovery policy

Important note

- Emergency controls are operational protection mechanisms, not discretionary rights to alter user assets.

12. Security and Operational Reliability

12.1 Product/System Threat Model

Key risks already identified:

- Reentrancy / unauthorized release / fee bypass (contract-level)
- Signature replay (for signed operations)
- Front-running / listing race conditions
- Entitlement mismatch (on-chain correct, game unlock missing)
- Indexer fragility (missed event, delayed sync)
- Tokenomics implementation mismatch (spec vs code)
- UX friction at wallet/tx step causing drop-off

12.2 Security Principles

- Internal security review before testnet
- Complete test matrix for fee split / burn / treasury release
- Integration tests across web/backend/indexer
- Sufficient logs and traces for incident investigation
- Least-privilege operator/admin model

12.3 Audit and Bug Bounty (TBD)

Public preview discloses baseline level:

- Audit intent and target scope (without over-claiming “audited” status)
- How audit findings will be published once plan is locked
- Bug bounty policy (if enabled)

Draft status

- TBD: audit vendor, timeline, and final scope are not locked in this public preview.

12.4 Beta Operational Targets (Draft)

Closed beta technical/operational KPI targets:

- Purchase success rate > 85%
- Entitlement sync success > 99%
- Time-to-unlock P95 < 60s

- 0 unresolved P0 incidents beyond 24h

13. Compliance, Policy, and Product Boundaries

13.1 Core Compliance Principles

- Do not position token as an investment product
- Do not promise profit
- Do not place crypto/trading UI inside mobile app
- Keep app experience and web crypto marketplace clearly separated

13.2 Minimum Legal Documents for Beta

- Terms of Service
- Privacy Policy
- Risk Disclosure
- Marketing Messaging Do/Don't
- Campaign Terms

13.3 Public Legal URLs (Release Gate)

- Risk Disclosure: <https://market.kitchentogether.com/risk-disclosure>
- Campaign Terms: <https://market.kitchentogether.com/campaign-terms>
- Terms of Service: <https://market.kitchentogether.com/tos>
- Privacy Policy: <https://market.kitchentogether.com/privacy-policy>

13.4 Locked MVP Policy Decisions (Internal)

- KYC: no mandatory product-layer KYC for MVP core flows; KYC is handled by exchange/on-ramp partners for cash-out/trading.
- Geo policy: no blanket block for core marketplace; apply targeted restrictions when legally/partner required.
- Age policy: no hard 18+ entry gate; age requirements for financial conversion follow partner/local law.
- App -> web deep link: allowlist-only and must pass signed-in + linked-account gate before sensitive actions.

13.5 Recommended Whitepaper Language Style

Public whitepaper should use neutral language:

- Explain utility and system behavior
- State risks and limitations clearly
- Avoid speculative/hype language
- Separate product content from token-launch marketing narratives

14. Implementation Roadmap

14.1 MVP Roadmap (12-Week Draft)

| Phase | Timeline | Objective | Primary Output | Gate |

| --- | --- | --- | --- | --- |

| P0. Scope + Compliance | Week 1-2 | Lock scope + red lines | MVP Scope, PRD draft, compliance checklist | No app/web policy contradictions |

| P1. Architecture + Spec | Week 2-3 | Lock buildable design | Architecture, contract spec, tokenomics spec | Team aligned on modules/API/events |

| P2. Contract + Backend Core | Week 4-6 | Run end-to-end on testnet | Contracts, indexer, API, DB schema | List/buy/sync works |

| P3. Web MVP + Game Integration | Week 5-8 | Complete UX + unlock | Web marketplace, entitlement sync | Buy -> unlock success |

| P4. Closed Beta | Week 9-10 | Validate operations | Bug fixes, monitoring, runbooks | Beta KPI threshold reached |

| P5. Mainnet Soft Launch (Future) | TBD | Conditional milestone (no committed date) | Launch checklist, support workflow | Open only after legal + KPI + ops readiness |

14.2 Rollout Phases Under Current Strategy

| Rollout Phase | Focus | Token/Airdrop Policy |

| --- | --- | --- |

| Phase 1 | Points-only off-chain + primary sale web | No token issuance, no airdrop commitment |

| Phase 2 | Conditional reward campaign | No guaranteed token promise for every user |

| Phase 3 | Controlled blockchain/on-chain expansion | Execute only when KPI + compliance gates pass |

14.3 Required Documentation Milestones

Documents expected before soft launch:

- MVP Scope v1
- PRD v1
- Tokenomics Spec v1
- Architecture v1
- Contract Spec v1
- Security Checklist v1
- Launch Checklist v1
- Operations Runbook v1

14.4 Post-MVP Directions (Non-Commitment)

Potential expansion areas:

- Advanced creator analytics
- Curation/rating/featured maps
- Mature staking mechanics
- Seasonal creator incentive programs
- Governance frameworks (only when operationally justified)

15. KPI and Success Measurement Framework

15.1 MVP/Beta Operational KPI

- Wallet connect rate
- Purchase success rate
- Time-to-first-purchase
- Time-to-unlock
- Entitlement sync success rate
- Incident severity distribution (P0/P1/P2)

15.2 Marketplace KPI

- Active creator count
- Number of listed maps
- Ratio of maps with at least one transaction
- Daily/weekly/monthly transaction volume
- Primary sale conversion rate

15.3 Ecosystem KPI (Later Stage)

- Creator retention
- Buyer repeat purchase rate
- Liquidity depth (if model expands)
- Incentive program effectiveness

16. Key Risks and Mitigation Strategy

| Risk | Description | Mitigation |

| --- | --- | --- |

| Compliance drift | Team accidentally adds non-compliant mobile crypto UX or messaging | Shared checklist + review flow across product/dev/marketing |

| Entitlement mismatch | On-chain ownership correct but game unlock missing | Indexer + reconciliation + re-sync tool + runbook |

| Indexer fragility | Missed events/lag overload support | Clear event schema, fallback scan, lag monitoring |

| Spec-code mismatch | Tokenomics spec differs from contract implementation | Detailed tokenomics spec + test vectors + review gates |

| UX friction | Users drop during wallet/tx flow | Clear state UX, onboarding, robust error handling |

| Abuse/wash trading | Fake volume to farm incentives | Policy controls, detection, moderation, anti-abuse rules |

17. Governance and Operational Control (Draft)

17.1 Early Stage (Operator-led)

In MVP/soft-launch phases, controlled operator governance is needed to:

- Protect users
- Resolve incidents quickly
- Coordinate compliance changes
- Tune systems based on real usage data

17.2 Gradual Decentralization Principles (If Applicable)

If ecosystem maturity justifies it, future progression may include:

- Greater treasury/reporting transparency
- Reduced admin privileges in selected modules
- Governance experiments for non-emergency decisions

Important note

- DAO/governance is not in MVP scope and no timeline should be promised before operational readiness.

18. Risk Statements and Disclosures (Draft)

18.1 Technical Risk

- Smart contract issues, integration bugs, indexer lag, or backend infrastructure incidents may impact purchase/unlock experience.
- Testing and controls reduce risk but cannot eliminate all technical risk.

18.2 Market and Liquidity Risk

- Token price and marketplace demand can be highly volatile.
- No guarantee is made for future liquidity or exchange value.

18.3 Legal/Policy Risk

- Regulatory and platform policies may change over time.
- Product features may require adjustments by region or rollout stage.

18.4 Not an Investment Offer

This document describes product/system design. It is not:

- an investment solicitation,
- financial advice,
- a profit commitment,
- or a guaranteed economic outcome.

19. Public Preview Release Checklist (v1)

19.1 Product / Compliance

- [x] Network positioning locked: currently testnet-only; mainnet future milestone with no committed date
- [x] Deep-link app -> web policy implementation locked
- [x] KYC / geo / age policy for MVP locked (low-friction + partner-side compliance for cash-out)
- [x] Minimum legal docs complete (Risk Disclosure + Campaign Terms + ToS + Privacy links)

19.2 Tokenomics

- [x] Tokenomics framework locked (treasury release + burn)
- [x] Fee split, vesting cadence, unlock schedule marked `Subject to Final Spec` for public preview
- [] Final fee split lock for mainnet phase
- [] Detailed vesting lock (cliff, cadence) for mainnet phase
- [] Treasury release authority/process lock for mainnet phase
- [] Detailed unlock schedule table (mainnet-ready)
- [] Unit economics examples and test vectors (mainnet-ready)

19.3 Technical / Security

- [x] Contract spec baseline v1 referenced in technical docs
- [x] Threat model baseline v1 published
- [x] Security checklist baseline v1 published
- [] Audit plan (vendor/timeline/scope)
- [x] Monitoring + incident response baseline

19.4 Communication / Publication

- [x] Language scope locked: EN only for this release (VN follow-up release note)
- [x] Public FAQ complete (utility, risk, roadmap, limitations)
- [x] Messaging do/don't package complete for marketing/community

20. Appendix A - Glossary

Term	Short Definition
---	---
MVP	Minimum viable release for validating product direction
UGC	User-generated content
Entitlement	Valid ownership/access right used by game unlock logic
Indexer	Service that reads blockchain events and syncs to DB
Reconciliation	Process that cross-checks snapshot data with on-chain data
Fee split	Transaction value distribution logic (seller/platform/burn/staking...)
Burn	Permanent token removal from circulation
Hard cap	Maximum supply that cannot be exceeded

21. Appendix B - Editorial Notes for Post-Public-Preview Step

To move from `Whitepaper v1 Public Preview` to `Whitepaper v1.0 Final`, recommended next steps:

1. Clearly separate "target design" from "already implemented."
2. Mark all still-changing parameters as `TBD` or `Subject to final spec`.
3. Move illustrative code samples out of public whitepaper into technical docs.
4. Add clean vector architecture diagrams and `Buy -> Unlock` sequence tables.
5. Publish a short `Litepaper` version (6-12 pages) for mainstream readers.

22. Related Public Documents

- Help & FAQ: <https://market.kitchentogether.com/help>
- Risk Disclosure: <https://market.kitchentogether.com/risk-disclosure>
- Campaign Terms: <https://market.kitchentogether.com/campaign-terms>
- Terms of Service: <https://market.kitchentogether.com/tos>
- Privacy Policy: <https://market.kitchentogether.com/privacy-policy>

Internal working documents are not listed in this public preview edition.

Document Version: 1.0 Public Preview (Testnet-Only)

Release Date: 2026-02-27

Status: Public preview release with conditional mainnet roadmap and Subject-to-Final-Spec tokenomics/audit details