ksUSD

Product Roadmap

Keystone Finance · July 2026 · pre-mainnet → beta → scale

ksUSD: deposit USDC, hold one share token, earn carry from perp funding, jitoSOL staking, and USDC lending. v1 hedges on Phoenix Perps (USDC margin via Ember), holds jitoSOL unlevered as the spot leg, and switches between normal basis and parked automatically: an on-chain smoothed funding signal plus a dynamic threshold (the lending-vs-staking spread) decides the mode and a keeper bot executes the switch, with no manual intervention. Reverse basis is out of v1 scope — the backtest shows it contributes ∼0 on Solana's long-biased funding.

This roadmap runs as parallel pipelines. Each advances on its own clock; the launch gate is where they converge. Status: • done · • in progress · • planned · • dormant.

Pipeline 1 — Integrations (the v1 stack)

The four on-chain primitives ksUSD composes. Three are live and wired; the perp hedge is the gating dependency.

`Jito (staking)`, `Kamino (lending)`, `Jupiter (swaps)` → vault plumbing wired (devnet) → Phoenix basis live (when available) → full v1

Primitive	Role in ksUSD	Provider	Status
Staking	jitoSOL held unlevered (the spot leg); ∼5.8% APR	Jito	• done (devnet)
Lending	USDC lending for parked NAV, liquidity buffer, and reserve fund	Kamino	• done (devnet)
Swaps	USDC ↔ jitoSOL on mode entry/exit, slippage-bounded on-chain	Jupiter V6	• done (devnet)
Perp basis	Short SOL-PERP at 1×, USDC margin via Ember (the hedge)	Phoenix Perps	• in progress (see below)

Phoenix — the port is done. The perp leg is now Phoenix-only (the old Drift prototype fully purged); it compiles and fork-deploys against real cloned Phoenix/Ember state, and the live Phoenix program accepts the vault's `register_trader`. The one remaining gap is a single Ellipsis capability grant — the KEYSTONE grant (Pipeline 3) — that lets the trader account post margin. Until the hedge is live, the vault runs parked (USDC lending on Kamino), which is fully proven. Switching the hedge on is additive, not a rewrite.

Pipeline 2 — Protocol (to mainnet)

The program work that turns the stack into a deployable vault.

Phoenix/Ember CPI port → automated keeper transitions → devnet round-trip → audit → mainnet

Stage	Detail	Status
Single-program vault	One Anchor program, one PDA vault, one share mint	• done (devnet)
Automated transitions	On-chain funding safety floor + keeper dynamic threshold (lending−staking spread) + guardrails (12h dwell, EMA staleness, OI cap, depeg auto-pause); keeper bot executes	• done (logic on-chain; dynamic rule in keeper)
Phoenix/Ember CPI	Perp leg ported off the Drift prototype to Phoenix + Ember margin (Drift purged; Phoenix-only compiles + fork-deploys)	• done
Devnet round-trip	deposit → normal basis → parked → withdraw, end to end	• planned (gated)
Audit	Single-program audit, top-tier Solana firm (∼$40–80K)	• planned
Mainnet	Deploy with a hard TVL cap	• planned

Pipeline 3 — Venue & capacity

The external caps and grants that gate how much TVL the vault can hold.

KEYSTONE grant (Ellipsis) → Phoenix OI headroom → Kamino borrow caps → multi-LST

Constraint	Owner	Ask	Unblocks
Capability grant (KEYSTONE)	Ellipsis Labs	`CAN_DEPOSIT/CAN_WITHDRAW` on the vault's trader account	The hedge at all
SOL-PERP OI headroom	Phoenix	Confirm the short fits at $50M+ TVL	Per-vault size ceiling
jitoSOL borrow cap (∼$26.8M)	Kamino governance	Cap-raise as dormant-reverse demand approaches it	Reverse leg if activated
Multi-LST collateral	Jito / Kamino	jupSOL, mSOL alongside jitoSOL	Removes single-LST binding

Critical path: the KEYSTONE grant. This is the one dependency that gates the whole launch, so it gets an owner action and a decision gate, not just a table row.

Why the name: Phoenix (built by Ellipsis Labs) gates trading behind a one-time builder approval. Because the grant is issued to Keystone, we call it the KEYSTONE grant. Mechanically, a bare `register_trader` (already proven on the fork) leaves the trader at capability flags = 6 (place-market on; deposit/withdraw off). The grant is an on-chain `OnboardTraderDelegated` authorized by Ellipsis's `risk_authority`, flipping the trader to flags = 63. It is signed by a delegate key we hold, not the vault PDA — so it needs no program change, just the grant. On the fork we self-grant by patching `risk_authority`; on mainnet Ellipsis issues it.

- Now: direct outreach to Ellipsis for the KEYSTONE grant; confirm the builder fee (it eats carry — check whether it's waivable). The full open/close round-trip can be proven on the fork before the real grant lands.
- Decision gate (end of Q3 2026): if the grant isn't in hand, escalate directly with Ellipsis. The venue doesn't change — ksUSD is built for Phoenix; the pre-mainnet posture is to be ready to go live the moment the grant lands.

Pipeline 4 — Composability (the collateral flywheel)

Once ksUSD is accepted collateral elsewhere, holding it earns carry and unlocks borrow, which is what makes TVL sticky. The sUSDe playbook.

Kamino listing → Marginfi → Jupiter routing pair → DAO treasuries

Track	Detail	Status
Kamino Lend	List ksUSD as a collateral asset, the first and highest-leverage listing	• in progress (post-beta)
Marginfi	Second lending venue	• planned (later)
Jupiter routing	Default USDC↔ksUSD routing pair; one-click deposit/redeem from any token	• planned (later)
DAO treasuries	Direct integrations for sticky $100K+ allocations	• planned (later)

Each is a governance proposal plus a risk-parameter negotiation. Conversations open now (• in progress); listings close post-beta (• planned), once there's a live track record to cite.

Pipeline 5 — Growth (TVL)

Private beta ($1M) → public launch ($5M → $25M) → $20M+ TVL

Milestone	Cap	Status
Private beta	$1M, invite-only	• planned (gated on audit)
Public launch	$5M → $25M ramp	• planned
Scale	$20M+ TVL	• planned

Where deposits come from. Caps set the ceiling; these fill it:

- Anchor DAO treasuries (Pipeline 6): day-one TVL from a named customer.
- Yield marketplaces / listings: surface ksUSD to yield-seekers via venues like Exponent and RateX (targets).
- Composability (Pipeline 4): collateral acceptance pulls sticky TVL.

Day-one liquidity for treasuries. Instant buffer redemption covers normal size, but allocators want a secondary exit on day one, so seed a USDC↔ksUSD pool at launch and bring Jupiter routing forward (Pipeline 4) to the public-launch window.

Pipeline 6 — Anchor capital (de-risking the seed)

A named customer DAO that precommits capital does two jobs at once: it turns the seed raise from a bet on demand into a bet against a signed order book, and it seeds day-one TVL into the private beta. This is the lighthouse-customer play, and it can move now, ahead of live code.

Target DAO shortlist → precommitment (soft-circle) → seed de-risked → beta anchor deposit

Stage	Detail	Status
Target shortlist	DAO treasuries with idle USDC seeking on-chain yield, the $100K+ sticky allocations from Pipeline 4	• planned
Precommitment	A signed soft-circle to deposit at beta; converts a warm lead into committed day-one TVL	• planned
Seed de-risk	A named anchor plus precommitted capital is the strongest signal in the seed raise: demand proven before code ships	• planned
Beta anchor deposit	Anchor funds first into the $1M private beta and becomes the public reference customer	• planned (gated on beta)

The precommitment can be soft-circled now as a commitment conditional on beta whenever it lands (not a fixed date), which is what lets it sit ahead of the KEYSTONE grant rather than behind it. A firm beta date is downstream of Pipeline 3, so anchor the soft-circle on the terms and the conditionality, not on a date you can't yet promise.

Pipeline 7 — Capital & audit booking

Capital-in-hand gates the audit, which gates beta, which gates everything. This pipeline is the money and the calendar.

audit-firm shortlist (now) → seed raise ($200–500K) → capital in hand → audit slot booked → audit

Stage	Detail	Status
Audit-firm shortlist	Top Solana firms book 2–3 months out. For a Q3–Q4 audit, reserving a slot is a now item, not "planned"	• next
Seed raise	$200–500K; funds the $40–80K audit + ∼12mo runway	• planned
Capital in hand	The gate on booking the audit	• planned (gated on raise)
Audit	Single-program audit (Pipeline 2)	• planned (gated)

Pipeline 6 (anchor capital) de-risks this raise; the raise funds the audit on the critical path. Firm selection runs concurrent with the raise, not after it.

Pipeline 8 — Ops & security

For a solo founder running a mainnet vault, this is the pipeline a DAO treasury risk committee asks about first.

multisig + pause authority → monitoring / alerting → incident runbook → bug bounty → reserve sizing

Track	Detail	Status
Admin / guardian authority	End-state is a multisig; keeper-key custody and pause authority defined before beta	• planned (before beta)
Monitoring	NAV, mode, Pyth oracle staleness, jitoSOL/SOL depeg, keeper liveness, drawdown counter	• planned (before beta)
Incident runbook	Depeg auto-pause, oracle-divergence breaker, `emergency_close`, keeper outage (permissionless cranks keep the vault safe + redeemable)	• planned (before beta)
Bug bounty	Live post-audit	• planned
Reserve fund	5% perf-fee skim, lent on Kamino; first-loss buffer	• in progress (in design)

Oracle: Pyth SOL/USD + jitoSOL/USD, gated by 5-min staleness + 2% confidence; the depeg trigger reads the jitoSOL/SOL ratio.

Pipeline 9 — Legal & compliance

A synthetic yield-bearing dollar needs this tracked before beta, not treated as an afterthought.

entity → regulatory posture → geo-fencing / access → terms & disclosures

Track	Detail	Status
Entity	Operating / issuing entity to be established	• planned (before beta)
Regulatory posture	ksUSD is a non-pegged share, not a stablecoin (no peg defense), the framing counsel works from	• in progress (framing set)
Counsel	Engage counsel before beta for the yield-bearing-dollar review	• planned (before beta)
Geo-fencing / access	App-level access controls; disclaimers already in docs	• planned (before beta)

Convergence — the launch gate

The pipelines run in parallel, but they gate each other in a specific order, and the critical path is the relationship-and-capital chain, not the engineering:

- To mainnet: KEYSTONE grant (Pipeline 3) → Phoenix hedge live (1) + devnet round-trip (2); seed raised (7) → audit booked and complete (2 / 7).
- To beta: audit complete and ops/security (8) and legal (9) all clear: a DAO treasury won't deposit without the last two.
- To scale: composability + growth (4, 5): conversations open now, listings and TVL close after beta produces a live track record.

The dates live on the KEYSTONE grant, the seed raise, and the audit slot. A slip on any of those three moves everything downstream; the engineering pipelines don't.

Indicative timeline

When	Convergence point
Now	KEYSTONE-grant outreach · audit-firm shortlist · seed raise · anchor soft-circle (the relationship + capital critical path)
Q3 2026	Perp open/close round-trip proven on fork · mainnet-ready pending the KEYSTONE grant
Q3–Q4 2026	Audit complete · ops/security + legal ready · private beta ($1M cap, invite-only)
Q4 2026	Public launch ($5M → $25M ramp) · USDC↔ksUSD pool + Jupiter routing · first composability listing
Q1 2027	$20M TVL, first proof the scale arc works

The critical path is the KEYSTONE grant, the seed raise, and the audit slot, not the engineering. A slip on any of those three moves everything after it.

Dormant / past v1 (not in launch scope)

- Reverse basis: • dormant — deferred to v1.1, not in the Phoenix program. The backtest shows it adds ∼0 on Solana's long-biased funding (the reverse hurdle is hit ∼2% of days and never sustains); Parked covers negative funding.
- Multi-LST collateral: jupSOL / mSOL / bSOL alongside jitoSOL.
- Marginfi as a second lending venue (v1 is Kamino-only).
- Risk-profile buckets: only if the market asks.

> −0.3% max drawdown through the 2025–26 funding compression; ∼5% net APY fully loaded in a thin-funding regime like today, scaling up to ∼11% when funding is rich (24-month backtest, v1 normal + parked set).