Funding Layer (FDLY) – Whitepaper v1.0

*Back the Builder. Not Just the App.*

Powered by Solana. Launching via Raydium.

# 1. Abstract

Funding Layer introduces a decentralized protocol that redefines early-stage capital allocation by enabling tokenized investments in individual builders rather than specific apps. Through a novel model of builder token launches, bonding curves, and revenue-sharing contracts, the protocol creates an “Unlimited Funding Loop” — where backers earn from all future success of the builders they support. Powered by $FDLY on Solana, the platform eliminates traditional VC gatekeeping, distributes risk, and empowers talent from day one.

# 2. Introduction

Traditional funding models are broken. They rely on predicting the success of specific products, not people. As a result, talented builders are often overlooked, and investors are left exposed to high-risk, binary outcomes.

Funding Layer flips this model by letting the market fund individual builders early — and earn from all their future projects if they succeed. Think of it as "investing in Elon Musk before he launched anything."

By enabling trustless revenue-sharing, community-driven discovery, and continuous capital loops, FDLY aims to become the base layer of talent funding in Web3.

# 3. Problem Statement

The Web3 ecosystem faces three critical challenges:

3.1 High Failure Rates

90%+ of new apps or protocols fail. Most early investors lose everything. There is no real safety net for builders or backers.

3.2 No Early Support for Builders

Funding only flows after traction. But builders need capital before they succeed — for experimentation, development, and iteration.

3.3 Speculation Without Substance

Airdrops and memes drive engagement, but rarely sustain meaningful ecosystems. The current system rewards marketing over actual innovation.

# 4. Protocol Overview

4.1 Builder Profiles

Builders create public, on-chain profiles on Funding Layer. They provide links to GitHub, past projects, and proofs of skill. Optional integration with a decentralized “Karma Score” or “Talent Passport” enhances credibility.

4.2 Builder Token Launches

Each approved builder can launch their own token on Solana via Funding Layer. Tokens are issued using a bonding curve, so early supporters pay less. These tokens represent a claim on the builder’s future success.

4.3 Revenue Sharing

When a builder launches dApps, a portion of their revenues (protocol fees, DeFi yield, retroactive funding, etc.) flow into a splits contract. Token holders receive distributions in $FDLY or $SOL, based on their share. Revenue streams are permissionless, transparent, and auditable on-chain.

4.4 The Unlimited Funding Loop

This is the core innovation:

1. Builders receive upfront capital via token sales

2. They launch projects, generating revenue

3. Revenue flows back to supporters via splits

4. Builders gain reputation → more backers → more capital

This creates a loop of trust, incentive, and capital that fuels long-term builder success.

# 5. Tokenomics

Total Supply: 100,000,000 FDLY

Bonding Curve Sale: 60% (60,000,000 FDLY)

Liquidity Pool Migration: 20% (20,000,000 FDLY)

Team Allocation: 10% (10,000,000 FDLY) – 1-month cliff, 12-month vesting

Community Airdrops/Promotions: 10% (10,000,000 FDLY)

There is no private sale. No tokens are allocated to investors or VCs. All token allocations are transparent and on-chain.

# 6. Technical Architecture

Smart Contracts: Built in Solidity and deployed on Solana-compatible chains using Neon EVM or native Rust if extended later. Uses OpenZeppelin templates.

Bonding Curve: Implemented using a simple linear bonding curve with adjustable slope. Token price increases as more tokens are bought.

Splits Contract: Collects revenue from builder apps and distributes it proportionally to token holders using predefined weightage in $SOL or $FDLY.

Frontend: Built with React + TailwindCSS. Backend is decentralized, using IPFS or Arweave for storage. Identity verification may use SelfKey or Talent Passport.

# 7. Builder Reputation System

Reputation metrics are sourced from on-chain history, GitHub contributions, verified social accounts, and user reviews.

Scoring methods like Karma Gap or decentralized Talent Passports will be used to assess builder credibility pre-token launch. This ensures the ecosystem remains quality-first.

# 8. Governance Design

FDLY will evolve into a DAO once the ecosystem has matured.

Holders of $FDLY can propose and vote on:

- Which builders to onboard

- Emission adjustments

- Treasury allocations

- System upgrades and integrations

# 9. Roadmap

Phase 1: Website + MVP Live (Completed)

Phase 2: FDLY Launch on Raydium

Phase 3: Builder Token Launches Begin

Phase 4: Revenue Contracts and Splits Live

Phase 5: DAO + FDLY Staking + Governance Tools

Phase 6: Multi-chain Expansion and Builder Analytics

# 10. Security & Audits

Smart contracts will undergo community auditing via tools like Code4rena or CertiK Community Audit. Contracts use battle-tested libraries from OpenZeppelin.

Bug bounty programs and progressive audits will ensure secure rollouts.

# 11. Regulatory Risks & Disclaimers

FDLY is an experimental utility token. No promise of profit is made. All participants use the platform at their own risk.

Builders and backers are encouraged to consult their own legal advisors.

None of this is financial advice. Tokens may be subject to securities regulations depending on jurisdiction.

# 12. Conclusion

Funding Layer reimagines early-stage funding by placing builder talent at the center. With tokenized profiles, transparent bonding curves, and revenue-sharing mechanics, it introduces a sustainable investment model that rewards vision and execution alike.

# 13. References

Inspired by mechanisms from Gitcoin, Juicebox, Mirror, and Farcaster.

Token bonding math based on open bonding curve whitepapers and community iterations.