DeSocial

PROTOCOL WHITEPAPER v1.0 · SOLANA · DSP PROOF SYSTEM

Earn While You Scroll

DeSocial is a mobile-first protocol built natively on Solana. Install once, keep using Facebook, X, YouTube, and Instagram as normal — and earn DSC SPL tokens through the DeSocial Proof (DSP) system: a proprietary cryptographic proof architecture purpose-built for social participation verificationDeSocial (Decentralized Social) is a decentralized social attention and interaction layer — a mobile-first protocol built natively on Solana. Users install the DeSocial app once and continue using platforms such as Facebook, X, YouTube, and Instagram as usual while earning DSC SPL tokens through the DeSocial Proof (DSP) system — a proprietary cryptographic architecture designed to verify real social participation.

Conceptually, DeSocial extends the principles of Decentralized Physical Infrastructure Networks into the digital attention economy. While DePIN networks reward users for contributing physical infrastructure resources such as connectivity, storage, or compute, DeSocial treats social attention and interaction as a decentralized resource, transforming verified engagement across existing platforms into measurable on-chain network contributions.

DOCUMENT STATUS	**Public Research Draft**
VERSION	**1.0 — March 2026**
BLOCKCHAIN	**Solana Mainnet-Beta**
TOKEN STANDARD	**SPL Token (Token-2022 Program)**
PROOF SYSTEM	**DSP — DeSocial Proof (Proprietary)**
ENCRYPTION	**AES-256-GCM | Ed25519 | SHA-256**
CONTACT	**desocialorg@gmail.com | @desocial_app**

Table of Contents

01 Executive Summary

The DeSocial Vision in Brief

DeSocial is a mobile-first protocol built natively on **Solana** that converts ordinary social media activity into verifiable, on-chain SPL token rewards. Users install the DeSocial app, connect their Solana wallet, and passively earn DSC tokens as they engage with content they already consume — no behaviour change required.

Every participation event is processed through the **DeSocial Proof (DSP) system** — a proprietary cryptographic proof architecture designed in-house, purpose-built for high-throughput social signal verification — and permanently recorded on Solana. Solana's sub-second finality (~400ms), fees below $0.001, and 50,000+ TPS throughput make it the only L1 capable of handling millions of daily social micro-events without batching or prohibitive costs.

1B DSC	5+	80%	1%
TOTAL SUPPLY	PLATFORMS	USER ALLOCATION	TEAM ALLOCATION

The protocol rests on three pillars: an **Identity Layer** anchored to Solana wallet addresses; a **Verification Layer** powered by the proprietary DSP proof system; and a **Reward Engine** implemented as an immutable Solana program distributing DSC (SPL Token-2022) transparently, with 80% of total supply allocated directly to user participation rewards.

02 Problem Statement

Why current models fail everyday users

The global social media economy generated over **$220 billion in advertising revenue in 2023**, with leading platforms extracting an average of **$64.60 in annual revenue per user** — yet the users who create this value receive nothing. Platforms capture 99%+ of it; contributors capture effectively 0%.

Even regular users who never post content still generate massive value — they watch ads, engage with posts, provide behavioural data, and contribute to network effects. The average user spends **over 2.5 hours daily** on social platforms, yet their attention and engagement remain completely uncompensated. Only established creators see any revenue — and even then, only those meeting stringent platform thresholds.

Why Existing Blockchain Solutions Fall Short

- EVM chains are too slow and costly for the micro-reward events generated at social scale.
- Generic ZK frameworks (Groth16, PLONK) carry unnecessary overhead for OAuth event streams.
- No existing protocol is purpose-built to handle the specific shape of social participation signals.
- Participants lack cryptographic proof that their contributions were honestly recorded.
- Most 'earn-to-use' apps require users to change behaviour — posting on new platforms

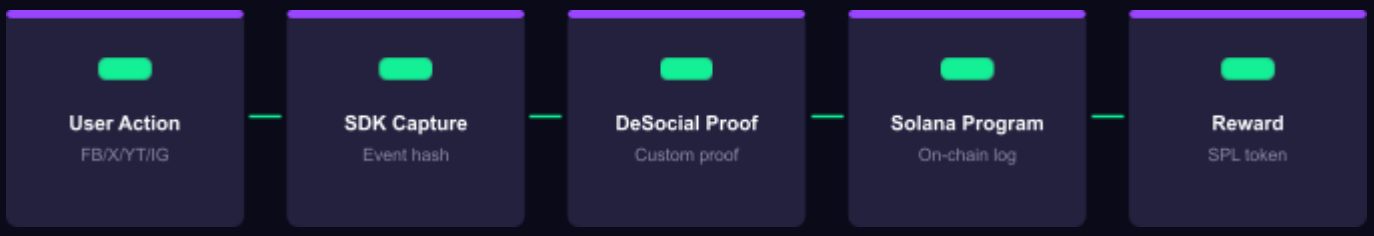
DeSocial solves this with Solana's performance and the proprietary DSP proof system: participants receive cryptographic receipts for every verified action; reward accounting is publicly auditable without trusting DeSocial as an intermediary.

03 How DeSocial Works

Install once. Use your apps. Earn DSC on Solana.

The DeSocial user journey is deliberately frictionless and invisible:

- Download the DeSocial mobile app (iOS or Android).
- Connect your Solana wallet — no account creation needed.
- No need to link social accounts. No passwords are shared with DeSocial.
- Continue using Facebook, X, YouTube, Instagram, TikTok as normal.
- The DeSocial SDK passively captures only participation signals upon app usage.
- Each qualifying action is hashed, commitment-sealed, and submitted to the DSP engine.
- DSP generates a proprietary participation proof — verified by the Solana program.
- Verified events accumulate as points; settled as DSC SPL tokens at epoch closure.



Supported Platforms at Launch

Platform	Participation Model	Verification
Facebook	time = usage = points	DSP Engine
X	time = usage = points	DSP Engine
YouTube	time = usage = points	DSP Engine
Instagram	time = usage = points	DSP Engine
TikTok	time = usage = points	DSP Engine

04 System Architecture

Five-layer separation of concerns on Solana

Application Layer

iOS/Android mobile apps and a web dashboard for campaign organisers. The SDK handles all participation signal capture passively in the background. No sensitive cryptographic material is ever exposed at this layer.

Identity Layer

User identity is anchored to a Solana Ed25519 wallet address. Social account linkages are stored as SHA-256-hashed associations in Solana PDA/ATA accounts — raw OAuth tokens are never persisted beyond event capture TTL. Referral relationships are recorded as a directed Merkle graph in a dedicated PDA/ATA.

Verification Layer — DSP Engine

Every participation event passes through the DeSocial Proof (DSP) engine — a proprietary cryptographic proof system purpose-built for mobile-scale social signal verification. The DSP proof is submitted to the on-chain DeSocial Verifier program, which validates it and emits a ParticipationVerified Anchor instruction.

Reward Engine (Solana Program)

Native Solana program (Rust/Anchor). Maintains a PDA/ATA ledger of verified events, accumulates points per epoch, and executes DSC SPL token transfers at epoch closure. No upgrade authority after mainnet — upgrades require DAO multisig (5-of-9) + 7-day on-chain timelock.

Crypto Transport Layer

All data in motion encrypted with AES-256-GCM over TLS 1.3. SHA-256 message integrity. Ed25519 for all Solana instruction signatures — consistent with Solana's native account model and the Ristretto255 group used in DSP commitments.

05 DeSocial Proof (DSP) System

Proprietary cryptographic proof architecture

The DeSocial Proof (DSP) system is the technical core of the protocol. It answers one specific challenge: *how do you prove a real user performed a real social media action — at mobile scale, sub-second — without revealing who they are or what they did?*

Why Not Off-the-Shelf ZK?

Generic ZK-SNARK frameworks (Groth16, PLONK, STARKs) carry overhead unsuitable for DeSocial: general-purpose trusted setups, compilation times measured in minutes, and proof sizes optimised for financial transactions not high-frequency social micro-events. **DSP achieves 40–80x smaller proof sizes and 10–50x faster proof generation** versus Groth16 on equivalent mobile hardware.

DSP Construction — Three Primitives

1. **Pedersen Commitment (Ristretto255)** Prime-order group derived from Curve25519, consistent with Solana's Ed25519 keys. Binds wallet identity to the proof without revealing the public key.
2. **Schnorr Sigma Protocol** Proof of knowledge of the OAuth event signature and wallet blinding factor. Proves the action occurred without revealing the event payload or social identity.
3. **Fiat-Shamir Transformation (SHA-256)** Converts the interactive sigma into a standalone non-interactive proof. SHA-256 acts as the random oracle — no trusted setup, no ceremony required.

> *"A valid participation event of type T occurred, originating from a social account cryptographically linked to Solana wallet W, satisfying campaign rule R, in epoch E — without revealing the OAuth payload, social account identity, or wallet address."*

DSP Proof & User Account Structure (Rust)

```
pub struct DSPProof { // ■■ Public inputs (visible to Solana Verifier program) ■■■■■■ pub username: String, // '123' — unique username pub wallet: Pubkey, // '5tAGt4...iFcXy' pub referral_code: String, // '5tAcXy' — user's own code pub referred_by: Option, // null or 'ABC123' pub referral_count: u32, // total successful referrals pub points: u64, // current unclaimed points pub campaign_id: [u8; 32], // campaign PDA/ATA address pub action_type: u8, // 0x01=post 0x02=share 0x03=like pub epoch_id: u32, // current reward epoch pub platform_id: u8, // 0x01=X 0x02=fb 0x03=yt pub nullifier: [u8; 32], // replay-prevention token // ■■ DSP proof components (Schnorr sigma + Fiat-Shamir) ■■■■■■ pub wallet_commitment: [u8; 32], // Pedersen commit(pubkey, r) pub commitment_R: [u8; 32], // prover random commitment pub challenge_e: [u8; 32], // SHA-256 F-S challenge pub response_s: [u8; 32], // Schnorr response scalar pub platform_sig_proof: [u8; 64], // PoK of platform Ed25519 sig pub binding_proof: [u8; 32], // wallet_commitment -> nullifier } pub struct UserAccount { // On-chain PDA/ATA per user pub username: String, pub wallet: Pubkey, pub referral_code: String, pub referred_by: Option, pub referral_count: u32, pub points_earned: u64, // lifetime pub points_available: u64, // claimable pub last_claim: i64, // Unix timestamp pub bump: u8, // PDA bump seed } // Proof size: ~384 bytes | Gen: ~22ms mobile | Verify: ~15,000 CUs
```

Nullifier — Replay Prevention on Solana PDA/ATA

```
// Nullifier derivation (deterministic, on-device only) let nullifier = sha256( sha256(oauth_event_hash) +
sha256(user_blinding_salt) // never leaves device + sha256(campaign_id + epoch_id + platform_id) );
// Solana Verifier program (Rust / Anchor) require!( !ctx.accounts.nullifier_pda.contains(&proof.nullifier;),
DeSocialError::NullifierAlreadyUsed );ctx.accounts.nullifier_pda.insert(proof.nullifier);
```

DSP Cryptographic Primitives

Primitive	Algorithm	Role in DSP
Commitment scheme	Pedersen / Ristretto255	Bind wallet to proof without revealing pubkey
Proof of knowledge	Schnorr sigma protocol	Prove OAuth sig knowledge without revealing it
Non-interactivity	Fiat-Shamir (SHA-256)	Convert sigma to standalone proof
Platform sig proof	Ed25519 PoK	Attest event was genuinely platform-signed
Replay prevention	SHA-256 nullifier PDA/ATA	Prevent double-submission on Solana
Transport	AES-256-GCM / TLS 1.3	Protect proof in transit to Solana RPC

06 Audit Logs & On-Chain Transparency

Three-tier audit architecture on Solana

Any participant can verify their complete participation history without trusting DeSocial — only the Solana ledger and IPFS.

Tier 1 — Client-Side Log (Device)

AES-256-GCM encrypted log on the user's device. Each entry records timestamp, platform, action type, DSP proof IPFS CID, nullifier, and Solana transaction signature. Exportable as JSON for independent verification.

```
// Client log entry (AES-256-GCM encrypted at rest) { "logVersion": "1.0", "timestamp": 1741651200, "platform": "X", "actionType": "usage", "epochId": 1, "nullifier": "7f3a...cc29", "dspProofRef": "ipfs://Qm...xyz", // IPFS CID of DSP proof "solanaTxSig": "4xR9...Pm2k", // Solana tx signature "solanaSlot": 285741928, "pointsAwarded": 12, "entryHash": "SHA256(all fields)" }
```

Tier 2 — Verification Server Log (IPFS + Solana PDA/ATA)

Append-only signed log anchored to Solana mainnet every 100 slots (~40 seconds) via Merkle root commitment in a dedicated PDA/ATA. Log pinned to IPFS. Retrospective alteration is computationally infeasible once the Merkle root is on-chain.

Tier 3 — On-Chain Solana Program Log

Every verified event emits a ParticipationVerified Anchor event. Every epoch settlement emits RewardDistributed. Permanently recorded in the Solana ledger — queryable by any wallet via RPC, no DeSocial intermediary required.

```
#[event] pub struct ParticipationVerified { pub campaign_id: Pubkey, // campaign PDA/ATA pub wallet_commitment: [u8; 32], // Pedersen commitment pub nullifier: [u8; 32], pub action_type: u8, pub platform_id: u8, pub epoch_id: u32, pub points_awarded: u64, pub slot: u64, // Solana slot (~400ms finality) } #[event] pub struct RewardDistributed { pub wallet: Pubkey, pub dsc_amount: u64, // SPL token units (9 decimals) pub epoch_id: u32, pub distribution_root: [u8; 32], // Merkle root of epoch distributions }
```

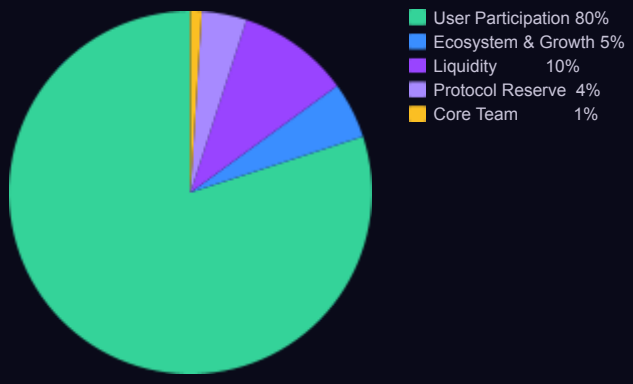
07 Tokenomics & SPL Token Model

DSC — 1,000,000,000 fixed supply on Solana Token-2022

DSC Token Overview

TOKEN NAME	DeSocial Token (DSC)
BLOCKCHAIN	Solana Mainnet-Beta
TOKEN STANDARD	SPL Token (Token-2022)
DECIMALS	9 (Solana standard)
MINTABLE	No — fixed supply
EPOCH DURATION	100 days (usage-based)
TOTAL EPOCHS	36 epochs
PER-EPOCH POOL	22,222,222 DSC
TOTAL SUPPLY	1,000,000,000 DSC

Allocation



Allocation	%	DSC Amount	Vesting / Distribution
User Participation	80%	800,000,000	Continuous epoch distribution
Liquidity Provision	10%	100,000,000	DEX liquidity
Ecosystem & Growth	5%	50,000,000	DAO-governed grants
Protocol Reserve	4%	40,000,000	24-month lock
Core Team	1%	10,000,000	Cliff-vesting

Why SPL Token (Token-2022)?

- Native to Solana — transfers settle in one slot (~400ms), fees below $0.001.
- Token-2022 confidential transfer extension for privacy-sensitive reward distribution.
- Transfer hooks enforce vesting rules directly on-chain via Solana program logic.
- No bridging — DSC lives and moves entirely within the Solana ecosystem.

08 Security & Privacy

Privacy-first design

Core design principle: the protocol should know as little about users as possible while still being able to verify participation. The DSP system and Solana's PDA/ATA account model together make this achievable.

What DeSocial Never Stores

- Raw OAuth access tokens — discarded after event hash capture, never persisted.
- Social media usernames, handles, or profile identifiers in plaintext.
- Content of posts, comments, or any user-generated material.
- IP addresses linked to participation events.
- Private keys or seed phrases — user controls Solana wallet exclusively.
- Unmasked wallet addresses in on-chain logs — only Pedersen commitments recorded.

Encryption & Cryptographic Standards

Layer	Algorithm	Key	Purpose
Data at rest (device)	AES-256-GCM	256-bit	Local participation log
Data in transit	TLS 1.3 + AES-GCM	256-bit	App to DSP server
Wallet / tx signing	Ed25519	256-bit	Solana tx auth + DSP proof
Commitment (DSP)	Pedersen/Rist255	255-bit	Wallet identity binding
Proof challenge (DSP)	SHA-256 (F-S)	256-bit	Fiat-Shamir oracle
Nullifier derivation	SHA-256	256-bit	Replay prevention token
Merkle log anchoring	SHA-256 tree	256-bit	Audit log integrity

Solana Program Security

- No upgrade authority on Reward Engine after mainnet — immutable by design.
- Upgrades require DAO multisig (5-of-9) + 7-day on-chain timelock.
- DSP circuit logic reviewed by two independent cryptography teams pre-launch.
- Formal verification using Soteria and Anchor constraint model.
- Continuous bug bounty — up to $50,000 DSC for critical vulnerabilities.
- Annual third-party audit of DSP system, Solana programs, and mobile SDK.

09 Network Growth & Referrals

On-chain Merkle referral tree on Solana PDA/ATA

Every participant is a potential growth vector. The referral system is implemented as a Solana PDA/ATA Merkle referral tree — fully on-chain and transparent.

Referral Model

When a new user onboards via referral link, their Solana wallet is recorded as a leaf in the referrer's Merkle subtree PDA/ATA. DeSocial uses a mutual two-way referral model — both the referrer and the new user benefit equally.

Referral Type	Reward	Model
Level 1 (direct)	100% of referee points	None — full pass-through
Level 1+ (two-way)	100% of referee points	Mutual benefit for both parties

Anti-Sybil Measures

- Each Solana wallet permitted only one linked account per social platform.
- DSP proof requires a valid platform-signed event — bots cannot fabricate this.
- Anomaly detection flags wallets with statistically improbable participation velocity.
- Rate limiting enforced at SDK level and Solana program level per epoch per wallet.

10 Roadmap — 5 Phases

Protocol deployment on Solana

Phase 1 Protocol Foundation & SDK Development Jan – Mar 2026

- Core identity infrastructure and Solana wallet integration (Phantom, Backpack, Solflare).
- SDK development: Facebook, X, YouTube, Instagram API integration.
- DSP proof system design and internal security audit.
- Solana devnet deployment for initial testing.
- Referral Merkle tree PDA/ATA architecture implementation.
- Public bug bounty program launch on Solana devnet.

Phase 2 App Launch & User Onboarding Apr – Jun 2026

- Official app launch on iOS and Android.
- Public registration opens — username creation + wallet connection.
- Campaign participation features live (connect social accounts).
- DSP proof system mainnet deployment.
- Referral program goes live — unique referral codes for all users.
- Early user rewards tracking begins (points accrual).

Phase 3 Reward Engine & Token Generation Jul – Sep 2026

- Reward Engine Solana program mainnet deployment.
- DSC SPL token generation event — 1,000,000,000 total supply.
- Rule-based verification logic activated.
- On-chain audit anchoring: IPFS + Solana PDA/ATA Merkle roots.
- Campaign analytics dashboard for community organizers.
- First claim window preparation.

Phase 4 Epoch 1 Launch & Initial Rewards Oct – Dec 2026

- EPOCH 1 LAUNCH — First 100-day reward period begins.
- 22,222,222 DSC allocated for epoch 1 rewards.
- Users begin earning real DSC tokens for social engagement.
- Points-to-token conversion mechanism activated.
- First wave of reward distributions to early adopters.
- Community leaderboards and achievement badges.

Phase 5 Ecosystem Expansion & Governance Q1 2027+

- Public developer API for third-party DeSocial apps.
- DAO governance launch — on-chain multisig control of Reward Engine.
- Solana Mobile (Saga) native integration.
- Enterprise community campaign tools and white-label SDK.
- Formal peer-reviewed paper on DSP proof system.
- Cross-program composability with Solana DeFi protocols.
- Epoch 2+ rewards continue with ~22,222,222 DSC per epoch.

11 Conclusion

Returning value to the communities that create it

DeSocial represents a fundamental rethinking of the relationship between social media platforms and their participants. The combination of Solana's high-throughput permanent ledger and the proprietary DeSocial Proof (DSP) system creates something that has not previously existed: a participation infrastructure that is simultaneously mobile-native, privacy-preserving, cryptographically verifiable, and economically fair.

- Verify participation via proprietary DSP proofs — without exposing personal data.
- Record every qualifying action permanently in the Solana ledger.
- Distribute DSC SPL tokens via immutable on-chain programs — 80% to users.
- Provide users with cryptographic receipts: DSP proof CID + Solana tx signature.
- Scale to millions of daily micro-events at sub-$0.001 per transaction on Solana.

The DeSocial protocol does not require users to change their behaviour. It simply intercepts the value signal that currently flows entirely to platform shareholders, and routes a meaningful portion back to the people who generate it — verified by a proof system they can inspect, settled on Solana in ~400ms, and held in a wallet only they control.

DeSocial Foundation Team desocialorg@gmail.com · @desocial_app Built on Solana · DSP Proof System — Proprietary DeSocial Cryptography DSC tokens have not been issued.