FUDCOIN: Credits, Cases & Creator Revenue Share

Version: 1.0
Date: 3 October 2025
Networks: Cardano mainnet

Abstract

FUDCOIN presents a game platform on Cardano where players purchase in-app credits with ADA, open themed “cases” with transparent odds, receive credit-based payouts, and can withdraw credits back to ADA. The platform includes two product lines. **Normal Cases**, **Jackpot Cases** and a **Creator Revenue Share** program that links specific cases to creator wallets with per-open fees and public earnings leaderboards. A special, live example of a creator case is the **SNEK Case**, whose fee stream accumulates in credits and is periodically used to market-buy and burn SNEK on-chain.

---

1. Introduction

The aim of FUDCOIN is to merge on-chain settlement with a responsive, low-latency game experience. Players fund gameplay using ADA, receive and spend in-app credits, and can redeem credits back to ADA. The platform emphasizes clarity around odds, payouts, and fee routing—particularly for Creator Cases— while providing public transparency via leaderboards and payout snapshots.

---

2. Product Overview

2.1 Core Loop

1. **Connect Wallet:** Player connects a Cardano wallet.
2. **Buy Credits:** Player purchases credits using ADA (or uses existing credits).
3. **Open Case:** Player selects a case; each case displays its **symbols**, **rarities**, and **payouts**.
4. **Instant Result:** A symbol is assigned in line with advertised odds; credits update immediately.
5. **Withdraw:** Player can request to withdraw credits for ADA via the **Stash** page.

2.2 Case Families

- **Normal Cases:** Fixed odds and symbol payouts.
- **Jackpot Cases:** A configurable percentage of each open contributes to a shared jackpot pot. If the jackpot symbol is hit, the pot is won in full and the pot resets to its seed value.

2.3 Creator Cases (Revenue Share)

- Each Creator Case is tied to a **unique creator wallet**.
- A **custom per-open fee** routes to that wallet (as credits) when the case is opened.
- A **public leaderboard** displays **lifetime earned credits** per creator for full transparency.
- Onboarding: via the **adaorca portal** for **69 ADA + 250 ADA worth of $FUDCOIN** (pricing subject to change).

---

3. Credits Economy

3.1 Unit of Account

- **Credits** are the in-app balance used to open cases and represent winnings.
- **Purchases:** ADA → credits conversion shown at point of purchase.
- **Redemptions:** Credits → ADA conversion shown at withdrawal initiation.

3.2 Balance Flow

- **Increase:** case wins, promotional credits, revenue share accruals (for creators).
- **Decrease:** case opens, fees where applicable, withdrawals.

3.3 Withdrawals

- Standard withdrawal requests are batched and processed periodically.
- **Creator claim threshold:** 200 credits (operational policy; subject to revision).
- Payout timing and minimums can vary based on network conditions and anti-abuse controls.

---

4. Case Mechanics

4.1 Symbols, Rarity, Payouts

- Each **case** defines a set of **symbols** Si , with:
  pi = probability of drawing Si ,
  xi = credit payout for Si .
- The platform displays symbol odds and payouts directly in the case UI.

4.2 Expected Value (EV)

For a case with open price C credits (net of explicit per-open creator/jackpot fees):

$$\mathrm{EV} = \sum_i pi \cdot xi.$$

Return-to-Player (RTP) is:

$$\text{RTP} = \frac{\text{EV}}{C}.$$

(Note: jackpot contributions and creator fees may be modeled as part of C or segregated as additive streams—see §5 and §6.)

4.3 Fairness & Result Determination

- Results are produced by a verifiable randomness pipeline with deterministic visualization in the client.
- Server assigns the outcome based on committed entropy and nonce; the UI then animates to that outcome.
- See **§7 Fairness & Transparency** for the commitment model and audit surfaces.

---

5. Jackpot Cases

5.1 Pot Accumulation

- Each open contributes a configured **jackpot fee** J credits into the pot.
- The pot increases monotonically until a jackpot hit occurs, then **resets to a seed** value J_0.

5.2 Jackpot Hit

- Hitting the **JACKPOT symbol** awards **the entire live pot** to the player as credits.
- After payout, the pot resets to $J0$ and begins building again.

5.3 Modeling EV with Jackpot

Let π be the probability of the jackpot symbol per open. Let **BaseEV** be the EV from non-jackpot symbols. Then incremental EV from jackpot is approximately:

$$\text{EV}_{\text{jackpot}} \approx \pi \cdot P,$$

where P is the live pot at open time. Full RTP must consider dynamic P and contribution J per open.

---

6. Creator Revenue Share

6.1 Overview

- **Per-open creator fee:** For a Creator Case, on each open the platform routes a configured fee to the **creator wallet** (credited in-app).
- **Public Leaderboard:** We publish a **Creators Earnings** leaderboard showing **lifetime earned credits** by creator.

6.2 Onboarding & Pricing

- **How to apply:** Through the **adaorca portal**.
- **Current onboarding price:69 ADA + 250 ADA worth of $FUDCOIN** (covers case creation, symbol design import, rarity calibration, QA).
- **Customization:** Creators propose symbols and rarities; the team composes a balanced case and publishes displayed odds/payouts.

6.3 Claims & Payouts

- **Minimum claim:** 200 credits by default.
- **Payout options:** ADA withdrawals to the creator's designated wallet.
- **Snapshots:** Periodic public snapshots of creator accruals and payouts.

---

7. Fairness, RNG & Transparency

7.1 Commit–Reveal (High-Level)

- The backend maintains a **server seed** and per-open **nonce**.
- Prior to use, the server may **commit** to a seed hash (e.g., HMAC or SHA-256).
- The result is derived from PRF(serverSeed,nonce) → mapped to the case's distribution.
- On rotation, the seed is **revealed** and can be verified against prior commitments.

7.2 Client Visualization

- The client UI is a **deterministic visualization** of a server-determined result (e.g., reel stopping at a predetermined index).
- Animation speed does **not** affect the drawn result; it only affects user experience.

7.3 Public Surfaces

- **Odds & payouts** per case are shown in-app.
- **Creators Earnings leaderboard** is public.
- **Payout snapshots and burn events** (e.g., SNEK burns) are published regularly.

(Exact cryptographic details, rotation cadence, and verification endpoints may be expanded in a technical appendix as they are finalized.)

---

8. Special Case: SNEK Case with Burns

8.1 Fee Routing

- **10% of every SNEK case open** is routed to the **$burnsnek** wallet **as credits**.
- The **$burnsnek** wallet **cannot withdraw credits directly**.

8.2 Buy-and-Burn Process

- Credits accumulate until a **target threshold** of ~**10,000 credits ≈ 100 ADA** is reached.
- The platform executes a **market buy of SNEK** with the accumulated amount and **burns** the purchased tokens.
- **Rationale:** Batch conversions reduce fees and operational overhead; timing is adjusted based on traffic and cost conditions.

8.3 Transparency

- We publish **burn transaction references** and **running totals burned**.
- The SNEK case thus functions as a **creator case that reduces circulating supply** through periodic buy-and-burn.

---

9. Fees, Limits & Policies

- **Case price:** displayed per case in credits.
- **Jackpot contribution:** configured per jackpot case, shown in UI.
- **Creator fee:** configured per Creator Case and shown in UI.
- **Creator minimum claim:** 200 ADA worth of credits (policy).
- **Withdrawals:** batched, with anti-abuse checks; processing windows may vary.
- **Price changes:** onboarding price and fees may change; any changes are announced in advance.

---

10. Security & Operations

- **Wallet connections:** Native Cardano wallets; standard permissions only.
- **Funds handling:** ADA inflows/outflows are tracked; credits ledger mirrors gameplay state.
- **Abuse prevention:** Rate limits, anomaly detection, freeze/review capabilities to protect the pool and players.
- **Key management:** Segregated credentials; least-privilege access; audit trails for payouts/burns.
- **Availability:** Redundant infrastructure and monitoring; planned maintenance windows communicated ahead of time.

---

11. Transparency & Reporting

- **Leaderboard:** Creator lifetime credits and case-level performance metrics.
- **Snapshots:** Periodic snapshots of jackpots, creator balances, and payout histories.
- **SNEK burns:** Public posting of accumulated credits, buy executions, and burn transactions.
- **Change logs:** Versioned updates to case odds, fees, and platform policies.

---

12. Governance & Roadmap

12.1 Governance (Initial)

- Operational control by the core team; community feedback via Discord and social channels.
- Creator program evolves based on measurable demand and quality criteria.

12.2 Roadmap (Indicative)

- **RNG verifiability expansion:** Public seed commitments & reveal endpoints with documentation.
- **Self-serve creator onboarding:** Streamlined case builder with automated rarity validation.
- **Expanded jackpots:** Multiple jackpot tiers and progressive pots.
- **On-chain attestations:** Optional proofs of payouts and burn events published to chain or IPFS.
- **Mobile UX:** Optimized flows and performance tuning for handheld devices.

---

13. Legal, Compliance & Risk Factors

- **Jurisdictional constraints:** Access and withdrawal features may vary by region and local regulation. Players are responsible for abiding by local laws.
- **Volatility:** ADA and any referenced tokens (e.g., SNEK) are volatile; credit valuations in fiat terms can change materially.
- **RNG & fairness:** While commit–reveal reduces manipulation incentives, improper implementation or delayed reveals can degrade assurances; audits and public checks help mitigate this.
- **Operational risks:** Downtime, queue backlogs, service provider failures, or adversarial activity may impact withdrawals or gameplay temporarily.
- **Policy changes:** Minimums, fees, and thresholds can change; such changes will be announced ahead of time when feasible.

---

14. Technical Architecture (Overview)

- **Frontend:** Case UI with deterministic animations; wallet connect; odds and payout displays.
- **Backend:** Result assignment, jackpot accounting, creator fee routing, credits ledger, withdrawals orchestration.
- **RNG:** Seed + nonce PRF mapping to case distributions; commit–reveal lifecycle.
- **Storage:** Relational store for credits/jackpots/leaderboards; object store for snapshots.
- **Observability:** Structured logs, metrics, and alerts; public metrics endpoints where appropriate.

---

15. Community & Contact

- **Creator applications:** DM **@acoustio** or visit Discord. Provide your **case theme**, **symbols**, **rarities**, and **payout vision**.
- **Players:** Try the **snek** case, share your pulls, and help accelerate the next burn.
- **Transparency:** Follow public leaderboards, payout snapshots, jackpot states, and burn notices.

---

16. Disclaimer

This document is for informational purposes only and is not investment, legal, or tax advice. Features, parameters, thresholds, and pricing are subject to change without notice. Access to the platform may be restricted in certain jurisdictions. By using the platform, you agree to abide by applicable laws and the platform's terms.

Appendix A — SNEK Burn Lifecycle (Example)

1. SNEK case opens route 10% of open price → **$burnsnek** credits.
2. Credits accumulate to **~10,000**.
3. Execute **market buy** of SNEK with the accumulated ADA value.
4. **Burn** the purchased SNEK (irreversible).
5. Publish **Tx references** and updated totals.
6. Continue accumulating anew.

Appendix B — Fairness Notes (Non-Normative)

- **Entropy:** Server seed periodically rotated; nonce increments per open.
- **Mapping:** PRF output → symbol index using rejection sampling or weighted cumulative mapping to preserve exact odds.
- **Audit:** On seed reveal, users can recompute prior outcomes for the cycle; discrepancies should not occur.

Appendix C — Glossary

- **Credits:** In-app accounting unit for gameplay and payouts.
- **Case:** A set of symbols with assigned odds and payouts.
- **Creator Case:** A case that routes a per-open fee to a designated creator wallet.
- **Jackpot Pot:** Accumulating pool funded by per-open contributions; paid out in full on jackpot hit.
- **RTP:** Return-to-Player; EV divided by case cost.
- **Commit–Reveal:** A method for verifiably binding randomness prior to use and revealing it later for verification.