

# Gruesome's Arcade

Classic browser games, on-chain rewards, real reputations

## WHAT IT IS

# Playable arcade with a real economy

- Play classic browser games in a unified shell.
- Earn credits + climb leaderboards (daily + weekly).
- Proof-of-Humanity gating for payout eligibility.
- Avatars + PRO membership (mint + renew).

### Core loop

Connect → get SYNC (credits/membership/avatar) → request run → submit result → leaderboard + payout

# Players + creators both win

## Players

- Instant play (no install).
- Credits + PRO unlocks.
- Skill + activity leaderboards.
- Weekly payouts (eligible humans).

## Creators

- Drop-in embed adapter.
- Metrics catalog + validation.
- Shared economy + payouts.
- Revenue share + treasury model.

# One shell, many games

- 1) Wallet connect (injected + optional WalletConnect).
- 2) Backend session creates a SYNC payload (credits/membership/avatar).
- 3) Game iframe requests a run; backend grants runId + cost.
- 4) Game posts RUN\_RESULT; backend validates + records metrics.
- 5) Leaderboards render from server-authoritative results.
- 6) Epoch settlement writes claim roots; users claim payouts on-chain.

# Daily + weekly leaderboards

## Weighted distribution

Split rewards between skill + activity (server policy).  
Proof-of-Humanity gates eligibility to keep payouts human.

## Revenue routing

mUSD purchases fund daily + weekly pots + ops.  
Non-mUSD (TBAG/RUSTYAI) defaults to Treasury for solvency; optional partial direct routing.

# Multiple metrics per game

Each game declares: defaultMetric + metrics[] + direction + clamps.

Leaderboards can show:

- Skill boards (accuracy, efficiency, speedrunMs...)
- Activity boards (attempts, durationMs...)

Backend remains authoritative: anti-cheat, caps, and eligibility filters.

# Spend-aware metrics

Primary ranked metric:

- waves (higher)

Supporting metrics:

- efficiency (higher)
- inRunSpendAC (lower)
- kills (higher)
- accuracyBp (higher)
- durationMs (context)

Purpose: reward skill without letting spend dominate.

# Admin mode + 3D data map

## Admin-only UI:

- Snapshot endpoint (pots, treasury, counts)
- Manual controls + auditing

## 3D map:

- Visualize users + payouts + activity
- Useful for debugging + community transparency

Admin wallets can appear on leaderboards but can be excluded from payouts.



# Consistent IDs + labels + rules

Central registry (arcade-metrics-library.json):

- id, label, description
- direction (higher/lower)
- type + unit + clamp
- category (skill/activity)

Why it matters:

- UI can render tooltips consistently
- Backend can validate + clamp results
- Future games plug in without breaking leaderboards

ON-CHAIN LAYER

# Contracts: money movement + receipts + claims

## What chain does

ArcadeEpochVaultV2: merkle claims for daily/weekly payouts (epoch ids).

ArcadePaymentsRouterV2: accepts mUSD + TBAG + RUSTYAI; routes to pots/ops/treasury; emits receipts.

ArcadePromo: promo claim receipts.

ArcadeProAvatar: PRO membership + Gen1 avatar NFT (SBT → tradeable unlock).

## What chain does NOT do

Leaderboard ranking + anti-cheat (server-authoritative).

Game logic (runs are verified off-chain, then committed).

Automatic token swaps: Treasury holds non-mUSD by default to keep mUSD payout solvency simple.