Shard Frontier — Wave-2 Submission Notes

Summary
Shard Frontier is the first end-to-end Web3 game built on BlockDAG, featuring an interconnected ecosystem of on-chain NFTs, a backend session engine, a crafting economy, token burning, and X1 miner integration.
Players earn in-game materials, forge them into tradable NFTs, and interact with a live backend powered by Railway + Postgres, while all assets are minted on the BDAG testnet.
This establishes a reusable blueprint for games, apps, and third-party developers to integrate with the BlockDAG ecosystem.

Problem
BlockDAG has millions of X1 miners, a growing developer base, and NFTs arriving — but no unified loop ties them together. Miners mine in isolation. NFTs exist in isolation. dApps don't yet leverage X1 mining sessions.

Solution
We built the first ecosystem loop where:
- X1 mining sessions (simulated endpoints today) trigger **in-game energy boosts**.
- A **Railway backend** processes gameplay → boosts → mining rewards.
- **NFT Shards** can be forged, refined, combined, and traded.
- All components run on **BlockDAG Awakening Testnet** with working smart contracts and RPC calls.

Technical Execution (Phase-2 delta)

Smart Contracts (Awakening Testnet)
- **ShardNFT (ERC-721)** — live at `0x0F2F6F22Aa68b11295e2FbEb07416c8910481c11`
  - Payable mint price: **5 BDAG**
  - **BaseTokenURI** rotation via script; **IPFS** metadata structure locked (no `.json` extension; `baseURI + tokenId`).
- **BonusBDAGToken (ERC-20)** + **BurnVault** (prototype burn engine)
  - Mint, approve, transfer, burn workflows proven on testnet (occasional RPC latency acknowledged).

Metadata & IPFS
- Rebuilt a correct folder layout:
  `/images/{n}.png` and `/json/{id}` (no extension).
- Inventory resolves **tokenURI → IPFS gateway** and displays **images + traits** from **Trait Matrix v1.1**.

Backend (Railway + Postgres)
- Public health endpoint: `/health`
- X1 simulation endpoints scaffolded:

  - `GET /x1/summary`
  - `GET /x1/wallet/:address/boosts`
  - `POST /x1/session/start`
  - `POST /x1/session/complete`
- Ethers provider wired to Awakening RPC; signer via env; CI/CD from GitHub to Railway.
- Database attached; schemas drafted (`users`, `sessions`, `boosts`, `burn_history`) for future waves.

Frontend (Vite + React + wagmi/Web3Modal)
- **Mobile-first 9:16** layouts (edge-to-edge canvas + letterboxing).
- Start → Home → Dashboard → Rooms (Forge, Refine, NFT Forge, Garage, Hover Bay, Medals, Profile, Map) with invisible hotspots mapped to concept art.
- **Inventory** screen: owned shard enumeration, tokenURI fetch, **IPFS metadata & images**, trait lists, and detail view.
- UI on-chain writes are **disabled for judging**; minting is performed by Hardhat scripts to avoid testnet instability.

Highlights
- Mobile-first **9:16** UI, fast, lightweight.
- **WalletConnect** integration on testnet (Chain ID 1043).
- **Shard Inventory** with **real IPFS metadata** and images.
- Validated **burn engine** direction (ERC-20 + BurnVault).
- **Railway** backend with clean latency and CI/CD.
- Minimal on-chain tx in demo path; off-chain gameplay for scale.

Scalability
- Backend runs **stateless** on Railway; auto-scales horizontally.
- Postgres connection pooling; caching and rate-limit plan documented.
- Off-chain gameplay with on-chain settlement (forge/mint) → cost-efficient and scalable.
- API endpoints designed to be **cache-friendly**; contract calls minimized.

Ecosystem Blueprint (reusable by BlockDAG projects)
- Gaming integrations with X1.
- Mining-boost loops and **burn-to-boost** mechanics.
- NFT-driven reward mechanisms and crafting economies.
- Real-time analytics and expansion via Telegram/WhatsApp mini-apps.
- Composable shard system across multiple future games.

What to Review (Judge Flow)
1. Open demo: **/home** → connect wallet → go to **/inventory**.

2. See **owned Shards** rendered with **IPFS metadata + images**.
3. (Optional) Inspect contract on explorer; check baseURI/tokenURI.
4. (Optional) Hit backend health and sample X1 endpoints.

Live Links
- **Demo (Railway):** https://shard-frontier-production.up.railway.app/home
- **Backend Health:** https://radiant-fascination-production.up.railway.app/health
- **Contract (Awakening):** 0x0F2F6F22Aa68b11295e2FbEb07416c8910481c11

Known Constraints (Wave-2 safety)
- On-chain mint buttons are intentionally disabled in UI to avoid testnet gas/RPC issues; mint via Hardhat scripts only.
- Occasional Awakening RPC latency is noted; retriable with scripts.
- Env vs fallback: UI reads contract from env with a safe fallback to the live address.

Credits
- Frontend UI/flows, Inventory IPFS integration, concept-aligned screens: **Jaime**
- Backend (Railway + Postgres), ERC-20 + BurnVault + X1 simulation endpoints: **Co-producer**