ECHOES OF THE SANKOFA

Comprehensive Design Overview — Updated with Autobattler Clarification

1. Core Concept

- Echoes of the Sankofa is a Ghanaian-inspired strategy autobattler roguelite about legacy, courage, and rebirth.

- The player, known as the Ase Keeper, summons spiritual heroes (Echoes) to reclaim lost realms, build the Sanctum, and guide generations of warriors bound by ancestral memory.

- Progress is shaped by courage, faith, and strategic resource use.

2. Combat System — Clarified Model

- Combat Type: Autobattler simulation with visible sequential rounds (no direct player commands).

- Internal Engine: Round-based AI system ('turn-based internally') for deterministic action resolution.

- Heroes act automatically by initiative, morale, and traits. Players watch battles unfold.

- Player Influence: Pre-battle squad composition, morale management, class synergy, and relic assignment.

- Future Expansion: Once the base is sufficiently upgraded, the player may plan multi-team strategies or set tactical 'plans' for squads during long Realm campaigns.

3. Core Gameplay Loop

- Summon heroes using Ase (core resource) and Relics.

- Form 1–6 member teams with complementary traits and classes.

- Deploy into procedural Realms with unique objectives and environments.

- Clear objectives to earn Ase, Ekwan, and Relics; failure results in wounds or death.

- Return to Sanctum to upgrade, craft, perform rituals, and assign heroes to duties.

- Repeat; over time, dead heroes influence future generations through the Legacy system.

4. Economic & Progression Systems

- Ase: Life■force currency; generated actively and passively; used for summoning, upgrades, and rituals.

- Ekwan: Combat-earned essence; used for rank-ups and research.

- Relics: Rare material from Realms; used for crafting, upgrades, or sacrifice rituals.

- Hero Tasks: Assign idle heroes to Meditation, Crafting, or Expeditions for background resource gain.

- Sanctum Upgrades: Long-term progression; maxing out the base requires ~120 hours casual or 80 hours intense play.

5. Realm & Objective System

- Procedural Realms include objective types: Purify Shrine, Protect Totem, Slay Spirit Lord, Recover Relic, Escort Soul.

- Each Realm offers different risk/reward structures with scaling Ase/Ekwan/Relic output.

- Environmental effects and fear zones interact with hero traits like Courage or Faith.

- Failure can cause permadeath or long-term debuffs; success yields morale bonuses and permanent growth.

6. Hero System

- Heroes have 6 traits: Courage, Virtue, Wisdom, Temperament, Faith, and Ambition.

- Classes define combat style; low-tier heroes learn classes over time, higher tiers start specialized.

- Legacy: Fallen heroes contribute fragments that empower future summons.

- Dialogue & Personality: Heroes express advice, warnings, or ambitions; personalities influence morale and success chances.

7. Sanctum & Base Management

- Ase Flame: Generates Ase, centerpiece of the Sanctum.

- Forge of Souls: Trait inheritance and hero synthesis.

- Shrine of Voices: Morale recovery, hero interaction, and event dialogues.

- Hall of Lineage: Stores ancestral echoes for future bonuses.

- Diviner's Library: Researches skills, relics, and crafting recipes.

- Marketplace: Trade Ase↔Ekwan; dynamic daily rates.

- Caretaker's Chamber: Handles hero wounds and recovery.

- Duties System: Idle heroes perform tasks that generate Ase, Relics, or healing benefits.

8. Future Combat Evolution (Player Agency Unlocks)

- As the Sanctum advances, players gain partial control through 'Tactical Planning'.

- This includes pre■battle role assignments, multi■team deployments for extended Realm campaigns, and optional auto■priority adjustments.

- This ensures progression from passive observation to light strategic oversight, mirroring the Keeper's growing wisdom.

9. Development Roadmap Snapshot

- Finalize Crafting & Research economy (next task).

- Model Combat Traits and AI logic numerically for MVP simulations.

- Run balance simulations to validate resource pacing (TODO before build).

- Build text-based MVP to validate economy, combat outcomes, and resource flows.
- Expand with UI prototypes and visual layers after MVP stability.