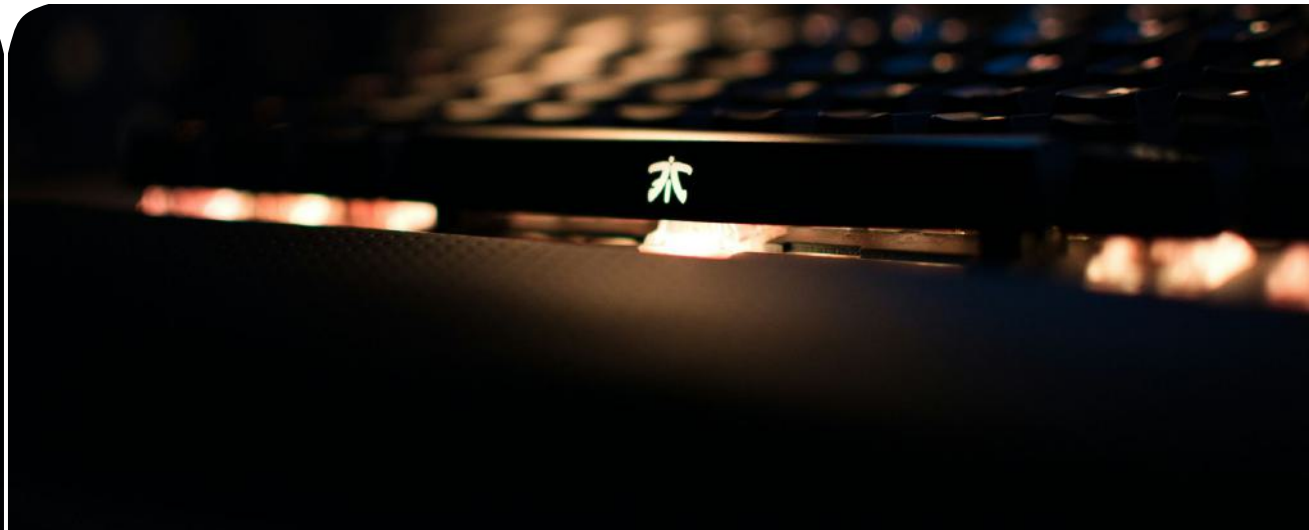
Hawamoni — Project Plan

BLOCKCHAIN TREASURY MANAGEMENT FOR TEAMS

HAWAMONI CORE TEAM



Hawamoni Project Overview

Hawamoni is a blockchain-native treasury management system designed for small groups and student organizations. It leverages the Solana blockchain and Solana Pay to provide secure, auditable financial governance with on-chain approvals and transparent transaction records. The platform enables groups to manage deposits, approve withdrawals, and maintain an auditable ledger while keeping UX simple for non-technical users.

STAKEHOLDER FOCUS

- Student groups
- Campus societies
- Project teams
- Small co-founder groups

Table of Contents

Team Roles

Akeem Jr

Frontend Developer and AI Engineer.

Henry

Backend Engineer

Joshua

Blockchain Developer

Phase 1 — Setup & Foundation (Day 1 and 2)

- Repository Setup: create GitHub repository, define branching strategy, and configure CI/CD with GitHub Actions.
- Environment Configuration: provision developer environments, cloud resources, and development databases; install Solana toolchain and wallets.
- Core Architecture: define API contracts between services, design database schema, and draft Solana program account layout.

Key Deliverables:

1. CI/CD pipeline active and passing initial builds.
2. Project board populated with component issues and milestones.
3. Solana development environment provisioned for all developers.
4. Initial database schema committed to repository.
5. API interface definitions published to repo (OpenAPI/README).
6. Program account structure drafted and reviewed.

PRIMARY WORKSTREAMS

- Set up GitHub repo, branching rules, and protected branches.
- Configure GitHub Actions workflows for build and test.
- Provision cloud dev environments and shared databases.
- Install and verify Solana CLI, local validator, and wallet integrations.
- Draft DB schema and migrate scripts.
- Define API contracts and document endpoints for frontend/backend integration.

Phase 2 — Core Functionality (Day 3, 4 and 5)

- Auth & Group Management: Wallet-based sign-in UI and group creation flows.
- Backend APIs: Authentication endpoints and group CRUD services with validation.
- On-chain Groups: Solana account mapping for group identities and membership.
- AI Intake: Initial telemetry capture for model training and privacy-safe logging.
- Transaction System: Frontend flows for deposits and withdrawal requests, including request lifecycle.

Key Deliverables:

- Wallet-signature login implemented and end-to-end verified.
- Group creation, join, and basic membership UX live.
- Transaction APIs and persistence with basic reconciliation in place.
- Treasury and approval logic scaffolded on-chain for further iterations.
- Monitoring hooks capturing transaction patterns for AI analysis.

KEY USER FLOWS

- Wallet sign-in and account linking.
- Create or join a group; manage roles.
- Add payment methods and view balances.
- Initiate deposit via Solana Pay QR and confirm on-chain.
- Submit withdrawal requests and track approvals.
- Receive transaction notifications and alerts.
- View transaction history and basic analytics.

Phase 3 — Integration & Features (Day 6 and 7)

1. Notification Center & Approvals (Frontend)
2. SMS Provider Integration (Backend)
3. Execute transactions when 80% approval threshold is met (Blockchain)
4. Anomaly detection for suspicious activity (AI)
5. Insights Dashboard (Frontend) with aggregated data endpoints (Backend)
6. Real-time alerts and approval UI components
7. SMS notifications dispatched for key events
8. Safe on-chain transaction execution flow
9. Basic anomaly flags and reporting

KEY DELIVERABLES

- Real-time alert system and approval workflow implemented
- SMS provider connected and message templates configured
- Secure transaction execution tied to 80% consensus
- Initial anomaly detection rules and alerting
- Spending insights cards and charts on the dashboard

Phase 4 — Pitch & Demo Prep

- Draft a concise pitch deck highlighting value proposition and demo flow.
- Record a short, clear video demo showing key features.
- Rehearse the live presentation with timing and role assignments.
- Prepare a production checklist for final assets and deployment.

Key Deliverables:

1. Finalized and approved pitch deck.
2. Demo video hosted and accessible via link.
3. Speaker notes with timings and cue points.
4. Distributed launch and production checklist.

PITCH & DEMO PREP OVERVIEW

- Outline the narrative: problem, solution, traction, and ask.
- Create slide visuals and one-page executive summary.
- Record demo clips: feature highlights, end-to-end flow, and fallback scenarios.
- Assemble a short edit with captions and call-to-action at the end.
- Run full rehearsals with timed transitions and backup plans.
- Prepare Q&A notes and technical fallback responses.
- Finalize distribution: upload video, share deck, and confirm presenter availability.