Canadian Music DNA – Hackathon Companion Summary

Team Member	Contact Information
Andrii Kavetskyi	andriykavetsky@gmail.com 672 472 0995
Taras Havrylyukh	taras.havrylyukh@gmail.com +1 236 979 1129

Project Overview

Discovery & Personas

Canadian Music DNA began as a challenge to translate a national survey into an experience personal and useful for any listener. Instead of building another dashboard of static charts, we imagined a studio where data guides an evolving creative journey. A short quiz distills survey insights into five archetypes; the interface reconfigures as an AI assistant narrates what responses reveal about each listening identity. Story-driven copy guides the user through every stage; animations reinforce that something alive is responding, and persona visuals draw directly from statistical clusters uncovered in the 1,006 response dataset.

Immersive Experience

Round four of the Vancouver AI Hackathon asked teams to blend diverse AI capabilities inside a unified UX. We positioned the platform as a collaborative music workshop: once a persona is identified, the resident AI companion proposes mood boards, playlist cues, and lyric fragments. A dedicated Lyrics Studio follows, where song briefs, ChatGPT-guided lyrics, and Suno-generated audio flow together in a three-step funnel. Transparency is core—the interface discloses which model is speaking, surfaces intermediate prompts, and archives every Suno job so experiments never disappear.

Community Insight

We surface aggregate trends—Canadian regions, discovery channels, emotional motivations—so each user knows how their habits compare with national peers. The interface encourages creative loops: remix prompts, re-run Suno generations with alternative moods, and export persona cards for sharing. The result respects underlying data while promoting imaginative play, demonstrating how music analytics and generative AI reinforce each other. The 2025 roadmap extends this spirit with collaborative jam sessions, open prompt libraries, and region-specific listening challenges that keep Canadians co-creating.

Technical Approach & Tools

Front-End Stack

Vite, React 18, and TypeScript deliver hot-module reloads with typed contracts safeguarding flows across the quiz, lyric editor, and Suno orchestration. TailwindCSS keeps persona badges, status chips, and

accessibility states consistent without bespoke CSS. Framer Motion stages the three-step Lyrics Studio, animating unlocks while honoring reduced-motion settings. Plotly.js renders persona analytics by binding Supabase datasets to responsive traces across devices. Storybook snapshots capture component variants and prevent regressions.

State & AI Services

A centralized Zustand store synchronizes quiz answers, generated lyrics, and Suno payloads so every surface sees one source of truth. OpenAI Chat Completions sit behind a typed wrapper that injects domain prompts, clamps temperature, enforces 30-second aborts, and normalizes errors for approachable messaging. Suno integration mirrors that discipline: we create jobs, poll with exponential backoff, hydrate tracks into downloadable object URLs, and persist transitions. Session logs feed the archive so users replay generations and restore flow.

Data & Persistence

Supabase hosts the `suno_requests` table with row-level security limiting writes to teammates while exposing aggregate reads to public charts. Database triggers timestamp updates, buckets cache persona exports, and SQL views project statistics into Plotly persona summaries. Supabase edge functions precompute trending moods so dashboard loads stay under two seconds, while local storage retains the active persona and Suno queue so visitors refresh without losing progress.

DevOps & Quality

GitHub Actions lint, type-check, and run unit tests before packaging assets for Vercel edge deployments. Playwright scripts simulate the Lyrics Studio funnel, verifying quiz gating with Suno hydration. Vitest covers prompt builders, state selectors, and Supabase mappers. A lightweight Sentry hook records latency, timeouts, and Suno codes so polling windows stay tuned. Manual QA checklists accompany pushes to verify accessibility cues and persona copy—keeping Canadian Music DNA resilient while enabling rapid iteration.
