



The Great Schema Cleanup

The shift to a One Companion Per User model collapsed our multi-agent infrastructure. 60-70% of the pure core carries over, but multi-agent routing dies.

v1 Schema

~10 Tables

Complex routing required three database joins just to answer: what did this user say?

v2 Schema

4 Core Tables (users, companions, messages, memories)

A flat stream of user-bound messages.

Insight: The entire multi-agent architecture collapsed into a single `user_id` foreign key constraint.



Severing the Fake Schedule

The emotion model was solid, but its triggers were artificial. We cut out the simulated routines to let emotions be driven purely by genuine conversation.



Scrapped: Fake-life triggers (e.g., feeling tired because it's 11 PM) and proactive preset files.



Kept: Core emotion model tracking valence, arousal, and discrete emotional states.



New Engine: Proactive messaging driven by memory gaps, actual time-awareness, and emotional continuity from the last chat.

The Real-Time Upgrade

A single WebSocket connection replaces fragmented endpoints, paving the way for real-time voice chat and native heartbeat monitoring.



SSE (v1 - The Old Way)

Server pushes to client. Strictly one-directional.

Required separate HTTP requests to send messages and application-level keepalives. Clunky polyfills needed for mobile.

WebSocket (v2 - The New Way)

Native bidirectional ping/pong.

Seamless proactive pushes on the same channel.

Native Expo support with built-in connection health frames.

Personality Through Conversation

Hundreds of lines of crafted JSON backstories are gone.
We deleted the preset persona files to make way for organic growth.

The Old Way: Presets

Reading static configuration files to decide who Xiaomeng the barista or Dashu the uncle is.



The New Way: Emergence

The companion is born through conversational onboarding. Identity is synthesized dynamically using user-customized parameters fed into the system-prompt builder.



A Native Destination

Telegram is retired. A chat history that doesn't sync across platforms is a broken experience. The UI paradigm shifts entirely to a single, focused native chat screen.

Backend (Untouched)

Hono (Server)

Supabase & Drizzle (DB)

pgvector (768-dim semantic search)

Frontend (Replaced)

Telegram & Next.js → Expo & React Native

UI Layer (New)

CSS Animations →

React Native Skia
(driving a breathing,
animated companion orb)