AI Development Log (Required)

Project Context

CollabBoard evolved into a collaborative whiteboard with an AI command assistant over the period covered by this submission. The AI layer was implemented as a board operation runtime that parses user commands, validates guardrails, executes deterministic board mutations, and emits tracing for each tool invocation.

Tools & Workflow

* **AI coding assistant:** ChatGPT/Codex (this environment) for planning, implementation, refactoring, and review.
* **Backend stack:** Next.js App Router, Firebase Auth/Firestore, and server routes under 'src/app/api/ai/*'.
* **AI runtime:** OpenAI Agents SDK integration ('@openai/agents') with deterministic fallback planner in place.
* **Tracing stack:** Langfuse (server tracing) and OpenAI traces (request/agent execution tracing).
* **Validation tooling:** 'playwright' (UI command probes), TypeScript tests ('vitest'), and manual command validation using the golden prompts.
* **Workflow used:** command-by-command incremental commits, each change validated by local checks/build, then merged into the AI path and UI integration before user-facing trials.

MCP Usage

* **Template MCP client:** 'src/features/ai/mcp/template-mcp-client.ts'.
* **Enabled MCP tools:** 'template.instantiate' and 'command.plan'.
* **What MCP added:**
- Canonical structured tool interface for template-specific planning.
- Fast route to generate deterministic template operations (for deterministic intents and fallback modes).
* **Current tradeoff:** MCP planning is retained for deterministic/template paths, while OpenAI Agents handles broad natural-language execution for most live commands. The MCP layer is still used where it improves structure and reliability.

Effective Prompts (Worked Well)

These prompts were directly used during iterative hardening and were reliable in the current implementation:

1. **'Add a yellow sticky note that says 'User Research''**
2. **'Create a blue rectangle at position 100,200'**
3. **'Add a frame called "Sprint Planning"'**
4. **'Create a 2x3 grid of sticky notes for pros and cons'**
5. **'Create a SWOT analysis template with four quadrants'**

Code Analysis: AI-Generated vs Hand-Written

* Estimated proportion of AI-assisted output: **~80% AI-assisted / ~20% manual** for this phase.
- AI assistance was strongest in:
- schema expansion and route wiring
- deterministic planner updates
- trace instrumentation and fallback routing
- Manual engineering focused on:
- edge-case fixes in UI/interaction behavior
- guardrail tuning
- final integration verification

Strengths & Limitations

Strengths

* Strong structured AI tool model with traceable execution path.