The Learning Systems Architect

Challenge: Acquire & Apply New Technology Skills

Summary:
Origin: Ask Pete (Capstone Project). This artifact is the Daydream Codebase (Rust/WASM). This artifact is the codebase itself: a full-stack web application built with Rust (Axum/Leptos) and compiled to WebAssembly. It demonstrates the acquisition of 'bleeding-edge' systems engineering skills to solve the 'Black Box' problem in EdTech, creating a local-first architecture that guarantees learner privacy by design.

Reflection:
I addressed the challenge to Acquire & Apply New Technology Skills by engaging in the rigorous acquisition and application of the Rust programming language and the Leptos (WebAssembly) framework. Unlike standard web technologies which often rely on server-side processing and opaque dependencies, Rust offers a strict, memory-safe environment that allows for a 'Local-First' architecture. I applied this skill to engineer a system where the entire application?including the narrative logic and the 'AI as a Mirror' components?compiles to WebAssembly, ensuring the code runs exclusively on the client's device.This technical acquisition was driven by a philosophy of 'Humanistic Engineering'?using the objectivity of code to scaffold the subjectivity of the learner. By mastering Rust's strict type system, I was able to solve the 'Black Box' problem inherent in many educational AI tools. Instead of relying on third-party APIs that introduce latency and potential data leakage, I built a transparent, predictable 'sandbox.' This creates a learning environment with objective 'physics,' where the rules are consistent and fair, allowing the learner to experiment without fear of surveillance.The application of these advanced engineering skills demonstrates a significant shift in the role of the Instructional Designer: from a consumer of tools to an Architect of Environments. By building the platform from the metal up, I ensured that the 'structural psychological development' of the learner remains a private process. This proves that high-level technical competency is not merely about building faster apps, but about having the agency to build ethical ones?systems that foster volition rather than compulsion by placing the full power of the computing environment directly into the hands of the learner.Learning Rust was arguably the hardest intellectual challenge of this project. It required me to unlearn habits from easier languages and think strictly about memory and safety. I see a direct parallel here to the discipline required in the Marine Corps?doing things the 'hard way' because it's the 'right way,' ensuring that the system (or the mission) doesn't fail when it matters most.

<div class=\