Project Issues, Escalations, Future Thinking, and Tech Stack Used

By: Ash Behrens

Objective: Automate the migration of a flat-file Smartsheet portfolio into a relational, bidirectional enterprise architecture on monday.com.

1. Technical Implementation & "Battle Scars"

- **The Bidirectional Handshake:** Engineered a two-pass migration logic to handle the API's sequential requirements—creating 27 deliverables first to collect their IDs before performing a "bulk link" update to the 6 parent engagements.
- **Relational Deduplication:** Identified "budget bloating" risks in the source flat-file; implemented a Python-based caching layer to ensure each project budget was assigned once at the portfolio level while maintaining many-to-one task relationships.
- **API Resilience:** Implemented staggered execution logic with `time.sleep` delays to respect platform rate limits and prevent 429 errors during high-volume mutations.
- **Sanitization Layer:** Developed a normalization function (`.strip().lower()`) to bridge the "handshake gap" between messy legacy CSV strings and strict GraphQL requirements.

2. Strategic Roadmap (Future Phase Growth)

- **Portfolio Intelligence:** Transition from list views to an **Executive Dashboard** utilizing the **Battery Widget** for portfolio-wide progress and **Chart Widgets** for Budget vs. Actuals.
- **Efficiency Automations:** Deploy "Parent-Child Status Syncs" (e.g., auto-completing an Engagement when all child tasks are 'Done') and automated "At Risk" alerts based on `due_date` proximity.
- **Resource Management:** Activate the **Workload View** to map `hours_estimated` against team capacity, identifying potential bottlenecks before they impact delivery.

3. Engineering Escalations & Feature Gaps

- **Validation Logic Debt:** Logged a bug in the automated "Orphaned Item" checker; successfully established links but identified a JSON parsing mismatch in the API's `board_relation` response. I am flagging this for **escalation to Engineering** to optimize the GraphQL fragments.
- **UX Visualization Gap:** Identified that native board views lack dynamic gradient heatmapping for numeric columns. Proposed a **Custom App build via the monday SDK** to provide bespoke financial visualizations for leadership.

4. Technical Stack & AI Workflow

- **AI Orchestration:** Leveraged **Gemini** for system architecture, **Claude** for high-fidelity GraphQL mutation refinement, and **Perplexity** for real-time 2024-01 API documentation research.
- **Software Engine: Python/Pandas** (Google Colab) for data transformation, **API Playground** for schema discovery, and the **Requests** library for secure bearer token authentication.