

Nithin,

I'm assigning you a focused 3-day build sprint for OneValue.

I want you to grow into leading OneValue. To do that, we need a working delivery operating system that gives us portfolio-wide truth without me pulling it together manually.

My support

- I'll be available all three days to unblock you. Escalate quickly when you hit blockers.

What you must ship (real product, not a PoC)

A gated Delivery Intelligence Console:

- Google OAuth + invite-only allowlist
- Supabase storage with RLS and auditability
- ingestion of PM meeting notes across all projects (starting today)
- ingestion of project comms from Slack/Teams where available (minimum: one working integration end-to-end; document the rest)
- join every project to its SOW/contract terms stored in Supabase (term dates, scope anchors, renewal window)

Dashboard outputs must include:

- Objectives, issues, blockers, owners, aging
- Delivery health metrics (portfolio + per project)
- Renewal readiness view post-June (signal-based, evidence-linked)
- Contract term visibility + scope alignment indicators
- Action queue: "what we do next," with owners and evidence links

Demo requirements

- Show the dashboard running on real inputs
- Show the SOW linkage working
- Show the access model (allowlist + roles)
- Show how it runs daily and how you handle failures

The bar is simple: an internal product we can rely on immediately.

**Platform.claude API Key :**

sk-ant-api03-xLRsAmquGXkB2kbuNL5cNMNIf0rX39e0hHF\_iMfcVapKAcjf9KtK-dmVdNRZSsjey  
TVFdIKmlgtlMRW8SxU63Q-t8gR2AAA

# Delivery Intelligence Console - PRD Document

**Product Requirements Document (PRD)** incorporates your specific instructions for SOW processing, error handling, and AI-driven ownership. This is a high-stakes, 48-hour build for the **OneValue Delivery Intelligence Console**.

---

## 1. Project Vision & Identity

The **Delivery Intelligence Console** is a unified "Delivery Operating System" designed to provide portfolio-wide truth<sup>11</sup>. It eliminates manual data pulling by automating the ingestion of project communication and contractual terms<sup>2222</sup>.

- **Target Bar:** A production-ready internal product, not a Proof of Concept (PoC)<sup>3333</sup>.
- **Design Standard:** "Tahoe OS26" (Apple-inspired) featuring liquid glass surfaces, semi-transparency, and rounded geometry.
- **Access:** Google OAuth restricted to the [oneorigin.us](https://oneorigin.us) domain via an invite-only allowlist<sup>4</sup>.

---

## 2. Infrastructure & Technical Stack

Component	Technology	Role
Frontend	Lovable	React-based UI with built-in Supabase integration.
Database	Supabase	Postgres with Row Level Security (RLS) and auditability <sup>55</sup> .
Automation	n8n	Orchestration of all workflows, historical dumps, and daily crons.

<b>Hosting</b>	<b>GitHub Projects</b>	Direct integration for deployment via GitHub Pages.
<b>AI Processing</b>	<b>Claude 3.5 Sonnet</b>	Logic via Platform.claude API key <sup>66</sup> .

### 3. Detailed Supabase Schema

The database ensures every delivery event is anchored to a contract (SOW).

#### Table: `sow_contracts`

- **Columns:** `id`, `project_name`, `start_date`, `end_date`, `scope_anchors` (text array), `renewal_window_start`, `contract_value`, `pdf_link`, `inferred_owner`.
- **Purpose:** Stores the "Truth Anchors" extracted from the standard format PDFs in Google Drive<sup>7</sup>.

#### Table: `delivery_intelligence`

- **Columns:** `id`, `project_id` (FK), `source` (Gmail/Chat), `event_type` (MOM/Comm), `content_raw`, `sentiment_score`, `extracted_blockers`, `aging_days`.
- **Purpose:** Central repository for project execution data<sup>8</sup>.

#### Table: `system_audit_logs`

- **Columns:** `id`, `workflow_name`, `execution_status`, `error_payload`, `timestamp`.
- **Purpose:** Ensures full auditability and tracks "Daily Run" success/failures<sup>999</sup>.

---

### 4. n8n Workflow Architecture

Workflows include a one-time historical dump and recurring daily intelligence gathering.

Workflow Name	Purpose
<code>historical_data_dump</code>	One-time pull of all Gmail (.edu domains) and Google Chat MOMs from Jan 1, 2024, to date.

<b>sow_pdf_auto_ingestor</b>	Watches Google Drive; parses standard-format PDFs to extract dates, scope anchors, and inferred owners.
<b>daily_delivery_poller</b>	Recurring cron to scrape MOMs from specific Google Chat Spaces and Direct Messages <sup>10</sup> .
<b>ai_project_analyzer</b>	Uses Claude to compare daily MOMs against SOW anchors to detect scope creep or health drops <sup>111111</sup> .
<b>onevalue_alert_manager</b>	<b>Central Error Handler:</b> Catches failures across all workflows and triggers an alert in the "OneValue Alerts" Space.

---

## 5. UI Requirements (Tahoe OS26 Standard)

The interface must be "**extremely intrusive**" to ensure critical data is never missed.

- **Portfolio Health Dashboard:** High-level metrics for renewal readiness post-June and overall portfolio health<sup>12</sup>.
- **Project Detail View:** Real-time linkage between execution data and SOW terms<sup>13</sup>.
- **Action Queue:** "What we do next" view with owners, evidence links, and aging indicators<sup>14141414</sup>.
- **The "Evidence Link":** Every AI-generated health signal must include a deep link back to the source Google Chat or Gmail message<sup>151515</sup>.

---

## 6. Advanced "Auto-Pilot" AI Features

- **The Inferred Owner Agent:** AI will analyze meeting notes and SOWs to automatically identify the OneOrigin owner and notify them via Google Chat when a blocker is detected.
- **Renewal Oracle:** A signal-based view that flags projects at risk of non-renewal based on sentiment in .edu domain emails<sup>16</sup>.

- **Automatic Scope Anchoring:** AI cross-references MOM outcomes against `scope_anchors` in the SOW; if a meeting discusses out-of-scope work, it triggers an immediate "Scope Creep" warning<sup>17</sup>.

---

## 7. Delivery & Demo Requirements

To meet the "3-day build sprint" bar<sup>18</sup>:

1. **Live Dashboard:** Must run on real inputs (not placeholder data)<sup>19</sup>.
2. **SOW Linkage:** Show the dynamic mapping between a Google Chat MOM and the original SOW terms<sup>20</sup>.
3. **Access Security:** Demonstrate the allowlist-only Google OAuth login<sup>21</sup>.
4. **Resilience Demo:** Show the "OneValue Alerts" workflow handling a simulated failure<sup>22</sup>.

**Next Step:** Would you like me to generate the **Supabase SQL setup script** for the tables and RLS policies defined above?