Product Requirements Document (PRD): Professional CMMS & Field Service Platform

Version: 1.0
Status: In Development
Date: 2026-02-09
Author: Senior Product Manager

---

1. Executive Summary

This Product Requirements Document outlines the vision and specifications for the **Professional CMMS (Computerized Maintenance Management System)**. The platform is designed to be the "Air Traffic Control" for modern field service operations, moving beyond simple ticketing to intelligent, real-time orchestration of assets, technicians, and work orders.
Core Value Proposition:

- **Real-Time Visibility:** Instant insight into field operations via map, timeline, and dashboard views.
- **Intelligent Automation:** Reducing manual dispatch overhead by 70% through rule-based auto-assignment and SLA monitoring.
- **User-Centric Design:** A premium, "consumer-grade" UX for complex industrial workflows.

---

2. Product Vision & Goals

Vision: To build the most responsive and intelligent operating system for field service teams, enabling proactive maintenance rather than reactive firefighting.
Primary Goals:

1. **Maximize Technician Utilization:** Reduce travel time and idle time through smart scheduling and routing.
2. **Enforce SLA Compliance:** Zero missed SLAs through proactive "at-risk" warnings and automated escalations.
3. **Complete Asset Lifecycle Visibility:** Track total cost of ownership (TCO) from procurement to disposal.
4. **Seamless Communication:** Bridge the gap between back-office and field via integrated Chat/WhatsApp.

---

3. Target User Personas

Persona	Role	Key Needs	"Aha!" Moment

Max (The Dispatcher)	Operations Manager	Real-time status of 50+ techs, drag-and-drop rescheduling, immediate bottleneck alerts.	Seeing the "Timeline View" update instantly as a tech completes a job.
Tanya (The Technician)	Field Service Tech	Mobile-first interface, offline access, clear priority instructions, minimal data entry.	Auto-assigned a job that is actually close to her current location.
Sarah (The Director)	Service Director	High-level KPIs, cost analysis, SLA compliance reports, "Air Traffic Control" oversight.	The "Live Operations Monitor" showing green across the board.

---

4. Detailed Feature Specifications

4.1. Mission Control Dashboard

Objective: Provide an at-a-glance health check of the entire operation.

- **KPI Cards:** Total Active WOs, Critical status counts, SLA Breach warnings.
- **Real-Time Activity Feed:** Live ticker of status changes, new tickets, and system alerts.
- **Visualizations:** Technician availability heatmaps, Work Order volume trends.

4.2. Intelligent Work Order Management

Objective: The core engine for creating, tracking, and completing service requests.

- **Views:**
  - **List View:** Sortable, filterable data grid with "zebra striping" for readability.
  - **Map View:** Geospatial view of open tickets vs. technician locations.
  - **Timeline (Gantt) View:**
    - Visual duration bars.
    - Color-coded status (🔵 New, 🟠 In Progress, 🟢 Completed, 🔴 Cancelled).
    - **Stagnation Indicators:** Warning icons for "Stuck" (In Progress > 5 days) or "Overdue" (> 7 days).
    - **Zoom Levels:** Day (Hourly), Week (Daily), Month.
- **Creation Flow:** Wizard-style form with customer selection, asset lookup, and priority setting (Critical, High, Medium, Low).

4.3. Automation Engine (The "Brain")

Objective: Remove manual decision-making for routine tasks.

- **Auto-Assignment Rules:**
    - **Scoring Algorithm:**
        - Distance (40%)
        - Workload (30%)
        - Specialization (25%)
        - Performance (5%)
    - **Triggers:** New Work Order -> "Ready" Status -> **Auto-Assign**.
- **SLA Monitoring:**
    - **Watchdog:** Runs every 15 mins.
    - **States:** On-track (<75%), At-Risk (75-100%), Breach (>100%).
    - **Escalation:** Auto-notify manager on "At-Risk"; Auto-reassign on "Breach".

4.4. Asset & Inventory Management

Objective: Full traceability of physical items.

- **Assets:**
    - QR Code generation/scanning.
    - Parent/Child hierarchy (e.g., HVAC Unit -> Compressor).
    - Downtime tracking & MTBF (Mean Time Between Failures) calculation.
- **Inventory:**
    - Multi-warehouse support ("Locations").
    - Low-stock alerts.
    - Part consumption tracking per Work Order.

4.5. "Air Traffic Control" Live Monitor

Objective: A dedicated screen for large operational centers.

- **Design:** High-contrast, dark mode optimized for wall displays.
- **Features:**
    - Masonry grid of active "Hot" items.
    - "Ghost" logic to fade out inactive cards.
    - Pulsing indicators for urgent/critical setups.

4.6. Integrated Communications

Objective: Keep everyone in sync without leaving the app.

- **Chat:** Internal team chat linked to specific Work Orders.
- **WhatsApp Integration:**
    - Automated customer updates ("Tech is en route").
    - Two-way manual messaging for customer inquiries.

---

5. Technical Requirements

5.1. Tech Stack

- **Frontend:** React (Vite), TypeScript, Tailwind CSS, Shadcn UI.

- **State Management:** React Query (Server state), Context API (App state).
- **Backend / DbaaS:** Supabase (PostgreSQL).
- **Edge Functions:** TypeScript (Deno) for Automation Logic.
- **Realtime:** Supabase Realtime (Subscriptions).

5.2. Performance Constraints

- **Dashboard Load:** < 1.5s for initial paint.
- **Map Rendering:** Support 500+ markers without lag.
- **Assignment Algorithm:** < 100ms execution time per ticket.

5.3. Security

- **Authentication:** Supabase Auth (JWT).
- **Authorization:** Row-level Security (RLS) enforcement at the database level.
- **Audit Logs:** Immutable `automation_logs` for all system actions.