**Claude Breakdown: Chunk 7 — System Monitoring, Device Status, and Environment Triggers**

**✅ Scope**

Define how ClubOS V2 integrates with systems like NinjaOne and Ubiquiti to monitor real-time device health, booking-related activity, and passive anomalies. Covers proactive flags, AI alerts, and human-aware triggers (e.g. no-shows, location offline).

**🔁 Cross-Referenced Corrections**

| **Early Plan** | **Final Decision** |
| --- | --- |
| No proactive AI monitoring | ✅ AI monitors device + location state but does not act autonomously |
| NinjaOne mentioned but undefined | ✅ NinjaOne used for PC uptime + service state tracking |
| No logic for booking dropouts | ✅ Booking no-shows flagged to operator, not resolved by AI |

**🔧 Dependencies**

* NinjaOne API access (PC online status, uptime, reset options)
* Ubiquiti access logs (door opened, customer connected)
* Booking data sync (from Skedda or Jason’s custom system)

**📋 Tasks**

**[7.1] Device Status Polling / Webhook Listening**

Each location’s core systems are tracked:

* TrackMan PC (via NinjaOne)
* Projector (optional future)
* Network connectivity (via Ubiquiti or ping logic)
* Booking activity (via internal DB or Jason’s system)

Track:

{

"location": "Dartmouth",

"bay": "4",

"pc\_online": false,

"last\_ping": "2025-08-01T14:03:00Z",

"booking\_active": true,

"auto\_flag": true,

"flag\_type": "device\_offline\_during\_booking"

}

**[7.2] Booking No-Show Detection**

If booking ends, and:

* No access log (door\_unlocked = false)
* No AI interaction or system usage

→ Thread is tagged as "booking\_no\_show"

Flag is surfaced:

* On operator dashboard
* Optional Slack notification
* Option to log refund, message customer, or mark complete

**[7.3] Location-Wide Offline Detection**

If all PCs in a location go offline within a short time window:

* AI flags it as potential\_power\_loss
* Escalation Slack alert sent to owner/operator role
* AI does **not take action** (no reset/reboot logic initiated)

Operator sees:

* Last heartbeat per bay
* Active bookings impacted
* Common failure reason suggestion (if known from past logs)

**[7.4] Passive Monitoring Triggers**

System watches for soft anomalies like:

* Multiple failed TrackMan resets within short window
* Recurring “no booking found” messages at a bay
* High escalation rate from a single customer

Claude proposes:

* Pattern: "trackman\_reset\_failure\_cluster\_03"
* Action: “Suggest reviewing SOP tech-010”
* Operator can review flagged cluster from logs

**[7.5] Alert Grouping + Noise Reduction**

* Alerts from same location or bay within X minutes are grouped
* Claude deduplicates recurring errors
* Optional thresholds per alert type:

{

"device\_offline\_timeout": 3,

"group\_similar\_alerts": true

}

**✅ Expected Behavior**

* AI never acts on system data alone — it flags and suggests only
* All device state changes are tracked and linked to active threads (if any)
* Operators get high-signal alerts only — grouped, annotated, and tied to booking context
* Claude builds passive pattern awareness to evolve logic over time

**Chunk 8: Frontend Layout, Mobile UX, and Desktop Controls**