**Claude Breakdown: Chunk 11 — Ticket System for Facilities and Tech**

**✅ Scope**

Define a lightweight but structured ticketing system for ClubOS V2. Supports ticket creation by AI or humans, tracks priority, type, resolution, and ties back to message threads or system events. Integrates into all interfaces.

**🔁 Cross-Referenced Corrections**

| **Early Plan** | **Final Decision** |
| --- | --- |
| Ticket system undefined | ✅ Use V1 model (Facilities + Tech categories, Low → Critical priority) |
| AI did not create tickets | ✅ AI can auto-create tickets from messages or system flags |
| No ticket access in thread view | ✅ Message page includes one-tap ticket creation and view per thread |

**🔧 Dependencies**

* Ticket table in DB (ticket\_id, status, source, etc.)
* Links to threads, users, and optionally locations
* Simple ticket modal component (mobile + desktop)

**📋 Tasks**

**[11.1] Ticket Schema**

{

"ticket\_id": "T-30981",

"created\_by": "OperatorGPT",

"submitted\_by": "customer\_1290",

"source": "message\_thread",

"linked\_thread": "msg\_9182",

"category": "facilities",

"priority": "medium",

"status": "open",

"title": "No balls in Bay 2",

"description": "Customer reported missing balls during 6PM session",

"location": "Bedford",

"timestamp": "2025-08-01T17:28:00Z",

"assigned\_to": null

}

**[11.2] AI Ticket Creation Logic**

AI auto-creates tickets when:

* Customer reports an **operational issue** outside the scope of reset/unlock (e.g. bathroom dirty, projector dim, balls missing)
* A system anomaly occurs with no SOP resolution path (e.g. persistent projector offline)
* No immediate fix possible or escalation is flagged

Ticket is:

* Categorized
* Prioritized based on pattern (e.g. critical = location-wide offline)
* Logged + attached to thread

Operator sees ticket preview in message view

**[11.3] Human Ticket Creation Flow**

From any customer thread:

* Tap “Create Ticket”
* Pre-filled fields: location, customer, category suggestion
* Operator can:
  + Type short description
  + Set priority (Low, Medium, High, Critical)
  + Submit instantly

Ticket linked to current thread + timestamped

Optional: create standalone ticket from Tickets tab

**[11.4] Ticket Visibility**

Tickets are visible in:

* **Main message thread view** (inline with conversation or header badge)
* **Dedicated Tickets tab**
  + Filter by: location, priority, category, assigned/unassigned
* **Dashboard / Desktop UI**
  + Kanban or list view of all open/resolved tickets

**[11.5] Status Management**

Statuses:

* open, in\_progress, on\_hold, resolved, archived

Operators can:

* Assign to staff
* Add resolution notes
* Mark complete

AI doesn’t auto-close tickets, but can suggest closure if:

* Operator resolved issue
* Customer confirms
* Related system is restored (e.g. projector back online)

**✅ Expected Behavior**

* Both humans and AI can open tickets quickly, with context pre-filled
* Each customer message thread shows relevant tickets
* Operators never lose track of facility or tech issues reported through messaging
* Priority and category ensure routing is clean and scalable
* Claude does not learn from tickets unless flagged as relevant