

# Communication Hub

**Doc type:** technical · **Version:** v0.1 · **Status:** published · **Module slug:** communication-hub  
**Exported:** 2026-05-15 11:11 UTC · **By:** anonymous

## Communication Hub – Technical Specification

### 1. Module Purpose & Scope (Authoritative)

---

Communication Hub is the governed communication layer within Primoro. It provides all internal staff communication, patient-facing messaging, and broadcast/acknowledgement primitives. It solves the problem of ad-hoc, ungoverned communication (e.g. informal WhatsApp use) by replacing it with auditable, role-owned, work-generating communication surfaces.

It governs:

- All conversation threads (internal staff, patient-facing, and lead-context)
- All broadcast and acknowledgement primitives (Announcements, Praise, Scheduled Events)
- Structured Internal Requests as a governed, task-backed capability
- Message-to-work conversion surfaces (callbacks, tasks, approvals, hand-offs)

It explicitly does not:

- Act as a CRM or perform lead acquisition or marketing (owned by a dedicated CRM/Lead module)
- Store documents outside of Document Hub (owned by Document Hub)
- Execute or lifecycle-track tasks and appointments (owned by Task Manager and Appointment Manager)
- Make autonomous AI decisions (AI intent detection and classification owned by AI Assistant – Aiden)

### 2. Ownership & Responsibilities

---

#### 2.1 Communication Hub IS Responsible For

- Providing and governing all communication primitives: Conversation Thread, Chat, Channel, Announcement, Praise, Acknowledgement, Structured Internal Request
- Owning delivery and audit of all system-initiated broadcasts (Scheduled Events)
- Surfacing AI-detected suggestions as a presentation and hand-off surface (not classification or learning)
- Exposing the message-to-work conversion surface so authorised users may create callbacks, tasks, approvals, or Appointment Manager hand-offs from any thread
- Full audit coverage for all communication events
- Serving as the authoritative store of all conversation history across all context types (Internal, Patient, Lead); downstream modules (e.g. Treatment Pipeline Manager) hold references and summaries only
- Executing all campaign step deliveries on behalf of Campaign Manager, including enforcing Campaign Manager's declared frequency caps and contact fatigue constraints at the point of delivery (see §6.1 and §6.6)

- Receiving and surfacing inbound patient replies to Recall & Reconnect messages, and triggering the appropriate sub-state transitions (Engaged, Needs Personal Follow-up) in collaboration with Recall & Reconnect
- Routing communications to the correct responsible party (patient, guardian, or other delegated actor) for patient records governed by Family Profiles, preserving patient attribution throughout (see §2.3)
- Acting as the sole channel through which Appointment Manager, Integrated Payments, Hygiene Subscriptions, Care Plan Subscriptions, Digital Forms, and all other consuming modules deliver messages to staff and patients; no consuming module may emit messages directly to recipients
- Receiving secure Document Hub references for inclusion in messages; raw document files **MUST NOT** be transmitted as message attachments
- Receiving payment notifications, receipts, and payment-exception events from Integrated Payments and routing them to the appropriate recipient (see §6.1)

## 2.2 Communication Hub IS NOT Responsible For

- Task execution and lifecycle tracking (owned by Task Manager)
- Appointment scheduling and lifecycle (owned by Appointment Manager)
- Document storage (owned by Document Hub)
- AI intent detection, classification, and learning logic (owned by AI Assistant – Aiden)
- Lead acquisition, CRM, or marketing activity (owned by the relevant CRM/Lead module)
- Access role definitions (owned by Access Manager)
- Form notification issuance or reminder scheduling — Digital Forms owns trigger logic; Communication Hub executes the delivery only
- Aftercare conversation ownership — Aftercare Manager owns aftercare-specific escalation threads; Communication Hub provides the underlying Conversation Thread artefact and messaging transport only
- Payment exception surfacing or failed-payment task creation logic — these are owned by Integrated Payments; Communication Hub receives and routes the resulting notification or task-creation event only
- Meeting transcript storage or note ownership — owned by AI Meeting Notes; Communication Hub provides the optional delivery surface (Channel or Thread) for sharing derived artefacts

## 2.3 Family Profiles — Delegated Communication Rules

Where a patient record is governed by Family Profiles, the following rules apply to all communication routing:

- Communications **MUST** be addressed to the responsible party designated in the Family Profiles record (guardian, parent, or authorised delegate) rather than directly to the dependant patient, unless the dependant holds independent consent status
- The underlying Conversation Thread **MUST** retain the dependant patient's `ContextId` to preserve patient attribution in audit records
- Escalations and hand-offs originating from a family-profile-governed thread **MUST** carry both the responsible-party identifier and the dependant patient identifier in the event payload
- Communication Hub **MUST NOT** suppress or redirect communications in a way that severs the audit link between the responsible party and the dependant patient record
- Family Profiles determines the responsible-party routing rule; Communication Hub enforces it at delivery

## 3. Core Objects (Normative)

---

### 3.1 Conversation Thread (Canonical Artefact)

A Conversation Thread is a governed digital artefact representing the canonical container for all messages within Communication Hub. Communication Hub is the authoritative store of all conversation history across every ContextType. Downstream modules that surface conversation data (e.g. Treatment Pipeline Manager for lead-context threads) MUST store references and read-only summaries only; they MUST NOT replicate or own the full conversation record.

Minimum required fields:

- ThreadId
- ContextType: `Internal` | `Patient` | `Lead` | `LabCase`
- ContextId (PatientId, StaffContextId, LeadId, or LabCaseId)
- OwnerRole (single authoritative owner at any time)
- Status: `Open` | `In-Progress` | `Waiting` | `Resolved` | `Archived`
- SLAState
- CreatedBy (user/role)
- CreatedAt
- AuditTrail (immutable)

The ContextType value `LabCase` is added to support Lab Manager's requirement to maintain an immutable communication timeline for each lab case (portal messages, email interactions, scanner events). Lab Case threads follow all base Conversation Thread rules; the ContextId carries the Lab Manager case identifier. Lab Manager MUST store a reference to the ThreadId only and MUST NOT replicate the thread content.

Treatment Pipeline Manager's Lead and Opportunity records each MUST reference exactly one ThreadId. When a Lead converts to an Opportunity, the associated Conversation Thread MUST be carried forward without interruption; Communication Hub guarantees that the thread identifier and ownership remain stable across this transition.

#### 3.1.1 CallThread (Sub-type)

Where a Conversation Thread is created or attached by AI Concierge in the context of a telephone call, it MUST carry additional call-specific fields in addition to the base Conversation Thread schema. These fields form a CallThread sub-type:

- CallMetadata: { `StartTime`, `EndTime`, `DurationSeconds`, `Mode` (`Inbound` | `Outbound` | `Transferred`) }
- Transcript (immutable; append-only once committed)
- DetectedIntents[ ] (populated by AI Assistant – Aiden; read-only within this module)
- FinalOutcome: { `OutcomeType`, `Summary` }
- LinkedTasks[ ] (references to Task Manager task identifiers)
- FormsSent[ ] (references to forms dispatched during or after the call)
- PaymentLinksSent[ ] (references to payment links dispatched during or after the call)
- TakeoverEvents[ ] ( { `EventType`: `TakeoverRequested` | `TakeoverCompleted` | `HandoverReturned`, `Actor`, `Timestamp` } )

Rules specific to CallThread:

- The Transcript field MUST be immutable once the call ends; no editing is permitted. This aligns with AI Concierge's canonical Call Thread definition, which likewise treats transcripts as immutable and append-only once committed. Communication Hub is the authoritative store of the CallThread record; AI Concierge holds a reference only.
- DetectedIntents[ ] MUST be populated exclusively by AI Assistant (Aiden) via the inbound integration contract; Communication Hub surfaces them but does not classify them
- TakeoverEvents MUST be logged in the AuditTrail with actor and timestamp
- The AuditTrail for a CallThread MUST be append-only; no event may be removed or modified after it is written
- All other base Conversation Thread rules (archival only, role ownership, state machine) apply

### 3.2 Conversation Thread State Machine (Authoritative)

States:

- Open
- In-Progress
- Waiting
- Resolved
- Archived

Rules:

- Every message belongs to exactly one thread
- Threads are never deleted; they are archived only
- Ownership is role-first; there are no personal inboxes
- State transitions are auditable and time-stamped

---

### 3.3 Announcement (Canonical Artefact)

An Announcement is a governed broadcast message.

Minimum required fields:

- AnnouncementId
- Title
- Body
- AudienceScope (Role | Site | Group)
- Priority: Normal | Important | Critical
- AcknowledgementRequired (bool)
- ScheduledAt (optional)
- ExpiryAt (optional)
- CreatedBy (user/role)
- CreatedAt

- AuditTrail (immutable)

### 3.4 Announcement State Machine (Authoritative)

States:

- Draft
- Scheduled
- Published
- Expired
- Archived

Rules:

- Announcements are not chat messages
- Announcements do not allow replies (reactions are optional where enabled)
- All announcements are auditable
- Acknowledgements (where required) are tracked per user
- Overdue acknowledgements MAY surface as tasks

---

### 3.5 Praise (Canonical Artefact)

Praise is a specialised Announcement type with stricter constraints.

Minimum required fields:

- Praiseld
- FromRole / FromUser
- ToUser / ToRole
- Reason (free text + optional category)
- Visibility: Team | Site | Organisation
- CreatedAt
- AuditTrail (immutable)

Rules:

- Praise is positive-only
- No anonymous praise permitted
- No replies (reactions optional)
- Included in audit log

---

### 3.6 Acknowledgement (Canonical Artefact)

Acknowledgements are explicit decision events, not chat replies.

Minimum required fields:

- AcknowledgementId

- `LinkedObject` (Announcement / Document / Task)
- `RequiredBy` (timestamp)
- `CompletedBy`
- `CompletedAt`
- `AuditTrail` (immutable)

Rules:

- Failure to acknowledge MAY generate a task
- All acknowledgements are auditable

## 4. Communication Primitives

---

### 4.1 Chat (Direct / Scoped Thread)

A Chat is a Conversation Thread scoped to a specific operational context.

Examples:

- Patient conversation
- Follow-up discussion on a task
- Internal 1-to-1 staff conversation

The module MUST:

- Limit the participant set to those relevant to the operational context
- Assign a single owner role at all times
- Enable message-to-work conversion from any chat thread (see §4.5)

The module MAY:

- Prompt users to convert free-form messages into Structured Internal Requests

The module MUST NOT:

- Support personal inboxes unbound from a role or operational context
- Permit chat deletion (archival only)

Chats replace informal direct messages and ad-hoc WhatsApp conversations.

### 4.2 Channel (Role-Scoped Thread) (Authoritative)

A Channel is a Conversation Thread scoped to a role or operational function.

Examples:

- Front-of-House channel
- Treatment Coordinator channel
- Management channel

Required:

- Participants = all members of the assigned role

- Ownership remains with the role, not individuals
- Messages MAY generate tasks for claiming

Optional (configurable):

- Which roles have channels enabled

Channels are governed role queues, not social chat rooms.

### **4.3 Structured Internal Requests (Optional CORE Capability)**

Communication Hub MAY expose admin-defined structured request types to staff as guided submission flows.

Examples:

- IT issue
- Supplies request
- Diary change request
- Training completion / attestation

The module MUST:

- Create a linked Conversation Thread for each submitted request
- Create a governed task in Task Manager for each submitted request

The module MAY:

- Allow free-form messages to prompt users to convert intent into a structured request

The module MUST NOT:

- Operate as a standalone ticketing system; execution and lifecycle tracking transfer to Task Manager

Request types, routing, and SLAs are configurable via admin controls.

### **4.4 Scheduled Events (System-Initiated Broadcasts)**

The system MAY generate Announcements automatically for:

- Birthdays
- Work anniversaries
- Practice events
- Rota-driven milestones

The module MUST:

- Mark all system-initiated events as system-initiated
- Own delivery and audit of all such events

The module MAY:

- Allow scheduled events to be disabled per site or per role

Source data for scheduled events may originate from other modules (e.g. Rota Manager for rota-driven milestones), but delivery is always owned by Communication Hub.

## 4.5 Message → Work Conversion (Non-Negotiable)

From any chat or channel thread, authorised users MAY:

- Create a callback
- Create a follow-up task
- Trigger an approval
- Hand off to Appointment Manager

When creating a task via message-to-work conversion, the event payload emitted to Task Manager MUST include at minimum:

- `ThreadId` — the originating Conversation Thread
- `ContextType` and `ContextId` — the subject context (e.g. `PatientId`, `LeadId`)
- `TaskType` — the type of work being created (e.g. `Callback`, `FollowUp`, `Approval`)
- `InitiatedBy` — role and user triggering the conversion
- `Timestamp` — time of the conversion event
- `MessageReference` — a reference to the specific message(s) from which the task was derived

Task Manager is the authoritative owner of the resulting task lifecycle; Communication Hub retains a `LinkedTasks[ ]` reference in the thread only.

When handing off to Appointment Manager, the hand-off event MUST carry sufficient booking context to allow Appointment Manager to initiate a governed booking flow. At minimum, the hand-off payload MUST include:

- `ThreadId` — the originating Conversation Thread
- `PatientId` (or `LeadId`) — the subject of the intended appointment
- `RequestedPractitioner` (optional) — preferred practitioner if stated in thread
- `RequestedAppointmentType` (optional) — appointment type inferred from thread context
- `RequestedTimeHint` (optional) — any date/time preference expressed in thread
- `InitiatedBy` — role and user triggering the hand-off
- `Timestamp` — time of hand-off event

Appointment Manager will use these fields to initiate a Find Next or direct booking flow in accordance with its own booking window, rota entry, and practitioner eligibility constraints. Communication Hub retains no booking execution lifecycle after the hand-off event is emitted.

The module MUST NOT:

- Own execution or lifecycle tracking of the resulting work; these transfer to Task Manager or Appointment Manager respectively

## 4.6 Staff App Mode — Consumption Contract

Staff App Mode surfaces Communication Hub primitives (Chats, Channels, Announcements, Praise, Acknowledgements, Structured Requests) on the tablet app delivery surface. The following rules govern this consumption:

- Staff App Mode MUST consume Communication Hub artefacts via the same API contracts as the Web Portal; it MUST NOT maintain its own copy of thread state or message history

- Staff App Mode MAY render a subset of fields appropriate to the mobile form factor, but MUST NOT suppress or transform fields in a way that changes the semantic content of a message or its state
- State transitions (e.g. marking a thread In-Progress, submitting an Acknowledgement) initiated in Staff App Mode MUST be applied to the canonical Communication Hub record immediately and reflected across all surfaces
- All interactions on Staff App Mode MUST be subject to the same RBAC rules enforced by Access Manager as on the Web Portal
- The feature matrix defined in §5.2 is authoritative for which primitives are available in Staff App Mode

## 5. Delivery Surfaces & Access (Authoritative)

### 5.1 Web Portal

Feature	Web Portal
Chats	■
Channels	■
Announcements	■
Praise	■
Acknowledgements	■
Structured Requests	■

### 5.2 Tablet App (Staff App Mode)

Feature	Staff App Mode
Chats	■
Channels	■
Announcements	■
Praise	■
Acknowledgements	■
Structured Requests	■

### 5.3 Patient Mobile App

Feature	Patient App
---------	-------------

Chats	■ (patient-facing only)
Channels	■
Announcements	■
Praise	■
Acknowledgements	■ (where relevant)
Structured Requests	■

## 5.4 Engagement Signals

*(no content captured in original — needs definition)*

## 5.5 Shared Device & Role-Scoped Delivery

Communication Hub MUST support notification delivery to shared tablet surfaces (e.g. Surgery and Decon wall tablets) used by Surgery & Decon Day View and similar role-scoped surfaces. The following rules apply:

- Notifications targeting a shared device MUST be scoped to the role currently active on that device, not to a named individual
- Where multiple roles share a device surface, Communication Hub MUST route to the role declared as the active session by Access Manager
- Urgent notifications (e.g. overdue task alerts surfaced by Surgery & Decon Day View) MUST override quiet hours regardless of device type
- Communication Hub is solely responsible for notifications transport to shared devices; Surgery & Decon Day View and equivalent modules MUST NOT emit notifications directly to recipients

## 5.6 Security & Privacy Controls on Delivery Surfaces

Communication Hub MUST enforce the notification privacy controls declared by the Security & Privacy module across all delivery surfaces, including:

- Screenshot and screen-recording blocking MUST be applied to any screen surface rendering sensitive patient-bound message content, in accordance with Security & Privacy policy
- Clipboard access MUST be restricted for message content classified as sensitive on all platforms where this control is technically available
- Message preview text shown in push notifications MUST be suppressed or redacted on lock-screen and notification-shade surfaces for patient-bound messages, unless the user has explicitly opted in and the practice-level policy permits it
- These controls apply equally to the Web Portal, Staff App Mode (Tablet), and Patient Mobile App delivery surfaces

# 6. Integration Contracts

## 6.1 Inbound (this module consumes from)

From module	What	Contract
AI Assistant (Aiden)	Intent classification suggestions (AllIntent payload — see §6.4) for surface presentation	async
Task Manager	Task context for thread linkage	async
Appointment Manager	Appointment context for hand-off	async
Rota Manager	Rota-driven milestone events (birthdays, anniversaries, schedule changes) for Scheduled Events (§4.4); shift and working-hours data for notification delivery gating (see §6.5) — staff notifications only	async
Product Shop	Order confirmation, payment receipt, shipment notification, refund, and delivery issue events requiring staff or patient notification; fulfilment exception events requiring staff task creation	async / event
Campaign Manager	Campaign step delivery requests with frequency cap and contact fatigue constraint declarations; all campaign steps MUST be submitted to Communication Hub for execution — no campaign step may deliver directly to recipients (see §6.6)	async / event
Recall & Reconnect	Inbound patient reply events originating from recall and reconnect journeys	async / event
Referral Manager	Patient milestone communication trigger events at key referral state-change points	async / event
AI Guardian	Alert and summary routing events directed to the appropriate role; finding-linked notifications	async / event

Smart Treatment Proposals	Proposal delivery requests (recipient, channel preference, proposal reference) for dispatch via Patient Mobile App, fallback email, and optional SMS; patient question and call-back request events originating from proposal interactions	async / event
AI Meeting Notes	Meeting summary and actions payloads for optional posting into a Conversation Thread or Channel as a governed delivery surface	async / event
Integrated Payments	Payment receipt delivery events, payment notification events, and payment exception events (failed payment, refund, approval request) requiring patient or staff notification or task creation	async / event
Hygiene Subscriptions	Subscription lifecycle events (enrolment confirmation, payment failure, pause, cancellation) requiring patient and/or staff notification	async / event
Care Plan Subscriptions	Payment exception notification events (failed payment, lapsed subscription) for routing to the appropriate staff role and patient; task creation for exceptions is owned by Integrated Payments via Communication Hub	async / event
Digital Forms	Form dispatch and reminder trigger events; Communication Hub executes delivery only — Digital Forms owns the issuance and reminder scheduling logic	async / event
Loyalty Insights	Action hand-off events for patient outreach (e.g. loyalty milestone reached, lapsed-patient reactivation prompt)	async / event

Rewards Manager	Outcome events (earn, redemption, referral state transition, charity donation) for patient notification delivery	async / event
AI Quality Monitor	Alert and structured remediation notification events directed to appropriate staff roles	async / event
Knowledge, Training & Learning	Learning announcement events, new course availability notifications, and completion reminder triggers for delivery as Announcements or staff messages	async / event
Family Profiles	Responsible-party routing rules for patient communications (see §2.3)	sync / config
External Provider API Gateway & Rate Governance	Rate-limit and throttle signals for outbound messaging provider calls (see §6.7)	sync / event

## 6.2 Outbound (this module emits to)

To module	What	Contract
Task Manager	Governed task creation from Structured Requests, message-to-work conversion, overdue acknowledgements, and payment or fulfilment exception events received from consuming modules	event
Appointment Manager	Hand-off trigger from message-to-work conversion (see §4.5 for required payload fields)	event
Audit & Compliance	Immutable event log for all communication events	event
Document Hub	Delivery notifications to staff and patients following document dispatch (e.g. in-app document delivery notifications)	event

Recall & Reconnect	Inbound reply routing events with sub-state transition signals (Engaged, Needs Personal Follow-up) following receipt of a patient reply to a recall or reconnect message	event
--------------------	--	-------

### 6.3 PMS Boundary

*(no content captured in original — needs definition)*

### 6.4 Aiden Intent Suggestion Payload Contract

When AI Assistant (Aiden) surfaces an intent classification suggestion to Communication Hub for presentation, the inbound event payload MUST carry at minimum:

- `IntentType` — the classified intent category (as defined by Aiden's AllIntent schema)
- `ConfidenceScore` — numeric confidence value (0.0–1.0)
- `RequiredEntities[ ]` — entities Aiden has identified as necessary to fulfil the intent
- `SupportedActions[ ]` — list of actions Communication Hub MAY surface to the user (e.g. "Convert to Structured Request", "Create Task")
- `ThreadId` — the Conversation Thread to which the suggestion applies
- `Timestamp` — time at which Aiden generated the suggestion

Where Aiden raises an escalation event, the payload MUST additionally carry:

- `EscalationReason` — textual description of why escalation is triggered
- `TargetRole` — the role to which the escalation is directed
- `EscalationTimestamp`

Communication Hub MUST log every suggestion event received from Aiden — including `IntentType`, `ConfidenceScore`, `ThreadId`, and whether the suggestion was accepted or rejected by a human — in the Audit & Compliance log (see §8).

Communication Hub MUST NOT act on any Aiden suggestion autonomously; all actions require explicit human approval.

The AllIntent suggestion contract also governs how Aiden feeds Communication Hub's message-to-work conversion flows. Where Aiden's `SupportedActions[ ]` payload includes a message-to-work action (e.g. "Create Task", "Convert to Structured Request"), Communication Hub MUST surface the action for human confirmation before emitting any event to Task Manager or Appointment Manager. No message-to-work conversion event may be emitted solely on the basis of an Aiden suggestion without explicit human approval.

### 6.5 Rota-Driven Notification Gating

Communication Hub consumes shift and working-hours data from Rota Manager to gate or modify notification delivery urgency for staff-facing messages. Rota Manager is the authoritative source for shift start/end times, quiet hours, and availability patterns; Communication Hub MUST treat this feed as read-only and MUST NOT maintain its own copy of rota state. The following rules apply:

- During a staff member's active shift, all notifications are delivered normally

- Outside shift hours, non-urgent notifications MAY be suppressed or queued until the next shift begins, subject to practice-level configuration
- Urgent or direct messages (as classified by the originating primitive or the sending module) MUST override quiet hours and be delivered immediately regardless of shift state
- The determination of whether a message qualifies as urgent is made at the point of delivery routing, based on the `Priority` field of the Announcement or the urgency flag of the originating event
- Where Rota Manager marks a staff member as unavailable (e.g. off-duty, on leave, or absent), work-related notifications (task assignments, message deliveries) for that staff member MUST be suppressed or queued in the same manner as out-of-shift notifications, subject to urgency override rules
- All suppression and override decisions MUST be recorded in the audit log

## 6.6 Campaign Frequency Cap & Contact Fatigue Enforcement

Campaign Manager declares frequency cap and contact fatigue constraints as part of each campaign step delivery request. Communication Hub MUST enforce these constraints at the point of delivery routing, prior to emitting any message to a recipient. The following rules apply:

- Communication Hub MUST evaluate the declared frequency cap for each recipient against the recipient's message history before dispatching a campaign step
- If a recipient has reached or exceeded the declared frequency cap, Communication Hub MUST suppress the campaign step delivery and record the suppression event in the audit log
- Contact fatigue constraints declared by Campaign Manager MUST be applied globally across all campaign steps for the same recipient, not per campaign
- Suppressed delivery events MUST be reported back to Campaign Manager so that campaign outcome tracking is accurate
- Communication Hub MUST NOT apply frequency cap or contact fatigue logic invented outside a Campaign Manager declaration; enforcement is always constraint-driven from the originating module

## 6.7 External Provider API Gateway & Rate Governance

All outbound messaging provider calls (e.g. SMS gateway, email relay, push notification service) made by Communication Hub MUST be routed through the External Provider API Gateway & Rate Governance module. The following rules apply:

- Communication Hub MUST declare a traffic class for each outbound messaging call: `Interactive` (user-initiated, latency-sensitive) or `Background` (system-initiated broadcast, batch)
- On receipt of a `429 Too Many Requests` or equivalent throttle signal from the gateway, Communication Hub MUST queue the affected messages and retry with exponential back-off, subject to the priority of the originating primitive
- Urgent messages (`Priority: Critical` or urgency-flagged events) MUST be placed in the highest-priority traffic class and MUST NOT be indefinitely queued; if delivery cannot be completed within the SLA window, an alert MUST be raised to the appropriate staff role
- All throttle, suppression, and retry events MUST be recorded in the audit log

## 7. AI Boundaries (Non-Negotiable)

---

Module does not embed AI surfaces directly; AI classification and learning are owned by AI Assistant (Aiden). Communication Hub's role is strictly presentational and governed hand-off.

AI MAY:

- Surface detected intent suggestions to staff (e.g. "This looks like a supply request") as a presentation layer
- Explain options to patients/staff
- Summarise thread activity for human review
- Suggest content from approved libraries (with explicit human approval before commit)

AI MAY NOT:

- Perform intent detection, classification, or learning logic within Communication Hub — this is owned by AI Assistant (Aiden)
- Take autonomous action on any communication event
- Bypass governance, audit, or access checks
- Make commitments on behalf of the practice
- Replace required clinical judgement

## 8. Audit & Compliance

---

The system MUST log:

- All state transitions on canonical objects (Conversation Thread, Announcement, Praise, Acknowledgement), with actor and timestamp
- All read/write actions on patient-bound message data
- All AI suggestions surfaced, including which were accepted or rejected by humans (including `IntentType`, `ConfidenceScore`, and `ThreadId` per §6.4)
- All cross-module events emitted or consumed
- All acknowledgement completions and overdue acknowledgements
- All system-initiated broadcast events
- All `CallThread` takeover and handover events (actor and timestamp)
- All notification suppression and urgency-override decisions made under rota-driven gating (§6.5)
- All message-to-work conversion events, including the target module (Task Manager or Appointment Manager) and the triggering actor
- All campaign step delivery suppressions made under frequency cap or contact fatigue enforcement (§6.6), including the suppressed recipient, the originating campaign step reference, and the constraint that triggered suppression
- All outbound messaging throttle, queue, and retry events processed via the External Provider API Gateway (§6.7)
- All inbound patient replies from Recall & Reconnect journeys and the sub-state transition triggered as a result
- All payment and subscription notification events received from Integrated Payments, Hygiene Subscriptions, and Care Plan Subscriptions, and the delivery or task-creation outcome for each

The final bullet above captures the message-to-work conversion audit requirement, which is implicit throughout §4.5 and §13.2 but not enumerated as an explicit audit event in the original §8.

Audit logs MUST be immutable and exportable for inspection.

## 9. Access Control

Access control within Communication Hub is enforced by Access Manager using role-based access control (RBAC). Communication Hub MUST delegate all role and permission definitions to Access Manager and MUST NOT maintain its own role registry.

### 9.1 Permission Matrix (Indicative)

The following table describes the permission model for each communication primitive. Exact role-to-permission mappings are owned by Access Manager; the entries below define the verbs that must be governed.

Primitive	Create	Read	Update (ownership/status)	Approve / Promote	Archive
Conversation Thread (Internal)	Authorised staff roles	Participants + OwnerRole	OwnerRole	N/A	OwnerRole or Admin
Conversation Thread (Patient)	Authorised clinical/admin roles	Participants + OwnerRole	OwnerRole	N/A	OwnerRole or Admin
Conversation Thread (Lead)	Authorised sales/admin roles	Participants + OwnerRole	OwnerRole	N/A	OwnerRole or Admin
Conversation Thread (LabCase)	Authorised clinical/lab roles	Participants + OwnerRole	OwnerRole	N/A	OwnerRole or Admin
Announcement	Authorised admin/management roles	AudienceScope members	Creator / Admin	Admin / Management role	Admin
Praise	Any staff role	AudienceScope members	No update permitted	N/A	Admin
Acknowledgement	System / linked primitive	Requester + recipient	N/A	N/A	System
Structured Internal Request	Any authorised staff role	Submitter + OwnerRole	OwnerRole	N/A	OwnerRole or Admin

Deletion is not a permitted verb for any primitive; archival is the only terminal state.

## 9.2 Session Revocation & Access Removal

Access Manager enforces immediate access removal across all modules, including Communication Hub. The following rules apply:

- When a user's access is revoked (e.g. leaver processing, locum expiry, forced logout following token revocation), Communication Hub **MUST** honour the revocation immediately across all active sessions and devices
- Any in-flight thread where the revoked user is the OwnerRole **MUST** have ownership automatically transferred to an appropriate fallback role (as configured in Access Manager) or flagged for manual reassignment by an admin
- Channel membership for the revoked user **MUST** be removed immediately upon revocation
- Locum and time-bounded access is governed by Access Manager; Communication Hub **MUST** enforce expiry of thread visibility and channel membership in line with Access Manager's time-bound role definitions
- MFA requirements for sensitive operations (e.g. publishing a Critical Announcement) are subject to Access Manager policy; Communication Hub **MUST** enforce any MFA gate declared by Access Manager before permitting the operation

## 9.3 Open Permission Questions

See §15 for outstanding questions relating to which specific roles are permitted to create Announcements vs. Praise vs. Structured Requests, and whether MFA is required for any specific communication operation.

## 10. Integration Summary

---

- **Task Manager** — receives governed task creation events from Structured Requests, message-to-work conversion, overdue acknowledgements, and payment or fulfilment exception events
- **Appointment Manager** — receives hand-off triggers from message-to-work conversion (with booking-context payload per §4.5); Appointment Manager does **NOT** emit messages directly to recipients — Communication Hub is the sole message delivery owner
- **AI Assistant (Aiden)** — owns intent detection and classification; Communication Hub consumes surface-level suggestions only (AllIntent payload shape defined in §6.4); the Aiden contract also governs message-to-work intent surfacing flows
- **Document Hub** — document storage is delegated; Communication Hub receives secure Document Hub references for inclusion in messages (raw files are never transmitted); Communication Hub emits delivery notifications to Document Hub for staff and patient document dispatch
- **Access Manager** — RBAC for read/write/approve across all communication primitives; enforces immediate session revocation and time-bounded access
- **Audit & Compliance** — immutable event log for all communication events
- **Product Shop** — Communication Hub receives and processes order lifecycle and fulfilment exception notification events for staff and patient delivery
- **Campaign Manager** — all campaign step delivery is executed via Communication Hub with frequency cap and contact fatigue constraints enforced at execution; no direct delivery by Campaign Manager is permitted
- **Recall & Reconnect** — all inbound patient replies from recall/reconnect journeys flow into Communication Hub; Communication Hub emits sub-state transition signals back to Recall & Reconnect
- **Referral Manager** — patient milestone communications at key referral state-change points are dispatched via Communication Hub; Referral Manager emits referral-state-change events inbound and Communication

Hub is the sole delivery owner

- **AI Guardian** — alerts and summaries are routed to the correct role via Communication Hub
- **Smart Treatment Proposals** — all outbound proposal delivery (Patient App, email fallback, SMS) and proposal-originated patient question / call-back request events flow through Communication Hub
- **AI Meeting Notes** — meeting summaries and actions MAY be posted into Communication Hub as a governed delivery surface; message delivery primitives and conversation storage are owned by Communication Hub
- **Rota Manager** — supplies rota-driven milestone data (§4.4 Scheduled Events) and shift/working-hours data for notification delivery gating (§6.5); Rota Manager is the authoritative source for shift times and availability — staff notifications only
- **Integrated Payments** — payment receipt, payment notification, and payment exception events are delivered via Communication Hub; failed-payment task creation is owned by Integrated Payments and executed via Communication Hub
- **Hygiene Subscriptions** — subscription lifecycle events (enrolment, payment failure, pause, cancellation) are delivered to patients and staff via Communication Hub; Hygiene Subscriptions does NOT send communications directly
- **Care Plan Subscriptions** — payment exception notifications for routing to staff and patient; task creation for exceptions remains with Integrated Payments via Communication Hub
- **Digital Forms** — form dispatch and reminder delivery is executed by Communication Hub on trigger from Digital Forms; reminder scheduling logic is owned by Digital Forms
- **Loyalty Insights** — action hand-off events for patient outreach are executed via Communication Hub
- **Rewards Manager** — reward outcome events (earn, redemption, referral, charity donation) are delivered to patients via Communication Hub
- **AI Quality Monitor** — alert and structured remediation notification events are routed to appropriate staff roles via Communication Hub
- **Knowledge, Training & Learning** — learning announcements and completion reminder notifications are delivered via Communication Hub
- **Family Profiles** — responsible-party routing rules are supplied to Communication Hub for patient communications involving dependants (§2.3)
- **External Provider API Gateway & Rate Governance** — all outbound messaging provider calls pass through rate governance; Communication Hub declares traffic class and handles throttle signals per §6.7
- **Surgery & Decon Day View** — notifications transport for shared device surfaces is owned by Communication Hub; Day View does not emit notifications directly

## 11. Explicit Non-Goals

---

- **CRM / lead acquisition / marketing** — Communication Hub is not a CRM; any such capability would be owned by a dedicated CRM or Lead module
- **Document storage** — owned by Document Hub; Communication Hub handles references only, never raw files
- **Task and appointment execution** — owned by Task Manager and Appointment Manager respectively
- **AI intent classification and learning** — owned by AI Assistant (Aiden)
- **Standalone ticketing system** — Structured Internal Requests are lightweight and task-backed; a full ticketing system would be a separate module

- **Aftercare conversation ownership** — Aftercare Manager owns aftercare-specific escalation threads; Communication Hub provides transport only
- **Form notification issuance and reminder scheduling** — owned by Digital Forms; Communication Hub executes delivery only
- **Payment exception task creation logic** — owned by Integrated Payments; Communication Hub routes the resulting notification or task-creation event
- **Meeting transcript storage or note ownership** — owned by AI Meeting Notes; Communication Hub is an optional delivery surface only

## 12. Versioning & Governance

---

This specification is owned by: *(role — needs definition)*

Changes to this spec require:

- Review by MVP module owner
- Impact analysis across declared related modules (see /propose)
- Version bump (patch / minor / major)

## 13. Build Contract (Engineering & QA)

---

### 13.1 Canonical Data Model

*(no content captured in original — needs definition)*

### 13.2 Core Behaviour Rules

1. Every message **MUST** belong to exactly one Conversation Thread.
2. Threads **MUST** never be deleted; archival is the only permitted terminal state.
3. Ownership of every thread **MUST** be assigned to a role, not an individual; there are no personal inboxes.
4. Announcements **MUST NOT** allow replies (reactions are configurable).
5. Praise **MUST** be positive-only and **MUST NOT** be anonymous.
6. Every Structured Internal Request **MUST** produce both a linked Conversation Thread and a governed task in Task Manager.
7. Message-to-work conversion (callback, task, approval, hand-off) **MUST** transfer execution ownership to Task Manager or Appointment Manager; Communication Hub retains no execution lifecycle.
8. All AI-surfaced suggestions **MUST** be logged, including whether they were accepted or rejected by a human.
9. Failure to complete a required Acknowledgement by its deadline **MAY** generate a task in Task Manager.
10. All system-initiated broadcast events **MUST** be marked as system-initiated in the audit log.
11. Every UX interaction **MUST** map to a defined communication primitive; no new message type may be introduced without a corresponding primitive.
12. All campaign step deliveries **MUST** be routed through Communication Hub; no consumer module may send directly to recipients.

13. All inbound patient replies from Recall & Reconnect journeys MUST be surfaced as Conversation Threads owned by the appropriate role.
14. When a Conversation Thread is created as a CallThread sub-type, the Transcript field MUST be treated as immutable once the call ends; no post-call editing is permitted.
15. Hand-off events emitted to Appointment Manager MUST include the full booking-context payload defined in §4.5.
16. Notification delivery for staff MUST be gated against Rota Manager shift data; non-urgent notifications MAY be suppressed outside shift hours; urgent messages MUST override quiet hours regardless of shift state.
17. Upon user access revocation, thread ownership for affected users MUST be transferred or flagged immediately; channel membership MUST be removed immediately.
18. Communication Hub is the authoritative store of all conversation history; downstream modules MUST NOT replicate or own full conversation records — references and summaries only are permitted.
19. Campaign step deliveries MUST be evaluated against Campaign Manager's declared frequency cap and contact fatigue constraints before dispatch; deliveries suppressed by these constraints MUST be logged and reported back to Campaign Manager.
20. All outbound messaging provider calls MUST pass through the External Provider API Gateway; throttle signals MUST trigger queuing and retry with exponential back-off; urgent messages MUST NOT be indefinitely queued.
21. Communications involving Family Profiles–governed patient records MUST be addressed to the designated responsible party whilst retaining the dependant patient's ContextId for audit attribution.
22. Message-to-work conversion events emitted to Task Manager MUST include the full task-creation payload defined in §4.5, including ThreadId, ContextType, ContextId, TaskType, InitiatedBy, Timestamp, and MessageReference.
23. All notification privacy controls declared by the Security & Privacy module (screenshot blocking, screen-recording blocking, clipboard restriction, notification preview suppression) MUST be enforced on all delivery surfaces for patient-bound message content.
24. CallThread AuditTrail records MUST be append-only; no audit event may be removed or modified after it is written.
25. Lab Case conversation threads (ContextType: LabCase) MUST be created and owned by Communication Hub; Lab Manager MUST store a ThreadId reference only.
26. Payments, receipts, and payment exception notifications received from Integrated Payments, Hygiene Subscriptions, and Care Plan Subscriptions MUST be routed to the appropriate recipient and MUST generate a task in Task Manager where human action is required; silent failures are not permitted.

### 13.3 Configuration Surfaces

- **Practice-level settings (Admin Control Plane):** Structured Request types, routing, and SLAs; scheduled event types (enable/disable per site or role); channel role assignments; Announcement acknowledgement requirements; notification suppression windows aligned to Rota Manager shift data; campaign frequency cap enforcement windows; notification privacy control policy overrides (within Security & Privacy limits)
- **Per-user preferences (Access Manager):** Notification preferences (where permitted by role policy)
- **Per-object overrides (Communication Hub):** Per-announcement priority and expiry; per-thread SLA state

### 13.4 Filtering & Views

*(no content captured in original — needs definition)*

## 13.5 Module Extension Map

*(no content captured in original — needs definition)*

## 13.6 Acceptance Criteria

The build of Communication Hub is complete when:

- [ ] All canonical objects (Conversation Thread, Announcement, Praise, Acknowledgement) can be created, read, and updated through the API
- [ ] CallThread sub-type fields are enforced, including Transcript immutability, append-only AuditTrail, and TakeoverEvent logging
- [ ] LabCase ContextType is supported as a valid Conversation Thread context; Lab Manager ThreadId reference contract is enforced
- [ ] State machine transitions enforce all rules in §3
- [ ] All integrations in §6 are wired, including all inbound sources (Product Shop, Campaign Manager, Recall & Reconnect, Referral Manager, AI Guardian, Smart Treatment Proposals, AI Meeting Notes, Rota Manager, Integrated Payments, Hygiene Subscriptions, Care Plan Subscriptions, Digital Forms, Loyalty Insights, Rewards Manager, AI Quality Monitor, Knowledge Training & Learning, Family Profiles)
- [ ] AI boundaries in §7 are enforced (negative tests pass)
- [ ] Aiden intent suggestion payload (§6.4) is correctly consumed, surfaced, and logged; no autonomous message-to-work conversion is permitted
- [ ] Audit log captures every event in §8
- [ ] Access control is enforced per §9, including immediate revocation and thread ownership transfer
- [ ] Message-to-work conversion correctly transfers execution to Task Manager / Appointment Manager, with full task-creation payload for Task Manager and full booking-context payload for Appointment Manager hand-offs
- [ ] Structured Internal Requests produce a linked thread and a governed task in Task Manager
- [ ] Delivery surface feature matrix (§5) is correctly enforced per platform
- [ ] Rota-driven notification gating (§6.5) correctly suppresses and overrides notifications per shift state, respecting Rota Manager availability patterns
- [ ] Campaign frequency cap and contact fatigue enforcement (§6.6) correctly suppresses, logs, and reports delivery events
- [ ] External Provider API Gateway integration (§6.7) correctly classifies traffic, handles 429 responses with queuing and back-off, and raises alerts for urgent messages exceeding SLA
- [ ] Family Profiles delegation rules (§2.3) are correctly applied to patient communications involving dependants
- [ ] Security & Privacy notification controls (§5.6) are enforced on all delivery surfaces
- [ ] Shared device / role-scoped delivery (§5.5) correctly routes notifications to the active role on a shared tablet surface
- [ ] Document Hub delivery notification events are emitted on document dispatch; raw file transmission is never performed
- [ ] Recall & Reconnect inbound reply events produce the correct sub-state transition signal

- [ ] Payment, receipt, and payment-exception notification events from Integrated Payments, Hygiene Subscriptions, and Care Plan Subscriptions are correctly routed and task-creation events are emitted where required
- [ ] All non-functional requirements in §14 are met

## 14. Non-Functional Requirements

---

- **Performance:** *(target latency and throughput — needs definition)*
- **Reliability:** *(availability target and graceful degradation behaviour — needs definition)*

The two items below scaffold canonical NFR categories present in the template; all values are left as stubs because the original is silent.

- **Scalability:** *(multi-site, multi-tenant constraints — needs definition)*
- **Security:** *(encryption at rest/in transit, key management, secrets handling — needs definition)*
- **Privacy:** *(GDPR rights this module honours, data retention policy for conversation history, CallThread transcripts, and patient-bound message data — needs definition)*
- **Observability:** *(metrics, logs, and traces this module exports — needs definition)*
- **Accessibility:** *(accessibility standards applicable to Web Portal, Tablet App, and Patient Mobile App delivery surfaces — needs definition)*

Accessibility added as a canonical NFR stub; the original is silent but it is a standard requirement for all patient-facing and staff-facing surfaces.

## 15. Open Questions

---

Outstanding decisions before this spec can be promoted from `draft` to `published`.

1. What roles are permitted to create Announcements vs. Praise vs. Structured Requests? (Awaiting Access Manager role definitions.)
2. Which module owns the source-of-truth for rota-driven milestone data used in Scheduled Events?
3. Is MFA required for any communication operation (e.g. publishing a Critical Announcement)?
4. What is the SLA policy for patient-facing conversation threads, and who configures it?
5. Are reactions on Announcements and Praise enabled at MVP or deferred?
6. What is the fallback ownership role when a thread owner's access is revoked and no explicit fallback is configured in Access Manager?
7. Should Campaign Manager delivery requests be surfaced as a distinct primitive or wrapped in an existing Announcement / Chat primitive?
8. What urgency classification rules apply to Product Shop fulfilment events — are delivery issue notifications treated as urgent?
9. Who owns this specification? (Role not yet captured — see §12.)
10. What are the target latency, throughput, and availability figures for Communication Hub? (Required to complete §14.)
11. What is the data retention policy for conversation history, CallThread transcripts, and patient-bound message data? (Required for GDPR compliance posture in §14.)

12. What is the exact event schema (field names, types, and versioning) for Hygiene Subscriptions lifecycle events? (Awaiting Hygiene Subscriptions finalised technical spec.)
13. What is the authoritative taxonomy of traffic classes for External Provider API Gateway declarations — is the `Interactive / Background` binary sufficient, or are additional tiers required?
14. How should Family Profiles delegation rules interact with patient consent status for minors who reach independent consent age mid-thread? What is the transition protocol for existing open threads?
15. Are Loyalty Insights action hand-off payloads delivered as Announcements, Chats, or a new primitive? (Awaiting Loyalty Insights technical spec finalisation.)
16. What notification privacy control overrides, if any, are permitted at the practice level for staff-facing surfaces (e.g. in-clinic shared screens where screenshot blocking may conflict with clinical workflow)?
17. Should AI Quality Monitor remediation alerts generate Structured Internal Requests (producing a governed task) or be delivered as direct Channel messages? (Awaiting AI Quality Monitor integration design.)