

# Referral Manager

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

## Referral Manager – Technical Specification

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

---

Referral Manager provides a governed, end-to-end system for managing all patient referrals — inbound and outbound — with clear ownership, progress tracking, automated patient communication, and full auditability. It exists to eliminate referrals being lost in inboxes, scans, or shared folders; ensure every referral has a named owner and next action; keep patients informed without manual chasing; and protect governance, compliance, and patient experience.

It governs:

- Creating and tracking Referral Records through governed lifecycles (both Outbound and Inbound directions).
- Maintaining a Referrer / Specialist Directory as a centralised, governed reference.
- Centralising all referral communications via the Communication Hub and a shared referrals mailbox.
- Creating and managing referral-linked tasks, SLA tracking, at-risk flags, and escalation tasks.
- Storing referral documents and outcome reports as structured records.

It explicitly does not:

- Deliver clinical treatment or manage external provider diaries.
- Act as a generic document archive or perform analytics and KPI visualisation (owned by Performance Dashboards).
- Issue, track, or redeem Refer-a-Friend rewards — patient Refer-a-Friend flows, including reward eligibility determination, reward issuance, and redemption state, are owned entirely by Rewards Manager.
- Make autonomous clinical or administrative decisions.

### 2. Ownership & Responsibilities

---

#### 2.1 Referral Manager IS Responsible For

- End-to-end lifecycle ownership of every Referral Record from creation to completion or cancellation.
- Enforcing named ownership on every active referral and preventing silent stalling via mandatory SLA rules.
- Maintaining the Referrer / Specialist Directory as a governed reference asset.
- Centralising all inbound and outbound referral communications in a single auditable record.
- Generating referral-linked tasks and escalation signals via Task Manager.
- Emitting referral-state-change events to Communication Hub for automated patient notifications.
- Emitting read-only metrics to Performance Dashboards.

- Providing a fully auditable trail of all referral interactions, state transitions, document uploads, and communications.

## 2.2 Referral Manager IS NOT Responsible For

- Clinical treatment delivery — owned by treating clinical modules outside this suite.
- External provider diary management — owned by external systems beyond the platform boundary.
- Generic document archiving — owned by the practice's document management system (PMS boundary).
- Analytics and KPI visualisation — owned by Performance Dashboards.
- Refer-a-Friend reward logic of any kind — owned entirely by Rewards Manager.
- Task lifecycle management beyond creation and escalation signalling — owned by Task Manager.
- Notification dispatch and channel fallback logic — owned by Communication Hub.
- Role and permission management — owned by Access Manager.

## 3. Core Objects (Normative)

---

### 3.1 Referral Record (Canonical Artefact)

A Referral Record is a governed digital artefact representing a single patient referral, in either an Outbound or Inbound direction.

Minimum required fields:

- ReferralID
- PatientID (FK to patient record)
- Direction (Outbound | Inbound)
- ReferralState (current lifecycle state — see §3.2 and §3.3)
- ReferringClinician or ExternalReferrer (contextually required per direction)
- ReceivingProvider / Service
- ReferralReason (structured)
- OwnerID (named staff member — mandatory, never null on an active referral)
- CreatedBy (user / role)
- CreatedAt (timestamp)
- AuditTrail (immutable append-only log of all transitions and actions)

Referral Records MUST NOT exist solely as emails, documents, or inbox items. Every referral interaction MUST be captured within a governed Referral Record.

### 3.2 Outbound Referral State Machine (Authoritative)

States:

- Draft
- Sent
- Awaiting Acceptance
- Info Requested

- Accepted
- Scheduled (optional sub-state of Accepted)
- Completed (Report Received)
- Rejected
- Cancelled

#### Rules:

- Mandatory states (Draft → Sent → Awaiting Acceptance) may not be bypassed.
- Every state transition is timestamped, attributed to an actor (user or governed system event), and written to the immutable AuditTrail.
- A referral in Completed or Cancelled state may not be returned to any earlier state.
- Only users with an authorised role (see §9) may trigger state transitions; system-initiated transitions are attributed to the governed system actor and logged accordingly.

### 3.3 Inbound Referral State Machine (Authoritative)

#### States:

- Received
- Awaiting Triage
- Accepted / Assigned
- Scheduled
- Completed (Report Sent)
- Rejected
- Cancelled

#### Rules:

- Mandatory states (Received → Awaiting Triage) may not be bypassed.
- Every state transition is timestamped, attributed, and written to the immutable AuditTrail.
- A referral in Completed, Rejected, or Cancelled state may not be returned to any earlier state.
- Inbound referrals created via email ingestion enter the Received state automatically; a named owner must be assigned before the referral may transition to Accepted / Assigned.

### 3.4 SLA / Escalation Lifecycle

- SLA Timer Starts — on transition to Sent (Outbound) or Received (Inbound).
- Reminder Triggered — system-generated, per configured threshold.
- At-Risk — SLA threshold exceeded; referral flagged visually.
- Overdue — escalation threshold exceeded.
- Escalation Task Created — Task Manager task generated and assigned.

SLA thresholds are practice-configurable (see §13.3). Referrals must not stall silently; an SLA always applies from the moment a referral enters an active state.

### 3.5 Referrer Directory Entry (Canonical Artefact)

A Referrer Directory Entry defines an external provider or clinic as a governed reference record.

Minimum required fields:

- ReferrerID
- Provider / Clinic Name
- Specialty or Service
- PreferredReferralMethod
- GovernedContactEndpoints
- ActiveStatus (`Active` | `Archived`) — to allow retirement of entries without deletion.

**Boundary note — Lab Manager's Lab Directory:** Lab Manager maintains its own Lab Directory as a separate canonical artefact governing external laboratory providers and their routing configuration (including email-based interaction for labs without portal access). The Referrer Directory owned by Referral Manager governs specialist clinicians, external clinics, and other non-laboratory receiving providers. Where an external laboratory is also the receiving provider for a patient referral, the Referral Record's `ReceivingProvider / Service` field MAY reference that laboratory, but the authoritative profile and contact configuration for that laboratory remains owned by Lab Manager's Lab Directory. Referral Manager MUST NOT duplicate or supersede Lab Manager's laboratory provider records; any cross-module lookup at the provider boundary is the responsibility of the integration layer and is subject to the same PMS/integration boundary rules described in §6.3. This boundary must be confirmed with the Lab Manager module owner before build if shared provider lookups are required.

## 4. Referral Workflows

---

### 4.1 Practice-Initiated (Outbound) Workflow

The module MUST:

- Allow authorised staff to create a new Referral Record manually via the Staff Web Portal.
- Require all mandatory fields (patient, direction, referral reason, named owner) before a record exits `Draft`.
- Support review and approval of AI-drafted referral content before the referral is sent (see §7).
- Support ownership reassignment with a full audit entry on each change.

The module MAY:

- Present AI-generated draft content for clinician review and approval.
- Surface missed-referral detection signals from AI Quality Monitor for staff action.

The module MUST NOT:

- Send, accept, or complete any referral autonomously without explicit human approval.

### 4.2 Inbound Referral Ingestion Workflow (Authoritative)

The module MUST:

- Ingest inbound referrals via secure email to the shared referrals mailbox, creating a `Received Referral` Record idempotently (duplicate ingestion must not create duplicate records).
- Expose a structured triage interface for assigning an owner and triaging the referral.
- Support accept, reject, and info-request responses to the external referrer via governed channels.

- Capture all inbound communications as part of the Referral Record's communication history.

The module MUST NOT:

- Allow an inbound referral to be actioned by a user without an authorised role.
- Expose patient data to the external referrer beyond the scope of their assigned referral.

### 4.3 SLA Tracking and Escalation (Authoritative)

The module MUST:

- Start an SLA timer on every referral at the point it enters an active state.
- Trigger reminder signals at practice-configured thresholds.
- Flag referrals as `At-Risk` and then `Overdue` at configured escalation points.
- Emit an escalation task to Task Manager when a referral becomes `Overdue`.

The module MUST NOT:

- Allow any active referral to persist without an SLA timer running.
- Manage task lifecycle beyond creation and escalation signalling; Task Manager owns that lifecycle.

## 5. Delivery Surfaces & Access (Authoritative)

---

### 5.1 Staff Web Portal

The primary surface for all referral management activity. Must provide:

- A Referral Dashboard with inbound / outbound separation, named owner, SLA / at-risk indicators, and filters by provider, patient, and status.
- A Referral Detail View showing full referral context, structured documents and reports, complete communication history, linked tasks, and related appointments.
- Referrer Directory management.
- All dashboard views and filters listed in §13.4 must be implemented on this surface.

### 5.2 Tablet App

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

### 5.3 Patient Mobile App

Patients receive automated milestone notifications dispatched by Communication Hub. The patient-facing surface for referral status visibility, if any, is governed by Communication Hub and the patient app shell; Referral Manager does not render a direct patient UI surface.

### 5.4 Engagement Signals

Referral Manager emits the following read-only metrics to Performance Dashboards:

- Inbound referral volume
- Outbound referral volume
- Average acceptance time

- Overdue referral count
- Referral completion rate
- Referral source distribution

Referral Manager does not analyse or visualise these metrics itself.

## 6. Integration Contracts

### 6.1 Inbound (this module consumes from)

From module	What	Contract
Communication Hub	Inbound referral email ingestion from shared referrals mailbox	Async / event
Task Manager	Task completion / update signals for referral-linked tasks	Async / event
Access Manager	Role and permission assertions for RBAC enforcement	Sync
AI Quality Monitor	Missed-referral detection signals; AI-drafted referral content	Async / event

### 6.2 Outbound (this module emits to)

To module	What	Contract
Communication Hub	Patient notification triggers at referral milestones	Event
Task Manager	Referral-linked task creation and escalation task creation	Event
Performance Dashboards	Read-only referral metrics (see §5.4)	Async / metric stream
Audit & Compliance	Immutable audit events for all state transitions and interactions	Event

### 6.3 PMS Boundary

The PMS may hold patient demographic data and appointment records that inform referral context. Referral Manager consumes patient identity and appointment linkage from the PMS but does not write clinical records back to the PMS. The exact integration contract at the PMS boundary — including whether data is pulled on demand or synchronised — is subject to the PMS integration layer and is outside Referral Manager's owned scope.

## 7. AI Boundaries (Non-Negotiable)

---

AI MAY:

- Draft referral content for clinician review and approval before any referral is sent.
- Surface missed-referral detection signals to staff for human action.
- Summarise referral activity and status for staff review dashboards.
- Suggest content from approved clinical libraries, with explicit human approval required before commit.

AI MAY NOT:

- Send, accept, reject, or complete any referral autonomously.
- Assign or reassign referral ownership without human confirmation.
- Bypass governance, audit, or access control checks.
- Make commitments on behalf of the practice or a clinician.
- Replace required clinical judgement at any point in the referral lifecycle.

## 8. Audit & Compliance

---

The system MUST log:

- All Referral Record state transitions, with actor identity and timestamp.
- All ownership assignments and reassignments, with actor identity and timestamp.
- All inbound and outbound referral communications captured against the Referral Record.
- All document uploads and outcome report attachments, with actor and timestamp.
- All task creation and escalation events linked to a referral.
- All AI-drafted content suggestions, recording whether each was accepted, modified, or rejected by a human actor.
- All cross-module events emitted or consumed (Communication Hub triggers, Task Manager events, Performance Dashboard metric emissions).
- All external referrer interactions (accept / info-request / reject responses) received via the shared mailbox or portal.

Audit logs MUST be immutable and exportable for inspection. No referral interaction is permitted outside governed channels.

## 9. Access Control

---

Access is governed by Access Manager using tenant-scoped RBAC. Illustrative role permissions:

Action	Clinician	Treatment Coordinator	Reception / FOH	Practice Manager	External Referrer
Create outbound referral	✓	✓	—	✓	—

View own referrals	✓	✓	✓	✓	assigned only
Manage workflows / follow-ups	—	✓	✓	✓	—
Reassign ownership	—	✓	—	✓	—
View SLA breaches / escalations	—	✓	—	✓	—
Update assigned referrals	—	—	—	—	✓
Access Referrer Directory (write)	—	✓	—	✓	—

All access is tenant-scoped. External referrer access is scoped strictly to their assigned referrals with no cross-referral visibility.

MFA requirements for sensitive operations — such as accessing patient-bound referral data or approving AI-drafted content — are governed by Access Manager policy and must be enforced at the platform authentication layer.

## 10. Integration Summary

- **Communication Hub** — outbound patient notification events at referral milestones; inbound email ingestion for the shared referrals mailbox.
- **Task Manager** — outbound task creation and escalation task events; inbound task completion signals.
- **Performance Dashboards** — outbound read-only referral metrics stream.
- **Access Manager** — RBAC enforcement for all read/write/approve operations; tenant scoping.
- **Audit & Compliance** — immutable event log for all referral interactions.
- **AI Quality Monitor** — inbound missed-referral detection signals and AI-drafted referral content for staff review.
- **Rewards Manager** — explicitly out of scope; Referral Manager emits no signals to and consumes no signals from Rewards Manager.

## 11. Explicit Non-Goals

- **Analytics and KPI visualisation** — Referral Manager emits metrics but does not own dashboards or visualisations. If added, would be owned by Performance Dashboards.

- **External provider diary management** — booking or managing external provider scheduling is out of scope and would require a separate integration surface.
- **Generic document archiving** — Referral Manager stores referral-specific structured documents only; general document management is a PMS-layer concern.
- **Refer-a-Friend reward logic** — any patient incentive or reward flow is owned entirely by Rewards Manager and must not be implemented here.
- **Autonomous clinical or administrative decision-making** — the module surfaces information and prompts for human action; it does not make decisions.
- **Laboratory provider directory management** — the authoritative directory of external laboratory providers is owned by Lab Manager. Referral Manager does not duplicate or supersede this data; see §3.5 for the boundary note governing cross-module provider references.

## 12. Versioning & Governance

---

This specification is owned by: *(operations suite module owner — needs assignment)*

Changes to this spec require:

- Review by the Post-MVP module owner.
- Impact analysis across all declared related modules (Communication Hub, Task Manager, Access Manager, Audit & Compliance, Performance Dashboards, AI Quality Monitor, Rewards Manager).
- Version bump (patch / minor / major) per change scope.

## 13. Build Contract (Engineering & QA)

---

### 13.1 Canonical Data Model

*(no schema captured in original — needs definition by engineering)*

The following canonical tables are implied by this specification and must be defined before build:

- `referral_records` — primary governed object (see §3.1 for required fields)
- `referral_state_transitions` — immutable audit of all state changes
- `referrer_directory` — governed Referrer / Specialist Directory entries (see §3.5)
- `referral_documents` — structured referral document and outcome report attachments
- `referral_communications` — captured inbound and outbound communication history per referral

### 13.2 Core Behaviour Rules

1. Every Referral Record **MUST** have a named owner (OwnerID **MUST NOT** be null on any active referral).
2. A referral **MUST NOT** exist solely as an email, document, or inbox item; it must be a governed Referral Record.
3. Mandatory lifecycle states may not be bypassed for either Outbound or Inbound directions.
4. Every state transition **MUST** be timestamped, attributed to an actor, and written to the immutable AuditTrail before the transition is confirmed to the caller.
5. SLA timers **MUST** start automatically when a referral enters an active state; no active referral may exist without a running SLA timer.

6. Inbound email ingestion **MUST** be idempotent — duplicate ingestion of the same referral email **MUST NOT** create duplicate Referral Records.
7. AI-drafted content **MUST NOT** be sent or committed without explicit human approval; the approval action **MUST** be logged in the AuditTrail.
8. External referrer access **MUST** be scoped strictly to their assigned referrals; cross-referral data **MUST NOT** be accessible to external users.
9. All patient notifications **MUST** be dispatched via Communication Hub; Referral Manager **MUST NOT** dispatch notifications directly.
10. Metrics emitted to Performance Dashboards are read-only signals; Referral Manager **MUST NOT** read back or act on visualised analytics data.

### 13.3 Configuration Surfaces

- **Practice-level settings (Admin Control Plane):** SLA threshold values (reminder trigger, at-risk threshold, overdue threshold); preferred referral methods in the Referrer Directory; shared referrals mailbox configuration.
- **Per-user preferences (Access Manager):** Role assignments and tenant scoping for all staff users and external referrers.
- **Per-referral overrides (this module):** Ownership reassignment; manual SLA override with audit justification (*if supported — needs decision; see §15*).

### 13.4 Filtering & Views

The Referral Dashboard **MUST** support the following filters and views:

- Filter by direction (Inbound / Outbound)
- Filter by status / lifecycle state
- Filter by named owner
- Filter by external provider / referrer
- Filter by patient
- Filter by SLA status (On Track / At-Risk / Overdue)
- Default saved view: "My Active Referrals" scoped to the authenticated user's OwnerID.
- Practice Manager saved view: "All Overdue Referrals" across all owners.

### 13.5 Module Extension Map

- Future: Portal-based lightweight external referrer interface (structured accept / info-request / decline, replacing email-link responses).
- Future: Enhanced inbound referral parsing (structured extraction from ingested emails).
- Future: Referrer Directory enrichment with performance ratings or outcome data.
- Extensions must not alter the canonical state machines in §3.2 and §3.3 or remove mandatory audit events in §8 without a major version bump to this spec.

### 13.6 Acceptance Criteria

The build of Referral Manager is complete when:

- [ ] All Referral Records can be created, read, and updated through the API for both Outbound and Inbound directions.
- [ ] State machine transitions enforce all rules in §3.2 and §3.3; mandatory states cannot be bypassed.
- [ ] SLA timers start automatically on entry to active states and trigger escalation tasks via Task Manager at configured thresholds.
- [ ] Inbound email ingestion is idempotent and traceable.
- [ ] All integrations in §6 are wired and verified end-to-end.
- [ ] AI boundaries in §7 are enforced; negative tests confirm AI cannot send, accept, or complete referrals autonomously.
- [ ] Audit log captures every event listed in §8, with actor and timestamp, and is confirmed immutable.
- [ ] Access control is enforced per §9; external referrer cross-referral access is verified to be blocked.
- [ ] Referrer Directory entries can be created, updated, and archived.
- [ ] All non-functional requirements in §14 are met.
- [ ] Metrics are emitted to Performance Dashboards and confirmed read-only from Referral Manager's perspective.

## 14. Non-Functional Requirements

---

- **Performance:** Referral state updates and audit writes must propagate near-real-time. Dashboard filter queries must return within an acceptable latency threshold to be defined by engineering, targeting interactive response times for practice staff.
- **Reliability:** Inbound email ingestion must be idempotent and traceable; ingestion failures must not silently drop referrals and must surface as observable errors. The module must degrade gracefully if downstream consumers (Task Manager, Communication Hub) are temporarily unavailable, queuing outbound events for retry rather than failing the referral state transition.
- **Scalability:** The module must operate correctly in a multi-tenant, multi-site deployment. Each tenant's referral data must be strictly isolated.
- **Security:** All referral data — including patient identifiers, clinical context, and communications — must be encrypted at rest and in transit. External referrer access tokens must be scoped and time-limited. Secrets and credentials for the shared mailbox integration must be managed via the platform's secrets management layer.
- **Privacy:** The module must honour patient GDPR rights applicable to referral records, including the right of access and the right to erasure where legally permissible. Data retention policy for Referral Records and associated communications must be defined and enforced; retention period is subject to clinical and regulatory requirements and must be confirmed before build (see §15).
- **Observability:** The module must export structured logs for all state transitions and integration events, metrics for referral volume and SLA breach rates, and distributed traces for inbound email ingestion and outbound notification flows. Health and readiness endpoints must be exposed for platform monitoring.

## 15. Open Questions

---

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

1. **SLA threshold defaults:** What are the default SLA thresholds (reminder, at-risk, overdue) for Outbound and Inbound referrals? Are they different per direction or per specialty?

2. **Manual SLA override:** Should practice staff (e.g. Practice Manager) be permitted to manually override or pause an SLA timer on a referral? If so, what governance and audit controls apply?
3. **External referrer portal:** The spec references an "optional lightweight portal access" for external referrers. Is this in scope for the initial build or deferred to a future version? What is the authentication mechanism for external portal users?
4. **AI Quality Monitor integration:** The module consumes missed-referral detection signals from AI Quality Monitor. Is AI Quality Monitor a confirmed module in the platform corpus, and what is the agreed event contract for these signals?
5. **Data retention policy:** What is the required retention period for Referral Records and associated communications, given clinical and regulatory obligations? This must be confirmed before build to implement §14 privacy controls.
6. **PMS integration contract:** Is patient demographic and appointment data pulled from the PMS on demand or synchronised? The exact contract at the PMS boundary must be agreed with the PMS integration layer before build.
7. **Scheduled sub-state ownership:** The `Scheduled` sub-state (Outbound: Accepted → Scheduled; Inbound: Accepted / Assigned → Scheduled) implies appointment linkage. Is this linkage to Appointment Manager, and if so, what is the integration contract?
8. **Lab Manager provider boundary:** Where an external laboratory is the receiving provider for a patient referral, should the Referral Record's `ReceivingProvider` / `Service` field reference Lab Manager's Lab Directory entry directly, or should a lightweight mirror entry be maintained in the Referrer Directory? The integration approach at this boundary must be agreed with the Lab Manager module owner before build.