MindLog

Mental Wellness Tracking Platform

Software Requirements Specification

Document Version: 1.0 DRAFT
Date: February 2026
Status: For Review

Document Type	Software Requirements Specification
Scope	Patient Mobile App + Clinician Dashboard
Platform	iOS, Android, Web (SPA)
Database	PostgreSQL 15+
Prepared By	MindLog Product Team
Confidentiality	CONFIDENTIAL — Clinical Use Only

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) defines the complete functional, non-functional, security, and compliance requirements for the MindLog platform. It is intended to serve as the authoritative reference for all design, development, testing, and deployment activities, and as the basis for regulatory submissions where required.

This document covers three interconnected components: the Patient Mobile Application (iOS and Android), the Clinician Dashboard (web application), and the shared Backend API and data platform that serves both.

1.2 Scope

MindLog is a mental wellness tracking platform that enables patients with mental health conditions to record daily mood, sleep, exercise, medication adherence, triggers, symptoms, and journal entries via a mobile application. Clinicians — including psychiatrists, psychologists, and general practitioners — access aggregated and individual patient data through a secure web dashboard, receive AI-generated clinical alerts when patient patterns warrant attention, and manage clinical workflows including notes, appointments, and reports.

The platform is designed for deployment within psychiatric and psychology services in clinical and private practice settings. Version 1.0 targets the Australian market; the architecture is designed to support additional jurisdictions in subsequent versions.

1.3 Intended Audience

Audience	Relevance
Product Team	Authoritative reference for feature decisions and scope management.
Engineering Team	Functional and technical specifications driving implementation.
QA & Test Engineers	Acceptance criteria for test case authorship.
Clinical Advisors	Review of clinical workflow accuracy and patient safety logic.
Security & Compliance	Input on regulatory obligations and risk assessment.
Regulatory Bodies	Evidence of structured requirements for SaMD classification.
Investors / Partners	Product capability and maturity evidence.

1.4 Definitions, Acronyms, and Abbreviations

Term / Abbreviation	Definition
SRS	Software Requirements Specification
SaMD	Software as a Medical Device (TGA/FDA classification)
TGA	Therapeutic Goods Administration (Australian medical device regulator)
AHPRA	Australian Health Practitioner Regulation Agency
RLS	Row-Level Security (PostgreSQL access control mechanism)
MRN	Medical Record Number
PHI	Protected Health Information

Term / Abbreviation	Definition
PII	Personally Identifiable Information
E2EE	End-to-End Encryption
RBAC	Role-Based Access Control
MFA	Multi-Factor Authentication
API	Application Programming Interface
REST	Representational State Transfer
JWT	JSON Web Token
WCAG	Web Content Accessibility Guidelines
ICD-10	International Classification of Diseases, 10th Revision
FCM	Firebase Cloud Messaging (Android push notifications)
APNs	Apple Push Notification service
GDPR	General Data Protection Regulation (EU)
Privacy Act	Privacy Act 1988 (Cth) — Australian privacy legislation
Audit Log	Immutable, append-only record of all data access and mutations
Rules Engine	Background service that evaluates patient data against alert thresholds
Care Team	The set of clinicians assigned to a patient at any given time
Safety Event	A logged instance of a safety symptom (e.g. suicidal ideation)
Tracking Streak	Consecutive days on which a patient completed a daily entry

1.5 References

- MindLog UX Specification v1.0 — Patient Mobile App (February 2026)
- MindLog Interactive Prototype v2.0 — All 11 screens (February 2026)
- MindLog Database DDL v1.0 — PostgreSQL schema (February 2026)
- MindLog Clinician Dashboard Prototype v1.0 (February 2026)
- TGA — Software as a Medical Device (SaMD): Regulatory Framework for Digital Health
- OWASP Application Security Verification Standard (ASVS) 4.0.3
- WCAG 2.1 Level AA — W3C Recommendation
- Australian Privacy Principles (APPs) — Schedule 1, Privacy Act 1988 (Cth)
- HL7 FHIR R4 — Base Specification
- ISO/IEC 27001:2022 — Information Security Management

1.6 Document Conventions

Requirements are prioritised using MoSCoW notation:

Priority	Definition
MUST	A mandatory requirement. The system cannot be released without it. Failure constitutes a critical defect.
SHOULD	A high-value requirement that is strongly recommended. Omission requires documented justification.
COULD	A desirable requirement included when time and resource permit.
WONT	Explicitly out of scope for Version 1.0. Documented to prevent scope creep.

Each requirement carries a unique identifier in the format [COMPONENT]-[CATEGORY]-[NNN], for example PAT-AUTH-001 (Patient app, Authentication, requirement 001).

2. Overall Description

2.1 Product Perspective

MindLog consists of three deployable components that share a single PostgreSQL 15+ database accessed through a RESTful JSON API. The Patient Mobile Application is the primary data collection surface. The Clinician Dashboard consumes that data for population health management and individual clinical review. The Backend API is the authoritative integration layer between them.

No component stores primary data locally beyond what is required for offline operation. The mobile app caches the current day's unsaved entry in device storage; all submitted entries are the canonical record in the database.

2.2 Product Functions — Summary

The table below summarises the major functional areas across all three components.

Function	Component	Description
Daily Check-In	Patient Mobile App	Mood, coping, sleep, exercise, medications — captured daily in under 90 seconds.
Wellness Strategies	Patient Mobile App	Daily logging of protective behaviours with tristate (Yes / No / N/A) and quality ratings.
Trigger Logging	Patient Mobile App	Active trigger identification with severity rating (1–10).
Symptom Logging	Patient Mobile App	Symptom presence and intensity, with safety event detection.
Journal	Patient Mobile App	Freeform daily entry with AI-generated prompts; optional clinician sharing.
Insights	Patient Mobile App	Patient-facing charts: mood trend, sleep, correlation analysis.
Population Heatmap	Clinician Dashboard	30-day mood heatmap across the clinician's entire caseload.
Clinical Alerts	Clinician Dashboard	Rule-generated alerts surfaced in priority order; acknowledgement workflow.
Patient Detail	Clinician Dashboard	Per-patient charts, daily entry history, trigger/symptom analysis.
Clinical Notes	Clinician Dashboard	Structured notes linked to patients and specific dates.
Population Trends	Clinician Dashboard	Aggregate mood, sleep, adherence, and risk distribution charts.
Report Generation	Clinician Dashboard	On-demand PDF reports: individual patient, population, and handover.
Rules Engine	Backend	Background job evaluating alert thresholds and generating ClinicalAlerts.
Notification Dispatch	Backend	Push, email, and SMS delivery of alerts and patient reminders.
Audit Logging	Backend	Immutable audit trail of all data access and mutations.
Consent Management	Backend	Full consent lifecycle including grant, expiry, and revocation.

2.3 User Classes and Characteristics

2.3.1 Patient

A person receiving treatment for a mental health condition, aged 16 and above. Patients interact exclusively with the mobile application. They range from highly technology-literate to minimal smartphone experience. The interface

must accommodate users who are in acute distress, whose cognition may be temporarily impaired, and who may be using the app at low points. Literacy levels vary; the app must avoid clinical jargon. The consent model places the patient in control of what data their clinician can access.

2.3.2 Clinician

A licensed health professional — psychiatrist, psychologist, general practitioner, care coordinator, or clinical nurse — responsible for the clinical care of one or more patients. Clinicians interact primarily with the web dashboard on desktop or tablet. They are time-pressured; the dashboard must surface the highest-priority information immediately. Clinicians are expected to understand clinical concepts but not necessarily technical software concepts.

2.3.3 Organisation Administrator

A non-clinical staff member managing the organisational configuration: adding clinicians, configuring alert routing, managing the wellness strategy catalogue, and generating aggregate reports. They do not access individual patient clinical data.

2.3.4 System / Rules Engine

An automated process that evaluates patient data on a schedule and generates ClinicalAlerts. This is not a human actor; it is documented here because it has specific data access requirements and produces observable side effects visible to all human actors.

2.4 Operating Environment

Patient Mobile Application

- iOS 16.0 and above, iPhone and iPad
- Android 10.0 (API level 29) and above
- Offline-capable: daily entries must be completable without network connectivity; sync occurs on next connection
- Accessible in low-light conditions; supports system Dark Mode
- Tested on devices with as little as 2 GB RAM

Clinician Dashboard

- Modern evergreen browsers: Chrome 110+, Firefox 110+, Safari 16+, Edge 110+
- Desktop-first, responsive to 768px minimum width
- No native desktop app required for Version 1.0
- Session timeout configurable per clinician (default 30 minutes inactivity)

Backend and Infrastructure

- PostgreSQL 15+ hosted on a managed cloud provider (AWS RDS, Google Cloud SQL, or Supabase)
- All data at rest encrypted with AES-256
- All data in transit encrypted with TLS 1.3 minimum
- Australian data residency for Version 1.0 (AWS ap-southeast-2 or equivalent)
- Infrastructure-as-code; reproducible environment provisioning

2.5 Design and Implementation Constraints

- The application must comply with the Australian Privacy Act 1988 (Cth) and the Australian Privacy Principles (APPs) from day one of any clinical pilot.

- Safety event detection and alert generation must occur within 5 minutes of a patient submitting an entry containing a safety symptom.
- Journal entries must default to private (not visible to clinicians) and must never be readable by clinicians without explicit, logged patient consent.
- The audit log is append-only. No UPDATE or DELETE operations are permitted on the audit_log table; this must be enforced at both the application and database levels.
- The mobile app must not store unencrypted PHI in device logs, crash reports, or analytics payloads.
- Third-party analytics SDKs must not receive PHI. Any event tracking must use anonymised identifiers only.
- The system must be architected to support FHIR R4 export in a future version; no design decisions should make this structurally impossible.

2.6 Assumptions and Dependencies

- Patients have access to an iOS or Android smartphone with sufficient storage for the application.
- Clinicians access the dashboard from a device on the organisation's network or via approved remote access.
- The organisation operates within a jurisdiction covered by the Privacy Act 1988 (Cth) for Version 1.0.
- Push notification delivery depends on third-party services (APNs, FCM) outside the system's control.
- ICD-10-AM codes are loaded from a licensed standard data feed; maintenance of that feed is out of scope.
- SMS delivery for critical alerts depends on a third-party SMS gateway (e.g. Twilio).

3. System Architecture

3.1 Architectural Overview

MindLog follows a three-tier architecture: a presentation layer (mobile app + web dashboard), a service layer (RESTful API), and a data layer (PostgreSQL). The API is the sole gateway to the data layer; no component connects directly to the database except the Rules Engine background service, which runs in the same trust boundary.

3.2 Component Architecture

Component	Technology	Responsibility
Patient Mobile App	React Native (Expo)	Cross-platform iOS/Android. WatermelonDB for local SQLite with sync. Expo Notifications for push.
Clinician Dashboard	React (Vite + TypeScript)	SPA. TanStack Query for data fetching. Recharts for visualisations. Real-time updates via WebSocket.
Backend API	Node.js + TypeScript (Fastify)	Stateless RESTful API. JWT authentication. Zod validation. Structured logging (Pino).
Database	PostgreSQL 15+	Primary data store. Row-Level Security for access control. Partitioned audit_log. Managed via Supabase or RDS.
Rules Engine	Node.js worker (BullMQ)	Background job queue. Nightly snapshot computation. Real-time safety event processing via DB trigger + webhook.
File Storage	S3-compatible (AWS S3 / Supabase Storage)	Generated PDF reports. Signed URLs with configurable expiry.
Email / SMS	Resend (email) + Twilio (SMS)	Critical alert delivery. Patient reminder emails. Configurable per organisation.
Push Notifications	Expo Push Service (wraps APNs + FCM)	Patient daily reminders. Medication reminders. Streak notifications.
Auth	Supabase Auth or Auth0	JWT issuance. MFA (TOTP). Session management. Separate tokens for patient and clinician roles.

3.3 Data Flow — Patient Check-In

1. Patient opens app; local WatermelonDB serves today's entry state if already started.
2. Patient completes entry sections; each section auto-saves locally as a draft.
3. On submission, the app calls POST /api/entries with the complete entry payload.
4. API validates, writes daily_entry and all child records in a single transaction.
5. The symptom_logs INSERT trigger evaluates safety symptoms; if present, inserts into safety_events and clinical_alerts atomically.
6. Rules Engine webhook receives a safety_event notification and initiates alert routing within 60 seconds.
7. Clinician receives in-app alert (WebSocket push) and, if severity = critical, push/email/SMS within 5 minutes.

3.4 Data Flow — Clinician Dashboard Load

8. Clinician authenticates; JWT includes organisation_id and clinician_id claims.
9. Dashboard calls GET /api/dashboard/overview — API reads from population_snapshots (pre-aggregated nightly) for KPI cards.

10. Heatmap data fetched from GET /api/dashboard/heatmap?days=30 — reads v_mood_heatmap_30d view.
11. Active alerts fetched from GET /api/alerts?status=unacknowledged.
12. WebSocket connection established; server pushes new alert events in real time.
13. Patient detail loaded on demand; raw log tables queried with RLS enforcing care team restriction.

4. Patient Mobile Application

4.1 Authentication and Onboarding

4.1.1 Functional Requirements

ID	Priority	Requirement	Acceptance Criteria
PAT-AUTH-001	MUST	The app shall support account creation with email address and password. Passwords must meet a minimum strength requirement of 12 characters including uppercase, lowercase, number, and symbol.	Weak passwords rejected with specific error messaging. Account created and JWT issued on success.
PAT-AUTH-002	MUST	The app shall support biometric authentication (Face ID / fingerprint) as the primary login method after initial password setup.	Biometric prompt presented on app open after setup. Falls back to PIN/password on biometric failure.
PAT-AUTH-003	MUST	The onboarding flow shall collect the patient's medication list during setup. Each medication shall be stored as a patient_medication record with name and optional dose.	Medications entered during onboarding appear on the daily medicine card on first check-in.
PAT-AUTH-004	MUST	The onboarding flow shall present a notification permission request with a preview of the notification content before invoking the system permission dialogue.	Preview shown before system prompt. Permission state recorded in patient_notification_preferences.
PAT-AUTH-005	MUST	Patients shall be able to log out explicitly. On logout, locally cached PHI shall be cleared from device storage.	Logout clears WatermelonDB local cache. Biometric prompt presented on next open.
PAT-AUTH-006	SHOULD	The app shall support passwordless authentication via a magic link sent to the patient's registered email.	Link expires after 15 minutes. Valid link issues a JWT with full patient scope.
PAT-AUTH-007	SHOULD	The app shall lock after a configurable inactivity period (default 10 minutes) and require biometric or PIN to re-enter.	Lock triggered after inactivity. Draft entry state preserved across lock/unlock.

4.2 Daily Check-In — Core Entry

4.2.1 Mood and Coping

ID	Priority	Requirement	Acceptance Criteria
PAT-ENTRY-001	MUST	The app shall present a 10-point mood rating using a colour-coded ring of selectable pips (1 = red, 10 = deep blue). Tapping a pip selects it; tapping again deselects.	Selected pip lifts with shadow. Badge updates with numeric value and semantic label. State persists on navigation.
PAT-ENTRY-002	MUST	The app shall present an equivalent 10-point coping rating using the same ring component as mood.	Coping ring behaviour identical to mood ring. Both values independently stored.
PAT-ENTRY-003	MUST	A completion ring in the Today screen header shall display the percentage of the day's entry that is complete, updating in real time as sections are filled.	Ring arc length proportional to completion_pct. Updates within 500ms of any section change.

4.2.2 Sleep

ID	Priority	Requirement	Acceptance Criteria
PAT-ENTRY-010	MUST	The app shall capture sleep duration using paired steppers for hours (0–12) and minutes (0, 15, 30, 45). Minutes shall wrap to hours at 60.	Hour stepper: 0–12. Minute options: 0/15/30/45. Wrap tested: 11h45 + 15m = 12h00.
PAT-ENTRY-011	MUST	The sleep large-display shall update in real time as steppers are adjusted.	Display reflects current value within one render frame of button press.
PAT-ENTRY-012	SHOULD	The app shall offer to import sleep duration from Apple Health or Google Fit if the patient has granted permission.	Permission requested once. Imported value pre-fills steppers. Patient may override manually.

4.2.3 Exercise

ID	Priority	Requirement	Acceptance Criteria
PAT-ENTRY-020	MUST	The app shall offer exercise duration quick-pick presets: 15, 30, 60, 90 minutes, and None. Only one preset may be active at a time.	Active preset highlighted in teal. Badge in card header updates. None sets duration to 0.
PAT-ENTRY-021	SHOULD	A custom duration input shall be available as an additional option beyond the presets.	Custom entry accepts integers 1–599. Invalid input rejected with inline error.

4.2.4 Medications

ID	Priority	Requirement	Acceptance Criteria
PAT-ENTRY-030	MUST	Each active patient medication shall appear as a named toggle in the medication card. Toggle state (taken / not taken) shall be persisted to medication_adherence_logs on save.	Medication names match patient_medications records where show_in_app = TRUE. One adherence log row per medication per day.
PAT-ENTRY-031	MUST	Toggle state shall animate smoothly (knob slides left/right within 250ms).	Animation measured at 250ms or less on reference device.
PAT-ENTRY-032	SHOULD	Patients shall be able to add a free-text note to a medication toggle (e.g. "forgot", "took late").	Note stored on medication_adherence_logs.notes column.

4.3 Wellness Strategies

ID	Priority	Requirement	Acceptance Criteria
PAT-WELL-001	MUST	Each wellness strategy shall present a three-state selector: Yes, No, N/A. Cycling through states shall update the wellness_logs record immediately on selection.	Three states rendered as distinct pill styles. State persists on back-navigation and app restart.

ID	Priority	Requirement	Acceptance Criteria
PAT-WELL-002	**MUST**	When a strategy with has_quality_rating = TRUE is set to Yes, a 1–10 quality slider shall expand inline below the strategy row with an animated transition (max-height animation, 300ms ease).	Slider hidden when state != Yes. Visible and interactive when state = Yes. Value stored to wellness_logs.quality.
PAT-WELL-003	**MUST**	Strategy names shall appear dimmed (reduced opacity) when state is N/A, providing visual confirmation that the strategy is not tracked for today.	Opacity of name text ≤ 0.4 when N/A. Full opacity when Yes or No.
PAT-WELL-004	**MUST**	The Wellness screen header shall display live counts for Yes (Done), No (Didn't), and N/A, recalculating on every state change.	Counts update within one render frame of tristate change.
PAT-WELL-005	**SHOULD**	Patients shall be able to add a custom wellness strategy via a dashed "Add strategy" button at the foot of the strategy list. Custom strategies are stored as wellness_strategies records with the patient_id set.	Custom strategy appears in patient's list immediately. Persists across sessions.

4.4 Triggers

ID	Priority	Requirement	Acceptance Criteria
PAT-TRIG-001	**MUST**	Each trigger shall present a two-state selector: On and N/A. The absence of a trigger is the baseline; there is no explicit "No" state.	Only On and N/A states rendered. No "No" option present.
PAT-TRIG-002	**MUST**	When a trigger is set to On, a severity slider (1–10, rose gradient) shall expand inline. The slider value shall be stored to trigger_logs.severity.	Slider not visible when N/A. Visible and interactive when On. Default value = 5 on first activation.
PAT-TRIG-003	**MUST**	The Triggers screen header shall display counts for Active triggers, N/A triggers, and average severity of active triggers.	Average severity = sum of active severity values / count of active triggers, to one decimal place.
PAT-TRIG-004	**SHOULD**	Patients shall be able to add custom triggers. Custom trigger records shall be stored with patient_id and category = "custom".	Custom trigger persists. Appears in trigger list on subsequent days.

4.5 Symptoms

ID	Priority	Requirement	Acceptance Criteria
PAT-SYM-001	**MUST**	Each symptom shall present a two-state selector: On and N/A. The intensity slider (1–10, lavender gradient) shall expand inline when On is selected.	Slider visible only when is_present = TRUE. Value stored to symptom_logs.intensity.
PAT-SYM-002	**MUST**	Any symptom with is_safety_symptom = TRUE (e.g. Suicidal Thoughts) shall, when activated, immediately display a safety resource card within the same screen. The card shall contain a clearly labelled tap-to-call link to a crisis service.	Resource card appears within one render frame of activation. Link tested for correct destination. Card remains visible until symptom is deactivated.

ID	Priority	Requirement	Acceptance Criteria
PAT-SYM-003	**MUST**	Activation of a safety symptom shall trigger the handle_safety_symptom() database function, creating a safety_events record and a critical ClinicalAlert, and updating the patient's status to "crisis" — all within the same database transaction as the symptom_log INSERT.	Safety_events row present in DB within 1 second of symptom submission. ClinicalAlert with severity = "critical" present. patient.status = "crisis".
PAT-SYM-004	**MUST**	The app shall NOT display alarming visual indicators (red flashing, warning sirens) in response to a safety symptom activation. The UI response shall be calm and supportive.	UX review with clinical advisor confirms tone. No alert-style UI elements within the resource card.

4.6 Journal

ID	Priority	Requirement	Acceptance Criteria
PAT-JOUR-001	**MUST**	The journal entry screen shall present a daily prompt selected at random from the journal_prompts catalogue. The same prompt shall persist for the entire day; it shall not change on re-entry.	Prompt consistent across app opens for the same calendar date. Changes on date rollover.
PAT-JOUR-002	**MUST**	A freeform textarea shall accept typed journal entries. Word count shall update in real time.	Word count matches whitespace-split word array length. Updates on each keystroke.
PAT-JOUR-003	**MUST**	Journal entries shall be private by default. The journal_entries.shared_with_clinician column shall default to FALSE. Sharing with clinician requires an explicit patient action and records a consent_records entry.	New entries created with shared_with_clinician = FALSE. Clinician API returns 403 for unshared entries.
PAT-JOUR-004	**MUST**	Past entries shall be listed in reverse chronological order below the current entry. Each past entry shall display a date label and a truncated preview of the first 80 characters.	List correctly ordered. Preview truncated with ellipsis at 80 characters.
PAT-JOUR-005	**SHOULD**	The journal shall support voice-to-text input using on-device speech recognition. Transcribed text shall be appended to the current entry.	Voice input tested on iOS (SFSpeechRecognizer) and Android (SpeechRecognizer). input_method = "voice" stored on the record.
PAT-JOUR-006	**SHOULD**	The journal shall support full-text search across all of the patient's past entries.	Search returns entries containing the query string. Results ordered by relevance.

4.7 Insights (Patient-Facing)

ID	Priority	Requirement	Acceptance Criteria
PAT-INS-001	**MUST**	The Insights screen shall display a 30-day mood and coping line chart. The mood line shall have a gradient fill. Both lines shall render from patient correlation_cache and daily_entries data.	Chart renders with correct data points. Missing days shown as gaps (no interpolation).

ID	Priority	Requirement	Acceptance Criteria
PAT-INS-002	**MUST**	The Insights screen shall display a 30-day sleep bar chart. Bars representing nights below the patient's sleep target (default 7 hours) shall render in a distinct warning colour.	Bars below target colour-differentiated. Target line rendered as dashed rule.
PAT-INS-003	**MUST**	A correlation impact panel shall display the top 5 factors positively or negatively correlated with the patient's mood, drawn from patient_correlation_cache. Each factor shall show a labelled bar and a signed delta value.	Correlation bars animated on load. Values match patient_correlation_cache.mood_delta_x10 / 10.
PAT-INS-004	**SHOULD**	An AI-generated insight card shall present a natural language summary of the most significant pattern in the current period. The insight shall be regenerated weekly.	Insight text present and clinically accurate (reviewed by clinical advisor). Refreshes on period change.
PAT-INS-005	**SHOULD**	A period selector (2W, 1M, 3M, Custom) shall filter all charts and the correlation panel to the selected date range.	All charts update on period change. Custom range supports arbitrary start/end date selection.

4.8 Date Navigation

ID	Priority	Requirement	Acceptance Criteria
PAT-NAV-001	**MUST**	All logging screens (Wellness, Triggers, Symptoms, Journal) shall share a date navigation strip with forward and back arrows. All screens shall reflect the same currently selected date.	Date change on one screen reflected on all others. Date label consistent across screens.
PAT-NAV-002	**MUST**	Patients shall not be able to navigate to a future date for the purpose of logging. The forward arrow shall be disabled when the currently selected date is today.	Forward arrow disabled/hidden on current date. API rejects future-dated entry submissions.
PAT-NAV-003	**SHOULD**	Tapping the date label shall open a calendar picker for non-sequential date navigation.	Calendar correctly restricts future selection. Selected date loads correct entry state.

4.9 Offline Capability

ID	Priority	Requirement	Acceptance Criteria
PAT-OFF-001	**MUST**	All daily check-in screens shall be fully functional without network connectivity. Entries shall be saved locally and synced when connectivity is restored.	Entry completable in airplane mode. Sync verified within 30 seconds of network reconnection.
PAT-OFF-002	**MUST**	Local drafts shall survive app termination and device restart.	Draft state intact after force-quit and reopen. Draft state intact after device restart.
PAT-OFF-003	**MUST**	Sync conflicts (entry modified on two devices while offline) shall be resolved in favour of the most recently modified version. The conflict shall be logged.	Conflict resolution tested with simultaneous offline modification. Correct version retained.

5. Clinician Dashboard

5.1 Authentication and Access Control

ID	Priority	Requirement	Acceptance Criteria
CLI-AUTH-001	**MUST**	Clinicians shall authenticate with email and password. MFA (TOTP) shall be enforced for all clinician accounts with access to patient data.	Login rejected without valid TOTP on accounts with mfa_enabled = TRUE. TOTP setup required on first login.
CLI-AUTH-002	**MUST**	Sessions shall expire after a configurable inactivity period (default 30 minutes). Expired sessions shall require re-authentication.	Session timeout enforced at the API layer (not just the client). Timeout value matches clinicians.session_timeout_min.
CLI-AUTH-003	**MUST**	Clinicians shall only be able to access data for patients on their active care team. RLS policies on the database and API-layer authorisation checks shall both enforce this.	Request for patient not on care team returns 403. Verified against both RLS and API middleware.

5.2 Population Overview

ID	Priority	Requirement	Acceptance Criteria
CLI-POP-001	**MUST**	The population overview shall display five KPI metric cards: Critical Alerts, Active Today (check-in count), Average Mood, Average Sleep, and Medication Adherence Rate. All values shall be drawn from the most recent population_snapshots record.	KPI values match population_snapshots. Delta indicators show change vs. previous day's snapshot.
CLI-POP-002	**MUST**	A safety alert strip shall appear at the top of the population overview whenever any unresolved safety_event exists for the clinician's patients. It shall name the patient, time of flag, and present a "Review Now" action.	Strip appears within 10 seconds of safety_event creation (via WebSocket). Disappears when event is acknowledged.
CLI-POP-003	**MUST**	A 30-day mood heatmap shall display one row per patient in the clinician's caseload. Each cell represents one calendar day. Cell colour maps to the patient's logged mood on that day (1 = red, 10 = deep blue; grey = no data; glowing red = safety flag).	Heatmap data matches v_mood_heatmap_30d view. Colour mapping verified against MOOD_COLORS array. Safety flag cells distinguished with border glow.
CLI-POP-004	**MUST**	Hovering any heatmap cell shall display a tooltip showing: patient name, date, mood value, and sleep hours for that day.	Tooltip appears within 100ms of hover. Contains all four data points. Disappears on mouse-out.
CLI-POP-005	**MUST**	Clicking any heatmap row label or cell shall navigate to the Patient Detail view for that patient.	Navigation occurs within 300ms. Patient detail loads correct patient.
CLI-POP-006	**MUST**	A mood distribution histogram shall display the count of patients at each mood value (1–10) for the current day.	Histogram counts match daily_entries for current date. Renders with correct colours.
CLI-POP-007	**MUST**	A top triggers bar chart shall display the 6 most prevalent active triggers across the population for the past 7 days, showing trigger name and count of patients who logged it.	Bars sorted descending by count. Counts match trigger_logs for date range.

5.3 Clinical Alerts

ID	Priority	Requirement	Acceptance Criteria
CLI-ALERT-001	**MUST**	The Alerts view shall display all clinical alerts for the clinician's patients, sorted by severity (critical first) then by creation time descending.	Critical alerts always appear before warnings; warnings before info. Within severity, newest first.
CLI-ALERT-002	**MUST**	Each alert shall display: patient name (tap navigates to patient detail), alert type icon, title, body text, timestamp, severity badge, and action buttons.	All fields present. Patient name navigation verified. Severity badge colour matches severity level.
CLI-ALERT-003	**MUST**	Critical-severity alerts shall include a "Crisis protocol" action button. Clicking it shall create a clinician_note of type "intervention" pre-populated with the crisis protocol template and open the patient detail.	Note created on click. Note body contains crisis protocol template. Patient detail opens.
CLI-ALERT-004	**MUST**	Clinicians shall be able to acknowledge any alert. Acknowledging sets acknowledged_at and acknowledged_by on the clinical_alerts record, and immediately removes the alert from the unacknowledged filter.	acknowledged_at set within 1 second. Alert disappears from unacknowledged view. Audit log records the acknowledgement.
CLI-ALERT-005	**MUST**	New critical alerts shall be delivered to the clinician's browser in real time via WebSocket without requiring a page refresh.	Alert appears in dashboard within 10 seconds of clinical_alerts row creation. Tested by inserting directly into DB.
CLI-ALERT-006	**SHOULD**	Acknowledged alerts shall remain visible in the Alerts view at reduced opacity with a tick indicator, to maintain a complete audit trail within the UI.	Acknowledged state visually distinct. Not hidden unless explicitly filtered out.
CLI-ALERT-007	**MUST**	The sidebar navigation shall display a live badge showing the count of unacknowledged critical alerts.	Badge count matches unacknowledged critical alert count. Updates in real time via WebSocket.

5.4 Patient List

ID	Priority	Requirement	Acceptance Criteria
CLI-LIST-001	**MUST**	The Patient List shall display all patients on the clinician's care team in a sortable table with columns: name/MRN, primary diagnosis, risk level badge, 10-day mood sparkline, today's mood dot, tracking streak, last check-in, and next appointment.	All columns present. Risk badge colour correct. Sparkline bars height-proportional to mood value.
CLI-LIST-002	**MUST**	The Patient List shall support one-click filter chips: All, Critical (risk), High Risk, Not Logged Today, Streak 7d+. Active filter persists across tab changes within the same session.	Each filter correctly subsets the patient list. Filter state maintained on return navigation.
CLI-LIST-003	**MUST**	The table shall be sortable by: name, risk level, today's mood, tracking streak, and last check-in date.	Each sortable column sorts correctly ascending and

ID	Priority	Requirement	Acceptance Criteria
			descending. Default sort is risk level descending.
CLI-LIST-004	**MUST**	Global search in the topbar shall filter the patient list by patient name, MRN, or diagnosis in real time as the user types.	Results update within 200ms of each keystroke. Search is case-insensitive. Matches on partial strings.

5.5 Patient Detail

ID	Priority	Requirement	Acceptance Criteria
CLI-DET-001	**MUST**	The Patient Detail header shall display: full name, MRN, age, gender, risk level badge, primary diagnosis, active medications, tracking streak, next appointment, and count of unacknowledged alerts.	All fields present. Alert count badge absent when count = 0.
CLI-DET-002	**MUST**	Patient Detail shall have five sub-tabs: Overview, Daily Entries, Charts & Trends, Triggers & Symptoms, and Clinical Notes.	All five tabs present and navigable. Active tab visually distinct.
CLI-DET-003	**MUST**	The Daily Entries tab shall render a full log table with one row per calendar day for the past 30 days. Each row shall show: date, mood dot, coping dot, sleep hours, exercise minutes, medication taken status, and flag icons for active triggers, symptoms, missed medication, and safety flags.	Table has 30 rows. Missing days shown at reduced opacity. Flag icons correct per daily_entries data.
CLI-DET-004	**MUST**	The Charts & Trends tab shall render: a 30-day mood and coping line chart, a 30-day sleep hours chart with target line at 7h, and a 30-day medication adherence bar chart (green = taken, red = not taken, grey = not logged).	Charts render from patient's daily_entries data. Target line at 7h confirmed. Adherence colours correct.
CLI-DET-005	**MUST**	The Clinical Notes tab shall display all notes for the patient in reverse chronological order. Each note shall show type badge, date, author name, and body. Deleted notes shall not be shown.	Notes ordered correctly. Soft-deleted notes (deleted_at IS NOT NULL) excluded.
CLI-DET-006	**MUST**	Clinicians shall be able to add a note from the Clinical Notes tab by entering text and selecting a note type (Observation, Intervention, Appointment Summary, Risk Assessment). Saving creates a clinician_notes record with the clinician's ID and timestamp.	Note created on save. Appears immediately in note list. clinician_id matches authenticated user.
CLI-DET-007	**MUST**	The Triggers & Symptoms tab shall display frequency bar charts for all triggers and symptoms logged in the past 30 days, sorted by frequency descending.	Bars sorted correctly. Bar length proportional to count. Count matches trigger_logs and symptom_logs.

5.6 Population Trends

ID	Priority	Requirement	Acceptance Criteria
CLI-TREND-001	**MUST**	The Population Trends view shall display three 30-day line charts rendered from daily population_snapshots: average mood, average sleep hours, and medication adherence percentage.	Charts render from population_snapshots. Missing snapshots shown as gaps.

ID	Priority	Requirement	Acceptance Criteria
CLI-TREND-002	**SHOULD**	A stacked bar chart shall show weekly risk level distribution (critical/high/moderate/low) across the past 12 weeks, enabling visual identification of caseload risk trends.	Stacked bars sum to total patient count for each week. Colours match risk level conventions.
CLI-TREND-003	**SHOULD**	The top 6 triggers across the population for the past 30 days shall be displayed as a horizontal bar chart with patient counts.	Bars match trigger_logs data for 30-day window. Sorted by count descending.

5.7 Reports

ID	Priority	Requirement	Acceptance Criteria
CLI-RPT-001	**MUST**	Clinicians shall be able to generate an individual patient PDF report covering a configurable date range. The report shall include: patient summary, mood chart, sleep chart, trigger frequency, symptom frequency, medication adherence, and clinician notes.	PDF generated and downloadable. Content accurate for selected date range. clinical_reports record created with status = "ready".
CLI-RPT-002	**SHOULD**	Clinicians shall be able to generate a population summary report covering their full caseload with aggregate statistics.	Report reflects population_snapshots data. Downloadable as PDF.
CLI-RPT-003	**SHOULD**	A handover report shall list all patients with active alerts, crisis status, outstanding clinical actions, and upcoming appointments, suitable for a covering clinician.	Report content matches real-time alert and appointment data at time of generation.

6. Backend API

6.1 API Design Principles

- All endpoints follow RESTful conventions: resources are nouns, HTTP methods express intent (GET, POST, PUT, PATCH, DELETE).
- All request and response bodies are JSON (application/json). Dates are ISO 8601 (YYYY-MM-DD). Timestamps are ISO 8601 UTC.
- All write operations validate input using Zod schemas before touching the database.
- All responses include a request_id in the response headers for tracing.
- Errors follow RFC 9457 (Problem Details for HTTP APIs): { type, title, status, detail, instance }.
- Pagination uses cursor-based pagination (after/before + limit) not offset pagination.
- API versioning is path-based: /api/v1/. Breaking changes require a new version prefix.

6.2 Authentication and Authorisation

ID	Priority	Requirement	Acceptance Criteria
API-AUTH-001	**MUST**	The API shall issue JWTs on successful authentication. Tokens shall include: sub (user ID), role (patient | clinician | admin), org_id, and exp (15-minute expiry for access tokens).	JWT claims verified. Expired token returns 401. Wrong role returns 403.
API-AUTH-002	**MUST**	Refresh tokens shall have a 7-day lifetime and be stored as HttpOnly, Secure, SameSite=Strict cookies. Refresh token rotation shall be implemented: each refresh issues a new refresh token and invalidates the old one.	Refresh token rotation verified. Old token returns 401 after rotation. Cookie flags correct.
API-AUTH-003	**MUST**	Every database connection made by the API shall set the PostgreSQL session variables app.clinician_id or app.patient_id before executing any query, enabling RLS enforcement.	Connection pooler verified to set session vars. Tested by querying with mismatched IDs.
API-AUTH-004	**MUST**	All authenticated endpoints shall write to the audit_log on every read or mutation of PHI. The audit record shall include actor_id, action, resource_type, resource_id, patient_id, and ip_address.	Audit log populated for all tested PHI endpoints. Missing audit entries treated as a critical defect.

6.3 Core API Endpoints

Endpoint	Description
POST /api/v1/auth/login	Authenticate user (patient or clinician). Returns access + refresh tokens.
POST /api/v1/auth/refresh	Exchange refresh token for new token pair.
DELETE /api/v1/auth/logout	Invalidate refresh token.
GET /api/v1/patients	List patients on clinician's care team. Supports filter and sort params.
GET /api/v1/patients/:id	Full patient record with today's entry, active alerts, and care team.
GET /api/v1/patients/:id/entries	Paginated daily entries for a patient. Clinician-accessible.
POST /api/v1/entries	Create or update today's entry for the authenticated patient. Idempotent per patient per day.

Endpoint	Description
GET /api/v1/entries/today	Return the authenticated patient's current day entry (creates empty draft if none exists).
POST /api/v1/wellness-logs	Upsert a wellness strategy log row. Patient-only.
POST /api/v1/trigger-logs	Upsert a trigger log row. Patient-only.
POST /api/v1/symptom-logs	Upsert a symptom log row. Triggers safety event detection if is_safety_symptom = TRUE.
PUT /api/v1/journal/:date	Create or update journal entry for given date. Patient-only.
GET /api/v1/alerts	List clinical alerts for clinician's patients. Filter by severity, status, date.
PATCH /api/v1/alerts/:id/acknowledge	Acknowledge an alert. Sets acknowledged_at and acknowledged_by.
POST /api/v1/notes	Create a clinician note for a patient.
GET /api/v1/dashboard/overview	Population KPIs from population_snapshots. Clinician-only.
GET /api/v1/dashboard/heatmap	Mood heatmap data from v_mood_heatmap_30d. Clinician-only.
POST /api/v1/reports	Enqueue a report generation job. Returns report ID.
GET /api/v1/reports/:id	Poll report status. Returns signed URL when status = ready.
GET /api/v1/patients/:id/insights	Correlation data from patient_correlation_cache.
WS /api/v1/ws	WebSocket endpoint for real-time alert delivery to clinician dashboard.

6.4 Rules Engine Requirements

The Rules Engine is a background worker (BullMQ) that evaluates patient data against predefined thresholds and inserts ClinicalAlert records. Version 1.0 implements the following rules:

Rule ID	Name	Logic
RULE-001	Missed Check-In	Fire WARNING after 3 consecutive days without a submitted entry; escalate to CRITICAL after 5 days. Suppress if patient status = "inactive" or "discharged".
RULE-002	Mood Decline	Fire WARNING when 7-day average mood drops ≥ 2.5 points below the patient's 28-day baseline. Fire CRITICAL when drop ≥ 3.5 points. Requires minimum 5 data points in each window.
RULE-003	Safety Flag	Handled in DB trigger (handle_safety_symptom). Rules Engine re-evaluates 24 hours later; if unresolved, escalates alert and notifies supervisor clinician if configured.
RULE-004	Medication Non-Adherence	Fire INFO when 1 dose missed in 7 days. Fire WARNING when 3 or more doses missed in 7 days. Per medication.
RULE-005	Trigger Escalation	Fire WARNING when a single trigger has severity ≥ 7 for 3 consecutive logged days.
RULE-006	Symptom Emergence	Fire INFO when a symptom that was N/A for the previous 14 days is logged as present for the first time.

Rule ID	Name	Logic
RULE-007	Streak Broken	Fire INFO when a tracking streak ≥ 7 days is broken. No alert for streaks under 7 days.
RULE-008	Risk Level Change	Fire WARNING when a clinician manually changes a patient's risk_level upward (e.g. moderate → high).

NOTE All threshold values (e.g. 2.5 point mood decline, 3 consecutive days) must be reviewed and confirmed by a clinical advisor before any clinical pilot deployment. These are engineering starting points, not clinically validated thresholds.

6.5 Notification Delivery Requirements

ID	Priority	Requirement	Acceptance Criteria
API-NOTIF-001	**MUST**	A critical-severity alert arising from a safety_event shall be delivered to the patient's primary clinician via in-app WebSocket, email, and SMS within 5 minutes of the safety_event being created.	Delivery latency measured end-to-end in test environment. All three channels deliver within 5-minute SLA.
API-NOTIF-002	**MUST**	Patient daily reminder push notifications shall be sent at the time configured in patient_notification_preferences.daily_reminder_time in the patient's timezone.	Notification delivered within 5 minutes of scheduled time. Timezone offset correct for AU/NZ timezones.
API-NOTIF-003	**MUST**	Failed notification delivery attempts shall be retried with exponential back-off (max 3 retries). Persistent failures shall be logged to the audit_log and surfaced in an operations dashboard.	Retry logic verified by mocking FCM failure. Failure logged after 3 attempts.

7. Non-Functional Requirements

7.1 Performance

ID	Priority	Requirement	Acceptance Criteria
NFR-PERF-001	**MUST**	The clinician dashboard population overview (KPI cards, alert list) shall load within 2 seconds on a standard broadband connection (50 Mbps), measured at the 95th percentile.	Lighthouse Performance score ≥ 85. API response time for /dashboard/overview ≤ 400ms at p95.
NFR-PERF-002	**MUST**	The mood heatmap shall render for a 24-patient, 30-day dataset within 1 second of data receipt.	Heatmap render time measured in browser performance trace. ≤ 1000ms from data arrival.
NFR-PERF-003	**MUST**	The patient mobile app shall launch to the Today screen within 3 seconds on a reference mid-range device (iPhone 12 / Pixel 5).	Time to interactive measured with React Native Performance Monitor. ≤ 3000ms cold start.
NFR-PERF-004	**MUST**	API endpoints serving patient data shall respond within 500ms at p95 under normal load (100 concurrent users).	k6 load test at 100 VUs confirms p95 ≤ 500ms for all core endpoints.
NFR-PERF-005	**SHOULD**	Nightly population snapshot computation shall complete within 30 minutes for a caseload of up to 500 patients per organisation.	Snapshot job execution time logged. ≤ 30 minutes for 500 patient dataset.

7.2 Scalability

ID	Priority	Requirement	Acceptance Criteria
NFR-SCALE-001	**MUST**	The system architecture shall support horizontal scaling of the API layer via stateless service instances behind a load balancer.	API deployed as at least 2 instances. Requests correctly distributed. No session state in API process.
NFR-SCALE-002	**SHOULD**	The audit_log table shall be implemented as a range-partitioned table, partitioned by month, to prevent unbounded table growth impacting query performance.	Partition creation verified in DDL. Query against single month does not scan other partitions (EXPLAIN ANALYZE).
NFR-SCALE-003	**SHOULD**	The system shall support at least 50 simultaneous WebSocket connections per clinician server instance without degraded performance.	Load test with 50 concurrent WS clients. No dropped connections. Alert delivery ≤ 10 seconds.

7.3 Availability and Reliability

ID	Priority	Requirement	Acceptance Criteria
NFR-AVAIL-001	**MUST**	The system shall target 99.5% monthly uptime, excluding scheduled maintenance windows announced 48 hours in advance.	Uptime calculated from synthetic monitoring. Incidents logged and postmortems published.

ID	Priority	Requirement	Acceptance Criteria
NFR-AVAIL-002	**MUST**	The database shall be configured with automated daily backups retained for 30 days, and point-in-time recovery enabled with a minimum RPO of 1 hour.	Backup policy confirmed in cloud provider console. Restore drill conducted quarterly.
NFR-AVAIL-003	**MUST**	The safety event detection path (DB trigger → alert creation → notification dispatch) shall be architected with no single point of failure. If the notification dispatch service is unavailable, the alert shall remain in the database and be retried on recovery.	Notification service failure simulated. Alert remains in DB. Delivery on service recovery confirmed.

7.4 Security

ID	Priority	Requirement	Acceptance Criteria
NFR-SEC-001	**MUST**	All data in transit shall be encrypted using TLS 1.3. TLS 1.0 and 1.1 shall be disabled. TLS 1.2 may be supported for compatibility with older devices.	SSL Labs scan score A or higher. TLS 1.0/1.1 confirmed disabled.
NFR-SEC-002	**MUST**	All data at rest in the database and file storage shall be encrypted using AES-256.	Cloud provider encryption configuration confirmed. Storage encryption verified.
NFR-SEC-003	**MUST**	The API shall implement rate limiting: 100 requests per minute per authenticated user, 10 requests per minute per IP for unauthenticated endpoints.	Rate limit headers present (X-RateLimit-*). 429 returned on breach. Limits verified in test.
NFR-SEC-004	**MUST**	The application shall pass an OWASP Top 10 vulnerability assessment with no Critical or High findings before any clinical deployment.	Third-party penetration test report with no Critical/High findings. Findings remediated and retested.
NFR-SEC-005	**MUST**	All API inputs shall be validated and sanitised server-side. SQL injection and XSS attacks shall be mitigated at the application layer, in addition to parameterised queries.	OWASP ASVS V5 input validation tests pass. Parameterised queries verified in code review.
NFR-SEC-006	**MUST**	The mobile application shall not log PHI to device logs, crash reporting services, or analytics platforms. All event tracking shall use anonymised patient identifiers.	Code review confirms no PHI in logging calls. Crash report payload inspected to confirm.
NFR-SEC-007	**MUST**	Clinician passwords shall be hashed using bcrypt (cost factor ≥ 12) or Argon2id. Plaintext passwords shall never be stored or logged.	Password hash confirmed bcrypt/Argon2id. Log search confirms no plaintext passwords in any log output.
NFR-SEC-008	**SHOULD**	The system shall implement a Content Security Policy (CSP) header on the clinician web dashboard to mitigate XSS risks.	CSP header present. Policy evaluated with CSP Evaluator (no "high severity" warnings).

7.5 Privacy and Compliance

ID	Priority	Requirement	Acceptance Criteria
NFR-PRIV-001	**MUST**	The system shall implement the Australian Privacy Principles (APPs) as a baseline for Version 1.0. This includes: purpose limitation, data minimisation, patient access rights, and correction rights.	Privacy impact assessment completed. Data flows mapped

ID	Priority	Requirement	Acceptance Criteria
			against APPs. Legal review sign-off.
NFR-PRIV-002	**MUST**	Patients shall be able to download all their personal data in a machine-readable format (JSON) via the mobile app. The export shall be processed within 24 hours of request.	Export includes all patient records across all tables. Delivered within 24h. Tested with data_export_requests.
NFR-PRIV-003	**MUST**	Patients shall be able to request account deletion. A deletion request shall anonymise all PII fields (name, email, DOB) and delete journal entry bodies within 30 days, while retaining anonymised clinical records for mandatory retention periods.	Deletion process documented. PII fields anonymised within 30 days. Journal bodies deleted. Audit records retained.
NFR-PRIV-004	**MUST**	Consent records shall be maintained for every instance of data sharing, including the exact version of the consent text shown to the patient, the timestamp, and the IP address.	consent_records table contains all required fields. Consent text snapshot stored verbatim.
NFR-PRIV-005	**SHOULD**	The architecture shall be designed to support GDPR compliance, enabling EU deployment in a subsequent version without architectural changes. This includes: data residency configuration, right to erasure, and data portability.	Architecture review confirms GDPR pathway. No structural blockers identified.

7.6 Accessibility

ID	Priority	Requirement	Acceptance Criteria
NFR-ACC-001	**MUST**	The patient mobile application shall conform to WCAG 2.1 Level AA. All interactive elements shall have accessible labels. Colour shall not be the sole means of conveying information.	Accessibility audit with VoiceOver (iOS) and TalkBack (Android). No critical or serious issues.
NFR-ACC-002	**MUST**	The clinician dashboard shall conform to WCAG 2.1 Level AA. Charts shall provide text alternatives (data tables or accessible descriptions).	Lighthouse Accessibility score ≥ 90. Manual audit with screen reader confirms chart data accessible.
NFR-ACC-003	**MUST**	All mood and severity rating components (mood ring, sliders) shall be operable via keyboard on the web dashboard and via assistive technologies on mobile. Colour coding shall always be paired with numeric labels.	Mood ring navigable by keyboard (Tab + Enter). Slider value announced by screen reader.
NFR-ACC-004	**SHOULD**	Text size throughout the mobile application shall respect the user's system font size settings (Dynamic Type on iOS, font scale on Android).	Tested at maximum system font size. No text overflow or clipping. Layout remains functional.

7.7 Maintainability and Developer Experience

ID	Priority	Requirement	Acceptance Criteria
NFR-MAINT-001	**MUST**	All code shall be written in TypeScript with strict mode enabled. No use of "any" type without explicit justification in a code comment.	TypeScript strict mode configured in tsconfig.json. CI fails on type errors.

ID	Priority	Requirement	Acceptance Criteria
NFR-MAINT-002	**MUST**	Database migrations shall be managed with a versioned migration tool (Flyway or node-pg-migrate). Migrations shall be additive only; destructive schema changes require a multi-step migration plan.	Migration history table present in DB. CI runs migrations against test database. Rollback procedure documented.
NFR-MAINT-003	**MUST**	Unit test coverage shall be maintained at ≥ 80% for all business logic modules, with 100% coverage required for the Rules Engine and safety event detection paths.	Coverage report generated in CI. Rules Engine coverage verified at 100% on each build.
NFR-MAINT-004	**MUST**	All API endpoints shall have integration tests covering: happy path, validation errors, authentication failures, and authorisation failures.	Test suite covers all four scenarios per endpoint. CI fails on any missing scenario.
NFR-MAINT-005	**SHOULD**	All API endpoints shall be documented with OpenAPI 3.1 specifications, auto-generated from route/schema definitions and served at /api/v1/docs.	OpenAPI spec accessible at /api/v1/docs. Spec validated with swagger-cli.

8. Clinical Safety Requirements

CRITICAL This section contains requirements that directly affect patient safety. All requirements in this section are classified as MUST. Any failure to implement, test, or maintain these requirements constitutes a patient safety incident. Clinical advisor sign-off is required before deploying changes to any component in this section.

ID	Priority	Requirement	Acceptance Criteria
SAF-001	**MUST**	Safety symptom detection, safety_event creation, ClinicalAlert creation, and patient status update shall occur within a single ACID-compliant database transaction. If any step fails, all steps shall be rolled back.	Transaction isolation confirmed in code review. Failure injection test: DB error mid-transaction leaves no partial records.
SAF-002	**MUST**	The safety resource card displayed to a patient on safety symptom activation shall contain the current Lifeline Australia contact (13 11 14) and the Crisis Text Line contact. These contacts shall be maintained and verified quarterly.	Resource card text confirmed accurate at each quarterly review. Review documented.
SAF-003	**MUST**	The app shall never claim to provide clinical diagnosis, treatment recommendations, or crisis counselling. All AI-generated insights shall carry a disclaimer stating they are for informational purposes only and not a substitute for professional clinical assessment.	Disclaimer present on all AI insight cards. Legal and clinical review of disclaimer text.
SAF-004	**MUST**	Critical clinical alerts (safety events and mood crisis thresholds) shall be delivered to at least one human clinician within 5 minutes via at least two delivery channels. If the primary clinician is unreachable after 15 minutes (no acknowledgement), the alert shall automatically escalate to a configured backup clinician.	Escalation path tested end-to-end. Primary non-acknowledgement triggers escalation at 15 minutes. Backup clinician receives alert.
SAF-005	**MUST**	The system shall maintain a complete, immutable record of every safety_event including: when it was created, which alerts were raised, who acknowledged them, when, and what action was taken. This record shall be retained for a minimum of 7 years in accordance with clinical record-keeping obligations.	Safety event audit trail complete in DB. Retention policy configured. 7-year retention confirmed.
SAF-006	**MUST**	The UI/UX response to a safety symptom activation in the patient app shall be validated by a registered mental health professional before deployment. The response must not increase patient distress or shame.	Written sign-off from clinical advisor on file. User testing with clinical supervisor present.
SAF-007	**MUST**	Alert rules and thresholds shall be reviewed by a qualified clinician before any change is deployed to production. A clinical change review process shall be documented and followed.	Clinical change review process documented. Clinician sign-off recorded for each threshold change.
SAF-008	**MUST**	The system shall include a mechanism for clinicians to flag a rule as producing excessive false positives (alert fatigue). Flagged rules shall trigger a clinical review within 14 days.	Feedback mechanism present in dashboard. Rule feedback stored. Review workflow documented.

9. Data Model Summary

The complete DDL is maintained separately (mindlog-schema.sql v1.0). The table below summarises each entity for requirements traceability.

Table	Domain	Purpose
organisations	Identity	Top-level tenant. All data is scoped to an organisation.
clinicians	Identity	Licensed practitioners. Authentication, professional details, MFA config.
patients	Identity	Patient identity, status, risk level, engagement metrics.
icd10_codes	Clinical Setup	ICD-10-AM reference. Loaded from standard data feed.
patient_diagnoses	Clinical Setup	Patient diagnoses; one or more per patient; supports primary flag.
medications_catalogue	Clinical Setup	Reference medication library. Fuzzy-searchable.
patient_medications	Clinical Setup	Prescribed medications per patient; drives daily medicine card.
care_team_members	Clinical Setup	Clinician-to-patient assignments with role and date range.
daily_entries	Daily Logging	Parent record per patient per day. Tracks completion state.
sleep_logs	Daily Logging	Sleep hours + quality. Child of daily_entries.
exercise_logs	Daily Logging	Exercise duration. Child of daily_entries.
medication_adherence_logs	Daily Logging	Per-medication taken/not taken. Child of daily_entries.
wellness_strategies	Wellness	Strategy catalogue (system + patient custom).
patient_wellness_strategies	Wellness	Patient's active strategy list.
wellness_logs	Wellness	Daily wellness state (yes/no/na) + quality. Child of daily_entries.
trigger_catalogue	Triggers	Trigger catalogue (system + patient custom).
patient_triggers	Triggers	Patient's active trigger list.
trigger_logs	Triggers	Daily active/inactive + severity. Child of daily_entries.
symptom_catalogue	Symptoms	Symptom catalogue with is_safety_symptom flag.
patient_symptoms	Symptoms	Patient's active symptom list.
symptom_logs	Symptoms	Daily present/absent + intensity.
safety_events	Safety	High-sensitivity. Created atomically with safety symptom log.
journal_prompts	Journal	Daily prompt catalogue. Org-specific or global.
journal_entries	Journal	Freeform text. Private by default. Consent-gated sharing.
clinical_alerts	Alerts	Rules-generated alerts with full lifecycle and acknowledgement.
alert_routing_rules	Alerts	Which clinicians receive which alerts via which channels.
patient_notification_preferences	Notifications	Patient push/reminder settings.
clinician_notes	Clinical Workflow	Structured notes; linked to patients and optional dates.
appointments	Clinical Workflow	Scheduled appointments with status lifecycle.
clinical_reports	Clinical Workflow	Generated report jobs with signed URL storage.

Table	Domain	Purpose
population_snapshots	Analytics	Nightly pre-aggregated KPIs per clinician and organisation.
patient_correlation_cache	Analytics	Mood correlation factors per patient. Computed weekly.
audit_log	Governance	Immutable append-only access and mutation log.
consent_records	Governance	Full consent lifecycle with text snapshots.
data_export_requests	Governance	Patient data portability requests.

10. Testing Strategy

10.1 Testing Levels

Level	Tool	Scope
Unit Tests	Jest (API/Rules Engine), Jest + React Testing Library (Dashboard)	All business logic functions, Zod validators, Rules Engine rule evaluations, React component rendering.
Integration Tests	Supertest (API), Playwright (Dashboard)	All API endpoints: happy path, validation, auth. Database trigger behaviour. RLS policy enforcement.
End-to-End Tests	Detox (Mobile), Playwright (Dashboard)	Critical user journeys: daily check-in, safety symptom activation, alert acknowledgement, report generation.
Performance Tests	k6	Load test at 100 VUs. Sustained 10-minute test. P95 latency against NFR thresholds.
Security Tests	OWASP ZAP, manual penetration test	OWASP Top 10. Authentication bypass. Authorisation escalation. Injection attacks.
Accessibility Tests	axe-core (automated), manual screen reader	WCAG 2.1 AA conformance on all screens. Keyboard navigation. Dynamic font sizes.
Clinical Safety Tests	Manual test scripts	Safety event end-to-end: submission → alert → escalation → acknowledgement → audit trail.

10.2 Critical Test Scenarios

Safety Event End-to-End

14. Patient activates "Suicidal Thoughts" symptom and submits entry.
15. Verify: symptom_logs row created with is_present = TRUE.
16. Verify: safety_events row created within 1 second.
17. Verify: clinical_alerts row with severity = "critical" created within 1 second.
18. Verify: patients.status = "crisis" within 1 second.
19. Verify: primary clinician receives WebSocket push within 10 seconds.
20. Verify: primary clinician receives email and SMS within 5 minutes.
21. Simulate no acknowledgement for 15 minutes.
22. Verify: escalation alert sent to backup clinician.
23. Clinician acknowledges alert. Verify: audit_log record created.

RLS Boundary Test

24. Authenticate as Clinician A (caseload: Patients 1–10).
25. Attempt to fetch patient data for Patient 11 (on Clinician B's caseload only).
26. Verify: API returns 403. No data returned. Audit log records the attempted access.

Offline Sync Test

27. Enable airplane mode on test device.
28. Complete a full daily entry (all sections).
29. Force-quit and reopen the app. Verify: draft state fully preserved.
30. Re-enable network. Verify: entry synced to server within 30 seconds.
31. Verify: server database contains correct entry data.

11. Regulatory Considerations

11.1 Software as a Medical Device (SaMD) Classification

MindLog captures symptoms including suicidal ideation and generates alerts intended to influence clinical decision-making. This is likely to trigger SaMD classification under the TGA's framework for Software as a Medical Device. Before any clinical deployment, a formal regulatory assessment must be conducted to determine:

- Whether the software meets the definition of a medical device under the Therapeutic Goods Act 1989 (Cth).
- The applicable risk classification (likely Class IIa or IIb given the intended use in mental health monitoring).
- Whether ARTG inclusion is required before clinical use.
- The applicable conformity assessment pathway (Self-Assessment, Third Party).

REGULATORY This SRS does not constitute a regulatory submission. Regulatory classification must be determined by a qualified regulatory affairs consultant before any patient-facing deployment, including clinical pilots.

11.2 Privacy Act 1988 (Cth) Obligations

The following Australian Privacy Principles are directly relevant to this system's design:

Principle	System Implication
APP 1 — Open and Transparent Management	A current Privacy Policy must be published and accessible from both the mobile app and web dashboard. The policy must accurately describe data collection, use, storage, and disclosure practices.
APP 3 — Collection of Solicited Personal Information	Only information reasonably necessary for the primary function (mental health monitoring and clinical support) may be collected. Each data field in the schema should have a documented clinical justification.
APP 5 — Notification	Patients must be notified at collection time of: what is being collected, why, who it will be disclosed to, and how to access or correct it.
APP 6 — Use or Disclosure	PHI may only be disclosed to clinicians with the patient's consent (consent_records) or where required by law (e.g. mandatory reporting obligations).
APP 11 — Security	Reasonable steps must be taken to protect PHI from misuse, interference, loss, unauthorised access, modification, or disclosure. The security NFRs in Section 7.4 are the technical implementation of this obligation.
APP 12 — Access	Patients must be able to access their personal information on request. The data export feature (data_export_requests) implements this right.
APP 13 — Correction	Patients must be able to request correction of inaccurate personal information. A correction request workflow must be documented.

11.3 Mandatory Reporting Obligations

Clinicians using MindLog remain subject to mandatory reporting obligations under applicable law (e.g. mandatory reporting of child abuse under the Children and Young Persons (Care and Protection) Act 1998 (NSW) and equivalent legislation). The system is not responsible for determining or discharging these obligations. However, the system must not impede a clinician's ability to access and act on relevant patient information in a timely manner.

11.4 Clinical Record Retention

Safety events, clinical notes, alert acknowledgements, and daily entries constitute clinical records. These must be retained for a minimum of 7 years from the date of last entry (longer for records relating to patients who were minors). The data retention policy must be documented, automated where possible, and reviewed annually.

12. Open Questions and Decisions Required

The following decisions must be made before development of the relevant components begins. Each is assigned an owner and target decision date.

ID	Question	Detail and Impact
OQ-001	Journal Encryption Model	Will journal entries be server-side encrypted (searchable, shareable with consent) or end-to-end encrypted (private by construction, no server-side search)? E2EE requires client-side key management. Affects: journal_entries schema, search feature, sharing consent flow.
OQ-002	Regulatory Classification	What is the TGA SaMD classification for MindLog? Does it require ARTG inclusion before any clinical pilot? Affects: timeline, quality management system requirements, labelling.
OQ-003	Real-Time Architecture	Will the dashboard use WebSockets (stateful, requires sticky sessions or Redis pub/sub) or Server-Sent Events (stateless, simpler, unidirectional)? Affects infrastructure complexity and horizontal scaling approach.
OQ-004	Alert Threshold Values	What are the clinically validated threshold values for each alert rule? Specifically: mood decline delta, consecutive missed days for escalation, trigger severity escalation days. Requires clinical advisor input.
OQ-005	Minor Patient Policy	What is the minimum patient age? Are parental consent flows required for patients under 18? What data access rights do parents/guardians have vs. the minor patient? Affects consent model and onboarding.
OQ-006	FHIR Export Scope	What is the target FHIR resource set for the future export feature? Minimum viable set likely: Patient, Observation (mood/sleep), Condition (diagnosis), MedicationStatement. Affects schema design constraints.
OQ-007	AI Insight Generation	Will the patient insight cards and clinician alert summaries use an on-premise ML model, a hosted LLM API (e.g. Anthropic), or rule-based heuristics for V1.0? Affects privacy risk profile, latency, and cost.
OQ-008	Biometric Auth for Clinician Dashboard	Will the web dashboard support biometric / passkey authentication (WebAuthn) in addition to TOTP MFA? Affects auth library selection.
OQ-009	Multi-Clinician Journalling Visibility	If a patient is on a care team with multiple clinicians, and they share a journal entry, is it visible to all care team members or only their primary clinician? Affects consent_records data model.
OQ-010	Population Snapshot Granularity	Should population_snapshots be generated per organisation only, or also per individual clinician's caseload? Per-clinician snapshots are faster for the dashboard but increase storage. Decision affects job design.

13. Implementation Roadmap

The following phasing is recommended based on dependency order and risk priority. Each milestone should conclude with a documented internal review before the next begins.

Phase	Name	Key Deliverables
Phase 0 2–3 weeks	Foundations	Resolve all Open Questions (Section 12). Establish version control, CI/CD pipeline, and IaC. Provision managed PostgreSQL. Run DDL migrations. Configure auth provider (Supabase or Auth0). Establish coding standards, linting, and commit hooks.
Phase 1 4–6 weeks	Backend API Core	Authentication endpoints. Patient and clinician CRUD. Daily entry submission (all child tables). Wellness, trigger, symptom log endpoints. Safety event detection (DB trigger + API validation). Basic alert creation. Audit logging. Unit + integration tests at 80%+ coverage.
Phase 2 4–6 weeks	Patient Mobile App	React Native (Expo) project setup. Onboarding flow (5 screens). Today screen (mood, coping, sleep, exercise, medications). Wellness, Triggers, Symptoms screens. Journal. Offline sync with WatermelonDB. Push notification registration. Basic Insights screen (charts from live data).
Phase 3 4–6 weeks	Clinician Dashboard	React + Vite project setup. Authentication (MFA). Population Overview (KPI cards, heatmap, alert strip). Alerts view with acknowledgement workflow. Patient list with filters. Patient detail (all 5 tabs). WebSocket real-time alert delivery.
Phase 4 3–4 weeks	Rules Engine + Notifications	BullMQ worker setup. All 8 alert rules (RULE-001 through RULE-008). Nightly population snapshot job. Weekly correlation cache computation. Email delivery (Resend). SMS delivery for critical alerts (Twilio). Escalation workflow.
Phase 5 2–3 weeks	Reports + Analytics	PDF report generation (individual, population, handover). Population Trends charts. Correlation cache patient insights. Data export (JSON/CSV/PDF).
Phase 6 3–4 weeks	Clinical Validation	Clinical advisor review of all safety flows. Usability testing with 3–5 clinicians. Usability testing with 3–5 patients (supervised). Penetration test. Accessibility audit. Performance testing. Regulatory assessment.
Phase 7 2–3 weeks	Pilot Deployment	Limited clinical pilot with 1 organisation, ≤ 5 clinicians, ≤ 20 patients. Daily monitoring. Incident response process active. Feedback collection. Retrospective and V1.1 planning.

NOTE Phase 6 (Clinical Validation) is not optional and must not be compressed. No production deployment should occur until an independent penetration test, clinical safety review, and regulatory assessment have been completed and their findings addressed.

Appendix A — Revision History

Version	Change Description
Version	Change
1.0 DRAFT — February 2026	Initial version. Covers patient mobile app, clinician dashboard, backend API, database model, safety requirements, privacy obligations, and implementation roadmap.

Appendix B — Requirement Traceability

The following table maps each requirement category to its primary design artefact, test type, and database entity.

Req. Group	Design Artefact	Test Evidence
PAT-AUTH	Onboarding prototype screens OB-1 through OB-5	Integration test: POST /auth/login
PAT-ENTRY	Today screen prototype	E2E test: daily check-in user journey
PAT-WELL	Wellness screen prototype	Integration test: POST /wellness-logs
PAT-TRIG	Triggers screen prototype	Integration test: POST /trigger-logs
PAT-SYM	Symptoms screen prototype	Integration + safety E2E test: SAF-001 scenario
PAT-JOUR	Journal screen prototype	Integration test: PUT /journal/:date
PAT-INS	Insights screen prototype	Integration test: GET /patients/:id/insights
CLI-AUTH	Clinician dashboard auth flow	Integration test: MFA enforcement
CLI-POP	Population overview prototype	Integration test: GET /dashboard/overview
CLI-ALERT	Alerts view prototype	E2E test: alert acknowledgement journey
CLI-LIST	Patient list prototype	Integration test: GET /patients with filters
CLI-DET	Patient detail prototype (5 tabs)	Integration test: GET /patients/:id/entries
API-AUTH	API architecture — Section 3	Integration: JWT, RLS, audit log
NFR-SEC	Security architecture	Penetration test, OWASP ZAP
NFR-PRIV	Privacy impact assessment	Legal review, consent flow testing

Req. Group	Design Artefact	Test Evidence
SAF-*	Safety requirements — Section 8	Safety E2E test + clinical advisor sign-off