Hospital Management System (HMS)

Executive Summary

The Hospital Management System is the healthcare vertical of the Operations platform — an integrated business suite that combines clinical workflows with inventory, accounting, HR, and CRM in a single product. In the application, the HMS appears as **Hospital** in the sidebar (technical module key: `clinic`). Alongside it, the **Pharmacy** module provides a full dispensary operations centre. Together they support the complete patient journey: registration and scheduling, clinical documentation, laboratory, inpatient care, surgery planning, prescribing, dispensing, billing, and insurance claims.

Unlike standalone clinic software that requires separate stock and finance systems, Operations HMS is built on shared platform foundations. Patients are customer records. Doctors are user accounts. Prescriptions and dispenses are invoice documents. Medicines come from the same inventory catalogue used across the organisation. Payments post to the general ledger automatically. This unified architecture reduces data duplication, improves auditability, and lowers total cost of ownership for hospitals, clinics, surgery centres, dental practices, and multi-specialty facilities.

The platform adapts to your facility type. A small outpatient clinic receives appointment, EMR, lab, and billing features without the complexity of wards and operating theatres. A full hospital adds departments, ward management, and inpatient admissions. A surgery centre enables operating room scheduling and surgical booking workflows. Pharmacy can run as a hospital dispensary, retail outlet, or standalone module — always connected to the same patient and product data.

Who This Is For

Role	How they use HMS
Hospital administrators	Configure facility type, activate modules, manage master data, review dashboards and reports
Reception and front desk	Register patients, book appointments, manage waitlists, check patients in
Doctors and clinicians	Work daily queues, document encounters (SOAP notes), diagnose, prescribe, order labs
Nurses and ward staff	Manage admissions, ward and bed occupancy, inpatient care (planned: nursing station)
Laboratory technicians	Receive orders, track samples, enter and verify results
Theatre coordinators	Schedule operating rooms, book surgeries, assign surgeons and anaesthetists
Billing and finance	Raise patient invoices, process prescription payments, reconcile dispensary sales
Insurance coordinators	Submit and track medical aid claims linked to encounters and invoices

Role	How they use HMS
Pharmacists	Verify prescriptions, dispense medicines, manage refills, print labels
Dispensary assistants	Process walk-in sales, manage the dispense queue, collect payments
Pharmacy managers	Monitor pharmacy dashboard, review compliance reports, oversee stock alerts

---

Platform Value Proposition

One system, not five

Traditional approach	Operations HMS
Separate patient registry, EMR, lab, pharmacy, and accounting systems	Single platform with shared patient, product, and financial data
Manual re-keying of prescriptions into pharmacy software	Clinic prescriptions flow to pharmacy patient search and dispensing
Stock levels updated only in pharmacy, not finance	FEFO stock deduction on every dispense, reflected in inventory and accounting
Doctors exist in HR; patients exist in CRM; no link	Doctors (`User`) and patients (`Customer`) are first-class platform entities
Insurance claims disconnected from billing	Healthcare insurance claims link directly to invoice documents

Facility-type flexibility

Organisations choose a **healthcare facility type** during setup. The type controls which menu items and features are available — so a dental practice is not overwhelmed by ward management screens, and a surgery centre sees operating room tools by default.

Facility type	Label	Key differentiators
`clinic`	Clinic	Outpatient focus: appointments, EMR, lab, billing, teleconsult, consent, scheduling
`hospital`	Hospital	Full inpatient: departments, wards, admissions, plus all clinic features
`surgery_center`	Surgery Centre	Hospital features plus **operating rooms**; teleconsult optional
`dental`	Dental Practice	Appointments, EMR, billing, consent, scheduling, departments
`multi_specialty`	Multi-Specialty	Full hospital capabilities including operating rooms

The facility type is stored in organisation settings (`healthcare_facility_type` in organisation custom arguments) and drives feature visibility in the user interface.

Module dependencies

Module	Depends on	Purpose
Hospital (`clinic`)	Accounting, Inventory	Clinical workflows, billing, lab products
Pharmacy	Accounting, Inventory	Dispensing, refills, stock deduction

Both modules are activated per organisation under **Organisation** → **Modules**.

Key Capabilities at a Glance

Hospital (HMS)

- Hospital Command Center dashboard with today's schedule and period KPIs
- Patient directory with bulk import and full patient chart
- Appointment calendar (week, day, month views) with waitlist automation
- Doctor's daily queue for outpatient workflow
- Electronic Medical Records (EMR) with SOAP notes, signing, and encounter lifecycle
- Prescriptions with MCAZ drug search, dosage cyphers, PDF generation, and email
- Laboratory orders, sample tracking, and inline result entry
- Inpatient admissions with ward and bed management
- Operating room registry and surgery booking (API-ready; UI in progress)
- Teleconsultation session management
- Digital consent forms and signature capture
- Doctor schedules and staff shift rosters
- Medical aid provider (health society) management
- Healthcare insurance claims workflow
- Master data: ICD-10 codes, CPT codes, EMR templates, lab test definitions
- Background waitlist allocation job (auto-books from waitlist against free slots)

Surgery

- Operating room registry (name, capacity, equipment notes, status)
- Surgery booking with surgeon, anaesthetist, patient, OR, CPT code, and scheduling
- Pre-operative and post-operative notes
- Links to admissions, appointments, and billing invoices
- Anaesthesia type tracking
- Booking status workflow (scheduled → in progress → completed → cancelled)

Pharmacy

- Pharmacy Command Center dashboard
- Quick Dispense workstation with patient search, FEFO batch allocation, and keyboard shortcuts
- Dispense queue (table and kanban views): Dispensed → Verified → Collected → Paid
- Dispensing records — full searchable audit history
- Pending refills with overdue alerts and reschedule actions
- Mixture recipe library for compounded preparations

- Dosage cypher settings with AI-assisted seeding
- Script attachment (upload/photograph prescriber prescription)
- Five operational reports: Dispensing Summary, Product Movement, Refill Compliance, Expiry Tracking, Patient History
- PDF dispense documents and printable medicine labels
- Point-of-sale payment integration after each dispense

Important Concepts

Term	Meaning
Patient	A `Customer` record flagged as a patient (`IsPatient`), with healthcare fields (allergies, chronic conditions, medical aid, blood type, national ID)
Doctor	A `User` with clinic metadata in custom arguments (license, specialty, consultation fee, `clinic_is_doctor`)
Encounter	A clinical visit documented in the EMR — outpatient, inpatient, emergency, or teleconsult
SOAP notes	Structured clinical documentation: Subjective, Objective, Assessment, Plan
Prescription (Rx)	An `Invoice` with `mode = "prescription"` and number format `Rx-{7-digit}`
Dispense	An `Invoice` with `mode = "dispense"` and number format `DISP-*`
FEFO	First Expiry, First Out — pharmacy stock deduction uses batches with earliest expiry first
Dosage cypher	Short code (OD, BD, TDS, PRN) that expands to full text on printed labels
Health society	Medical aid provider / insurer, stored as a supplier record
Insurance claim	Healthcare claim (`ClinicInsuranceClaim`) linked to an invoice — distinct from the standalone Insurance brokerage module
Ward	Inpatient unit (general, ICU, maternity, pediatric, surgical, isolation)
Admission	Inpatient stay linking patient to ward, bed, attending doctor, and discharge
Surgery booking	Scheduled surgical procedure with assigned OR, surgeon, anaesthetist, and CPT code
Waitlist allocation	Automated background job that books waitlisted patients into free doctor slots

---

Table of Contents

Getting Started

Activating HMS for your organisation

1. Sign in as an organisation administrator.
2. Open **Organisation → Modules**.
3. Activate **Hospital** (`clinic`) and, if needed, **Pharmacy** (`pharmacy`).
4. Confirm **Accounting** and **Inventory** are also active (required dependencies).
5. Set your **healthcare facility type** in organisation settings (clinic, hospital, surgery centre, dental, or multi-specialty).

First-time setup checklist

Before going live with patients, complete these steps with your administrator:

Step	Where	Why it matters
Set facility type	Organisation settings	Controls which HMS features appear
Configure locations	Stock → Locations	Branches, wards, and stock deduction points
Add doctors	HR / Users	Assign `clinic_is_doctor` metadata; doctors appear in appointment and queue pickers
Set up departments	HR → Departments	Clinic reads departments from HR user groups
Register medical aid providers	Hospital → Administration → Master Data → Health Societies	Patients link to funders for claims
Seed ICD-10 and CPT codes	Hospital → Administration → Master Data	Diagnosis and procedure coding

Step	Where	Why it matters
Configure lab test definitions	Hospital → Administration → Master Data	Test catalog for lab orders
Set up dosage cyphers	Pharmacy → Settings	Standard frequency codes for prescribing and labels
Load medicine inventory	Stock → Inventory	Products, batches, expiry dates, and pricing
Configure payment methods	Accounting	Collect payments at prescription pay or pharmacy till
Assign roles and permissions	Organisation → Roles	Grant clinic and pharmacy permissions to staff

Opening the Hospital module

1. Sign in to Operations.
2. Open **Hospital** in the sidebar.
3. You land on the **Hospital Command Center** (`/clinic/dashboard`).

Navigation Structure

The Hospital menu is organised into six operational areas. Pharmacy has its own top-level menu.

Hospital menu

```
Hospital
├── Dashboard                          → /clinic/dashboard
├── Front Desk
│    ├── Patient Directory             → /clinic/patients
│    ├── Appointments                  → /clinic/appointments
│    ├── Waitlist                      → /clinic/appointments/waitlist
│    └── Triage                        → (planned; currently routes to
Appointments)
├── Outpatient (OPD)
│    ├── My Queue                      → /clinic/outpatient/my-queue
│    ├── Encounters                    → /clinic/emr/encounters
│    ├── Teleconsultations             → /clinic/teleconsult
│    └── Consents                      → /clinic/consent-forms
├── Inpatient (IPD)
│    ├── Admissions                    → /clinic/inpatient/admissions
│    ├── Wards & Beds                  → /clinic/inpatient/wards-beds
│    ├── Nursing Station               → (planned)
│    ├── Operating Rooms               → /clinic/surgery/operating-rooms (UI in
progress)
│    └── Surgery Bookings              → (planned; API available)
```

```
├── Laboratory
│   ├── Lab Orders                     → /clinic/lab/orders
│   ├── Sample Tracking                → /clinic/lab/sample-tracking
│   └── Results                        → (entered inline from Lab Orders)
├── Billing
│   ├── Invoices                       → /accounting/invoices
│   └── Insurance Claims               → /clinic/billing/insurance-claims
└── Administration
    ├── Doctor Schedules               → /clinic/schedule/doctors
    ├── Master Data                    → /clinic/admin
    ├── Drug Interactions              → (planned)
    └── Clinic Reports                 → (planned)
```

Pharmacy menu

```
Pharmacy
├── Dashboard              → /pharmacy/dashboard
├── Quick Dispense         → /pharmacy/dispense
├── Dispensing Records     → /pharmacy/dispensing-records
├── Pending Refills        → /pharmacy/pending-refills
├── Patients               → /accounting/customers (shared CRM)
├── Products               → /stock/inventory (shared stock)
├── Mixtures               → /pharmacy/mixtures
├── Reports                → /pharmacy/reports
└── Settings               → /pharmacy/settings/dosage-cyphers
```

Context menu quick actions

From anywhere in the application, right-click or use the context menu to:

- **Add Patient** (requires Hospital module)
- **Book Appointment** (requires Hospital module)
- **Create Lab Order** (requires Hospital module)
- **Dispense Medicine** (requires Pharmacy module)

Hospital Command Center

The Hospital Command Center (`/clinic/dashboard`) is the operational nerve centre for clinical managers and administrators.

What you see

Metric	Description
Today's appointments	Count of patients scheduled today
Completed today	Appointments marked completed

Metric	Description
Encounters (period)	EMR encounters in the selected date range
Lab orders (period)	Laboratory orders in the selected date range
Today's status breakdown	Chart of appointments by status: pending, confirmed, arrived, in progress, completed, missed, cancelled
Today's schedule table	Time-ordered list with patient name, doctor, type, status, and reason

Date range controls

Use presets (7 days, 30 days, month, quarter) to change the reporting window for period statistics. Today's schedule always reflects the current day.

Quick links

Jump directly to patients, appointments, encounters, lab orders, admissions, and reports from the dashboard banner.

API endpoint: `GET api/Dashboard/hms?from=&to=`
Permission required: `clinic:admin:view`

Front Desk

Patient Directory

Route: `/clinic/patients`

The patient directory is the central registry for everyone receiving care at your facility.

Features:

- Paginated search by name, phone, national ID, or medical aid number
- Create and edit patients in a side panel (offcanvas)
- Bulk import from spreadsheet (`Import Patients`)
- Navigate to full patient chart from any row

Patient record fields:

- Demographics: name, date of birth, gender, national ID, phone, email, address
- Clinical: blood type, allergies, chronic conditions
- Medical aid: provider (health society), membership number
- Emergency contact
- Custom clinic arguments (extensible via `clinic_*` keys)

Patients are stored as **Customer** records with `IsPatient = true`, so the same person can appear in CRM, accounting, and pharmacy without duplicate registration.

Patient Chart

Route: `/clinic/patients/:id`

The patient chart is a tabbed clinical hub:

Tab	Content
Summary	Demographics, allergies, chronic conditions, medical aid, emergency contact
Encounters	Clinical visit history with SOAP summaries
Medical History	Longitudinal history entries
Lab Results	Completed laboratory results
Admissions	Inpatient stay history
Appointments	Past and upcoming appointments
Consents	Signed consent records

From the chart you can edit demographics, open encounters, and navigate to the prescription list.

Appointments

Route: `/clinic/appointments`

Full calendar scheduling with week, day, and month views powered by `angular-calendar`.

Appointment fields:

- Patient, doctor, date and time, duration
- Appointment type (outpatient, procedure, teleconsult, etc.)
- Reason for visit
- Status workflow

Status workflow:

Status	Meaning
`pending`	Booked, not yet confirmed
`confirmed`	Patient confirmed attendance
`arrived`	Patient checked in at front desk
`in_progress`	Currently being seen
`completed`	Visit finished
`missed`	Patient did not attend
`cancelled`	Appointment cancelled

Actions: Create, edit, update status, delete.

Waitlist

Route: `/clinic/appointments/waitlist`

When no appointment slot is available, add patients to the waitlist.

Status	Meaning
`waiting`	Awaiting a slot
`notified`	Patient informed a slot is available
`booked`	Converted to a confirmed appointment
`expired`	Waitlist entry timed out

Automation: A background job (`WaitlistAllocationJob`) runs on a schedule, scanning doctor availability over a 14-day lookahead and automatically booking waitlisted patients into free slots. This reduces manual rescheduling work for front desk staff.

Outpatient (OPD)

My Queue

Route: `/clinic/outpatient/my-queue`

A doctor-specific daily view of scheduled patients. Designed as the starting point for a clinician's workday.

Features:

- Lists today's appointments for the logged-in doctor
- Status summary counts (pending, arrived, in progress, completed)
- Update appointment status inline
- Start an encounter directly from a queue item

API endpoint: `GET api/Clinic/Appointments/my-queue`

Encounters (EMR)

Route: `/clinic/emr/encounters`

The core electronic medical record for documenting clinical visits.

Encounter types:

- Outpatient
- Inpatient
- Emergency
- Teleconsult

SOAP documentation fields:

- Chief complaint

- History of present illness (HPI)
- Subjective
- Objective (examination findings)
- Assessment
- Plan

Status workflow:

Status	Meaning
`open`	Draft — being documented
`signed`	Clinician signed — locked for editing
`closed`	Visit complete

Related clinical data (per encounter):

- **Vitals** — blood pressure, pulse, temperature, respiratory rate, SpO_2, weight, height, pain level
- **Diagnoses** — ICD-10 coded conditions
- **Procedure records** — CPT coded procedures with optional billable product link
- **Encounter audits** — field-level change trail on signed encounters

EMR templates: Pre-built note structures by specialty speed documentation while maintaining consistency. Templates are managed under Administration → Master Data.

Teleconsultation

Route: `/clinic/teleconsult`

Manage virtual consultation sessions linked to appointments.

Status	Meaning
`scheduled`	Session created, awaiting start
`in_progress`	Consultation active
`completed`	Session finished
`no_show`	Patient did not join

Sessions include meeting URL, session token, recording consent flag, and audio-only option. Clinicians join via the session URL (opens in browser).

Consent Forms

Route: `/clinic/consent-forms`

Digital consent management for clinical and surgical procedures.

Features:

- Consent form templates (including surgery-specific types)

- Captured patient signatures linked to encounters or appointments
- Consent type categorisation for audit and compliance

Prescriptions

Prescriptions bridge clinical care and pharmacy dispensing. They are modelled as **invoice documents** with `mode = "prescription"` and reference numbers `Rx-{7-digit}`.

Prescription list

Route: `/clinic/patients/:patientId/prescriptions`

View all prescriptions for a patient with status, date, doctor, and line items.

Create / edit prescription

Route: `/clinic/patients/:patientId/prescriptions/new`
Route: `/clinic/patients/:patientId/prescriptions/:prescriptionId`

Features:

- MCAZ drug search for Zimbabwe-registered medicines
- Line items with product, quantity, dosage instructions
- Dosage cyphers (OD, BD, TDS, etc.) and administration route (oral, IV, IM, topical, etc.)
- Per-line instructions for pharmacy labels
- Doctor assignment
- Status workflow: Draft → Pending → On Hold → Cancelled → Paid

Outputs:

- PDF prescription document (download or print)
- Email prescription to patient
- Public download link (token-based, no login required)
- Payment collection with journal posting to accounting

API endpoints:

- `GET/POST/PUT/DELETE api/clinic/Prescriptions`
- `GET api/clinic/Prescriptions/{id}/download` — PDF
- `POST api/clinic/Prescriptions/{id}/pay` — payment + journal
- `POST api/clinic/Prescriptions/{id}/email` — email to patient
- `POST api/clinic/Prescriptions/import` — bulk import

Once a prescription is saved, the patient and medicines appear in pharmacy patient search and pending refill workflows.

Inpatient (IPD)

Available for **hospital**, **surgery centre**, and **multi-specialty** facility types.

Admissions

Route: `/clinic/inpatient/admissions`

Manage the full inpatient lifecycle.

Admit a patient:

1. Select patient, attending doctor, ward, and bed
2. Record admission reason and notes
3. System marks bed as occupied and creates admission record

Admission statuses:

Status	Meaning
`admitted`	Currently inpatient
`transferred`	Moved to another ward/bed
`discharged`	Discharged from facility

Actions:

- Admit (`POST api/Clinic/Admissions/admit`)
- Discharge (`POST api/Clinic/Admissions/{id}/discharge`)
- Transfer (service layer ready; HTTP endpoint planned)

Wards and Beds

Route: `/clinic/inpatient/wards-beds`

Tabbed management of ward capacity and bed inventory.

Ward types:

- General
- ICU
- Maternity
- Pediatric
- Surgical
- Isolation

Bed statuses:

Status	Meaning
`available`	Ready for admission
`occupied`	Patient assigned
`maintenance`	Out of service
`reserved`	Held for upcoming admission

Features:

- Create and edit wards with department link
- Create beds within wards
- Visual occupancy overview
- Update bed status and patient assignment

Surgery and Operating Rooms

Surgery capabilities are available for **surgery centre** and **multi-specialty** facility types, and partially for **hospital** types. The backend API is fully implemented; the dedicated user interface is in active development.

Operating Rooms

Operating rooms are registered facilities where surgical procedures take place.

Fields:

- Room name and code
- Location link
- Capacity
- Equipment notes
- Status (available, in use, maintenance)

API: Full CRUD at `api/Clinic/OperatingRooms`
Permission: `clinic:admissions:view|create|update|delete`

Surgery Bookings

Surgery bookings schedule procedures with full theatre coordination.

Booking fields:

Field	Description
Booking number	Unique reference per organisation
Patient	Customer (patient) record
Surgeon	User (doctor)
Anaesthetist	User (optional)
Operating room	Assigned OR
Department	User group (department)
Procedure name	Description of surgery
CPT code	Procedure coding for billing
Anaesthesia type	General, regional, local, sedation, etc.
Scheduled start / end	Theatre time block

Field	Description
Actual start / end	Recorded when procedure begins/ends
Pre-op notes	Pre-operative assessment and preparation
Post-op notes	Post-operative findings and instructions
Status	scheduled, in_progress, completed, cancelled
Links	Optional appointment, admission, and invoice

Status workflow:

```
scheduled → in_progress → completed
                ↓
            cancelled
```

Integration points:

- Links to **admission** for inpatient surgical patients
- Links to **appointment** for scheduling coordination
- Links to **invoice** for surgical billing
- CPT codes connect to procedure billing products

API: Full CRUD at `api/Clinic/SurgeryBookings`
Permission: `clinic:admissions:view|create|update|delete`

Surgery workflow (planned UI)

The intended end-to-end surgical workflow:

1. **Referral** — surgeon orders procedure during encounter (procedure record with CPT)
2. **Pre-op assessment** — admission created if inpatient; consent form signed (type: surgery)
3. **Theatre booking** — coordinator schedules OR, surgeon, anaesthetist, and time block
4. **Day of surgery** — status updated to in_progress; actual start recorded
5. **Post-op** — post-operative notes documented; encounter updated; status completed
6. **Billing** — invoice raised for procedure, theatre time, and consumables
7. **Discharge** — if admitted, patient discharged from ward after recovery

Consent for surgery

Consent forms support a `surgery` consent type. Templates are managed under **Consents**, and signatures are captured digitally before procedures proceed.

Laboratory

Lab Orders

Route: `/clinic/lab/orders`

Dashboard of incoming test requests from clinicians.

Order lifecycle:

Status	Meaning
`ordered`	Test requested
`sample_collected`	Specimen obtained
`in_progress`	Being processed
`completed`	Results available
`cancelled`	Order cancelled

Features:

- Create orders with patient, ordering doctor, priority, and test selection
- Status transitions via workflow actions
- Inline result entry in offcanvas panel
- Link to lab test definitions (catalog)

Sample Tracking

Route: `/clinic/lab/sample-tracking`

Track specimens from collection through processing.

Sample statuses:

Status	Meaning
`collected`	Sample obtained from patient
`received`	Received in laboratory
`rejected`	Sample unsuitable (hemolysed, insufficient, etc.)

Features:

- Barcode tracking per sample
- Pending orders view (awaiting collection)
- Collected samples view (in processing)

Lab Results

Results are entered from the Lab Orders screen (inline offcanvas) rather than a separate page.

Result fields:

- Test name and value
- Unit of measure
- Reference range

- Abnormal and critical flags
- Verified by (technician)

Results appear on the patient chart under the **Lab Results** tab.

Lab Test Definitions

Route: `/clinic/admin/lab-test-definitions`

Configure the test catalog available for ordering.

Fields:

- Test name and code
- Sample type (blood, urine, swab, etc.)
- Normal range
- Turnaround time (hours)
- Critical low/high thresholds
- Optional link to billable inventory product

Billing and Insurance

Patient Invoices

Route: `/accounting/invoices` (shared accounting module)

Hospital billing uses the platform's standard invoicing engine. Consultation fees, procedures, lab tests, and other billable services appear as invoice line items linked to products in inventory.

Prescriptions (`mode = prescription`) and dispenses (`mode = dispense`) are specialised invoice types with their own workflows but share the same financial backbone.

Healthcare Insurance Claims

Route: `/clinic/billing/insurance-claims`

Submit and track claims to medical aid providers (health societies).

Claim fields:

- Patient and medical aid membership
- Linked invoice (amounts and line items)
- Claim reference number
- Submission date
- Status workflow

Claim statuses:

Status	Meaning
`draft`	Being prepared

Status	Meaning
`submitted`	Sent to funder
`approved`	Funder accepted
`partially_paid`	Partial reimbursement received
`paid`	Fully reimbursed
`denied`	Funder rejected
`appealed`	Under appeal

Medical aid providers are managed as Health Societies under Administration → Master Data. They are stored as supplier records with type `medical_aid_provider`.

> **Note:** The standalone **Insurance** module in Operations is for insurance brokerage (policies, reinsurance, regulatory reporting). Healthcare medical aid claims are a separate feature within HMS.

Administration and Master Data

Route: `/clinic/admin`

The administration area provides configuration tools for clinical operations.

Doctor Schedules

Route: `/clinic/schedule/doctors`

Tabbed screen for:

- **Doctor schedules** — recurring weekly availability (day, start time, end time, slot duration)
- **Staff shifts** — shift roster for ward and department staffing

Schedules feed into appointment booking and the waitlist allocation job.

Master Data (`/clinic/admin`)

Screen	Route	Purpose
Health Societies	`/clinic/admin/health-societies`	Medical aid providers and funders
ICD-10 Codes	`/clinic/admin/icd10-codes`	Diagnosis code reference (~70k codes)
CPT Codes	`/clinic/admin/cpt-codes`	Procedure code reference
EMR Templates	`/clinic/admin/emr-templates`	Note templates by specialty
Lab Test Definitions	`/clinic/admin/lab-test-definitions`	Laboratory test catalog

Departments

Departments are managed in **HR → Departments** (user groups with type `Department`). The clinic module reads them read-only for encounter, admission, and surgery assignment.

Drug Interactions

A drug interaction reference database exists in the API (`DrugInteractionsController`). A dedicated management UI is planned for the administration area.

Pharmacy Module

The Pharmacy module is a complete dispensary operations centre. It can run alongside HMS (hospital dispensary) or independently (retail pharmacy). For full pharmacy documentation, see Pharmacy. This section summarises pharmacy capabilities in the context of HMS.

How clinic and pharmacy connect

```
Clinician writes prescription (Rx invoice)
        ↓
Patient appears in pharmacy search with allergies and history
        ↓
Pharmacist dispenses medicines (DISP invoice)
        ↓
Stock deducted (FEFO) → Accounting updated → Refills scheduled
```

Connection	Detail
Shared patients	Same `Customer` record; allergies and chronic conditions visible at dispense
Shared products	Medicines from inventory catalogue with batch and expiry tracking
Shared dosage cyphers	Same codes used in prescribing and label printing
Prescription → dispense	Rx creates the clinical order; dispense fulfils it with stock deduction
Refills	Scheduled during dispensing; appear in Pending Refills worklist

Pharmacy Command Center

Route: `/pharmacy/dashboard`

KPI	Description
Today's dispenses	Count and total value
Pending refills	Scheduled and overdue
Verification backlog	Awaiting pharmacist sign-off
Awaiting collection	Verified but not collected
Low stock alerts	Below reorder level

KPI	Description
Expiring batches	Within 90 days

Quick Dispense

Route: `/pharmacy/dispense`

The main dispensing workstation organised in three columns:

1. **Patient panel** — search, allergy alerts, chronic conditions, pending refills, dispensing history
2. **Add line form** — product search, quantity, dosage cyphers, route, directions, refill scheduling
3. **Dispense basket** — line items, subtotal, fees/taxes, total

Workflow:

1. Select stock location and patient
2. Add medicine lines with dosing instructions
3. Attach prescription script (upload or photograph)
4. Save & Print → stock deducted (FEFO), PDF generated
5. Collect payment (optional) → print medicine labels

Keyboard shortcuts: P (patient), M (medicine), Ctrl+S (save), ? (help)

Dispense Queue

Tracks each dispense through internal workflow stages:

Stage	Meaning
Dispensed	Saved, stock deducted; awaiting pharmacist verification
Verified	Pharmacist confirmed accuracy
Collected	Patient picked up medicines
Paid	Payment recorded
On Hold	Paused pending clarification

Available in **table view** (searchable, filterable) and **kanban view** (drag between columns).

Pending Refills

Route: `/pharmacy/pending-refills`

Refills are created when staff tick "Schedule refills" during dispensing and specify count and interval (7, 14, 28, or 30 days).

Filter tabs: All, Overdue, Due Today, Due Soon, Scheduled

Actions: Process refill (creates new pre-filled dispense), reschedule, cancel

Auto-refreshes every 5 minutes.

Mixtures

Route: `/pharmacy/mixtures`

Compound preparation recipe library. Each mixture lists ingredient products with quantities and units, plus label text for printing.

Dosage Cyphers

Route: `/pharmacy/settings/dosage-cyphers`

Manage short codes (OD, BD, TDS, QDS, PRN, NOCTE, OM) with AI-assisted seeding for initial setup.

Pharmacy Reports

Route: `/pharmacy/reports`

Report	Purpose
Dispensing Summary	Volume and value by period
Product Movement	Which medicines move fastest
Refill Compliance	Patients collecting on schedule
Expiry Tracking	Batches approaching expiry
Patient History	Individual patient's dispense timeline

End-to-End Clinical Workflows

Workflow 1: New patient outpatient visit

```
Front Desk                    Clinician                    Lab / Pharmacy
    │                              │                             │
    ├─ Register patient            │                             │
    ├─ Book appointment            │                             │
    ├─ Check in (arrived)          │                             │
    │                              ├─ Open My Queue              │
    │                              ├─ Start encounter            │
    │                              ├─ Document SOAP notes        │
    │                              ├─ Add diagnoses (ICD-10)     │
    │                              ├─ Order lab tests ──────────┼─► Lab order
created
    │                              ├─ Write prescription ───────┼─► Rx available
in pharmacy
    │                              ├─ Sign & close encounter     │
    ├─ Collect payment (if cash)   │                             │
    │                              │                             ├─ Dispense
medicines
    │                              │                             ├─ Verify &
collect
```

Workflow 2: Inpatient admission and discharge

```
Admission Clerk             Ward Nurse                  Billing
     │                           │                           │
     ├─ Admit patient            │                           │
     ├─ Assign ward & bed        │                           │
     │                           ├─ Monitor occupancy        │
     │                           ├─ Record vitals (planned)  │
     │                           │                           │
     ├─ Discharge patient        │                           │
     ├─ Free bed                 │                           │
     │                           │                           ├─ Final invoice
     │                           │                           ├─ Insurance claim
```

Workflow 3: Surgical procedure

```
Surgeon                    Theatre Coordinator          Billing
     │                           │                        │
     ├─ Order procedure (CPT)    │                        │
     ├─ Request consent          │                        │
     │                           ├─ Book OR & team        │
     │                           ├─ Schedule time block   │
     │                           ├─ Update status         │
     ├─ Document post-op notes   │                        │
     │                           │                        ├─ Raise surgical
invoice
     │                           │                        ├─ Submit claim
```

Workflow 4: Laboratory test

```
Clinician                 Lab Technician              Patient Chart
     │                        │                           │
     ├─ Create lab order      │                           │
     │                        ├─ Collect sample           │
     │                        ├─ Track barcode            │
     │                        ├─ Enter results            │
     │                        ├─ Flag abnormal/critical   │
     │                        │                           ├─ Results visible
```

Workflow 5: Insurance claim

```
Billing Clerk              Funder (external)
     │                          │
     ├─ Create claim from invoice │
```

```
      ├─ Submit (status: submitted)├─ Review
      │                           ├─ Approve / Deny
      ├─ Update status            │
      ├─ Record payment           │
```

Permissions and Security

Permissions follow the pattern `{module}:{resource}:{action}`. Access is granted through organisation roles.

Hospital (clinic) permissions

Permission	Typical use
`clinic:patients:view`	Patient list, detail, chart
`clinic:patients:create`	Register, import patients
`clinic:patients:update`	Edit patients, clinic custom args
`clinic:appointments:view`	Appointments, waitlist, teleconsult, schedule
`clinic:appointments:create`	Book appointments, waitlist, shifts
`clinic:appointments:update`	Status changes, teleconsult updates
`clinic:appointments:delete`	Cancel/delete appointments
`clinic:encounters:view`	EMR, vitals, diagnoses, procedures, Rx list
`clinic:encounters:create`	Create encounters, prescriptions, medical history
`clinic:encounters:update`	Edit encounters, Rx payment
`clinic:encounters:delete`	Delete encounters/Rx
`clinic:encounters:sign`	Sign and download prescriptions
`clinic:admissions:view`	IPD, wards, beds, surgery, operating rooms
`clinic:admissions:create`	Admit, create ward/bed/surgery/OR
`clinic:admissions:update`	Discharge, update inpatient resources
`clinic:admissions:delete`	Delete ward/bed/surgery/OR records
`clinic:lab:view`	Lab orders, samples, results, catalog
`clinic:lab:create`	Create orders, test definitions
`clinic:lab:update`	Record results, update status
`clinic:lab:delete`	Delete lab entities
`clinic:billing:view`	Insurance claims list

Permission	Typical use
`clinic:billing:create`	Submit claims
`clinic:billing:update`	Update claim status
`clinic:billing:delete`	Delete claims
`clinic:admin:view`	Templates, codes, departments, reports, HMS dashboard
`clinic:admin:create`	Create admin reference data
`clinic:admin:update`	Edit admin reference data
`clinic:admin:delete`	Delete admin reference data
`clinic:consent:view`	Consent forms and signatures
`clinic:consent:create`	Record consent
`clinic:consent:update`	Edit consent
`clinic:consent:delete`	Delete consent

Pharmacy permissions

Permission	Use
`pharmacy:dispense:view`	Queue, records, refills, patient medication history
`pharmacy:dispense:create`	Dispense, verify, collect, mixtures, dosage cyphers
`sales:customers:view/create/update`	Pharmacy patient search
`sales:invoices:delete`	Cancel/delete dispenses
`sales:invoices:print`	Dispense PDF and labels
`stock:inventory:view`	Pharmacy products view

Route-level access

The UI enforces permissions at the menu and route level. Users without the required permission do not see the menu item or receive an access denied response when navigating directly.

Integration Architecture

```
┌─────────────────────────────────────────────────────────────────┐
│                          ORGANISATION                             │
│  healthcare_facility_type, facility name, registration number   │
└──────────────────────────────┬──────────────────────────────────┘
                               │
      ┌────────────────────────┼────────────────────────┐
      ▼                        ▼                        ▼
```

```
CUSTOMER               USER (doctor)          USERGROUP
(Patient)              clinic_* metadata      (Department)
   │                       │                       │
   ├──── ClinicAppointment ◄── doctor_id ──────────┤
   ├──── ClinicEncounter ──────────────────────────┤
   ├──── ClinicAdmission / Ward / Bed ─────────────┤
   ├──── ClinicSurgeryBooking ─────────────────────┤
   │                       │                       │
   ▼                       ▼                       ▼
INVOICE ◄── journal posting ───────────────► ACCOUNTING
 ├─ mode: "prescription" (Rx-*, clinic)
 └─ mode: "dispense"   (DISP-*, pharmacy)
        │
        ├── InvoiceItem ──► PRODUCT (inventory)
        ├── InvoiceItemInstruction ──► DosageCypher
        ├── InvoiceItemRefill ──► Refills workflow
        └── BATCH / STOCK (FEFO deduction)

PHARMACY ◄── depends on ──► INVENTORY + ACCOUNTING
HOSPITAL ◄── depends on ──► INVENTORY + ACCOUNTING

SUPPLIER (medical_aid_provider) ──► Patient.MedicalAidProvider
ClinicInsuranceClaim ──► InvoiceId
LabTestDefinition ──► ProductId (billable tests)
ProcedureRecord ──► ProductId (billable procedures)
```

Data reuse philosophy

The HMS follows a "reuse before create" architecture:

HMS concept	Platform entity	How
Patient	`Customer`	`IsPatient = true` + healthcare fields
Doctor	`User`	`clinic_*` custom arguments
Department	`UserGroup`	`GroupType = Department`
Prescription / Dispense	`Invoice`	`mode` field distinguishes type
Medicine	`Product`	Inventory catalogue with batches
Medical aid funder	`Supplier`	`SupplierType = medical_aid_provider`
Branch / site	`Location`	Extended for wards, beds, ORs

Dedicated clinical tables (in SQL schema `clinic`) exist only where no platform entity adequately models the workflow: appointments, encounters, lab orders, admissions, surgery bookings, and insurance claims.

Dashboards and Reports

Hospital Command Center

Source	Endpoint	Permission
HMS Dashboard	`GET api/Dashboard/hms`	`clinic:admin:view`

Metrics: Appointments (total, today, completed, cancelled, missed), encounters count, lab orders count, today's status breakdown, today's schedule table.

Pharmacy Command Center

Source	Endpoint	Permission
Pharmacy Dashboard	`GET api/Dashboard/pharmacy`	`pharmacy:dispense:view`

Metrics: Today's dispenses/value, pending/overdue refills, verification backlog, awaiting collection, low stock, expiring batches.

Clinic reports (planned)

Report	Status	Endpoint
Doctor revenue	API available	`GET api/Clinic/Reports/doctor-revenue`
Lab turnaround	API available	`GET api/Clinic/Reports/lab-turnaround`
Clinic reports UI	Planned	`/clinic/reports`

Pharmacy reports (implemented)

See Pharmacy Reports for full detail on all five operational reports.

Implementation Status

This section provides an honest assessment of feature maturity for proposal and planning purposes.

Production-ready

Area	Features
Front desk	Patient directory, import, patient chart, appointments (calendar), waitlist
Outpatient	My queue, encounters (SOAP), teleconsult, consent forms
Prescriptions	Create, edit, MCAZ search, PDF, email, payment
Laboratory	Lab orders, sample tracking, inline results, test definitions
Inpatient	Admissions, wards and beds
Billing	Insurance claims; invoices via accounting module
Administration	Doctor schedules, shifts, health societies, ICD-10, CPT, EMR templates
Pharmacy	Quick dispense, queue, records, refills, mixtures, cyphers, reports, dashboard

API-ready, UI in progress

Area	Backend	Frontend
Operating rooms	Full CRUD API	Stub component; route not wired
Surgery bookings	Full CRUD API	Stub component; route not wired
Drug interactions	Full CRUD API	No management UI
Clinic reports	Doctor revenue, lab turnaround APIs	Stub component

Planned

Area	Description
Triage	Quick intake for vitals and chief complaint before clinician
Nursing station	Inpatient vitals tracking and care routines
Standalone vitals/diagnoses	Dedicated screens (currently managed within encounters)
Patient transfer	Service method exists; HTTP endpoint not yet exposed
Embedded teleconsult video	Currently opens session URL in new browser tab
Clinic-specific invoice UI	Uses shared accounting invoices today

---

Technical Foundation

Architecture

Layer	Technology
Frontend	Angular (lazy-loaded `clinic` and `pharmacy` modules)
Backend	ASP.NET Core Web API with area-based routing
Database	Microsoft SQL Server; clinical entities in `clinic` schema
Authentication	Platform auth with permission-based authorisation (`RequirePermission`)
Background jobs	Quartz scheduler (waitlist allocation)
Documents	PDF generation for prescriptions and dispense labels

API structure

Area	Route prefix	Controllers
Clinic	`api/Clinic/*`	30+ controllers (patients, appointments, encounters, lab, admissions, surgery, etc.)
Pharmacy	`api/pharmacy/*`	Dispense, Refills, Patients, Prescriptions (scripts), Mixtures, Dosage Cyphers

Area	Route prefix	Controllers
Shared	`api/Dashboard/hms`, `api/Dashboard/pharmacy`	Dashboard KPIs
Prescriptions	`api/clinic/Prescriptions`	Clinic prescribing (invoice-based)

Key services

Service	Responsibility
`ClinicService`	Central clinic business logic, dashboard, admissions, wards, ORs
`ClinicCustomArgsService`	Typed read/write of `clinic_*` / `healthcare_*` organisation and user metadata
`PharmacyDispenseService`	Dispense creation, FEFO batch allocation, workflow transitions
`PharmacyRefillService`	Refill scheduling, overdue detection, reschedule
`PrescriptionDocument`	Prescription PDF template generation

Clinical database entities

Category	Tables
Scheduling	`clinic_appointments`, `clinic_doctor_schedules`, `clinic_waitlist_entries`, `clinic_shifts`
EMR	`clinic_encounters`, `clinic_vitals`, `clinic_diagnoses`, `clinic_procedure_records`, `clinic_encounter_audits`
Laboratory	`clinic_lab_orders`, `clinic_lab_samples`, `clinic_lab_results`, `clinic_lab_test_definitions`
Inpatient	`clinic_admissions`, `clinic_wards`, `clinic_beds`
Surgery	`clinic_operating_rooms`, `clinic_surgery_bookings`
Billing	`clinic_insurance_claims`
Reference	`clinic_icd10_codes`, `clinic_cpt_codes`, `clinic_drug_interactions`, `clinic_templates`
Telemedicine	`clinic_teleconsult_sessions`
Consent	`clinic_consent_forms`, `clinic_consent_signatures`

Related Documentation

- Pharmacy — Full pharmacy module guide
- Pharmacy Home Delivery — Delivery logistics for pharmacy
- Accounting & Sales — Invoicing, payments, and journal posting
- Inventory & Stock — Products, batches, and stock management

- HR & Payroll — Doctors, departments, and shift management
- Organisation — Module activation and settings
- All Modules — Complete platform module index

This document describes the Hospital Management System, Surgery, and Pharmacy capabilities of the Operations platform. Feature availability depends on organisation module activation, healthcare facility type, and user role permissions.