

PRODUCT · OPESCARE.COM

The Platform Behind the App

How OpesCare Runs Every Desk

Jude Nshome on OpesCare's web platform: one patient index, a FHIR ingest layer and pharmacy, lab, blood and front-desk workflows on one shared record.



BY

Jude Nshome

Published 19 October 2026

CONTENTS

- 01** What the platform is
- 02** The Master Patient Index: who this person is
- 03** The interoperability layer
- 04** The front office and the clinic
- 05** The pharmacy: not a stock list — a workflow
- 06** The laboratory: sample to release
- 07** The blood bank: supply meets demand
- 08** The governance spine
- 09** Under the hood (for engineers)
- 10** Where the honest limits are
- 11** The platform in one paragraph
- Frequently asked questions
- Sources
- About the author

KEY TAKEAWAYS

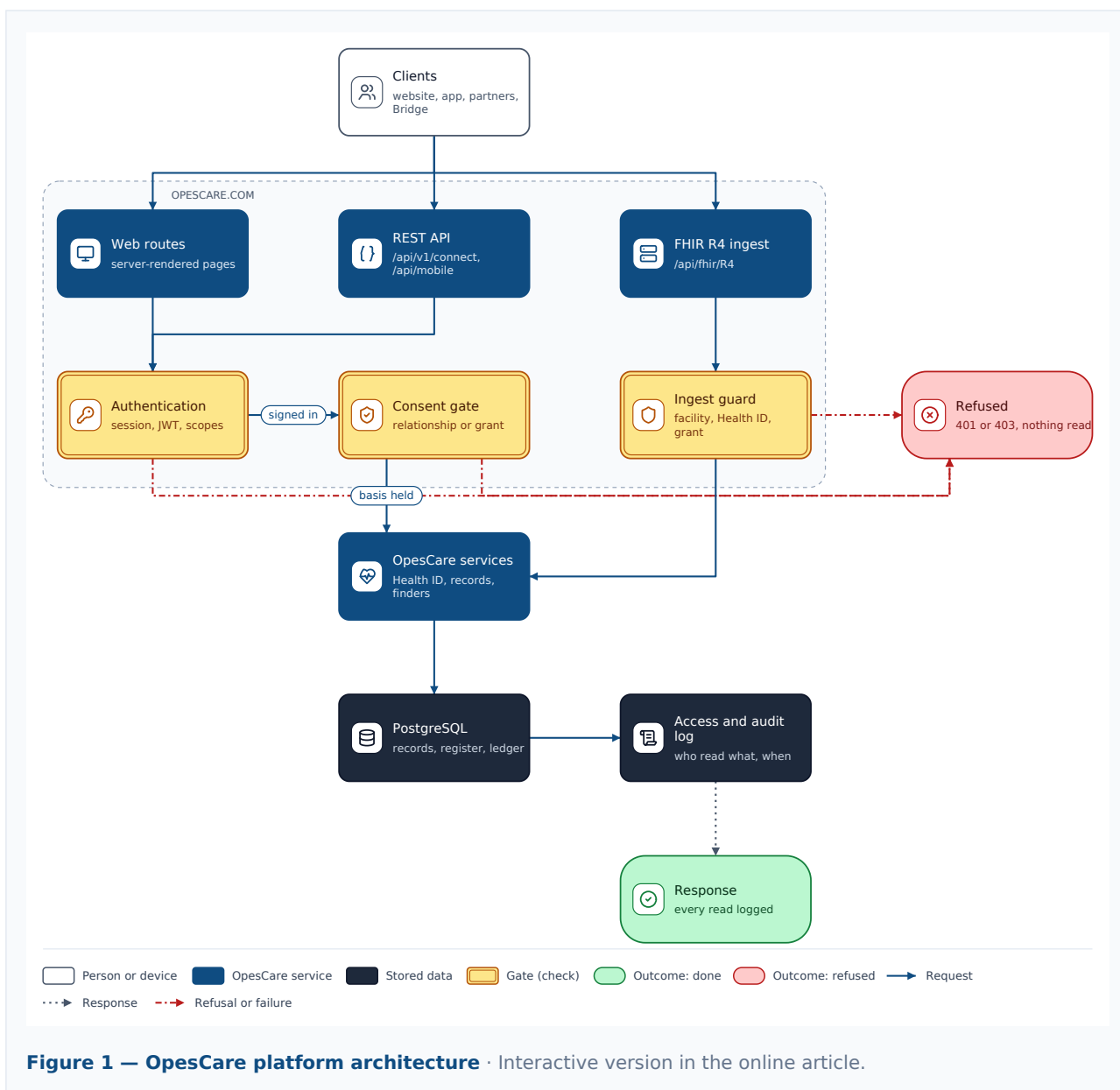
- The web platform is the institutional half of OpesCare: the same record the patient carries on their phone is the one the pharmacy, laboratory, blood bank and front desk work on.
- The Master Patient Index finds likely duplicates every night, but nothing merges automatically: a person reviews each pair, and every merge can be undone.
- Interoperability is inbound first: partners send FHIR R4 records through a scoped, audited ingest that refuses what it cannot place. Patients can export their own record as FHIR; there is no outbound FHIR server.
- Clinical codes come from published vocabularies loaded as data — ICD-10, LOINC, ATC and UCUM — rather than typed as free text.

- Each desk is a workflow with states: prescriptions are accepted, readied and dispensed; lab results are entered, validated and released to the patient and the ordering clinician at once; blood requests are decided, readied and fulfilled.
- The limits are stated plainly: DHIS2 reporting is aggregate-only and waits on MINSANTE access, payments wait on merchant accounts, and OpesCare is not a hospital information system.

The web platform is not an admin panel bolted onto an app. It is the other side of the same record: one patient index, one way in for partner data, and the everyday workflows of pharmacies, laboratories and blood banks, rebuilt so that they share a record instead of copying one.

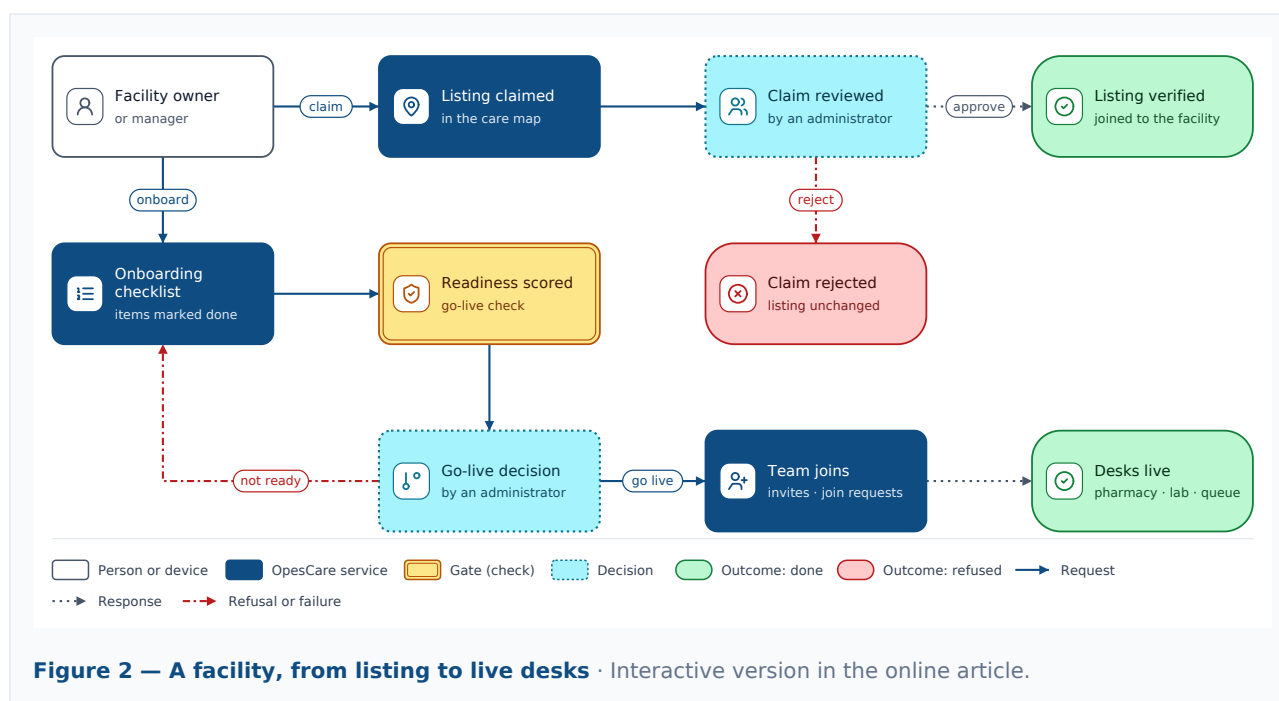
The patient-facing app, described in [The Record Is the Product](#), is the visible half of OpesCare. The other half is the **web platform** at opescare.com, which serves institutions: the independent pharmacy, the district clinic, the diagnostic laboratory, the imaging service, the blood bank — and the hospital that combines several of them.

This essay walks inside it: the **Master Patient Index** that settles who a patient is, once; the **interoperability layer** that lets partner systems send records in; and the **desk workflows** — registration, triage, queue, prescribing, dispensing, sample-to-release and blood requests — built around the same record the patient holds on their phone. It is also careful about what the platform is not. OpesCare is a discovery, identity and coordination product. It is not a hospital information system, and it does not pretend to be one.



1. What the platform is

One Laravel application on PostgreSQL serves the public website, the institutional consoles, and one /api surface shared by the mobile app and by partner integrations. Each institution is onboarded, worked through a go-live checklist and readiness score by a platform administrator, and from then on takes part in the shared patient index while running its own desks.



Who	What they use on the web today
Facility administration	Team invitations and roles, facility settings, a ward and bed register for occupancy counts, invoices issued
Front office	Triage desk with reassessment; the day's queue — check-in, call, start, complete, no-show
Clinician	Prescribing with an interaction check, lab ordering, referrals, immunisation records, a consent-scoped chart
Pharmacy	Prescriptions and dispensing, medicine orders from patients, stock, batches, receipts, adjustments, reorder notes, prices, stock import, a controlled-medicines register, reports
Laboratory	Test requests, collection, results, validation, release, a manager dashboard with critical-value call-backs
Imaging	Imaging orders scheduled, performed and reported, and a public imaging-exam finder
Blood bank	Blood inventory by product and group, and a blood-request desk
Platform administration	Master Patient Index desk, facility onboarding and readiness, partner credentials, integration certifications, DHIS2 aggregate reporting, breach register, legal documents

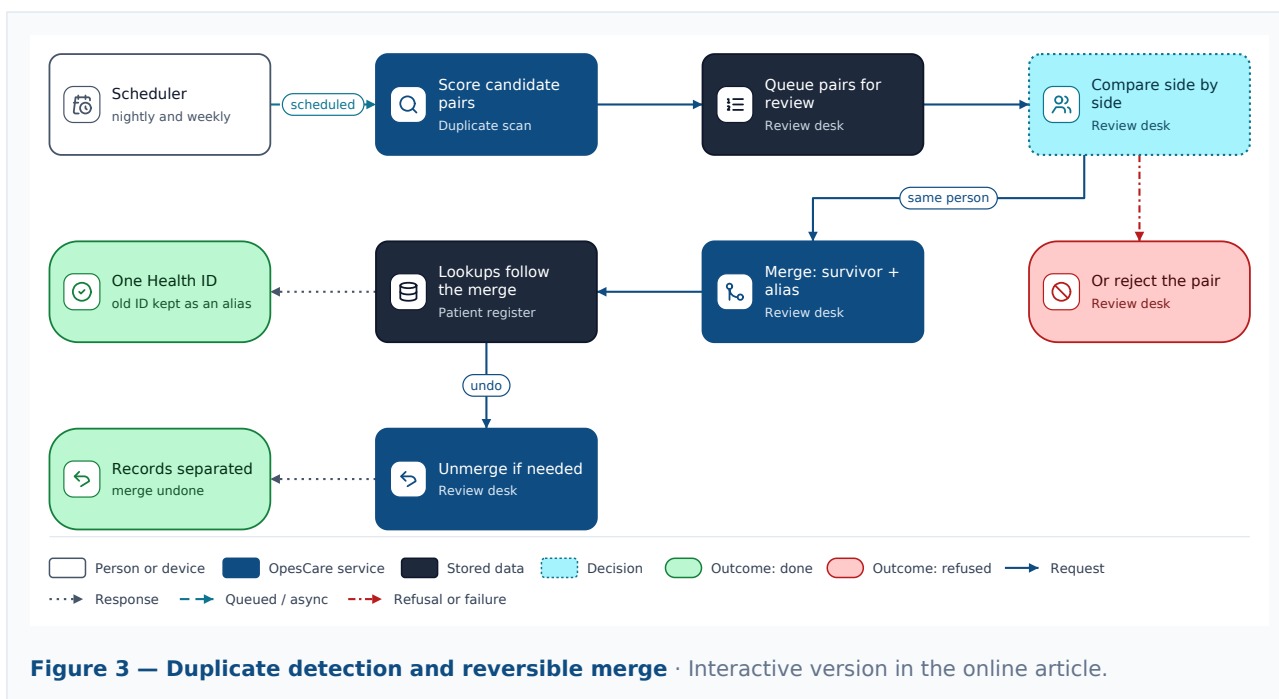
Who	What they use on the web today
Developers	Applications and credentials, webhook subscriptions, delivery history and replay, documentation

None of these is a mock-up. Each has a registered route, a controller, a page and a link in the navigation for the role that uses it. The common thread is that they are not separate products: one patient, one consent model, one audit trail.

2. The Master Patient Index: who this person is

A national network fails quietly if the same person exists twice. The Master Patient Index (MPI) is therefore a governed process, not a search box.

- **Exact identifiers at the desk.** Front-desk registration and booking work on the exact Health ID. A desk cannot browse the population by typing part of a name; if a registration looks like an existing identity, it is refused rather than duplicated. Every path that creates a patient goes through the same duplicate check.
- **Detection runs on a schedule.** A nightly scan looks at recent registrations and a weekly scan looks wider. Both raise *candidate pairs*; neither merges anything.
- **A person decides.** At the MPI desk an administrator reviews each pair with the evidence side by side and either merges or rejects it. Staff can also raise a pair by hand when they spot one.
- **Merges are reversible.** A merge records which record survived and keeps the other identifier as an alias, so an old Health ID or QR code still resolves to the right person. A merge can be undone, most recent first, from the same desk.
- **The card maps to one row.** QR tokens are short-lived and the Health ID resolves through the alias table, so the card in the patient's hand points to exactly one entry in the index.



This carries the "one person, many facilities" argument. The questions a national index must answer — *who is this?* and, a year later, *are they in here twice?* — are answered by a process with human review and an audit trail, not by whoever happened to type the name.

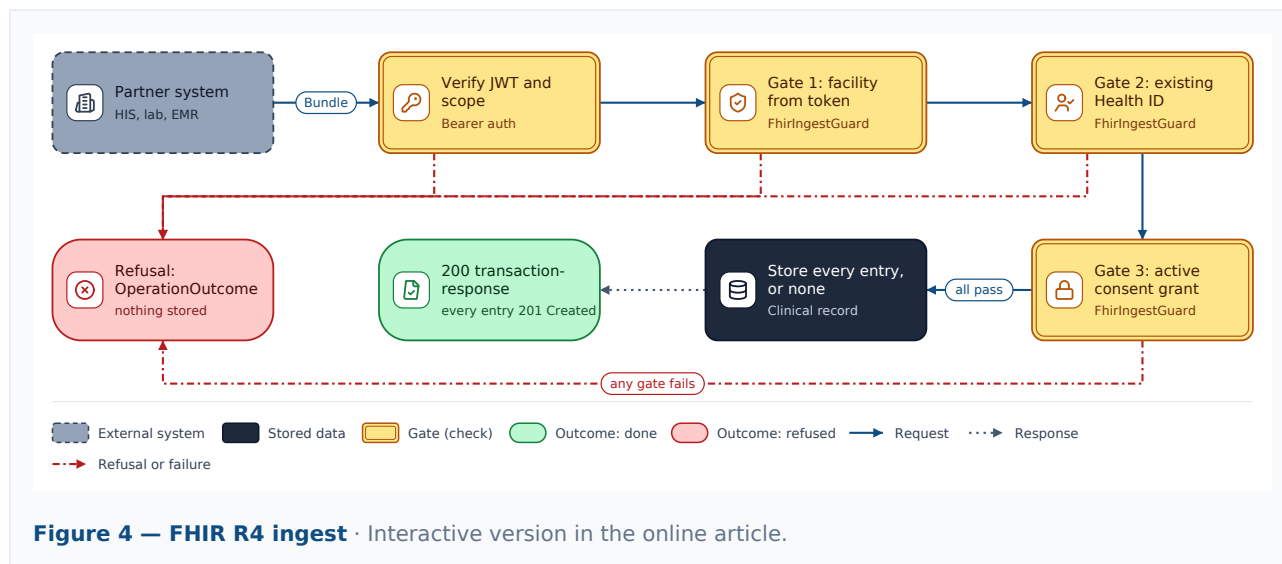
3. The interoperability layer

OpesCare treats interoperability as a discipline that starts with what comes in.

FHIR in, and the patient's own export out

- **Partner ingest.** Approved partners send FHIR R4 resources — DiagnosticReport, Observation, Immunization, and transaction bundles — to a scoped endpoint with a published capability statement. Each partner holds bearer credentials limited to the scopes an administrator granted. A guard checks that the partner may write about that patient and that the resource can be mapped; when it cannot, the ingest **refuses loudly** instead of storing something in the wrong place.
- **The patient's export.** From the mobile app a patient can export their own record as a FHIR bundle, and import one.

- **What is absent.** There is no outbound FHIR server for third parties to query, and no HL7 v2 interface. That is a deliberate choice: records leave OpesCare when the patient takes them, not when a system asks.



Vocabularies as data, not literals

Diagnoses, tests, medicines and units are coded against published vocabularies imported as data: **ICD-10** from the World Health Organization, with French titles; **LOINC** for laboratory observations; **ATC** for medicines; and **UCUM** for units of measure, so that milligrams and micromoles per litre are never confused at a handover. A code dictionary sits on top of them, so that a code arriving through ingest and a code chosen at a desk mean the same thing.

A developer surface with guard rails

The developer portal lets an organisation register applications and obtain credentials, subscribe to **webhooks**, see each delivery and **replay** a failed one. A partner can also push pharmacy and blood stock levels through the Connect API, which feeds the finder only when the source is on the availability allow-list. An integration certification checklist is kept by platform administrators. Documentation lives at opescare.com/docs. A laboratory system vendor can integrate against a permissioned, audited surface rather than an email thread.

National reporting, aggregate only

OpesCare can report its own monthly **aggregate counts** — never patient-level data — to the national DHIS2 instance run by MINSANTE. Facilities are mapped to organisation units and counts to data elements; a draft is generated on the first of each month, and a platform administrator reviews and approves it before anything is sent. Today the drafts are generated, but nothing is pushed: that waits on MINSANTE granting access to the national instance. This is one-way reporting, not a health-data exchange.

4. The front office and the clinic

For a clinic, the platform covers the parts of the day that touch OpesCare patients:

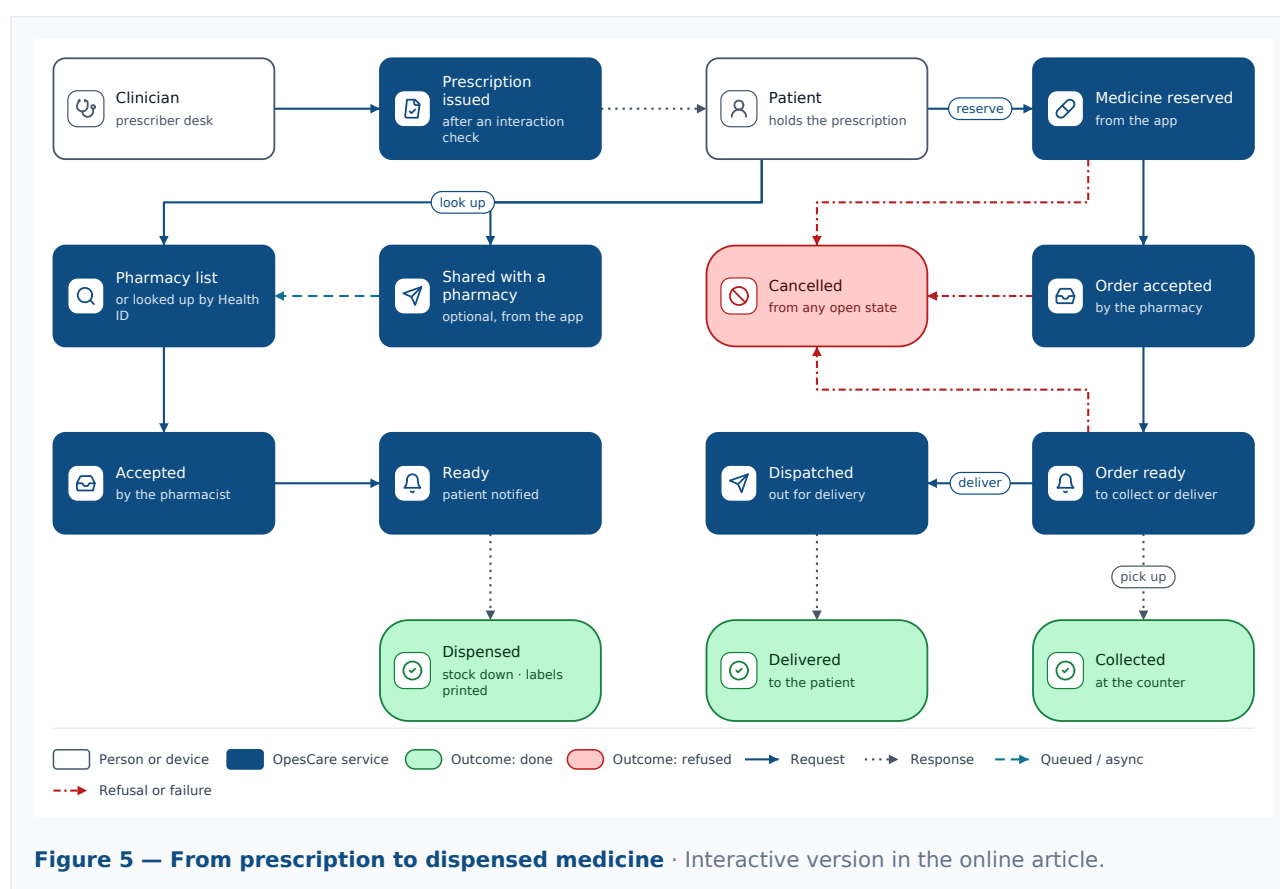
- **Triage** records vital signs against adult reference bands and supports reassessment. It re-orders the queue by urgency; it does not shorten anyone's wait, and it never blocks care.
- **The queue** takes a patient from check-in to call, start and completion, or marks a no-show, and is sorted on the server so every screen sees the same order.
- **Prescribing** runs an interaction check before a prescription is issued, and the printed prescription carries a verification code.
- **Referrals** are sent, accepted, completed or declined between clinicians; patients can request one from the app.
- **Immunisations** are recorded against the patient and can be voided, never silently edited.
- **Invoices** can be issued and paid by mobile money — unavailable until the merchant accounts are connected — and care is never gated on payment.

What the clinic side is *not*: full clinical documentation, admissions, discharges and transfers, bed-management workflows, insurance claims and pre-authorisation. A ward and bed register exists so a facility can report occupancy; it is a register, not an admissions system.

5. The pharmacy: not a stock list — a workflow

An independent pharmacy gets a complete operational console, built around dispensing.

- **Prescriptions arrive in one list** — from clinicians on OpesCare and from patients who share them — and move through *accepted* → *ready* → *dispensed*. A prescription written at one facility can be filled at another; the patient carries it.
- **Patient orders** — medicine reservations placed from the app — follow their own path: accepted, ready, then collected or dispatched and delivered, or cancelled.
- **Stock is what the pharmacy actually holds**: stock levels and batches, receipts of incoming deliveries, adjustments with a reason, reorder notes, a price list with history, and spreadsheet import with a preview, column mapping and an error report before anything is committed.
- **Controlled medicines** have their own register.
- **Reports** export dispensing and stock figures as CSV or PDF.

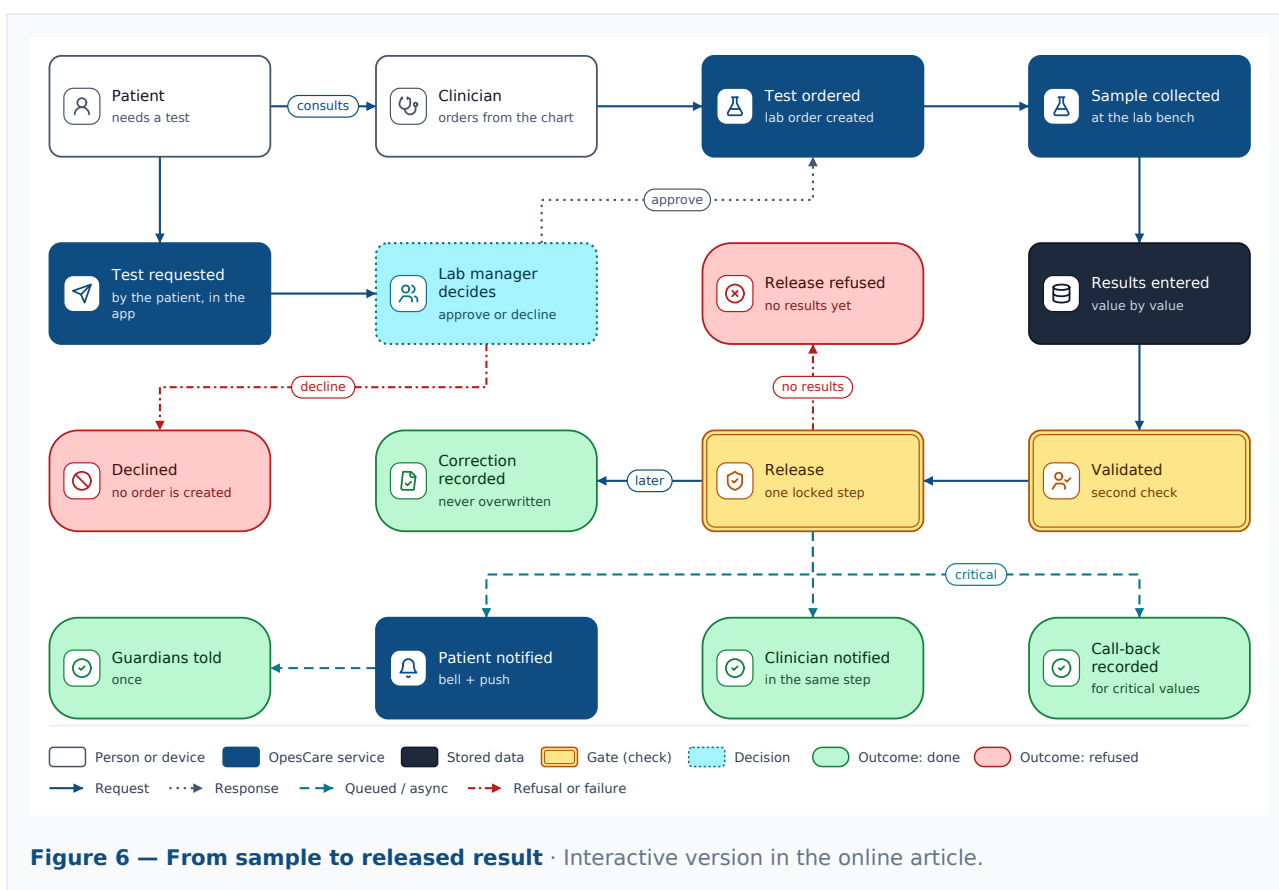


Two limits matter. Stock reported by a pharmacy becomes *finder* availability only when its source is on the allow-list, so an empty finder is the honest answer rather than a demo filled with invented stock. And procurement — purchase orders, supplier contracts — is not part of OpesCare; a simple supplier list exists so receipts can say where stock came from.

6. The laboratory: sample to release

Laboratories run a release workflow, not an inbox.

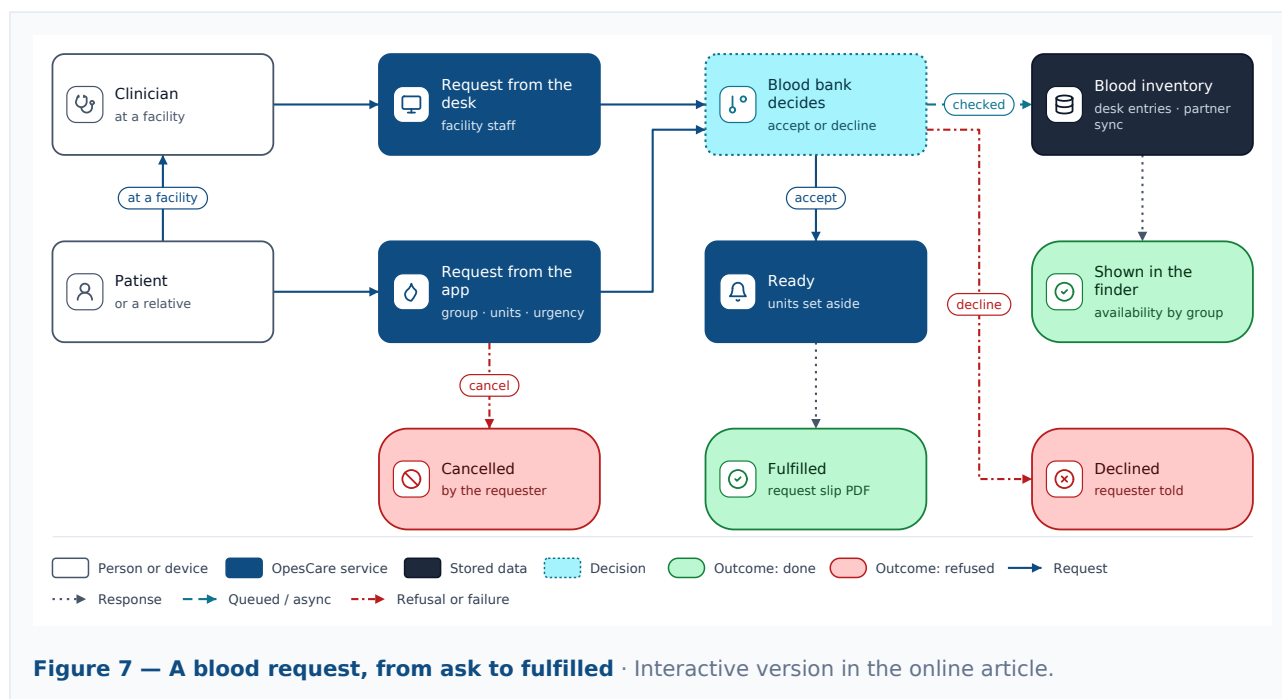
- **Requests** from clinicians or patients are approved or declined by the laboratory.
- **Orders** move through collection, result entry, validation and **release**. A released result can be corrected, but the correction is recorded, never overwritten.
- **Release notifies both sides at once.** The patient receives the result in their app and the clinician who ordered the test is told in the same step. A critical value is flagged as high priority, and the laboratory records its telephone call-back to the clinician.
- **A manager dashboard** shows turnaround, pending work and critical alerts.



Imaging follows the same shape for orders: requested, scheduled, started, completed, reported, then finalised and distributed, with an amendment path. OpesCare stores the report and its attachment; it is not an image archive and has no PACS or DICOM viewer.

7. The blood bank: supply meets demand

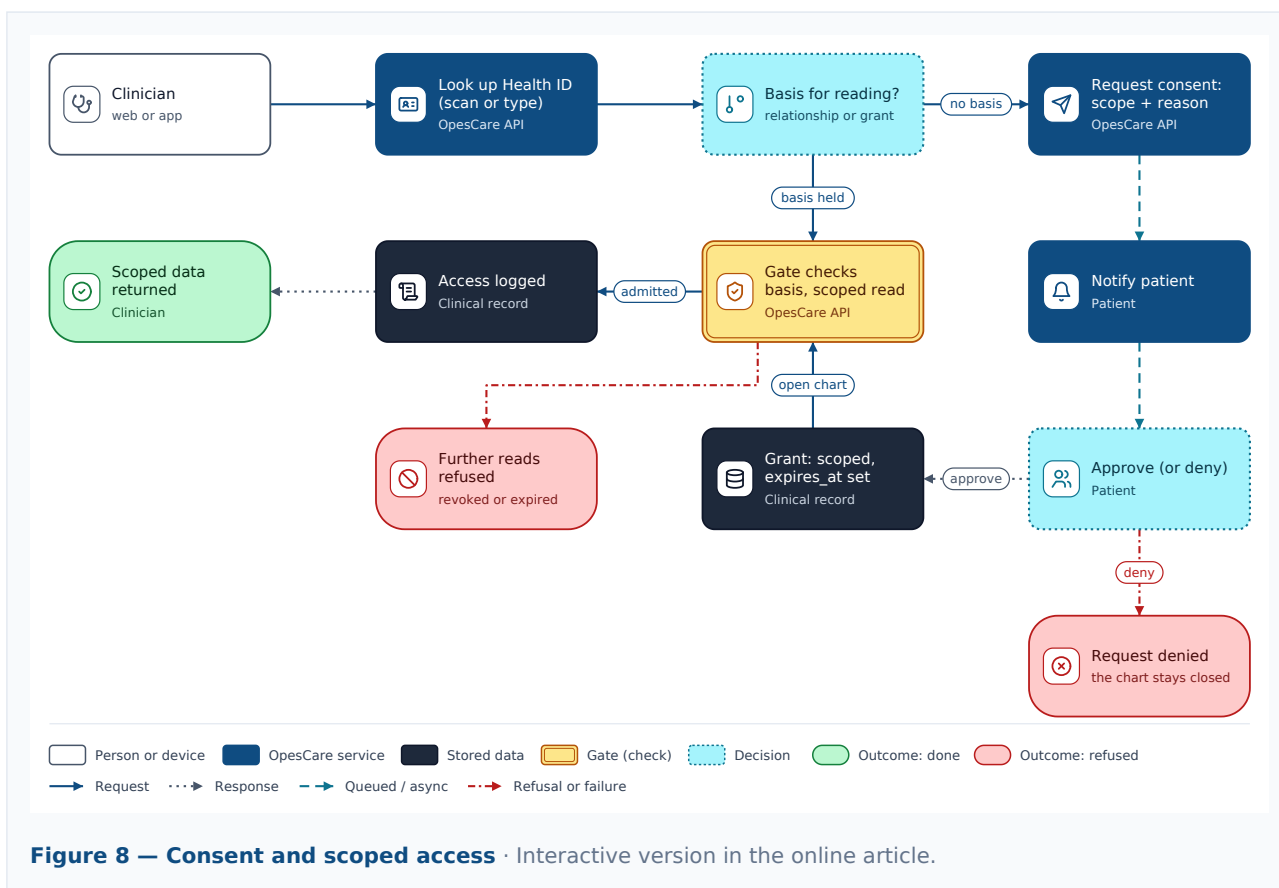
- **Inventory** is kept by blood product and group, with adjustments and flags, and can also be synced from a partner system.
- **The request desk** takes each request from a clinician or a patient through decision, ready and fulfilled, and prints a request slip.
- **Donors** can register their willingness to donate from the app.
- **The patient side closes the loop.** The blood finder a patient searches is fed by the same inventory the blood bank maintains, and a request made in the app arrives at the same desk.



8. The governance spine

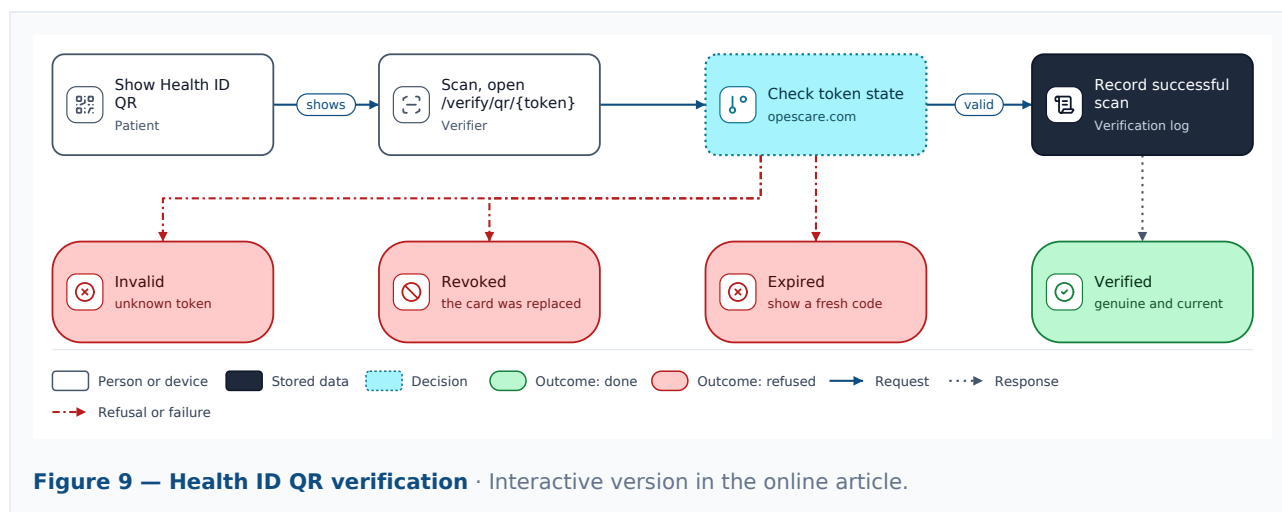
Everything above runs on the same governance as the app.

- **Consent as a protocol.** Every clinician read of a patient passes a gate that requires a care relationship — an appointment, a care team, an open episode at that facility — or a consent grant the patient approved.
- **Access logs.** Reads are logged, and the patient's own access-log view is the same log.
- **Break-glass.** Emergency access requires a reason, opens a limited profile, alerts the patient and is reviewed.
- **Security operations.** Administrators keep a breach register in which an incident is opened, contained, notified and closed.
- **Legal documents** — terms and policies — are versioned, published from the administration console and readable by anyone at a public URL.



9. Under the hood (for engineers)

- **Backend:** Laravel on PHP 8.3 and PostgreSQL. Background work runs on a queue worker managed by the operating system, with separate queues for imports and webhooks.
- **Documents:** PDFs — prescriptions, lab requests, invoices, receipts, labels — are rendered on demand on a secure sheet with a verification QR code, so a printed page can be checked against the server.
- **API:** the mobile app is an API client like any other; the OpenAPI description is generated from the code.
- **Audit:** patient-record access from every desk is written to the audit trail with the reason and the actor.



10. Where the honest limits are

- **Production, not proof of scale.** OpesCare has served opescare.com in production since 19 October 2026. We do not publish adoption figures here.
- **One country.** OpesCare is built for Cameroon, in English and French. There is no multi-country rollout.
- **DHIS2 waits on MINSANTE.** Monthly drafts are generated; nothing is sent until access to the national instance is granted.
- **Payments wait on merchant accounts.** Mobile-money payment is built and unavailable until those accounts are connected.
- **Push notifications** are not yet in the store build; they wait on store credentials.
- **Not a hospital information system.** No admissions, no inpatient nursing, no insurance claims, no procurement, no PACS. Those belong elsewhere, and saying so keeps the platform honest.

11. The platform in one paragraph

One patient index — checked nightly, merged only by a person, reversible, addressed by Health ID — feeds **one interoperability layer** — FHIR ingest from partners, the patient's own FHIR export, ICD-10, LOINC, ATC and UCUM as data, scoped credentials and replayable

webhooks, aggregate reporting to DHIS2 once it is opened — which serves **the workflows that already exist**: triage, queue, prescribing, dispensing, sample-to-release, imaging reports and blood requests. The patient's phone talks to the same spine the pharmacy counter talks to.

The patient's record does not need transcribing at any desk, because every desk is on the same record.

OpesCare is built by OPES Health Systems for care in Cameroon. It is available in English and French, for patients and for staff, on Android and the web today, with iOS on the roadmap. This essay is the companion to [The Record Is the Product](#), which describes the patient's side. Read more about [how hospitals join OpesCare](#), [the Health ID](#), [FHIR](#) and the [Master Patient Index](#), or the [developer documentation](#).

Frequently asked questions

What is the difference between the OpesCare app and the OpesCare platform?

The app is what a patient — or a clinician on the move — holds in their hand. The platform is the web application at opescare.com that institutions use: pharmacy, laboratory, imaging, blood-bank, front-desk and administration consoles, plus the API the app and partners talk to. Both work on the same record.

How does OpesCare avoid creating duplicate patients?

Registration and booking at the desk use the exact Health ID, and every patient-creation path checks for an existing identity first. A nightly scan raises likely duplicates for review at the Master Patient Index desk. A person decides each pair; merges are logged and reversible.

Does OpesCare exchange data with hospital systems using FHIR?

Inbound, yes: approved partners can send FHIR R4 diagnostic reports, observations, immunizations and transactions with scoped credentials. OpesCare does not run an outbound FHIR server. Patients can export their own record as a FHIR bundle from the app.

Can a pharmacy manage its stock in OpesCare?

Yes, for what it needs to be findable and to dispense: stock levels, batches, receipts of incoming stock, adjustments, reorder notes, price lists and spreadsheet import. Purchase orders, procurement and supplier contracts are out of scope.

Who is told when a lab result is released?

Both the patient and the clinician who ordered the test, in the same step. A critical value is flagged as high priority, and the laboratory records its telephone call-back to the clinician.

Does OpesCare send data to DHIS2?

Only aggregate counts, never patient-level data, and only after a platform administrator approves each monthly draft. Drafts are generated today; nothing is sent until MINSANTE provides access to the national instance.

Does OpesCare handle insurance claims or hospital admissions?

No. Insurance claims, pre-authorisation, admissions, discharges, transfers and bed management workflows are deliberately out of scope. A ward and bed register exists only so a facility can count occupancy.

Can a developer integrate with OpesCare today?

Yes. The developer portal issues application credentials, lets you subscribe to webhooks, inspect deliveries and replay them, and the documentation at opescare.com/docs covers the API, authentication, webhooks and the consent, QR, FHIR and MPI flows.

Sources

1. Platform architecture — OpesCare documentation — OpesCare
<https://opescare.com/docs/architecture> (accessed 19 October 2026)
2. Master Patient Index flow — OpesCare documentation — OpesCare
<https://opescare.com/docs/flows/mpi> (accessed 19 October 2026)
3. FHIR R4 ingest flow — OpesCare documentation — OpesCare
<https://opescare.com/docs/flows/fhir> (accessed 19 October 2026)
4. Consent flow — OpesCare documentation — OpesCare
<https://opescare.com/docs/flows/consent> (accessed 19 October 2026)

5. Webhooks — OpesCare documentation — OpesCare
<https://opescare.com/docs/webhooks> (accessed 19 October 2026)
6. HL7 FHIR Release 4 specification — HL7 International
<https://hl7.org/fhir/R4/> (accessed 19 October 2026)
7. LOINC — Logical Observation Identifiers Names and Codes — Regenstrief Institute
<https://loinc.org/> (accessed 19 October 2026)
8. International Classification of Diseases, 10th revision (ICD-10) — World Health Organization
<https://icd.who.int/browse10/2019/en> (accessed 19 October 2026)
9. ATC/DDD Index — WHO Collaborating Centre for Drug Statistics Methodology
https://atcddd.fhi.no/atc_ddd_index/ (accessed 19 October 2026)
10. The Unified Code for Units of Measure (UCUM) — Regenstrief Institute
<https://ucum.org/> (accessed 19 October 2026)



ABOUT THE AUTHOR

Jude Nshome

Jude Nshome is the founder of OpesCare, the bilingual health platform built by OPES Health Systems in Cameroon. He writes about patient-owned health records, consent and building digital health infrastructure for low-resource settings.

Read online: <https://opescare.com/blog/the-platform-behind-the-app>