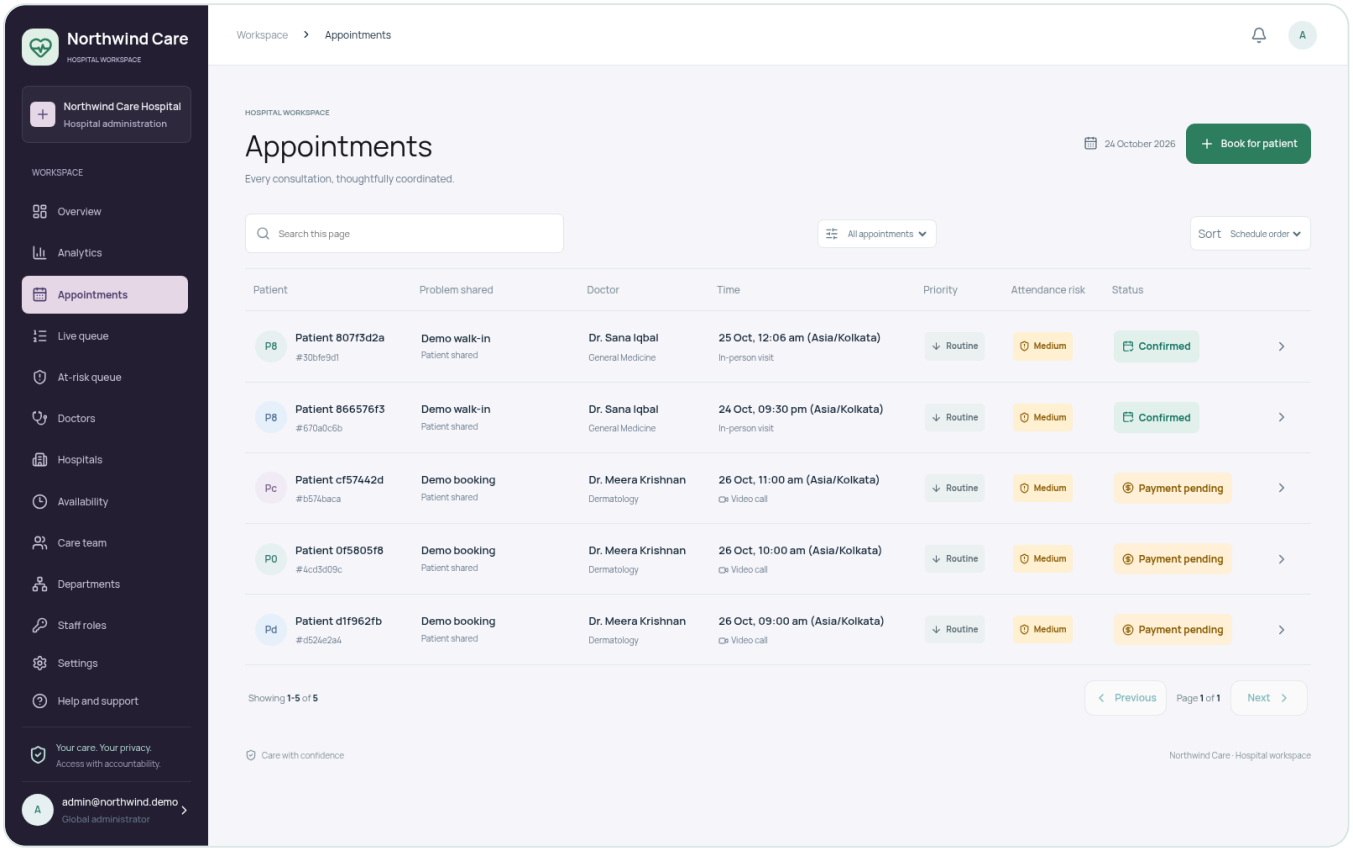
Telemedicine your hospital runs under its own brand.

Patient and doctor mobile apps, a hospital administration console, video consultations, scheduling, queues, payments, clinical documentation and AI-assisted care coordination. One shared core engine, deployed per hospital, in the hospital's own cloud account.

Patient and doctor apps | Hospital admin console | Video consultations | AI-assisted coordination | White-label



The live admin console: appointments with explicit hospital timezone labels. Northwind Care is the fictional demo brand; all data shown is synthetic.

See it running

Live demo workspace and a patient, doctor and admin walkthrough video at matrytech.com/products/telemedicine.

Honest status

This guide separates what is working and verified today from what is in progress or roadmap. Page 8 is the engineering status table.

Scoped pricing

Package pricing is agreed on a discovery call and confirmed in a written proposal before work starts.

One platform, three connected apps, one back office.

A hospital gets its own branded telemedicine service: patients book and attend video or in-person visits, doctors run consultations and prescriptions, and staff run the front desk, scheduling, payments and reporting from a web console.

3
Connected apps: patient, doctor, hospital admin console

36
Backend modules covering clinical, operational and commercial flows

603 + 78
Automated unit tests and end-to-end API tests on the deployed build

5
AI-assisted capabilities implemented, covered by automated tests

Appointments & scheduling

Doctors publish availability; a schedule generator fills whole weeks in one step. Patients book video or in-person visits. All times are stored as exact UTC instants and shown in the hospital's own timezone, explicitly labelled. Daylight-saving gaps are flagged, never silently moved.

Video consultations

In-app video visits with a consultation room, pre-call camera and microphone check, doctor notes and a visit summary. Instant-consult requests route to the least-loaded available doctor.

Implemented and test-covered; live-call verification happens per deployment.

Live queue & front desk

Walk-ins and booked patients check into a live queue. Staff admit, complete or mark no-shows; urgency levels and an at-risk queue surface patients likely to miss visits.

Clinical records

Visit summaries, prescriptions with medicine search, patient history, dependents and consent management built in.

Payments & invoices

Per-visit online payment with signature verification, webhook confirmation, automatic invoice PDFs and full refunds. Additional payment providers are planned; the provider-adapter groundwork is built and tested on the development branch, not yet deployed.

Covered by automated API integration tests; live provider checkout is not yet verified.

Hospital administration

Staff roles and permissions, doctors and departments, analytics, notification campaigns, audit log, support tickets and patient registry: a full back office.

Multi-hospital ready

Tenancy, per-hospital timezones, staff scoping and data separation are designed for groups running several facilities.

White-label

Name, logo, brand colors and terminology are configuration, not code. The same build serves any hospital under its own identity and domain.

Patient app and doctor app, built from one mobile core.

Both apps are Flutter builds from a shared, tested core with per-brand flavors, so a hospital's pair of apps stays consistent and a second hospital is a new flavor, not a fork.

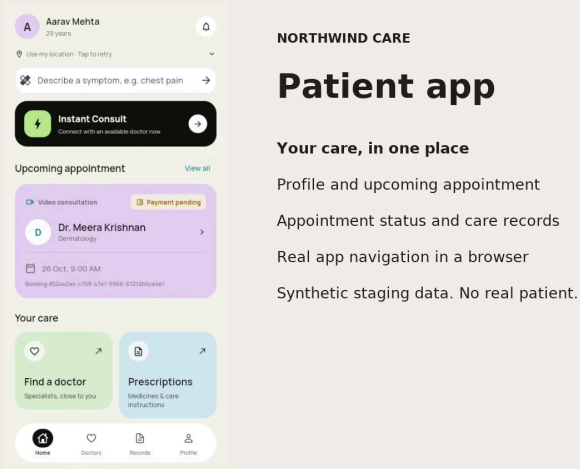
Patient app

- Phone OTP sign-in with per-market international calling codes
- Book video or in-person appointments against real doctor availability
- Dependents: book and manage visits for family members
- Purpose-specific consent captured before teleconsultation, with a working withdrawal path
- Video visit with pre-call camera and microphone check
- Visit summaries and prescriptions downloadable as PDFs
- Per-visit online payment with automatic invoice PDF (API test coverage; live provider checkout not yet verified)
- Patient reports and support tickets in-app

Doctor app

- Today's schedule and live queue of checked-in patients
- Consultation screen for clinical notes during the visit
- E-prescriptions with medicine search from the hospital's catalog
- Prescription templates for repeatable treatment plans
- AI-drafted visit summary the doctor reviews and approves
- Availability publishing with a week-at-a-time schedule generator
- Leave management that respects existing bookings

Platform honesty, mobile: the walkthrough video shows actual patient and doctor app navigation in Flutter browser builds connected to synthetic staging. It is not an iOS or Android device test. Android debug APKs compile and their package signatures and demo API configuration have been checked; installation and device flows are not yet verified. iOS distribution requires a Mac signing environment and the hospital's own Apple account.

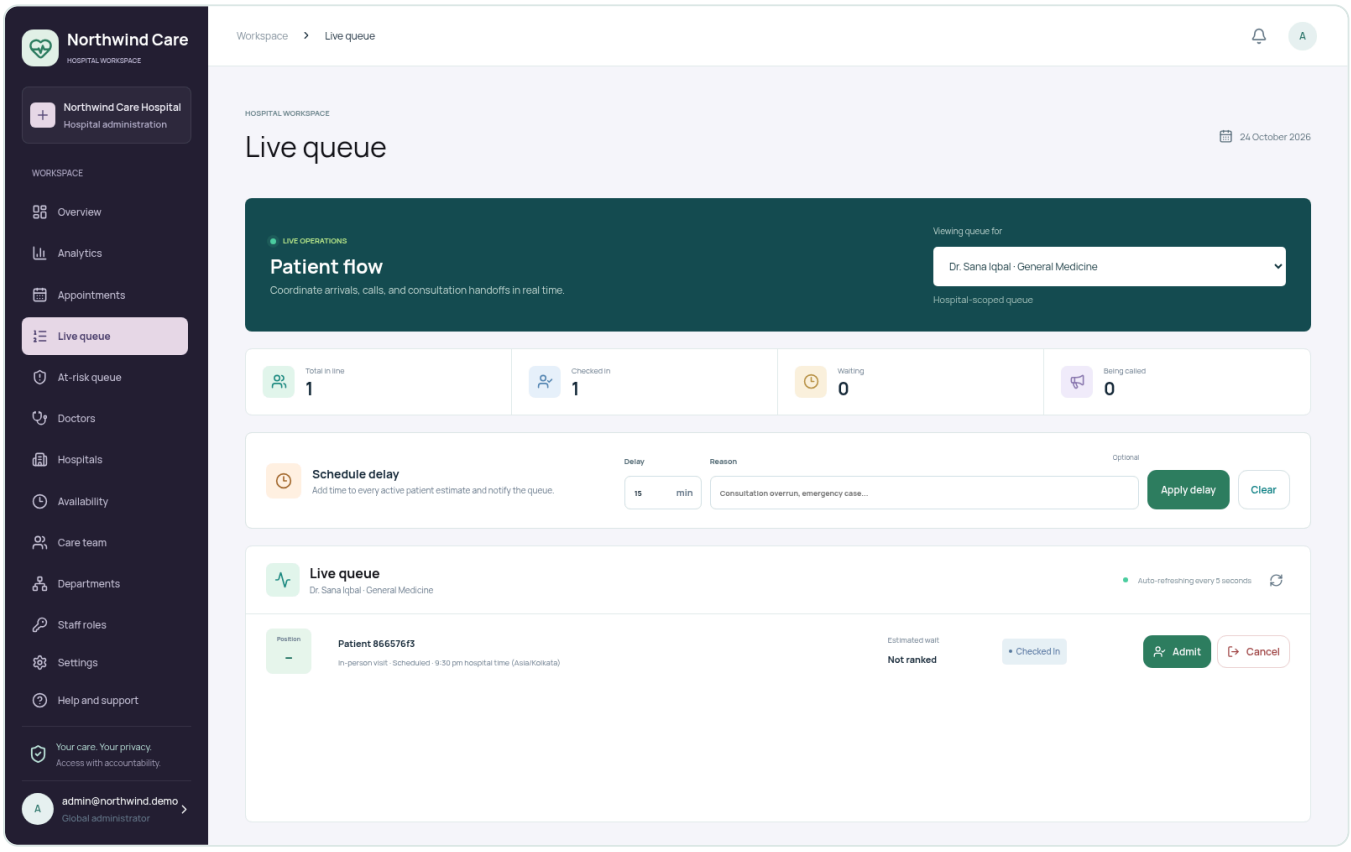


Frames from the combined patient, doctor and admin walkthrough. Synthetic demo data.

The back office your staff actually runs.

A secure web workspace for appointments, patient flow, clinical teams and hospital operations, with role-based access for front desk, doctors, and administrators.

- **Overview dashboard** with today's appointments, queue and stats
- **Appointments** with full lifecycle: booked, checked in, in consult, completed, no-show, cancelled
- **Live queue** for walk-ins and arrivals, with delay broadcast to waiting patients
- **Doctors and departments**, including doctor lifecycle and profile photos
- **Availability and slots**, week schedule generation, leave handling
- **Staff and roles**: role-based access control with auditable permissions
- **Patient registry** with bulk import
- **Medicine catalogue** and hospital-specific medicine lists
- **Symptom mappings** the triage and search features use
- **Payments**: reconciliation, refunds and stale-order cleanup
- **Analytics**: bookings, utilization and no-show patterns
- **Notification campaigns** with a composer for patient messaging
- **Audit log** of sensitive actions across the workspace
- **Support tickets** and multi-hospital management



Live queue with a checked-in synthetic patient, urgency levels and hospital-timezone labels. Synthetic demo data.

Practical AI that assists staff, not a demo chatbot.

The AI features are aimed at the two places hospitals lose time and money: missed visits and consultation admin. Each one is implemented in the product and covered by automated tests (54 tests across these five modules, passing 24 October 2026). Runtime validation on a hospital's own deployment is part of every go-live.

AI visit summaries IMPLEMENTED

After a consultation, a structured draft summary is generated for the doctor to review and approve. Only the clinical text is sent to the AI provider; patient name, phone number and record IDs are never included. A template-based fallback works with no external AI dependency.

No-show risk IMPLEMENTED

A risk model scores upcoming bookings and feeds an at-risk queue in the admin console, so staff can confirm or reschedule the appointments most likely to be missed. Attendance-risk badges already render on appointments in the demo console.

AI triage IMPLEMENTED

A protocol-based triage module captures symptoms before booking and routes patients to the right specialty, with urgency levels surfaced to the front desk.

Symptom search IMPLEMENTED

Patients describe symptoms in plain words and reach the right department; staff manage the symptom-to-specialty mappings from the console.

Doctor load balancing IMPLEMENTED

Instant-consult requests are routed to the least-loaded available doctor instead of a static assignment list.

AI patient-history search ROADMAP

Natural-language search across a patient's history. The feature flag and architecture exist; it is not enabled and is scoped as phase 2 work per deployment.

A note on AI and patient data: data-touching vendors are declared per deployment and signed off by the hospital's compliance reviewer before real patient data flows through them. The AI summary provider is configured at the account level for no training and short retention before go-live. Triage follows a clinical rules protocol and no-show risk is a scoring model; neither is generative AI, and none of these features makes an autonomous diagnosis or replaces clinical judgment.

Your brand, your cloud, your data.

Your brand, not ours

App name, logo, colors, legal entity name, terminology (what you call patients, doctors, facilities and appointments), support contacts and policy display are all set in a per-client configuration manifest. The same engine serves every hospital under its own identity and domain.

Your market configuration

Country, currency, locale, IANA timezone, calling-code options, measurement system, default appointment duration, cancellation window, refund mode, consent version and data-retention period are configured per client.

Your cloud, your data

The platform deploys into the hospital's own cloud account using the included Terraform stack. Patient data lives in the hospital's own database, not in a Matrytech-controlled one. This is the foundation of the platform's data-residency and DPDP-compliance story.

Your store presence

Patient and doctor apps ship as per-client release builds with their own icons, bundle IDs and store listings under the hospital's own Apple and Google accounts.

What "white-label" means commercially

- **You own the deployment.** It runs in your cloud account on your own provider accounts, and source-code and IP terms are agreed in your contract. Our fees are setup plus a flat monthly retainer - they do not meter per patient or per visit.
- **Setup plus a monthly retainer.** A one-time setup covers configuration, branding, deployment and verification; a monthly fee covers management, support and platform updates.
- **Third-party services at cost.** Video, SMS, payment, push and maps providers run on the hospital's own accounts, billed by the provider without markup.
- **Groups and chains.** Multi-hospital tenancy, per-hospital timezones, staff scoping and data separation are designed in, so a clinic group runs several facilities on one deployment.

> **Against rent-a-platform alternatives:** the deployment, data and brand stay with the hospital instead of a rented seat on someone else's platform. Source access and rights to extend the system are licensing terms agreed in your contract.

From discovery call to a running hospital workspace.

STEP	WHAT HAPPENS
1. Discovery call	We map your current booking and consultation flow, markets and must-haves. A short call is usually enough; bring your current process even if it is a phone and a register.
2. Written proposal	Scope, price and timeline confirmed in writing before development. No hourly billing surprises.
3. Configuration	Brand manifest, market settings, provider identifiers and feature flags for your hospital. Configuration work, not new engineering, for a standard deployment.
4. Deployment	Your isolated cloud stack is stood up with the included Terraform configuration; the admin workspace, backend and app builds point at it.
5. Verification	Booking-to-payment, video, queue and consent flows are exercised end to end on your deployment before handover.
6. Run and improve	Monthly retainer covers management, support and platform updates. New platform features land as they are released.

Honest scope boundaries

- **HMIS integration is real engineering, every time.** Every hospital's HMIS is different; there is no generic adapter. Integration time is budgeted per client.
- **Regulatory requirements beyond the base compliance pack** (for example a specific country or state regime) are scoped per deployment.
- **Deviations from the core feature set** (for example live HMIS sync on day one) are priced as separate work with their own timeline.
- **Not built yet, and not implied:** a self-serve onboarding console, multi-client billing automation, and runtime tenant discovery by custom domain. Onboarding is engineering-led today.

Ready-made

The platform described in this guide, configured and branded for your hospital or clinic group on the shared core engine. The monthly retainer covers management, support and platform updates; hosting and third-party usage are billed to you directly by the providers at cost. Price scoped on a call.

Custom build

For requirements the core platform does not cover. Scope and price confirmed in your written proposal before development; management, support and updates run as a monthly retainer after launch.

What is verified, and what is not.

Split between the deployed demo build and the current development branch. We update this table as verification lands.

AREA	STATUS
Backend platform	**Deployed demo build:** 603 automated unit tests and 78 end-to-end API tests covering booking, payment, queue, consent, video and administration flows, including hospital-timezone slot scheduling with explicit daylight-saving handling. The development branch adds payment provider-routing (607 + 81 tests there), not yet deployed.
Admin console (web)	**Deployed and verified live** on the demo environment, including timezone-labelled scheduling, live queue and availability generation.
Shared mobile core	**39 automated tests passing;** static analysis clean. Patient and doctor flavors build from one codebase.
Android apps	**Debug APKs built** for patient and doctor flavors; package signatures and demo API endpoint checked. Installation and device flows not yet verified on hardware.
iOS apps	**Prepared** (schemes, bundle IDs, permissions declared); store builds require a Mac signing environment and the hospital's Apple account. Not yet run.
Push notifications	**Per-deployment wiring** with Firebase/APNs required; not connected on the demo.
Payments	**Booking-to-payment flow covered by automated API integration tests** using an injected test gateway; no live provider calls, no real charges. Live charging activates with the hospital's production payment keys.
Video consultations	Provisioned on the demo with Zego; token issuance covered by automated tests. End-to-end live-call verification is part of each go-live.

> **What a demo includes:** a working hospital workspace with synthetic demo data: live appointments, a running queue, availability generation, and booking and payment-state screens covered by automated API integration tests. Provider checkout is not verified end to end. The demo contains no real patient data.

Where the platform goes next.

Roadmap items are labeled as such everywhere we present them. Nothing below is claimed as working today.

Market readiness (in progress)

Payment and SMS provider routing

The payment provider interface is being generalized on the development branch so providers such as Stripe can be added; SMS routing follows the same pattern. No Stripe or Twilio adapter exists yet; specific providers are scoped per market.

Regional compliance packs

Deployment-specific compliance requirements (for example US or UAE regimes and data residency) scoped and packaged per market.

iOS store builds

Schemes, bundle IDs and permissions are prepared; release signing runs on a Mac build environment with the hospital's Apple account.

Production hardening

Demo conveniences (like the fixed demo OTP) are removed and push notification wiring is completed as part of each go-live.

Phase 2 platform features (scoped per client)

Live HMIS / FHIR sync

Two-way integration with the hospital's HMIS. A per-hospital adapter with real integration time budgeted; the booking model already supports it.

AI patient-history search

Natural-language search across patient history. Architecture and feature flag exist; enabling is phase 2 work.

Medicine delivery tracking

Prescription-to-door tracking. Flag exists; not enabled.

Remote patient monitoring

Wearable and RPM data feeds for chronic-care programs. Early design stage.

> **Geotagged doctor allocation** across a group's hospitals is designed into the allocation module and activated where a client needs it.

What your launch needs.

The known launch steps are listed below and scoped in your written proposal, so they are planned and priced before you sign.

- **Provider accounts on your side:** video (Zego or equivalent), SMS, payments, push (Firebase/APNs) and maps, on the hospital's own accounts, billed at cost.
- **Store presence:** hospital-owned Apple and Google developer accounts; iOS release builds prepared on a Mac signing environment arranged for the engagement.
- **Device verification:** patient and doctor apps tested on real phones before public release.
- **Compliance review:** your compliance reviewer signs off the declared data-touching vendors before real patient data flows.
- **Production cutover:** production keys, demo conveniences removed, deployment verified end to end, staff onboarded.

See it running today.

Watch the combined patient, doctor and admin walkthrough, open the live demo workspace, or book a free discovery call.

matrytech.com/products/telemedicine

Demo sign-in: telemedicine.matry.tech · info@matrytech.com · +91 70044 88728

> **Our promise on honesty:** this document separates working and verified features from in-progress and roadmap items. If a capability matters to your decision, ask us to show it running on the demo before you sign.