Everyday Dental Surgery — Comprehensive Technical Overview

Release summary

Latest update: 14 October 2026 • Latest verified implementation: 5 October 2026

September 2026 • Independent build • Client and developer edition

Everyday Dental presents a dental-practice website, an interactive sample patient journey and an evidence-based engineering plan. It gives reviewers a working interface and inspectable implementation without asking them to submit real health information. The public deployment is available at Open the dental site.

The release includes bilingual presentation, responsive layouts, sample privacy controls, explicit access-denial examples, first-party media, restrictive browser headers and an isolated static origin. It also preserves a historical Supabase clinical backend for technical review. All 11 historical Edge Function endpoints are now disabled by a deployed guard following explicit owner approval. Existing database records were preserved; no migrations or database mutation commands were executed.

This is not a working clinical patient portal, HIPAA/GDPR certification, full WCAG conformance statement or proven million-user platform. The value for a client is the separation of implemented behavior, tested evidence, known defects and future acceptance gates. A fresh-context review has approved the public documentation publication. Independent clinical implementation and security review remain required before real patient use.

October 2026 release update

Two public static releases on 5 October improved how the site is shared and identified. LinkedInBot can fetch a link preview while the wildcard robots rule still disallows other crawlers; the page retains noindex metadata and an X-Robots-Tag header. LinkedIn Post Inspector fetched the page title and description, but an image warning remains open.

The configured site origin now produces the only canonical link on each page. Old page-level links to a retired host and a placeholder domain were removed. The retired-host CORS fallback was removed from preserved backend source, but the 11 guarded Edge Functions were not redeployed and continue returning the previously verified 503 boundary. Neither release changed clinical data, migrations, DNS or gateway configuration.

GitHub publication and documentation update

Publication verified on Sep 24, 2026. The frontend, guarded backend and public documentation are published at commit f758b68 on both the default master branch and the existing hardening branch. All 255 published file hashes match the local committed tree. The repository includes nine READMEs and 45 public Markdown documents; all 173 checked relative documentation links resolve.

Open the GitHub repository, published commit or documentation index. The repository description, site homepage and project topics have also been updated.

The publication changes documentation and publishes the existing source snapshot. It does not activate clinical services, apply migrations or redeploy the live website. Earlier deployment measurements in this report remain historical evidence; they were not rerun as part of the GitHub publication. The latest publication checks and their limits are recorded in the verification section below.

The new sections extend the earlier report with Bangladesh launch planning, detailed capacity qualification, reliability and recovery proposals, accountable roadmap milestones and public governance requirements. Existing technical explanations, US/EU research, threat model, deployment history, clinical blockers and media provenance remain part of this report.

A practical walkthrough for reviewers

Start on the home page and review the clinic presentation. Open the sample patient journey from the site notice. The page uses a fixed sample patient and makes its nonclinical status explicit. The role selector illustrates policy decisions; visitors are not authenticating as real staff.

With the patient role, review the sample record, change optional sharing, cancel the sample appointment and download a FHIR-shaped Patient bundle. Optional sharing begins off. Try erasure while the legal hold is active: the operation is deferred. Switch to the sample admin role to release the hold, return to the patient role and erase the sample. Export is then denied until reset. Switch to reception to observe that clinical access and patient consent changes are denied.

These interactions are browser-memory transitions, not authoritative clinical decisions. The activity list is a temporary illustration, not an immutable audit trail. Closing or resetting the page clears the sample state. The downloaded file contains the fixed sample only. Real identity, care assignment, tenant boundaries, lawful retention decisions and durable audit records belong to the future server implementation.

Visit the privacy and engineering page to examine the stated limits and accessibility behavior. The contact, registration, appointment and login layouts remain read-only previews. No clinical client, real payment, email, SMS or live patient intake is enabled by the public application.

Product surfaces and technical structure

The presentation retains home, about, services and service-detail pages; pricing; articles; FAQ; contact; gallery; community; conferences; appointment previews; account-form previews; and privacy/terms routes. Clinic profiles, testimonials, offers, treatment figures and pricing are sample content, not a real clinic's. They must be replaced with verified, jurisdiction-appropriate business content before a real clinic uses the design.

React handles page composition and state. React Router provides navigation and restricted-portal redirects. Vite produces versioned assets. Tailwind and existing motion libraries preserve the design; reduced-motion preferences are respected. The obsolete animation dependency that produced an eval warning was removed. Forty-seven remote stock-image variants are now served from the same site, with source and license information retained.

The new configuration boundary validates the public HTTPS origin and accepts sample-data mode only. Sample records, workflow messages, disclosures and image mappings have maintained data files. Deployment settings own the pinned server image, resource bounds and cache settings. Tooling settings own local test limits and capacity assumptions. Legacy clinical configuration and older embedded presentation content still need consolidation before clinical reuse.

The public request path is browser → CDN → HTTPS gateway → loopback static container. No patient query is needed to render a page or execute the sample journey. The origin runs as a non-root user with a read-only filesystem, dropped capabilities and explicit resource limits. Only the static build and generated server configuration are deployed; credentials, SQL, source and recovery files are excluded from the web root.

Historical clinical architecture and engineering debt

The preserved backend contains 11 migrations and 11 Edge Functions covering form submission, patient data, cancellation, administrative queries and role management, request resolution, FHIR access/export, notifications, breach checks and payments. Earlier modules also sketch encrypted fields, database authorization, consent, retention and audit concerns. Source presence is not evidence that those concerns are correctly implemented end to end.

The prior implementation had material contract and security gaps: plaintext writers did not consistently match the encrypted schema, patient-controlled query selection and FHIR ownership needed repair, privileged notification/breach operations required stronger caller authority, payment verification and idempotency were incomplete, and rate limiting could fail open. The release guard contains these functions; it does not repair those defects or establish that the database REST/Auth/Storage configuration is safe for real records.

Clinical activation requires its own isolated test database, verified schema upgrade path, complete server-side authorization, operational contracts, region decisions and independent security review. Removing a guard is not an acceptable shortcut. The clinical-release section later in this report lists the repair and verification obligations in detail.

Engineering approach and handoff

The implementation uses small, testable policy functions and explicit boundaries rather than simulated production credentials. Real browser checks complement unit tests. Deployment work begins with a live snapshot and local comparison, then a versioned local build, runtime checks and deployed-file hashes. Recovery archives and project checkpoints allow another developer to resume after workstation power loss.

The remaining sections provide the legal research, security model, staged capacity and availability design, operational runbook, clinical release gates and exact verification limits. They should be read together: a law citation does not prove compliance, a passing test does not prove an SLA, and a planning calculation does not prove production capacity.

Frontend, backend and developer documentation

The frontend guide maps routes, components, configuration ownership and the local workflow. Node 24+ and the committed lockfile support npm ci, npm run dev, npm run check and npm run preview. Local development requires no clinical credentials. The browser suite runs against the local preview by default; testing a deployed target requires authorization.

The backend guide, function inventory and migration guide describe the 11 guarded Deno/TypeScript functions and 11 historical SQL migrations. The backend is not a one-command clinical bootstrap: isolated configuration, test fixtures, fresh/upgrade migration tests and all clinical blockers must be addressed before activation.

The API reference specifies the current handler-level 503 guard and proposed future contracts for identity, authorization, input, idempotency, pagination, errors, caching, jobs, webhooks and audit. Hosting gateways may reject a request before the handler. A generic 503 without the expected body is not proof that the release guard executed.

The configuration guide distinguishes public runtime settings, test-only settings, maintained content and dormant backend configuration debt. VITE_ variables are browser-visible and must never contain secrets. The architecture separates public delivery from future authenticated clinical services, bounded database connections, transactional outbox/jobs, restricted audit and recovery.

Use testing, contributing, security reporting and support for reproducible changes and private vulnerability handling. No project-wide open-source license has been selected; see licensing and reuse. Publishing source does not establish permission to reuse every third-party asset commercially.

US and EU privacy, health-data and accessibility readiness

Executive assessment

Everyday Dental is an independent portfolio build, not a clinic's live website. Its useful professional claim is that it shows selected privacy and security decisions, documents their limits, and provides a reviewable route to a production system. It is not a certified clinical platform. The sample patient journey uses fixed sample records in browser memory. Public intake is disabled, and the shipped frontend cannot connect to the historical clinical backend through environment variables.

This distinction is important commercially. A client can examine the design, tests and release evidence without trusting invented compliance badges. The architecture supports a future separated clinical service, but deploying that service is a new engineering and governance milestone. It is not a configuration toggle. Findings below distinguish legal requirements, applicability questions and project recommendations. Research was checked against primary sources in September 2026; jurisdiction and product changes require reassessment.

Scope and applicability

The baseline covers US federal health/privacy law, California and Washington examples, EU GDPR and ePrivacy, accessibility, and future European health-data interoperability. It does not purport to cover every US state or every EU member state. A real launch needs a named operating entity, patient locations, service locations, data flows, vendors and purposes before counsel can complete the applicability analysis.

- Deployment: Public portfolio • Recommended boundary: Sample content, no real patient intake, no payment or messaging • Additional review before launch: Truthful claims, accessibility, hosting metadata and asset rights
- Deployment: Clinic brochure/contact site • Recommended boundary: Minimum administrative contact data • Additional review before launch: Advertising, privacy notice, retention, tracking, patient expectations
- Deployment: Patient portal • Recommended boundary: Verified identity and server-side access controls • Additional review before launch: Health confidentiality, contracts, consent/legal basis, rights and incident handling
- Deployment: Multi-clinic SaaS • Recommended boundary: Tenant-scoped clinical service • Additional review before launch: Processor/business-associate obligations, regional hosting, auditability, vendor chain
- Deployment: Clinical AI or diagnostic software • Recommended boundary: Separate product classification • Additional review before launch: Medical-device and AI-specific assessment; no inference from this site

Sample records do not eliminate all privacy considerations: hosting providers still handle network traffic. The release avoids application analytics and automatic third-party media loads. An operator must still understand provider security logs, contractual roles, retention and cross-border processing. “No patient database” is narrower and more defensible than “we process no personal data.”

United States

HIPAA applicability and contracts

HIPAA applies to covered entities and business associates, not automatically to every health-themed website. A healthcare provider conducting covered electronic transactions may be a covered entity. A vendor handling protected health information on its behalf may be a business associate. Cloud storage can create business-associate responsibilities even when a provider does not hold the decryption key. Appropriate BAAs and a shared responsibility allocation are essential before a relevant cloud service handles ePHI.[1]

Project decision: do not accept real patient records on the site. Do not describe HTTPS, encryption libraries or a policy folder as proof of HIPAA compliance. A real engagement should maintain a vendor matrix recording service, purpose, data classes, BAA status, subcontractors, access roles and termination/export process.

Supabase’s documented HIPAA configuration requires a signed BAA and its HIPAA add-on for PHI, with designated high-compliance projects. Its shared-responsibility documentation identifies the required plan level. The old repository instruction that an ordinary Pro subscription alone enables HIPAA use was therefore unsuitable. No paid plan or add-on is activated by this work.[2]

Security Rule and evidence

HHS describes administrative, physical and technical safeguards, risk analysis, access management, contingency planning and evaluation. The Security Rule’s documentation retention requirement is distinct from a clinic’s medical-record retention schedule. A universal “delete all health records after six years” rule would be an unsafe interpretation.[3]

Recommended clinical evidence: maintain an asset inventory and risk register; require staff MFA; verify role and patient/tenant scope on every request; rotate and recover encryption keys; review audit access; rehearse restoration; train workforce members; review vendors; and record incident decisions. Access-control code needs negative tests, including another patient, another tenant, expired sessions, revoked roles and failed audit persistence. Browser role selectors and inactivity timers are not server enforcement.

HHS still describes the Security Rule cybersecurity update as a proposed rule and states that the current rule remains in effect during rulemaking. Proposed MFA/encryption requirements should not be represented as enacted solely because they are desirable design choices. Track the rulemaking at release time.[4]

Health-data breach response

HIPAA breach notification generally requires individual notification without unreasonable delay and no later than 60 days after discovery; reporting obligations differ with breach size and circumstances. This is an outer deadline, not a recommendation to wait. Document assessment, discovery, containment, affected data, recipient decisions and the basis for notification.[5]

The FTC’s Health Breach Notification Rule can cover qualifying personal-health-record vendors and related entities outside HIPAA. The 2024 amendments clarify coverage and unauthorized disclosures. A health app should not assume that exclusion from HIPAA means no health-breach obligations. Applicable organizations need an incident path aligned with the rule, alongside any state duties.[6]

Recommended implementation: retain a restricted incident register with discovery time and jurisdiction-specific deadlines, identity of the incident owner, evidence links and approval history. Compute deadlines by jurisdiction; do not hardcode a single “60-day compliance timer” for all incidents. Protect evidence from routine purges and keep notification drafts separate from ordinary product email.

California

The CCPA applies to covered businesses under its applicability tests; rights and exceptions are contextual. Relevant concepts include disclosure, access, deletion, correction, sale/sharing opt-out, sensitive-information limits and non-discrimination. Medical-information exemptions do not justify assuming every dataset held by a clinic is exempt. Advertising and website activity need their own analysis.[7]

California’s Confidentiality of Medical Information Act separately restricts disclosure of medical information and defines authorization conditions and exceptions. CCPA checklists do not replace medical-confidentiality analysis.[8]

Project decision: no application sale/sharing for advertising, analytics pixels or replay. A Global Privacy Control signal cannot trigger tracking because no application tracking path exists. A real clinic must supply verified privacy contacts, request verification and response procedures, authorized-agent handling and a record-category exemption analysis. Do not publish invented contact details as a working rights-request channel.

Washington

Washington’s My Health My Data Act addresses consumer health data beyond traditional HIPAA coverage. Its text includes consumer-health notices, consent rules, rights, security practices, processor duties, separate authorization for sale and healthcare geofencing restrictions. Scope includes certain residents and health data collected in Washington; exemptions need a data-specific assessment. A broad terms checkbox is not a substitute for the required consent.[9]

Project decision: no geolocation/geofencing, no advertising sale, no embedded map and optional sharing off in the sample. A real product should inventory inferred health data and acquisition sources as well as direct clinical fields. Evaluate the exact consent, deletion, backup and appeal workflow under the statute; do not assume a generic GDPR screen is enough.

European Union

GDPR legal bases and responsibility

GDPR applicability depends on establishment and relevant offering/monitoring activities, not merely whether a site is reachable in Europe. Health data are a special category. A real clinical service needs an Article 6 basis and a valid Article 9 condition, with national healthcare/secrecy requirements considered. Consent is not the universal basis for all treatment processing. Privacy by design, minimization, transparency, retention, processor contracts and risk-appropriate safeguards must be reflected in the actual data lifecycle.[10]

Recommended design: separate treatment records from marketing preferences, retain the purpose and legal basis of each processing activity, and require explicit approval when adding a new use. Record why a field is necessary. An intake form should not demand clinical history when a

scheduling request needs only an appointment category and a secure callback channel. Avoid sending clinical text in URLs, analytics labels, emails or support screenshots.

The sample's optional sharing toggle intentionally does not control the legal basis for all treatment. Its legal hold illustrates why an erasure request may need review. These are educational workflow examples; they do not decide a real person's legal entitlement.

Security, incidents and international transfers

The EDPB's small-business guidance connects security to organizational and technical measures, including access control, backups and incident response. Under GDPR, an applicable controller generally notifies the authority within 72 hours after becoming aware of a qualifying personal-data breach, unless the risk exception applies; high-risk individual communication is a separate assessment. Keep the reasoning and evidence, including decisions not to notify.[11]

Hosting in an EU region alone does not settle international transfers. Review support access, subprocessors and onward transfers, then identify a valid transfer mechanism where required. The EDPB describes mechanisms and safeguards for transfers outside the EEA; assess the actual vendor arrangement and supplementary measures rather than treating geography as certification.[12]

Recommended clinical implementation: region-specific storage boundaries, least-privilege support access, encryption with documented key ownership, restricted exports, auditable access requests and a vendor exit plan. Test restoration into the authorized region. Do not put genuine patient records into a public object cache or an unapproved observability provider.

ePrivacy and browser storage

CNIL guidance explains prior consent for non-essential trackers, with exceptions for necessary functions, and requires a meaningful ability to refuse and withdraw. Broad acceptance of terms does not establish valid tracker consent. Language preferences may be an expected interface function, but that is not permission for analytics or fingerprinting.[13]

Implemented design: first-party stock images, no automatic external map, no advertising/analytics scripts, no patient information in persistent browser storage, and a language-preference clearing control. The sample resets when the tab reloads. Because there are no optional tracking technologies, adding a decorative "accept all cookies" banner would misdescribe this product. If tracking is added later, inventory it first, gate it before consent where required, and test refusal and withdrawal equally carefully.

Accessibility and member-state variation

US DOJ guidance identifies accessibility obligations for covered public accommodations and discusses barriers such as missing alternatives, poor contrast and inaccessible forms. A private clinic's duties should not be confused with the separate Title II rules for state/local government services.[14]

The European Accessibility Act applies to specified products and services, including relevant e-commerce services; it is not an automatic blanket rule for every brochure website. Its scope, microenterprise service exemptions and national implementation need assessment. National disability and healthcare duties may still apply independently.[15]

WCAG 2.2 provides a technical standard for evaluating accessibility. Keyboard operability, visible focus, labels, status announcements, text alternatives, contrast, reflow and motion preferences are practical test targets. Automated checks only cover part of accessibility; manual keyboard and assistive-technology evaluation remain necessary.[16]

Implemented/recommended boundary: the new review pages include semantic headings, keyboard controls, visible focus, a skip link and announced workflow results. Reduced motion is supported. Legacy cinematic pages need broader manual evaluation, including zoom, screen readers, vestibular comfort and accessible error recovery. Do not display a conformance badge based only on an automated scan.

Before onboarding a clinic in a named member state, confirm professional registration/advertising requirements, minor-consent rules, record-retention periods, secrecy obligations, accessibility rules, required site legal notices, supervisory authority and relevant national health-hosting requirements. These remain a launch checklist because no target country or actual clinic was specified.

EHDS and future product expansion

The European Commission describes phased application of the European Health Data Space, including major primary-use milestones in 2029 and 2031. A FHIR-shaped JSON export does not prove EHDS interoperability, conformity assessment, national integration or patient identity matching.[17]

The site makes no clinical AI decision, diagnosis, prescription or medical-device claim. Adding AI or clinical decision support requires a separate intended-purpose and regulatory-classification review. The project should not invent an AI Act or medical-device certification for functionality it does not implement.

Implementation matrix

- Concern: Patient data collection • This public release: Disabled; fixed sample workflow • Production evidence still required: Final schema, lawful purpose, data minimization and negative authorization tests
- Concern: Identity and access • This public release: Sample role-policy model • Production evidence still required: Verified identity, MFA, tenant/patient scope, session revocation, care-assignment policy
- Concern: Consent • This public release: Optional sharing off; reversible sample toggle • Production evidence still required: Versioned notices, proof, purpose separation and downstream withdrawal propagation
- Concern: Erasure/retention • This public release: Sample hold and erasure transition • Production evidence still required: Legal schedule, restricted review, backup strategy and verified deletion/exception response
- Concern: Export • This public release: Sample Patient bundle • Production evidence still required: Partner implementation guide, authorized scope, terminology, validation and secure delivery
- Concern: Audit • This public release: Temporary sample outcomes • Production evidence still required: Restricted durable server events, tamper evidence, monitoring and retention
- Concern: Tracking • This public release: No application analytics or advertising • Production evidence still required: Provider inventory and approval before adding new processors
- Concern: Encryption • This public release: HTTPS and no clinical database on the site • Production evidence still required: Key management, encrypted storage/backups, recovery and rotation tests
- Concern: Incidents • This public release: Documented response design • Production evidence still required: Named responders, tested deadlines, evidence handling and exercises

- Concern: Availability • This public release: Stated target and repeatable probes • Production evidence still required: Sustained measurements and funded redundancy for stronger guarantees

Operating model and client acceptance

The operator should own legal applicability, approved vendors, retention decisions and notices. Engineering owns the implemented controls and reproducible evidence. A privacy/security lead owns risk review and incident coordination. These responsibilities need named people in a real engagement; a repository cannot appoint an absent compliance officer.

A client acceptance review should walk through one normal and one denied action per role, one withdrawal, one export, one hold/deletion request and one restore. Review actual network requests and logs. Verify that a changed role takes effect immediately and that failure of a dependency does not open access. Budget and contracts must be approved before sending real data to any new service.

Useful commercial wording is: “Dental-workflow build with privacy-conscious delivery, tested sample access policies, documented production gates and a staged scaling design.” Avoid “HIPAA certified,” “GDPR compliant worldwide,” “SOC 2 audited,” “one million concurrent users supported,” or “guaranteed uptime” without the corresponding independent and operational evidence.

Sources

1. HHS. Guidance on HIPAA and Cloud Computing.
2. Supabase. HIPAA Projects and Shared Responsibility Model.
3. HHS. Summary of the HIPAA Security Rule.
4. HHS. HIPAA Security Rule NPRM. Proposed rule; current rule remains applicable during rulemaking.
5. HHS. Breach Notification Rule.
6. FTC. Complying with the Health Breach Notification Rule. Includes amendments effective in 2024.
7. California DOJ. California Consumer Privacy Act.
8. California Legislature. Civil Code, Confidentiality of Medical Information, Chapter 1.
9. Washington Legislature. RCW 19.373, My Health My Data Act.
10. European Union. Regulation (EU) 2016/679, especially Articles 3, 5, 6, 9, 17, 25, 28, 32–35 and 44 onward.
11. EDPB. Secure personal data.
12. EDPB. International data transfers.
13. CNIL. Cookies et traceurs : que dit la loi ?.
14. US DOJ. Guidance on Web Accessibility and the ADA.
15. European Union. Directive (EU) 2019/882, European Accessibility Act.
16. W3C. Web Content Accessibility Guidelines 2.2.
17. European Commission. European Health Data Space Regulation.

Bangladesh legal and clinical launch planning

Research date: Sep 24, 2026. This is a source-backed engineering launch checklist, not a legal opinion or a statement that the site operates a licensed clinic. Review Bengali statutory text, current regulations, gazettes and the actual service model with qualified local counsel before patient intake.

Current legal baseline

The official database lists the Personal Data Protection Act, 2026 (Act 63), replacing the 2025 ordinance. Section 1 treats most provisions as effective from Nov 6, 2025, while sections 23 and 31–35 require later commencement by gazette after the specified 18-month period. Health data is sensitive personal data. The Act addresses consent, children, rights, security, retention and processors. Section 29 regulates cross-border transfers and specified large transfers requiring notification. It does not justify copying the superseded ordinance's cloud-localization wording into a current plan. Exact classification, regulations and commencement must be checked at launch. Official Act.

The Cyber Protection Act, 2026 (Act 81) replaced the 2025 cyber ordinances. The official text also records a 2026 amendment (Act 99). Review the consolidated law, not an old Cyber Security Act 2023 checklist. Current opening provisions, amended offence provisions, repeal provisions.

Proposed data-protection implementation

The following are project engineering requirements to validate against the Act and implementing rules, not quotations of universal legal deadlines:

- Workflow: Intake • Proposed project control: Collect minimum contact/clinical fields; show reviewed Bangla/English notice before submission • Launch evidence: Approved field/purpose inventory and notice versions
- Workflow: Consent • Proposed project control: Separate purposes; record notice version and positive action; allow withdrawal • Launch evidence: Consent/withdrawal tests and legal basis review
- Workflow: Children/guardians • Proposed project control: Verify authority through a designed workflow; avoid excessive identity collection • Launch evidence: Clinical/legal guardian policy and denial tests
- Workflow: Rights • Proposed project control: Verified access/correction/export/deletion request queue with jurisdiction-specific deadlines • Launch evidence: Request simulation, exceptions and escalation records
- Workflow: Confidentiality • Proposed project control: Staff MFA, care assignment, tenant/subject scope and minimized audit payloads • Launch evidence: Negative authorization and privileged-access review
- Workflow: Retention • Proposed project control: Approved category-specific schedule and legal holds; no blanket automatic record deletion • Launch evidence: Retention decision record and dry-run/restore tests
- Workflow: Transfers • Proposed project control: Inventory primary/backup/log/support locations and each processor • Launch evidence: Approved transfer analysis and contractual protections
- Workflow: Incident • Proposed project control: Record discovery and preserve evidence; assess current local notification requirements • Launch evidence: Named local response owner and exercised contact procedure

Do not import GDPR's 72-hour notification timer as Bangladesh law. Do not treat consent as a blanket permission for any overseas service. Authority notifications, registration/officer obligations and implementing procedures need current local review, including any delayed commencement.

Professional duties and content

BMDC publishes a professional ethics code with medical-record confidentiality and professional-information/promotion sections. Use it to review actual clinician identity, qualifications, clinical claims, patient images, testimonials and disclosure workflows. The official site also publishes telemedicine guidelines; adding video/chat diagnosis or prescribing requires a separate clinical and professional-scope assessment. A booking website does not establish permission for telemedicine. BMDC, ethics code, telemedicine guidelines.

Proposed project gates: verify practitioners against the relevant register; replace sample profiles with approved facts; obtain specific permissions for patient media; review advertising and outcome statements; define record access, amendment and emergency referral; and confirm whether any proposed remote dental service is within professional scope. No production medical advice or prescribing is enabled by this repository.

Facility and business operation

The 1982 Medical Practice and Private Clinics and Laboratories Ordinance regulates medical practice and private clinics/laboratories. Its definition of private clinic concerns facilities where patients are admitted and kept for treatment; do not automatically classify every outpatient dental chamber identically. DGHS publishes private-facility registration guidance. Confirm the actual facility category and applicable licensing route. Official ordinance, DGHS registration guidance.

Local launch decisions still required: operating entity and trade permissions, tax/VAT applicability, clinic registration/renewal, equipment/radiology and waste permissions where relevant, clinician supervision, records/cash receipts, advertised charges, cancellation/refunds and complaint contacts. These are a review inventory, not a finding that each permit applies to every deployment.

Hosting, payments and million-user expansion

Proposed architecture keeps clinical data segregated from public CDN delivery and records all transfer/backup paths. Select regions only after the legal data-classification decision. Assess whether scale changes regulatory classification, audit/officer requirements or risk-assessment scope. A million-user target cannot be used to justify collecting extra identity data.

Verify payment-provider country availability, merchant eligibility, currency/settlement, refund rules and contracts before implementation. The preserved Stripe integration is not evidence that a Bangladesh operator can onboard to Stripe. Use provider sandbox tests only until the commercial and clinical release gates are approved.

Open decisions before launch

A named operator must approve the final statutory version/commencement register, implementing-rule review, data classification/transfer position, professional/facility licensing, rights/incident deadlines, notices, retention, vendor terms and clinical escalation. This task does not establish those operator-specific approvals. Recheck the official legal database at each launch and material scope change; the 2025-to-2026 replacement shows why old checklists can become unsafe.

Threat model and release boundaries

Assets and trust boundaries

The public release contains sample clinic content, stock media, a sample Patient fixture, deployment configuration and source code. It holds no clinical account session, payment key or patient database connection. A visitor controls their browser, URL and role selector. Browser state must therefore never be trusted as production identity or permission.

Trust boundaries are: visitor → CDN; CDN/direct-origin visitor → TLS gateway; gateway → loopback static container; static files → browser; operator workstation → pinned SSH deployment; and maintainer → Git/CI. The preserved Supabase backend is a separate, unapproved clinical surface. Its runtime controls cannot be inferred from the public static site.

Threat and control matrix

- Threat: Accidental patient intake • Implemented control: Disabled form previews; no backend client; fixed fixture-only sample • Verification / remaining boundary: Browser checks find no enabled personal-data inputs; guard regression tests
- Threat: Unauthorized clinical backend activation • Implemented control: Unconditional shared release guard on every historical Edge entrypoint • Verification / remaining boundary: Local boundary tests; remote deployment must be recorded separately
- Threat: XSS / injected dependencies • Implemented control: No inline scripts/eval permission; React escaping; dependency updates; strict CSP • Verification / remaining boundary: Header/browser checks and npm audit; style-src permits inline styles for current UI
- Threat: Tracking and third-party disclosures • Implemented control: First-party image variants; no automatic fonts/maps/analytics; no-referrer • Verification / remaining boundary: Browser request inventory; network metadata still reaches hosting/CDN
- Threat: Client patient-data persistence • Implemented control: Retired storage mock; sample memory only; old app cache retirement • Verification / remaining boundary: Source and browser tests; exported sample file deliberately downloads to user device
- Threat: Clickjacking • Implemented control: frame-ancestors none and X-Frame-Options DENY • Verification / remaining boundary: HTTP headers
- Threat: Secret/source exposure • Implemented control: Static artifact allowlist, dotfile/source/config exclusions, no source maps • Verification / remaining boundary: Negative HTTP path tests and staged/history secret scan
- Threat: Origin/container compromise • Implemented control: Non-root, read-only filesystem, dropped capabilities, no-new-privileges, bounded resources, loopback-only publishing • Verification / remaining boundary: Compose validation and actual runtime inspection
- Threat: Resource exhaustion • Implemented control: Static-only request surface, GET/HEAD, request/body time limits, bounded resource allocation • Verification / remaining boundary: Does not provide a contractual DDoS guarantee; no large shared-origin load test
- Threat: Supply-chain regression • Implemented control: Lockfile, dependency audit, gitleaks, typed new boundary, policy tests and lint • Verification / remaining boundary: Review registry updates and independent review before production
- Threat: Unsafe deployment/rollback • Implemented control: Pre-edit drift snapshot, versioned release, exact hashes, off-server archive • Verification / remaining boundary: Verify restore and record release before advertising completion
- Threat: Misleading compliance claims • Implemented control: Research-based limits, clinical blocker register, measured vs target separation • Verification / remaining boundary: No certifications, million-user claim or observed monthly uptime without evidence

Security assumptions and residual risks

TLS gateway administration is shared with other applications. The VPS remains a single failure domain, and the existing infrastructure permits origin web access. Cloudflare does not replace origin authorization. This site has no sensitive API to authenticate, but a future clinical service must enforce authorization even when the CDN is bypassed.

The sample audit list is temporary and editable by a browser owner. It is an illustration, not a protected audit trail. The sample staff policy is intentionally simplified and does not model care assignment, tenant membership or emergency access. Those are production release blockers.

Original clinical modules and migrations remain readable for design review, with known defects documented. Their presence is not shipped clinical functionality. A source scanner score from compliance/ is a historical pattern check; it does not measure legal compliance, attack resistance or runtime correctness.

Incident and change procedure

For a suspected compromise: preserve evidence privately; disable the affected route/service using a reviewed local change; identify affected assets and versions; compare deployment hashes; rotate exposed credentials through approved stores; assess legal notification obligations; restore a verified release; and add a defect-log entry identifying the missed gate. Do not publish secrets or patient content in issues or commit messages.

Before changes: update the project checkpoint, research the approach, inventory configurable values and write acceptance tests. After changes: run relevant gates, record before/after evidence, update the dossier first, and derive public statements from verified results.

Reference framework: OWASP Application Security Verification Standard. The project borrows review concepts; it does not claim full ASVS verification.

Capacity and availability design

Current release

The public release serves static HTML, JavaScript, CSS and images. The sample patient journey executes in browser memory and causes no clinical database requests. This keeps the site independent of a paused database and avoids maintaining patient data merely to prove design. It does not prove the original clinical backend is scalable.

The initial Supabase CLI inventory found the linked dental project inactive. The owner subsequently restored it through the dashboard. It has not been deleted or replaced. The public site deliberately does not depend on that service. A free service that can pause on inactivity cannot be assumed to deliver a sustained availability commitment. See Supabase project pausing.

Workload definitions

“Simultaneous users” is ambiguous. Open tabs, actively browsing readers, authenticated sessions, concurrent HTTP requests and database transactions have different resource requirements. Agree on the workload before capacity acceptance.

The following is planning arithmetic: each active user performs one action every 30 seconds, with three HTTP requests per action. At a 95% cache hit rate, five percent of those requests reach the origin. A clinical API cannot apply that public-cache assumption to personalized responses.

- Active readers: 10,000 • Edge requests/second: 1,000 • Origin requests/second at 95% hit: 50
- Active readers: 100,000 • Edge requests/second: 10,000 • Origin requests/second at 95% hit: 500
- Active readers: 1,000,000 • Edge requests/second: 100,000 • Origin requests/second at 95% hit: 5,000

node scripts/capacity.mjs reproduces the table from config/tooling.json. The numbers are a workload model, not tested capacity. At zero cache hits, the million-reader scenario reaches 100,000 origin requests/second. If each response is 100 KB, that is approximately 10 GB/second before protocol overhead. Cold caches, launches and cache purges are meaningful failure scenarios.

Cloudflare does not cache HTML or JSON by default; eligible assets and origin directives must be checked separately. An origin public header alone does not prove edge HTML caching. Personalized records, authentication, exports and error responses must not enter a shared public cache. Verify cache behavior with response headers and separate cold/warm measurements. Cloudflare cache behavior.

Staged architecture

Stage A: current site

Use a CDN for eligible immutable assets and a bounded static origin. Keep hashed asset filenames, compression, explicit routes, a small health endpoint, no public write API and no database dependency. Limit the origin container’s CPU, memory and process count and bind it to loopback behind HTTPS. Retain versioned releases and an off-server archive.

Admission gate: browser flows, read-only form boundary, headers, isolated deployment, health checks and restore/parity evidence. Run a small loopback HTTP test; do not stress a shared production server. Current measurements belong in VERIFICATION.md, not in this planning table.

Stage B: measured 10,000-reader target

Create an isolated test environment. Generate the agreed traffic mix including images, page transitions, cold requests and slow connections. Measure p50/p95/p99 latency, errors, origin CPU/memory, egress and cache hit rate. Repeat with a deployment in progress. Define pass thresholds before the test and retain generator saturation evidence; an overloaded load generator cannot establish server capacity.

If introducing real clinical workflows, split public delivery from the authenticated service immediately. Use a separately tested identity provider, staff MFA, short-lived sessions, server-side role/patient/tenant scope, bounded payloads, timeouts, idempotency and audit persistence. Cache public service descriptions; never use those caches for patient responses.

Stage C: 100,000-reader and transactional growth

Introduce multiple origins across failure domains when measurements or availability requirements justify them. Regional ingress directs public requests to healthy origins. Clinical services remain stateless where practical; authoritative patient data stays in its approved region. Use a bounded connection pool, indexed tenant/subject queries, keyset pagination, query deadlines and per-operation concurrency limits.

Do not map one user to one database connection. PostgreSQL allocates resources based on connection limits, so indiscriminately increasing max_connections is not scaling. Benchmark the pool against actual query latency and database capacity. Replica lag must not make permissions, consent withdrawal or appointment confirmation stale. PostgreSQL connection settings.

Move notifications, large exports and noninteractive work into queues with idempotent consumers, retry budgets, backoff and a dead-letter path. A booking confirmation must be based on a committed authoritative record, not optimistic email delivery. Apply backpressure and return a controlled failure when a queue or downstream service is saturated.

Stage D: million-reader design target

At this scale, architecture and operations need separately funded capacity testing, regional delivery, traffic engineering and a staffed response model. Test hot keys, cold cache storms, failed regions, certificate/DNS incidents, database failover, queue backlog and restoration. Partition clinical data only after tenant boundaries, access patterns, regional law and recovery behavior are understood.

A million open static tabs is not a million simultaneous clinical transactions. The latter demands a different budget, workload model and data architecture. This repository offers a migration path and reproducible small-scale tools; it does not make either workload “easy” by assertion. Kubernetes is optional and is not installed merely to make the repository look sophisticated.

Availability objectives

A service-level indicator measures behavior; an objective defines its target; an SLA is a contractual commitment. Keep these separate. Google SRE: Service Level Objectives.

Proposed public availability objective: 99% of one-minute observation intervals over a rolling 30-day window have a valid TLS response and expected health body within the agreed latency limit. Count planned maintenance as unavailable unless a contract explicitly says otherwise. A 30-day window contains 43,200 minutes; one percent is 432 minutes, or 7 hours 12 minutes. Do not mix this time-based budget with a request-success denominator.

Clinical availability needs separate indicators for login, reading a permitted record, submitting an appointment and performing an authorized write. A static health page says nothing about database correctness. Measure from at least two independent locations and investigate disagreements; do not mask a regional outage by averaging it away.

The release includes scripts/probe.mjs, which emits a timestamp, success and latency without credentials or a patient identifier. Scheduling it on an independent always-on monitor is an operator task. A laptop that loses power is not an availability monitor. No 30-day observation period or external monitoring SLA is claimed by this deployment.

Recovery design

- Recovery concern: State • Public site: Static release + configuration • Future clinical service: Database, uploads, keys, identity and audit state
- Recovery concern: Backup • Public site: Off-server release archive with SHA256 manifest • Future clinical service: Application-consistent encrypted backups and protected key recovery
- Recovery concern: Restore test • Public site: Extract archive and compare all release files • Future clinical service: Restore to isolated region-approved environment; test application reads/writes
- Recovery concern: RPO • Public site: No user-generated clinical state in this release • Future clinical service: Proposed target set by client; validate against backup/replication cadence
- Recovery concern: RTO • Public site: Measure actual release rollback/restore • Future clinical service: Proposed target set by clinical impact; include DNS, keys, identity and staffing
- Recovery concern: Failure domain • Public site: One shared origin • Future clinical service: Multiple independently recoverable domains if funded

Do not advertise an unmeasured recovery time. Keep the previous release available. When restoring a clinical backup, reapply deletion/consent restrictions and legal holds where required; restoring a database must not silently resurrect permissions or erased data.

Cost and acceptance boundary

The site reuses existing hosting and free tools. Existing hosting still has the owner’s pre-existing cost; no new paid service is necessary for this static release. Free tiers do not promise unlimited capacity, contractual uptime or health-data processing approval. Before any paid upgrade, obtain explicit approval for price, scope and recurrence.

A production proposal should budget regional hosts, database capacity, backup storage, monitoring, security support, contractual vendor options, test infrastructure and operations. Provider price changes belong in configuration/procurement records, not hardcoded business logic. Progress from one stage only when measurements show why the next investment is necessary.

Capacity qualification: 10k to 1M+ simultaneous users

This is a proposed engineering qualification plan. No large-scale load run is represented as completed. The existing calculator and scalability design explain the current public-reader model.

Define the workload before sizing

Open tabs, active sessions, in-flight requests and database transactions are different quantities. For the public planning model, each active reader performs one action every 30 seconds and each action generates three HTTP requests:

- Edge requests/second = active readers × requests per action ÷ seconds between actions.
- Origin requests/second = edge requests/second × (1 − measured cache-hit ratio).
- Transfer bytes/second = requests/second × measured mean response bytes.
- Average requests in flight ≈ throughput × mean response time, only under a stable measured workload.
- Active readers: 10,000 • Edge requests/s: 1,000 • Origin at 95% hits: 50 • Origin with no cache: 1,000
- Active readers: 100,000 • Edge requests/s: 10,000 • Origin at 95% hits: 500 • Origin with no cache: 10,000
- Active readers: 1,000,000 • Edge requests/s: 100,000 • Origin at 95% hits: 5,000 • Origin with no cache: 100,000
- Active readers: 1,250,000 • Edge requests/s: 125,000 • Origin at 95% hits: 6,250 • Origin with no cache: 125,000

The first three rows are reproduced by node scripts/capacity.mjs; the fourth applies the same assumptions as a proposed 25% headroom target. A 95% hit ratio must be measured by route/content type. At 100 KB average response size, 100k requests/s means approximately 10 GB/s before overhead. Assets and navigation mixes therefore materially affect cost.

Clinical capacity is a separate model

Do not put personalized responses in a shared cache. Define active clinical users, actions/user/minute, API calls/action, read/write ratio, rows/query, transactions/write, authentication refreshes, attachment sizes and export frequency. Budget separate auth, database, object-storage, queue and vendor limits.

Example assumption: 1M active clinical users making one API call every 30 seconds produce about 33,333 API requests/s. An 80/20 read/write mix would yield about 26,667 reads/s and 6,667 writes/s before internal queries, audit writes and retries. This arithmetic is not a selected production workload or proof that the existing backend can support it. Agree on real workflows before sizing.

Measure transaction latency, connection wait, locks/deadlocks, replica lag, hot tenants and audit volume. Bound pool size from measured database capacity; one user must not imply one database connection. Use indexes and keyset pagination, enforce transaction constraints, and partition only after queries and tenant boundaries justify it.

Proposed test profile

At each stage, use isolated infrastructure, sample records and approved spending/traffic limits. Define regions, devices, network conditions and route weights before the run. Suggested starting profile: 10-minute ramp, 60-minute steady load, a 10-minute 2× arrival-rate burst, then a 4-hour soak at the target. These durations are acceptance proposals, not implemented load-script defaults.

- Dimension: Public HTTP • Proposed pass gate: p95 ≤ 500 ms and p99 ≤ 1.5 s in each agreed test region
- Dimension: Clinical reads • Proposed pass gate: p95 ≤ 500 ms and p99 ≤ 1.5 s, measured by operation
- Dimension: Clinical writes • Proposed pass gate: p95 ≤ 1 s and p99 ≤ 2 s through committed result
- Dimension: Unexpected errors • Proposed pass gate: < 0.1% of valid in-scope requests; count overload rejections against admitted capacity
- Dimension: Correctness • Proposed pass gate: Zero cross-tenant disclosures, duplicate committed bookings or lost acknowledged writes
- Dimension: Resources • Proposed pass gate: No unbounded memory/queue growth; proposed 30% steady-state resource headroom
- Dimension: Recovery • Proposed pass gate: Queue drains and latency returns to baseline within the agreed recovery window
- Dimension: Generator validity • Proposed pass gate: Offered/achieved load, dropped iterations, generator CPU/network and clock sync recorded

These latency/error thresholds must be approved for actual workflows and recorded before testing. Intentional authorization failures are separate negative tests; do not remove legitimate-user failures from the success denominator. Read/write counts alone do not establish user-visible performance.

Failure and overload tests

Exercise cold caches, cache purge, hot content/tenant, retry storms, rolling release, expired token bursts, database pool saturation, queue backlog, failed workers, region loss, DNS/TLS failure and restore. Prove backpressure, idempotency, bounded retries with jitter and recovery of queued work. Verify privacy after cache changes and permission withdrawal during replica lag.

Use public immutable caching, healthy independent origins and regional routing for public growth. For clinical growth, keep authoritative writes in an approved data region, use stateless APIs and bounded jobs, and define consistency/replication choices explicitly. Kubernetes and multi-region writes are optional architecture decisions, not prerequisites by name.

Cost model and decision record

Estimate monthly/request-based cost for CDN requests and egress, origins, database compute/IO/storage, replicas/pooling, backups, object storage, queue jobs, monitoring/log retention, security support, load generators and on-call staffing. Price from current provider quotes; record currency, tax, region, included quota, overage and recurrence. This plan does not provision resources or authorize purchases.

A stage report must contain commit/configuration, workload assumptions, test environment, offered/achieved traffic, per-region latency/errors, correctness results, bottleneck traces, failure recovery, cost estimate and approval. Store raw sensitive traces privately; publish scrubbed aggregates. Advance only after a reviewed pass. See Supabase production guidance for staging load testing and PostgreSQL connection settings for connection resource considerations.

Availability, monitoring and recovery objectives

Baseline objective

Proposed public SLO: 99% availability over a rolling 30-day window. This is a target, not a contractual SLA or a measured history. Record objectives separately for public content, clinical authentication, permitted record reads and committed appointment writes. A static health check cannot prove a database transaction worked.

- Time-based target: 99% • Unavailable time per 30 days: 432 minutes = 7 h 12 min • Intended use: Initial public objective
- Time-based target: 99.9% • Unavailable time per 30 days: 43.2 minutes • Intended use: Future staffed/redundant service candidate
- Time-based target: 99.95% • Unavailable time per 30 days: 21.6 minutes • Intended use: Stronger candidate after measured resilience

99% can permit substantial downtime for a clinic; a real clinical service must select a target from patient impact and support commitments. Stronger targets are roadmap decisions, not automatic guarantees. Definitions follow Google SRE.

Measurement contract

Proposed time-based SLI: one observation each minute from at least two independent regions; TLS, expected body and response within the configured latency budget must pass. Maintain regional results. Count a minute good only when all designated required probes pass; treat missing observations as unavailable/unknown conservatively and report coverage separately. Count planned maintenance in the budget. Changes to scope or probe locations require a recorded review, not retroactive removal of outages.

Availability = good observed intervals ÷ all scheduled intervals. Keep request-based indicators separate: successful valid operations ÷ all eligible operations, with per-operation latency and correctness checks. Do not mix interval and request denominators. Record failures at the user boundary as well as origin/database signals.

The repository's probe is a one-shot tool. Scheduling, multiple locations, durable storage, dashboards and paging are future operating work. A local laptop or a server probing itself is not independent evidence.

Alerting and error-budget policy

Proposed policy: page the on-call owner when a critical user journey fails repeatedly from independent probes or security/correctness is compromised. Use multi-window burn-rate alerts so brief noise does not hide sustained failures. At 50% budget consumption, review risks and capacity; at exhaustion, pause discretionary releases until recovery work and review justify resuming. Tune thresholds against actual traffic before adopting them. Google's error-budget policy provides the operating pattern.

A production contract must name service scope, support hours, exclusions, reporting, escalation and any credits. No response-time or support SLA is currently offered by this repository.

Recovery proposals

- Service: Public static release • Proposed RTO: 1 hour • Proposed RPO: Last reviewed artifact • Required evidence: Restore immutable artifact; verify content, TLS and hashes
- Service: Clinical transactional records • Proposed RTO: 1 hour • Proposed RPO: 15 minutes • Required evidence: Funded backup/replication, isolated restore, integrity and access tests
- Service: Jobs and notifications • Proposed RTO: 4 hours • Proposed RPO: Replay from committed outbox • Required evidence: Deduplication, backlog recovery and delivery reconciliation
- Service: Identity, keys and audits • Proposed RTO: Set from clinical risk assessment • Proposed RPO: Set from legal/security needs • Required evidence: Key recovery, revoked access and audit continuity

These are planning candidates requiring owner approval and proven tooling. No provider backup/PITR plan is assumed to be enabled. RPO does not authorize silently losing an acknowledged appointment: reconcile from durable records and follow an approved incident/manual workflow if a recovery gap occurs.

Backups must cover database, uploads, identity configuration, audit state and decryption-key recovery. Keep an independent protected copy, test restoration, and reapply holds/deletion/withdrawal restrictions. Run a restore before clinical launch, then proposed quarterly drills and after material storage changes. Record actual RTO/RPO and failure-domain results. See operations and continuity.

Product and engineering roadmap

Planning baseline: Sep 24, 2026. Milestones are dependency-based, not delivery promises. The current application is the starting point. The destination is a regional clinical platform and a measured 1M+ concurrent-user public experience, with separately qualified clinical traffic.

Milestones and exit criteria

- Stage: A — public foundation • Deliverable: Cohesive source/docs, guarded backend, local tests and safe publication • Accountable role: Maintainer • Exit evidence: Reviewed source, public-safe tree and reproducible verification
- Stage: B — clinical correctness • Deliverable: Typed settings, schema reconciliation, tenant/subject scope, consent, payments and audit • Accountable role: Backend/security leads • Exit evidence: Every clinical blocker closed; migration, RLS, integration and negative tests
- Stage: C — controlled pilot • Deliverable: One approved jurisdiction/operator, support model and recovery • Accountable role: Product/privacy/operations leads • Exit evidence: Legal/vendor approval, staff training, restore rehearsal, limited test-data then approved clinical pilot
- Stage: D — 10k active readers • Deliverable: Measured CDN/API boundaries, performance budgets and monitoring • Accountable role: Performance lead • Exit evidence: Agreed mixed workload passes warm/cold, burst and sustained tests
- Stage: E — 100k active readers • Deliverable: Multi-origin resilience, pooled database, asynchronous jobs and tenant limits • Accountable role: Platform lead • Exit evidence: Failure-domain drill, queue/backpressure and database saturation evidence
- Stage: F — 1M+ active readers • Deliverable: Regional delivery and capacity headroom with rehearsed failover • Accountable role: Platform/operations leads • Exit evidence: Distributed load-generator evidence, 1M baseline plus 25% proposed headroom, regional failure and restore tests
- Stage: G — mature clinical service • Deliverable: Independently qualified clinical workloads and stronger availability • Accountable role: Clinical/product/operations leads • Exit evidence: Contracted workloads, measured SLOs, reviewed access controls and continuing incident exercises

Roles are responsibilities to assign, not claims that a staffed team exists. Clinical launch does not depend on reaching a million public readers; public scale does not establish clinical safety. Dates and budgets follow measured estimates and owner approval.

Workstreams

- Workstream: Product • Next work: Define real clinic, tenant boundaries, care assignment, scheduling and support scope • Dependency / acceptance: Approved requirements and sample journey acceptance
- Workstream: Frontend • Next work: Full typing, localized notices/errors, bundle budgets and accessible form recovery • Dependency / acceptance: Keyboard/screen-reader/zoom and English/Bangla review
- Workstream: Backend/data • Next work: Fixed operation contracts, encryption alignment, booking constraints, outbox/idempotency • Dependency / acceptance: Fresh/upgrade schema tests and denied cross-tenant access
- Workstream: Security • Next work: ASVS mapping, staff MFA, audit integrity, keys and dependency lifecycle • Dependency / acceptance: Independent security review and exercised incident controls
- Workstream: Legal • Next work: US, EU and Bangladesh applicability and current national/state requirements • Dependency / acceptance: Named operator, current sources, approved notices and contracts
- Workstream: Reliability • Next work: Independent probes, service indicators, error-budget policy and recovery drills • Dependency / acceptance: Measured rolling windows and owner-approved objectives
- Workstream: Capacity • Next work: Instrument, model, test and size public and clinical workloads separately • Dependency / acceptance: Capacity plan acceptance report

- Workstream: Commercial • Next work: Ownership/license, vendor terms, support hours, paid infrastructure estimates • Dependency / acceptance: Explicit owner decisions; no automatic paid upgrades

Release gate for real patient data

No pilot opens until the clinical blocker register, governance checklist and tested recovery plan pass. A failed security/correctness gate blocks clinical expansion even if latency is excellent. A breached availability budget prioritizes reliability work before feature releases.

Decision log to maintain

Record workload mix, jurisdiction, data location, tenant isolation model, SLO and support hours, backup/RPO/RTO, provider contracts, budget, licensing and responsible owners. Track unresolved decisions explicitly. Verification records measured outcomes; scalability and reliability own the design.

Deployment, monitoring and recovery

Build and inspect

Use Node 24 or newer within the supported toolchain. Run npm ci, npm run check, npm audit, npm run test:browser against a local preview, and node scripts/security-check.mjs. Run gitleaks over history and staged changes. Retain raw logs privately. node scripts/build-deploy.mjs generates Compose, nginx and gateway-site files from deployment settings and the canonical header policy.

The release consists only of dist/, compose.yaml, nginx.conf, security-headers.conf and the project gateway file. Do not upload source, environment files, credentials, memory, source maps, SQL or local research artifacts to the web root. Settings identify the hostname, pinned image, loopback port and resource bounds in one place.

Deploy safely

- Read the shared infrastructure inventory, then inspect current DNS, listeners, containers and capacity. Confirm project ownership and compare any existing project files with local snapshots.
- Build a new versioned release locally. Generate a SHA256 manifest, save an off-server archive and verify extraction against the manifest.
- Upload only the allowlisted release to its unique release directory. Validate Compose and nginx before starting the project container. Verify loopback health and inspect actual limits/bindings.
- Prepare only this project's gateway site locally. Preserve the shared main configuration and unrelated sites. Validate the whole gateway configuration before reloading.
- Create/update only the confirmed project DNS record. Verify direct-origin certificate validity using hostname SNI, then public TLS, expected content and security/cache headers.
- Compare every deployed release file hash with local. Record image digest, release, gateway hash and DNS change evidence privately.

Rollback

Retain the previous release and its manifests. Compare current project configuration first to avoid overwriting newer work. Start the previous project release with its own verified Compose file, then verify health, public content and hashes. Restore only this project's gateway/DNS values when they changed. Never remove unrelated containers or volumes, run a global prune or use down -v as a routine rollback.

This static release has no patient-generated persistent state. A later clinical release requires an application-consistent database and upload backup, keys, restored permissions/holds and compatible migrations; restoring static files does not recover patient data.

Monitoring

Run node scripts/probe.mjs https://your-host/healthz from an independently hosted scheduler and retain its JSON output. Configure a named responder and alert policy before making an operating SLA. The probe requires valid TLS and the expected body, returns nonzero on failure and does not print sensitive URLs. Do not count a one-time probe or server-local healthcheck as a month of availability.

The loopback load tool refuses remote targets. Increase load only in an explicitly isolated test environment after agreeing on resource limits. Never run a million-user test against this shared VPS or a free third-party project.

Recovery records

The private checkpoint stores deployment state before/after each significant action. The dossier stores durable incident lessons. CREDENTIALS.md is gitignored and must be reconciled privately when credentials change. Public verification contains counts and limits, not passwords, account IDs or server access details. The public documentation has passed fresh-context review; clinical implementation approval remains outstanding.

Public launch and governance checklist

This document defines proposed operating responsibilities for a future real clinic. The site's sample content and workflows are not a deployed governance program. Assign named owners privately before launch.

Jurisdiction and accountability

- Area: US • Decision/evidence required: Covered entity/business-associate analysis; applicable state health/privacy, dental licensing and advertising rules • Owner role: Legal/privacy lead
- Area: EU • Decision/evidence required: Article 3 scope, controller/processor roles, Articles 6/9 basis, country health/confidentiality rules, transfers and supervisory authority • Owner role: Legal/privacy lead
- Area: Bangladesh • Decision/evidence required: Current data/cyber laws, BMDC professional duties, facility category, permits and local operating obligations • Owner role: Local legal/clinical lead
- Area: Multi-region • Decision/evidence required: Actual patients, operator locations, data regions, processors and transfer routes • Owner role: Product/privacy/platform leads
- Area: Clinical safety • Decision/evidence required: Intended purpose, care responsibility, escalation and downtime procedures • Owner role: Clinical lead

Read US/EU research and Bangladesh research. Launch review must cover the actual US state and EU member state; a regional overview cannot settle every local rule. Keep signed agreements and legal opinions outside public Git.

Required public surfaces

Publish the actual operator identity, verified contact channel, service scope, accurate clinician qualifications, current pricing/cancellation/refund terms, privacy notice, terms and accessibility/help contact. Explain data categories, purposes, applicable legal bases, vendors/transfers, retention, rights and complaints. Use reviewed translations; do not reuse sample identities or assume a translated privacy notice resolves legal differences.

Keep optional marketing consent separate from treatment/administration and terms acceptance. Do not embed advertising pixels or session replay on clinical pages. Explain cookies/storage according to actual runtime behavior. An emergency disclaimer must point users toward appropriate local care and must not replace a clinical escalation process.

Data and patient rights

Maintain a record-by-record retention schedule with jurisdiction, legal basis, period, trigger, hold exception and deletion owner. Verify the requester proportionately, scope exports, support corrections and complaints, document refusals/exceptions, propagate consent withdrawal and review backup restoration. Avoid blanket immediate deletion of medical records that must be retained. Keep staff access and purpose histories reviewable.

Children, guardians, deceased patients, emergency access and disputed authority require explicit product/legal workflows. Do not infer guardian rights from a shared email address or a browser checkbox. Define care assignment and tenant boundaries before clinical API implementation.

Accessibility and language

Proposed target: WCAG 2.2 AA, using W3C criteria. Test keyboard-only operation, focus, screen-reader names/status, error recovery, contrast, zoom/reflow, motion preferences and touch targets across public and clinical journeys. Check English/Bangla language tags, date/number meaning and translations. Automated axe results are one part of evidence, not a conformance certificate. Publish known barriers and a real assistance channel when operated.

Vendor and operational readiness

Approve data flow, processor terms/BAA where applicable, security evidence, permitted regions, sub-processors, support, retention, deletion/export and exit strategy for every vendor. Verify actual purchased features instead of assuming a plan includes backups or health-data processing eligibility. Review payment-provider availability for the operator's country before committing to a vendor.

Before launch, rehearse backup restoration, incident notification, offboarding, role revocation, downtime/manual scheduling and recovery reconciliation. Define support hours, on-call coverage, clinical escalation and contractual commitments. The policy catalog, reliability plan and roadmap provide the proposed structure.

Sign-off package

Require product acceptance, completed clinical blockers, security and accessibility findings disposition, applicable legal/contract approval, cost authorization, measured capacity and recovery, named responders and fresh-context review. Preserve dated evidence and exceptions privately; publish only scrubbed status. Reassess after jurisdiction, vendor, clinical purpose or data-flow changes.

Clinical backend release gate

The supabase/ directory preserves historical clinical backends. Every Edge Function entrypoint now returns a shared 503 boundary before processing a request. No environment variable can bypass it. The original function source was downloaded and all 15 recovered source files matched the Git baseline. The owner restored the historical project. After explicit owner approval, all 11 guards were deployed and each returned the expected HTTP 503 response. All 16 downloaded deployed source files matched local SHA256. No database records or migrations have been changed.

The following defects were recorded in the previous dossier and reconfirmed in inspected source. They are not falsely relabeled as repaired by disabling public access. Removing the release guard requires a dedicated clinical implementation, isolated test database and independent review.

- Blocker: Plaintext writers versus migration 010 • Required repair and verification: Reconcile every encrypted write/read with the final schema; empty and upgrade migration tests
- Blocker: Patient-selected query tables • Required repair and verification: Fixed operation/column allowlists, tenant/subject scope, row limits and negative authorization matrix
- Blocker: FHIR subject ownership • Required repair and verification: Enforce ownership for every patient-linked resource and search/export; test IDOR and care-assignment boundaries
- Blocker: Notification and breach authority • Required repair and verification: Verified caller/scheduler identity, fixed recipient/template policy, audited admin authority, no public service-role utility
- Blocker: Rate limiter • Required repair and verification: Atomic shared limiter, trusted proxy policy, fail closed on dependency errors and test concurrent requests
- Blocker: Consent • Required repair and verification: Purpose-specific grant/withdraw journey with legal-basis distinction, RLS and downstream enforcement
- Blocker: Payments • Required repair and verification: Server-owned catalog/amount, verified provider signatures, replay/idempotency handling; sandbox-only before separate approval
- Blocker: Retention • Required repair and verification: Jurisdiction-specific schedule, legal holds, dry run, approval, scheduler and restoration behavior
- Blocker: MFA/sessions • Required repair and verification: Actual staff MFA enforcement, revoked-role/session tests, token expiry and inactivity policy
- Blocker: Audit • Required repair and verification: Durable authorized server events, verified failure behavior, redaction, access review and tamper evidence
- Blocker: Configuration • Required repair and verification: Central typed server settings; inventory all Deno variables, limits, provider endpoints and templates
- Blocker: Operations • Required repair and verification: Vendor agreements, keys, encrypted off-server backups, restore rehearsal, monitoring and incident ownership

Do not apply historical migrations to an unknown live database. Inventory and export the schema/functions first, compare them with local source, preserve existing records, and build a tested upgrade/rollback plan. Never place service-role credentials in frontend configuration. Never send real patient data to a free-tier provider without verified legal and contractual suitability.

The frontend sample is deliberately not a substitute for this gate. Browser-selected roles only illustrate policy decisions. The static site proves neither database RLS nor backend encryption nor clinical FHIR interoperability. These limits should remain visible to clients and developers.

Release verification and limitations

October 2026 static release verification

On 5 October, the latest static release passed the recorded pre-deployment drift check and post-deployment file parity check (138/138 files), 21/21 public HTTP conditions and 24/24 public browser checks. Four live routes each had one canonical URL from the configured site origin. The earlier LinkedIn preview release preserved the page’s noindex metadata and header; LinkedIn Post Inspector read its title and description but reported that no preview image was available. These are release-specific checks, not clinical approval or measured uptime.

Latest publication verification

Checks completed on Sep 24, 2026 for the documentation and GitHub publication scope. All eight publication acceptance criteria were met. This result is separate from the earlier clinical implementation criteria retained below.

- Gate: Automated tests • Latest publication result: 15/15 passed
- Gate: Scoped type check • Latest publication result: Passed for the configured runtime, policy and shared guard modules
- Gate: Lint and build • Latest publication result: Passed; Node 24.20.0 build reported a plugin-timing warning only
- Gate: Real browser flow • Latest publication result: 24/24 passed on the local production preview: routes, disabled intake, sample policy/workflow, protected routes and limited accessibility
- Gate: Dependency audit • Latest publication result: Zero reported vulnerabilities at the point-in-time check; no new dependencies
- Gate: Secret scanning • Latest publication result: Nine baseline commits, the outgoing 255-file snapshot and staged diff scanned with zero detected leaks
- Gate: Security tools • Latest publication result: Edit hook exit 0; Semgrep 23 rules across 107 application/tooling targets, zero findings or parser errors; Bandit not applicable to shipped JS/TS
- Gate: Documentation checks • Latest publication result: Nine READMEs, 45 public Markdown files and 173 local links verified; 11 guards and 11 migrations confirmed
- Gate: Fresh-context review • Latest publication result: Approved for public documentation publication with no blocking defects; not clinical or legal certification
- Gate: GitHub parity • Latest publication result: Both branches at f758b68; all 255 file blob hashes match the local commit; default README and repository metadata read back

No clinical database integration, full Deno typing, million-user load, monthly uptime observation or new live deployment was tested during this publication. The earlier public-URL browser and deployment counts below belong to the previous release. No real secrets were detected in the outgoing publication; private operational material remains excluded. Dormant configuration debt remains a clinical release gate.

Evidence scope

This section preserves the earlier September release evidence for the portfolio build, not clinical production approval. Its independent clinical review was deferred. The latest documentation publication received a separate fresh-context approval, as recorded above.

- Gate: Automated regression tests • Result: 15/15 pass: access decisions, consent, hold/erasure, sample export, runtime boundary and source regressions
- Gate: Type check • Result: Pass; strict scope covers new runtime/policy/guard modules, not all historical JSX or Deno functions
- Gate: Lint • Result: Pass, zero errors/warnings in configured application/tooling scope
- Gate: Build • Result: Pass with locked Node/Vite toolchain; removed obsolete eval-producing animation dependency
- Gate: Dependency audit • Result: Zero reported npm vulnerabilities in the final lockfile audit; point-in-time evidence
- Gate: Gitleaks • Result: Zero findings in baseline history and scanned working source; final staged scan also passed with zero findings
- Gate: Security edit hook • Result: Exit 0 for application, tooling, automated tests and function sources
- Gate: Semgrep • Result: 23 security-audit rules, 106 scanned targets, zero findings; not a penetration test
- Gate: Bandit • Result: Not applicable to shipped code: no Python application source; private operator tooling is outside the shipped application
- Gate: Browser • Result: 24/24 local checks and 24/24 explicit-public-URL checks pass
- Gate: Accessibility • Result: Two axe checks on new main-content surfaces pass; mobile width/keyboard behavior exercised; full site conformance unverified
- Gate: Backend containment • Result: 11/11 deployed endpoints return HTTP 503 with expected guard response; all 16 downloaded source files match local SHA256
- Gate: Database preservation • Result: Only Edge Function code was deployed; no SQL, migration, record deletion or database mutation command was executed
- Gate: Independent review • Result: Earlier clinical review was deferred; the later documentation publication review approved its own scope. No clinical approval verdict exists

The public browser suite checks 20 routes, no automatic third-party requests from application pages, disabled intake inputs, the sample patient workflow, sample export, denied operations, held erasure, reset and protected-route redirection. Cloudflare/network operational metadata remains outside the application's no-tracking claim.

Deployment and recovery

The origin runs a pinned nginx image as a non-root user with a read-only filesystem, dropped capabilities, no-new-privileges, loopback binding and explicit CPU/memory/process limits. DNS and gateway publication reused existing resources. No new paid plan, server or domain was purchased.

Before changes, legacy HTML and six entry assets matched the local baseline, and 15 recovered backend files matched Git. Versioned static archives are kept off-server and extraction is checked against SHA256 manifests. Each staged release has 138 allowlisted files, with local/remote hashes checked. Five unrelated gateway site files were preserved.

Public TLS initially failed while the new DNS record propagated; certificate issuance retried and public HTTPS recovered. HTTP checks found an nginx location-precedence error that prevented immutable caching on three entry assets. The redundant extension regex was removed locally and a new versioned release prepared. The corrected release passed all 138 file hashes. Direct-origin and public TLS were verified. Container recreation reached healthy status and the expected response in 37.15 seconds; this measures that command sequence, not a disaster-recovery guarantee. Actual runtime inspection verified user 101:101, read-only root, all capabilities dropped, no-new-privileges, 128 MiB RAM, 0.5 CPU and 64 processes. The container port was unreachable externally and five unrelated gateway files retained their hashes. Final public HTTP checks passed 21/21 conditions; see the retained release evidence.

Capacity evidence

A bounded localhost Vite HTTP microbenchmark used 20 clients and 400 requests: zero failures, about 1,735 requests/second, p95 21.51 ms over about 0.231 seconds. This tiny warm local sample does not establish production capacity, endurance, CDN behavior or clinical transaction throughput. It must not be extrapolated to 10,000 or one million simultaneous users.

The workload calculator models 10,000 / 100,000 / 1,000,000 active readers, giving 1,000 / 10,000 / 100,000 edge requests per second under its stated assumptions. These are calculations only. No large-scale production load test or 30-day availability measurement was performed. The 99% target permits 432 unavailable minutes per 30-day window; there is no contractual SLA.

Configuration and secret audit

Real secrets were present in private local configuration and recovery sources; they were kept outside the public artifact. No real secret was detected in scanned shipped source or Git history. The private credentials, dossier, environment files and memory directory are ignored. Credential recovery and any required rotations remain private operator responsibilities.

New runtime origin/mode and inactivity settings, sample workflow data, privacy disclosures, image mapping, deployment image/ports/limits and test/load assumptions have central owners. Provider clients cannot be enabled through environment flags in the public release. Protocol syntax, HTTP status codes and static routing syntax remain fixed intentionally.

This is not a complete refactor of every historical module: dormant clinical Deno environment reads, provider URLs, timeouts, encryption/role contracts and older presentation copy still require consolidation before clinical activation. The scoped type check does not prove these modules correct. See CLINICAL-RELEASE.md for the mandatory repair gates.

Acceptance accounting

Of the original 14 draft criteria, 10 are addressed by this release (1, 2, 6–11, 13, 14); document synchronization is recorded in the final handoff. Criteria 3–5 are only partially satisfied: the site is isolated, but full clinical authorization/schema/configuration remediation remains blocked from production use. Criterion 12 remains open for clinical implementation review. The later documentation publication met its own eight acceptance criteria and received fresh-context approval; it does not close the clinical gates or establish 14/14 clinical readiness.

Media licensing and provenance

The application serves 47 local stock-image variants. The repository media manifest preserves the exact source-to-file mapping. People shown are stock subjects, not verified practitioners or patients of the sample clinic. See the Unsplash license.

- Stock photograph 1
- Stock photograph 2
- Stock photograph 3
- Stock photograph 4
- Stock photograph 5
- Stock photograph 6
- Stock photograph 7
- Stock photograph 8
- Stock photograph 9
- Stock photograph 10
- Stock photograph 11
- Stock photograph 12
- Stock photograph 13
- Stock photograph 14
- Stock photograph 15
- Stock photograph 16