

Assessing the PIRE Cloud Platform Against Interoperable Digital Hospital Requirements

Leonardo Juan Ramirez-Lopez*

Universidad Militar Nueva Granada, Bogota, Colombia

Abstract

Hospital digitization is often described through requirements catalogues without systematic verification against operational platforms, leaving implementation feasibility uncertain. This study identifies requirements for an interoperable digital hospital and evaluates their implementation in PIRE (Interoperable Electronic Records Platform), an HL7 FHIR- and AWS-based platform. A mixed-methods approach combined a PRISMA-guided systematic review, analysis of Colombian regulations, and semi-structured interviews with three stakeholder groups, yielding 65 functional requirements across 21 modules and 8 technology categories. These requirements were compared with PIRE through read-only inspection of its running pilot instance. The results showed that 12% of the requirements were confirmed, 40% were partially covered, 43% had no coverage, and 5% could not be verified without load testing. The study demonstrates a traceability approach that connects documented requirements with empirically verified implementation, providing evidence to guide the technical evolution of interoperable digital health platforms.

Keywords

Digital hospital, HL7 FHIR, Requirements engineering, Health interoperability, Production verification, Cloud health platforms

1. Introduction

The digital transformation of hospital systems has stopped being a technological aspiration and has become a regulatory and operational requirement. In Colombia, Law 2015 of 2020 establishes the interoperable electronic health record as state policy, and Resolution 1888 of 2025 mandates the adoption of the Digital Summary of Care -RDA within the electronic health record interoperability framework. However, the gap between what regulation requires, what health professionals and patients clinically need, and what a concrete technology platform can actually sustain in production is rarely documented explicitly and verifiably.

Most requirements engineering (RE) exercises for digital hospitals end up as catalogues of functional and non-functional requirements that remain reference documents, with no later contrast against a real implementation; and when that contrast does exist, it tends to rely on architecture documentation -diagrams, technical memos- rather than on the system as actually deployed, which can produce false positives of compliance when the implementation evolves without the documentation being updated. This leaves an open question of interest to both researchers and development teams: of a set of requirements obtained with methodological rigor, which are already resolved, which are partially resolved, and which represent a real engineering gap?

This paper answers that question in two parts. The first reports the elicitation of requirements for an interoperable digital hospital, through a mixed method that combined a systematic literature review under PRISMA methodology, a review of current Colombian regulation, and semi-structured interviews with three stakeholder groups (telecommunications engineers, physicians, and patients), from which 65 functional requirements (RF01-RF65) were derived, organized into 21 functional modules and regrouped into 8 technology categories.

ICAIW 2026: Workshops at the 9th International Conference on Applied Informatics 2026, October 13–17, 2026, Orlando, United States

*Corresponding author.

✉ leonardo.ramirez@unimilitar.edu.co (L. J. Ramirez-Lopez)

ORCID 0000-0002-6473-5685 (L. J. Ramirez-Lopez)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

The second part, which constitutes the main contribution of this work, contrasts those 65 requirements against PIRE (Plataforma Interoperable de Registros Electrónicos, Interoperable Electronic Records Platform), a hospital platform natively built on HL7 FHIR, deployed on AWS and in active pilot operation. PIRE operates as a pilot in a controlled environment rather than in a real clinical setting; in this paper, "production instance" refers to the live server on which this pilot runs. Verification did not rely on architecture documentation, but on direct, read-only inspection of the production instance: running containers, active processes, the FHIR server's own metadata endpoint, real counts of clinical resources, read-only queries against the operational database schema, and static searches over the backend source code. This approach, whose procedure is described in Section 3, made it possible not only to confirm or refute each requirement, but to correct two classifications that a prior, documentation-based assessment had established incorrectly or incompletely (Section 6).

The rest of the paper is organized as follows: Section 2 reviews the regulatory framework and relevant standards; Section 3 describes the requirements elicitation methodology and the technical verification procedure; Section 4 summarizes the 65 functional requirements obtained; Section 5 describes PIRE's verified architecture with quantitative evidence; Section 6 presents the comparative gap analysis; Section 7 discusses the findings; and Section 8 concludes with future work.

2. Related Work and Regulatory Framework

2.1. State of the art in FHIR interoperability

HL7 Fast Healthcare Interoperability Resources (FHIR) has become the de facto standard for exchanging structured clinical information, through resources such as Patient, Observation, Condition, or MedicationStatement exposed via a REST API HL7 International [1]. Bender and Sartipi [2] presented one of the earliest evaluations of FHIR's RESTful, agile approach against previous standards (HL7 v2/v3, CDA), pointing to its shallower adoption curve as a key advantage for resource-constrained development teams. Lehne et al. [3] argued that semantic -not merely syntactic- interoperability is the enabling condition for digital medicine, and that its absence is the most frequent cause of failure in electronic health record projects. FHIR is further supported by complementary specifications such as SMART on FHIR for OAuth 2.0-based authorization [4], and clinical terminologies such as SNOMED CT [5] and LOINC [6] for the semantic coding of diagnoses and observations.

More recent systematic reviews temper the initial optimism about FHIR adoption. Ayaz et al. [7] reported that most documented FHIR implementations correspond to proofs of concept or academic pilots, with scarce evidence of real production deployments and even less independent verification of the underlying code or infrastructure. Vorisek et al. [8] confirmed the same pattern specifically in health research, noting that most studies report a FHIR system's declared architecture without contrasting it against its observed behaviour. This line of findings directly motivates the central methodological contribution of this paper: extending gap analysis beyond declared architecture, toward direct verification of the running system.

2.2. Requirements engineering in digital health

Requirements engineering is framed by ISO/IEC/IEEE 29148:2018 [9], which requires well-formed requirements to be singular, traceable, and verifiable, and distinguishes requirements verification, which confirms that the statements are well-formed, from requirements validation, which confirms that they define the system the stakeholders intend. Both activities operate on the requirements artefact; establishing whether a deployed system satisfies those requirements is a different task, which requires evaluating the implemented system in its operational environment. In the health domain, ISO 10781:2023 [10], which adopts the HL7 Electronic Health Record-System Functional Model Release 2.1, provides a reference list of the functions of an electronic health record system with conformance criteria, and supports the derivation of functional profiles for specific care settings and national realms. The catalogue presented in Section 4 is operationalized as such a profile for the Colombian realm:

its functions come from the elicitation described in Section 3, while the obligations that make them mandatory come from Colombian regulation.

Kinast, Ulrich, Bergh and Schreiweis [11] showed that document-based elicitation complements stakeholder-based methods and broadens the scope of the resulting requirements; the mixed design adopted here follows that logic, with interviews capturing situated needs and the literature and Colombian regulation acting as documentary sources. A catalogue, however, specifies the functions sought and provides no evidence about the behaviour of a specific deployment. Shivers, Amlung, Ratanaprayul, Rhodes and Biondich [12], who translated World Health Organization recommendations into FHIR implementation guides, likewise identified validation and real-world application as pending steps. This paper therefore addresses two related gaps, between elicitation and evaluation and between documented requirements and the running system, by treating the requirements catalogue as a measurement instrument applied to a deployed platform, with PIRE as the case study and a systematic protocol based on direct inspection of its production instance.

2.3. Colombian regulatory framework

In Colombia, the regulatory framework for the interoperable electronic health record is composed of Law 2151 of 2020; Law 1581 of 2012, on personal data protection; Resolution 866 of 2021, which defines the clinical data relevant to interoperability; and Resolution 1888 of 2025, which adopts the Digital Summary of Care (RDA). This framework is complemented by international reference standards for specific subdomains: DICOM for diagnostic imaging [13], IHE XDS.b for the exchange of clinical documents between institutions IHE International [14], ISO/IEEE 11073 [15] for personal health devices IEEE [15], and WCAG 2.2 for interface accessibility W3C [16].

3. Requirements Elicitation and Verification Methodology

3.1. Requirements elicitation

Requirements elicitation combined three complementary sources. First, a systematic literature review under PRISMA methodology (identification, screening, eligibility, and inclusion), searching Scopus, IEEE Xplore, PubMed, and EBSCO. Second, a review of current Colombian regulation, described in Section 2.3. Third, semi-structured interviews with three stakeholder groups: three telecommunications engineers (ENG1-ENG3), two physicians (PHY1-PHY2), and five patients with heterogeneous digital access profiles (PAT1-PAT5). Participants were selected by purposive sampling, according to the profile defined for each group: engineers with experience in the field and knowledge of HL7 FHIR and digital hospitals, physicians with clinical practice experience, and patients with different levels of access to digital channels.

The systematic review followed the four PRISMA stages. Five research questions guided the search: which requirements engineering methodologies have been applied to digital hospital systems; what the main interoperability challenges among Hospital Information System, Laboratory Information System, and Radiology Information System in developing countries are; how HL7 FHIR has been implemented in Latin American hospitals; which performance metrics are used to evaluate interoperable clinical information systems; and which data quality audit models exist for FHIR systems. Each question was translated into its own Boolean equation, combining terms such as requirements engineering, hospital information system, interoperability, FHIR, and data quality audit. The retrieved records were centralized in Mendeley, which supported duplicate removal and source traceability, and a separate flow diagram was produced for each question. The inclusion criteria were: peer-reviewed publications from the last five years; studies reporting requirements, architectures, or implementations of interoperable clinical information systems; and documents available in full text in English or Spanish. The exclusion criteria were: works without an explicit technical or methodological component, publications limited to the commercial description of a product, and records duplicated across databases.

The searches were conducted in March 2026. Across the five research questions, 210 records were identified; after removing 25 duplicates, 185 records were screened by title and abstract, of which 95 were excluded. The remaining 90 were assessed in full text, 24 were rejected, and 66 were included in the synthesis: 13 for the first question, 11 for the second, 13 for the third, 15 for the fourth, and 14 for the fifth. Figure 1 aggregates the flow diagrams of the five questions.

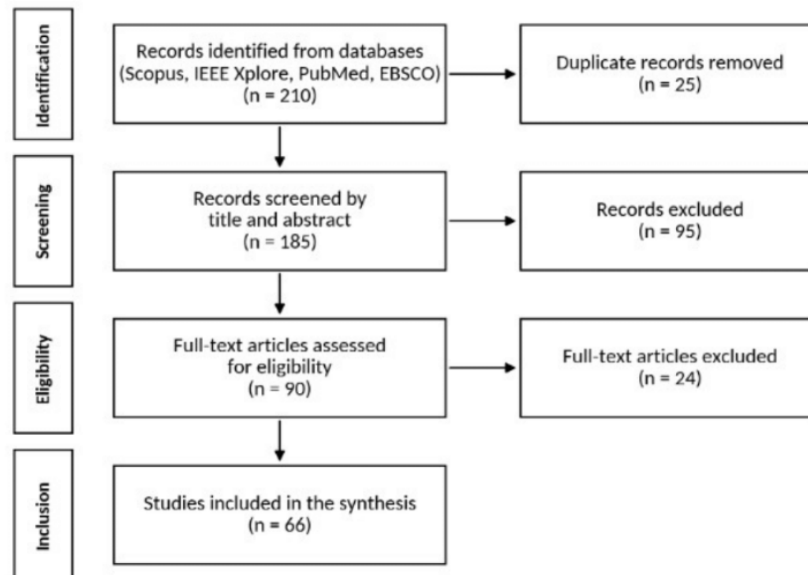


Figure 1: PRISMA flow diagram of the systematic review, aggregated across the five research questions.

Interviews with the engineers surfaced technical consensus (interoperability under HL7 FHIR as a non-negotiable requirement) and relevant disagreements (the choice between a pure-cloud or a hybrid architecture with local priority under connectivity failures), translated into infrastructure and availability requirements. Interviews with physicians contributed the clinical perspective: organizing information according to the care context, the need for the interface to follow the logical flow of the consultation, and a reliability indicator for lab results after reports of values described as “falsely normal”. Interviews with patients revealed the heterogeneity of available digital access channels, giving rise to the multichannel accessibility requirements.

The interviews followed a semi-structured guide with a standardized opening protocol in which the researcher introduced herself, explained the purpose of the study, and requested consent to record the session, in compliance with Law 1581 of 2012 on personal data protection. After two characterization questions on age and role or specialty, each guide was organized into thematic blocks specific to the group: system usability, patient monitoring, consultation channels, and data quality and integrity for the physicians; system usability, integration and interoperability, access media and infrastructure, and metrics and auditing for the engineers; and system usability, monitoring and follow-up, access media, and privacy and trust for the patients. Every guide closed with an open block of recommendations.

The qualitative information was coded into 65 functional requirements (RF01-RF65), organized into 21 modules (Table 1) and regrouped into 8 technology categories. The coding proceeded in three passes over the transcripts and the documentary corpus. The coding was performed by one researcher and reviewed by the project director, who validated the resulting requirements and their assignment to modules. The first pass extracted units of meaning that expressed a need, a constraint, or an expectation. The second pass grouped equivalent units, removed duplicates, and reformulated each surviving unit as a verifiable requirement statement, assigning it a code and a functional module. The third pass regrouped the modules into the technology categories used in Section 6, so that every requirement traces to a concrete technology or standard and, therefore, to something that can be searched for in a running system. Requirements that could not be traced to a verifiable technical element were reformulated or

discarded.

3.2. Technical verification of PIRE

Unlike a comparison based on architecture documentation, PIRE's verification was carried out through a systematic protocol of direct, read-only inspection of its production instance on AWS, performed via administrative SSH access: (i) querying the FHIR server's own CapabilityStatement endpoint (/fhir/metadata) to establish the engine and version actually running; (ii) counting active resources by type through the FHIR Search API (_summary=count parameter); (iii) inventorying Docker containers (docker ps) and systemd services, including the Flask process's unit file, to determine the real application server; (iv) read-only SQL queries against the full schema of the operational database on AWS RDS, including counts of audit events, users, and remaining tables from discarded FHIR engines; and (v) static pattern search over the backend source code to confirm or refute the presence of specific technologies (OAuth2/SMART on FHIR, DICOM/PACS, MQTT/Twilio/WebRTC, production WSGI servers). This protocol was applied against the current production instance, and its quantitative results are reported in Section 5.

Each functional module was assigned one of four statuses, which applies to all the requirements it contains. A module was classified as confirmed when the principal function described by its requirements was observed operating in the production instance through at least one of the inspection steps (i) to (v); partial, when that function was observed only in part, or when it was present in the source code or database schema without being exercised in practice (as with the audit log, whose columns for clinical resources exist but contain no clinical events); not implemented, when none of the inspection steps produced evidence of the function; and not verifiable, when confirming or refuting the requirements depends on a test that non-invasive inspection cannot provide, such as a load test. Each technological category in Table 2 may group requirements from different modules among the 21; consequently, the status of a category is not derived from the status of any individual module.

4. Results: Identified Functional Requirements

Several qualitative findings illustrate the nature of the 65 requirements obtained and their traceability to concrete technical decisions. ENG2 summarized the centrality of interoperability with the analogy “interoperability is like knowing English in the job market”, and recounted a real case of system downtime during a surgery to justify a maximum tolerable failure time of 15 to 30 minutes (RF33-RF34). ENG3 distinguished confidentiality from data integrity using the clinical risk of deleting an allergy record, and pointed to the lack of data homogenization across systems -“cédula” (national ID) versus “ID”- as a limitation rarely discussed in the literature (RF61-RF63).

PHY2 was the only interviewee to explicitly point to the research value of clinical data, requesting that information be retrievable by diagnosis, age, gender, ethnicity, and family history simultaneously, and proposed a reliability indicator per lab result (RF09). Among patients, PAT3 formulated the most complete results-visualization proposal, combining color alerts, trend charts, and alphanumeric records, and considered a wait of up to 15 days to obtain a clinical history unacceptable; PAT4 was the only interviewee who confirmed having repeated a test because the physician lacked access to previous results; and PAT5, whose only digital channel is WhatsApp, named appointment availability -not information handling- as her main complaint, directly reinforcing the real-time scheduling requirement (RF10-RF11).

Two cross-cutting themes emerged that do not belong to a single module. The first is the asymmetry between what the institution needs and what the patient perceives: the engineers and the physicians formulated their requirements around data integrity, availability, and traceability, whereas the patients formulated theirs around access, waiting time, and comprehension, which is the origin of the multi-channel accessibility module (RF50-RF51) and of the results presentation requirements (RF06-RF09). The second is the recurring distinction between having a function and having it usable under real conditions: ENG1 described the audit trail as a requirement that is usually declared and rarely exercised,

and PHY1 insisted that information organized by care context is not the same as information merely present in the record. Both themes anticipated the partial coverage findings reported in Section 6.

Table 1

Distribution of the 65 functional requirements by module.

Module	No. of RF	Codes
EHR management	7	RF01-RF05, RF55-RF56
Test results lookup	4	RF06-RF09
Appointment scheduling	2	RF10-RF11
Remote monitoring	4	RF12-RF15
Alerts and notifications	6	RF16-RF20, RF60
Teleconsultation	2	RF21-RF22
Imaging (RIS/PACS)	5	RF23-RF24, RF52-RF54
Audit and traceability	4	RF25-RF27, RF59
Role management	2	RF28-RF29
Security and privacy	3	RF30-RF32
Availability	2	RF33-RF34
Performance and latency	3	RF35-RF37
Usability	3	RF38-RF40
Interoperability	3	RF41-RF43
Scalability	3	RF44-RF45, RF65
Regulatory compliance	2	RF46-RF47
Maintainability	2	RF48-RF49
Multichannel accessibility	2	RF50-RF51
Infrastructure	2	RF57-RF58
Data quality	3	RF61-RF63
Monitoring and metrics	1	RF64

These 65 requirements, with their traceability to the qualitative source that originated them, constitute the comparison input for Section 6. The full catalogue is provided in Appendix A.

The 21 modules were regrouped into 8 technology categories, each one associated with the technical element that would have to exist in a platform for the corresponding requirements to be satisfied: HL7 FHIR core (resources, REST API, search and subscription mechanisms); cloud infrastructure (compute, network isolation, load distribution and failover); databases (clinical persistence, operational persistence and caching); medical terminologies (SNOMED CT, LOINC, ICD-10 and ICD-11 coding) [5]; communications and IoT (biomedical device channels, telemedicine transports and notification transports); security and privacy (authentication, authorization, auditing, encryption and personal data protection); development and DevOps (version control, containerization, production servers and client applications); and imaging management (DICOM, PACS and visualization). This regrouping is what makes the comparison of Section 6 possible, because it converts each requirement into an observable question about the deployed system.

5. The PIRE Platform: Verified Architecture

PIRE (Spanish: Plataforma Interoperable de Registros Electrónicos, English: Interoperable Electronic Records Platform) is a hospital platform natively built on HL7 FHIR, deployed on AWS infrastructure and in active pilot operation. This section reports findings obtained through the verification protocol

described in Section 3.2, executed directly against the current production instance.

The FHIR server’s own `/fhir/metadata` endpoint confirmed that the platform ran on Blaze (Samplify), version 1.9.0, FHIR R4 (4.0.1), containerized with Docker (docker ps confirmed the active blaze container on the `samplify/blaze: latest` image). The HAPI FHIR instance, described in earlier documentation as the persistence engine, existed only as a discarded prototype: the `hfj_resource` table retained 18 historical resources, and the terminology tables (`trm_concept`) remained at zero rows, evidence of a real FHIR engine migration during development that was not reflected in earlier documents. The development team reported that this migration was driven by two reasons: Blaze’s greater operating capacity compared to HAPI FHIR under the compute resources available for the pilot instance, and an interoperability requirement with an external entity still in the process of being formally onboarded to the project, whose technical connection specifications conditioned the choice of engine. The full PostgreSQL schema on AWS RDS retained 62 tables, split between remnants of the HAPI schema (`hfj_*`, `trm_*`) and the application’s own schema (`audit_log`, `users`, `pacientes_datos_locales` (local patient data), `solicitudes_firma_consentimiento` (consent signature requests)).

The backend actively built and sent eight FHIR resource types, with counts verified via the FHIR Search API: Patient (13), Practitioner (3), Observation (215), Condition (11), FamilyMemberHistory (6), MedicationStatement (5), RelatedPerson (8) and RiskAssessment (2); 263 active clinical resources in total. Physiological observations were coded directly in LOINC (heart rate 8867-4, respiratory rate 9279-1, physical activity 41950-7, posture 8361-8), confirmed by inspecting the source code of `observation_service.py`. The clinical history module integrated complete catalogues, verified by direct count: 12,552 ICD-10 codes and 35,340 ICD-11 codes loaded as local catalogues, more complete than earlier documentation described. SNOMED CT remained only referenced, without the catalog loaded.

A finding not reported in previous evaluations of the platform was the real integration of a portable biomedical device: the source code (`static/js/t800_capture.js`) invoked the browser’s native Web Bluetooth API (`navigator.bluetooth.requestDevice`) to connect to a “T800 Ultra 2” device, recording heart rate, oxygen saturation, blood pressure, and electrocardiogram sessions directly from the clinical browser. The backend (`t800_observation_service.py`, `app.py`) validated that the captured sample was no more than 30 seconds old before accepting it and converted it into a FHIR Observation resource coded in LOINC with UCUM units. This evidence contradicted a previous classification that assumed the biomedical device channel had been retired due to lack of use. It reclassified the communications and IoT category from “not implemented” to “partial” (Section 6).

For security, the Flask backend (version 3.0.0) implemented role-based access control through route decorators (`login_required`, `require_permission`, confirmed in `auth/decorators.py`), active CSRF protection (Flask-WTF), login-attempt limiting (Flask-Limiter, with at least one real `login_rate_limited` event recorded), and salted password hashing. The database recorded 5 active roles: `super_admin`, `auditor`, `medico` (physician) (2), `enfermero` (nurse), and `psicologo` (psychologist) (2). The last two, together with the dedicated auditor role, were not contemplated in the admin, physician, nurse, and patient matrix that ENG1 described as a minimum requirement (RF28), which evidences a greater granularity in the separation of duties. HTTPS was verified at the web server level: nginx exposes the application only on port 443 with a Let’s Encrypt certificate, and plain HTTP requests are not served.

The audit log (`audit_log` table) contained 227 real events at the time of verification, with a schema prepared to record action, resource type and ID, IP address, and outcome (success/failure) for each event. However, the breakdown by action type showed that all recorded events corresponded to authentication and user administration (`login`: 165, `logout`: 37, `login_failed`: 14, `create_user`: 6, `reset_password`: 4, `login_rate_limited`: 1); no create, update, or delete event on clinical resources (Patient, Observation, Condition, etc.) was being audited, even though the `resource_type` and `resource_id` columns existed in the schema. This finding, obtained by direct inspection of the table’s actual content and not only its definition, refined the audit status from “implemented” to “partially implemented” (Section 6).

At the deployment level, the `pire.service` systemd unit ran `/opt/pire/flask/venv/bin/python app.py` directly, and the production virtual environment had no gunicorn installed, confirming that the application ran on Flask’s built-in development server (with debug explicitly disabled in `app.run`) rather than a production WSGI server. The active codebase comprised 67 Python files and 9,299 lines, alongside 50

timestamped manual backup directories (evidence of a workflow based on pre-change backup copies rather than commit-based version control), confirming the absence of a mature DevOps practice (Section 6).

6. Comparative Gap Analysis

Table 2 summarizes the status of the 8 technology categories based on the verified evidence. Each functional requirement was paired with a proposed technical solution, expressed as a non-functional requirement specifying the technology or standard through which it should be implemented, such as SMART on FHIR for access control or FHIR Subscriptions for real-time updates. The 8 technology categories group these technical solutions; therefore, Table 2 indicates whether PIRE uses the proposed technologies rather than whether it performs the corresponding functions. At the functional-module level (Table 1, 21 modules, 65 RF), the aggregate distribution, updated after the verification in Section 5, is: 8 RF (12%) with confirmed implementation -role management, security and privacy, and interoperability-; 26 RF (40%) partial -EHR management, test results lookup, remote monitoring, audit and traceability, regulatory compliance, maintainability, infrastructure, and monitoring and metrics-; 28 RF (43%) with no implementation identified -appointment scheduling, alerts and notifications, teleconsultation, RIS/PACS imaging, availability, usability, scalability, multichannel accessibility, and data quality-; and 3 RF (5%) -performance and latency- could not be verified without a real load test.

The audit and traceability module (RF25-RF27, RF59) is reclassified in this paper from “confirmed implementation” to “partial” relative to a preliminary assessment: the audit mechanism exists, is active, and is correctly indexed, but its real coverage -verified by direct event count- is limited to authentication and user management, without yet reaching the clinical CRUD auditing that ENG1 identified as an unresolved priority.

Two findings exceed what previous, documentation-based assessments had established. First, LOINC coding of vital signs, the ICD-10/ICD-11 catalogues with over 47,000 combined codes, and the dedicated Digital Summary of Care builder show that medical terminologies and regulatory compliance are more advanced than documented. Second, the Web Bluetooth integration with the T800 Ultra 2 confirms real biomedical monitoring from the browser, reclassifying “Communications and IoT” from “not implemented” to “partial”, although it still lacks MQTT, WebRTC, Twilio, or FCM for external telemetry, teleconsultation, or push/SMS notifications.

At the other extreme, the sharpest gaps remain on three fronts: the total absence of diagnostic imaging management (RIS/PACS, no trace of DICOM, PACS, or OHIF); patient multichannel accessibility (the interface remains exclusively a web panel for clinical and administrative staff, with no portal, mobile app, SMS, or WhatsApp for the patient); and infrastructure availability/scalability (a single compute and database instance, no gunicorn or load balancer, running Flask’s development server).

7. Discussion

The pattern that emerges from the comparative analysis is not random. The categories with the best coverage -role-based access control, CSRF, session attempt limiting, and the core FHIR interoperability layer- correspond to the layer a small research team can build first, and that is most critical to get right from a clinical safety standpoint. The categories with the largest gaps -diagnostic imaging, patient multichannel accessibility, and availability/scalability- require integration with regulated external vendors, a patient-facing product surface that goes beyond the clinical backend, or an infrastructure investment typical of a maturity stage beyond a research pilot.

Two methodological findings deserve particular discussion. First, the real scope of the audit log -active but limited to authentication, not clinical CRUD- could only be discovered through direct inspection of the audit_log table’s content, not merely its existence or schema; a verification limited to “does the audit table exist?” would have produced a false positive of compliance for RF25-RF27. Second, the Web Bluetooth integration with the T800 Ultra 2, not reported in previous evaluations, illustrates the inverse

Table 2

Status of PIRE by technology category, verified through direct inspection of the production instance.

Technology category	Status	Key evidence
HL7 FHIR core	Partial	Blaze FHIR engine 1.9.0 (via /fhir/metadata) in production; 263 active resources across 8 types; no confirmed use of Subscriptions or _include/_revinclude.
Cloud infrastructure (AWS)	Partial	VPC confirmed at the network level; compute and database services on a single instance; no load balancer or failover mechanism was identified during inspection; boto3 present only for email via AWS SES; no S3/KMS/HealthLake integration in the FHIR pipeline.
Databases	Partial	PostgreSQL on AWS RDS (62 tables); FHIR resources persisted in Blaze's own volume, not in PostgreSQL; no Redis.
Medical terminologies	Partial	LOINC in active use; ICD-10 (12,552) and ICD-11 (35,340) catalogues loaded and verified by count; SNOMED CT only referenced; no catalogue loaded.
Communications and IoT	Partial (reclassified)	Web Bluetooth integration confirmed (navigator.bluetooth.requestDevice) with the T800 Ultra 2 wearable for vitals and ECG, coded in LOINC/FHIR; no MQTT/WebRTC/Twilio/FCM for external telemetry or teleconsultation.
Security and privacy	Partial (reclassified)	RBAC, CSRF, login attempt limiting, salted passwords, forced HTTPS, 5 roles with separation of duties (incl. dedicated auditor role); audit active but limited to authentication (227 events, 0% clinical CRUD); no OAuth2/SMART on FHIR.
Development and DevOps	Not implemented	Systemd runs 'python app.py' directly (no gunicorn); only the FHIR engine is containerized; 50 manual backup directories were observed; no version-control repository or workflow was identified during inspection; no Kubernetes or mobile/PWA client.
Imaging management (RIS/PACS)	Not implemented	No evidence of DICOM, a PACS server, or an image viewer in the inspected code.

risk: a requirement marked as “not implemented” based on a code review that became outdated within a few weeks of active development can turn into a false negative. Both cases confirm that verification must reach the actual content and behaviour of the system, and must be repeated as development progresses.

This distinction has direct practical value for prioritizing PIRE's technical roadmap. Some gaps are of relatively low cost and high impact -replacing Flask's development server with a WSGI server is a configuration change, not an architectural one (RF44-RF45, RF65); extending the existing audit logging to clinical CRUD operations reuses an already-built mechanism (RF25-RF27)-, while others are structural and costly -integrating a complete RIS/PACS flow (RF23-RF24, RF52-RF54) requires adopting a full standard (DICOM) and typically a third-party component.

Ordered by cost and dependency, these gaps define a phased roadmap. A first phase groups low-cost changes that reuse mechanisms already in place, such as running the application on a production WSGI

server and extending audit logging to clinical CRUD operations. The three sharpest gaps identified in Section 6 belong to later phases because each requires more than extending the current code. Diagnostic imaging (RF23-RF24, RF52-RF54) requires adopting the full DICOM standard and, typically, a third-party PACS server and viewer. Patient multichannel access (RF50-RF51) requires building a new patient-facing channel, such as a portal, a progressive web app, or SMS, since the current interface serves only clinical and administrative staff. Availability and scalability (RF33-RF34, RF44-RF45, RF65) require an infrastructure investment, such as redundancy across availability zones, load balancing, and auto-scaling, which corresponds to a maturity stage beyond the pilot.

This work has limitations. Verification was limited to what could be directly and non-invasively inspected on a single production instance, at a specific point in active development (50 manual backup directories in the month prior, indicating frequent changes); the performance and latency requirements (RF35-RF37) could neither be confirmed nor ruled out without a controlled load test. Both the requirements and the platform correspond to a Colombian pilot context, so generalization should be made with caution. In addition, the interview sample was small (ten participants: three engineers, two physicians, and five patients), so the elicited needs may not capture the full diversity of stakeholders.

8. Conclusion

This paper presented two complementary contributions: a catalogue of 65 functional requirements for an interoperable digital hospital, obtained through a mixed method of literature review, regulatory review, and semi-structured interviews; and a comparative analysis of those requirements against PIRE, verified through a systematic protocol of direct inspection of its production instance -FHIR server metadata, resource counts, real content of audit tables, and static code search- rather than its architecture documentation.

Results show that PIRE solidly covers role-based access control and core FHIR interoperability, partially covers audit, medical terminologies, remote monitoring, and regulatory compliance, and presents significant gaps in diagnostic imaging management, patient multichannel accessibility, and infrastructure availability/scalability. Verification also corrected two prior classifications: audit, from “confirmed” to “partial”, as it does not yet reach clinical CRUD; and communications and IoT, from “not implemented” to “partial”, upon confirming a functional Web Bluetooth integration not previously documented.

Future work includes: (i) the detailed design of the missing modules, prioritized by cost/impact, starting with hardening the deployment layer (production WSGI server) and extending audit logging to clinical CRUD events; (ii) a controlled load test to conclusively verify the performance and latency requirements (RF35-RF37); and (iii) adopting this direct verification protocol -repeated periodically, not as a one-off event- as a standard practice in PIRE’s requirements engineering lifecycle, given that this same paper documented both a false positive and a false negative that a single, outdated verification would have missed.

Beyond the specific case, the exercise suggests a practice that may be useful in other settings: a requirements catalogue is more useful as an inspection instrument than as a reference document, provided that each requirement is formulated so that a concrete observation over the running system can confirm or refute it. The eight technology categories used here are the mechanism that makes that formulation possible, and other teams could adapt them to establish, at a given moment, how far their platform actually stands from the requirements it declares to pursue.

Acknowledgements

The author thanks the Nueva Granada Military University for funding project INV-ING-4355.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

- [1] HL7 International, HL7 fast healthcare interoperability resources (fhir), release 4 (v4.0.1), 2019. URL: <https://hl7.org/fhir/R4/>.
- [2] D. Bender, K. Sartipi, HL7 fhir: An agile and restful approach to healthcare information exchange, in: Proceedings of the 26th IEEE international symposium on computer-based medical systems, IEEE, 2013, pp. 326–331.
- [3] M. Lehne, J. Sass, A. Essenwanger, J. Schepers, S. Thun, Why digital medicine depends on interoperability, NPJ digital medicine 2 (2019) 79.
- [4] HL7 International, Smart application launch framework release 1.0.0 (smart on fhir), 2021. URL: <http://hl7.org/fhir/smart-app-launch/>.
- [5] SNOMED International, Snomed ct editorial guide, 2024. URL: <https://confluence.ihtsdotools.org/display/DOCEGU>.
- [6] Regenstrief Institute, Logical observation identifiers names and codes (loinc): User guide, 2024. URL: <https://loinc.org/downloads/loinc-table/>.
- [7] M. Ayaz, M. F. Pasha, M. Y. Alzahrani, R. Budiarto, D. Stiawan, The fast health interoperability resources (fhir) standard: systematic literature review of implementations, applications, challenges and opportunities, JMIR medical informatics 9 (2021) e21929.
- [8] C. N. Vorisek, M. Lehne, S. A. I. Klopfenstein, P. J. Mayer, A. Bartschke, T. Haese, S. Thun, Fast healthcare interoperability resources (fhir) for interoperability in health research: systematic review, JMIR medical informatics 10 (2022) e35724.
- [9] ISO/IEC/IEEE, Iso/iec/ieee 29148:2018 – systems and software engineering: Life cycle processes: Requirements engineering, 2018. URL: <https://www.iso.org/standard/72089.html>.
- [10] ISO, Iso 10781:2023 – health informatics: HL7 electronic health record-system functional model, release 2.1 (ehr fm), 2023. URL: <https://www.iso.org/standard/84722.html>.
- [11] B. Kinast, H. Ulrich, B. Bergh, B. Schreiweis, et al., Functional requirements for medical data integration into knowledge management environments: requirements elicitation approach based on systematic literature analysis, Journal of medical internet research: the leading peer-reviewed eHealth journal 25 (2023).
- [12] J. Shivers, J. Amlung, N. Ratanaprayul, B. Rhodes, P. Biondich, Enhancing narrative clinical guidance with computer-readable artifacts: authoring fhir implementation guides based on who recommendations, Journal of Biomedical Informatics 122 (2021) 103891.
- [13] D. Standard, Digital imaging and communication in medicine, Strateg. Doc. Available online: [\(https://www.dicomstandard.org/\(accessed on 25 March 2024\)\)](https://www.dicomstandard.org/(accessed on 25 March 2024)) (2018).
- [14] IHE International, It infrastructure technical framework: Cross-enterprise document sharing (xds.b), 2023. URL: <https://profiles.ihe.net/ITI/TF/Volume1/ch-10.html>.
- [15] IEEE, Iso/ieee 11073-20601:2019 – health informatics: Personal health device communication, 2019. URL: <https://standards.ieee.org/ieee/ISO-IEEE-11073-20601/7109/>.
- [16] W3C, Web content accessibility guidelines (wcag) 2.2, 2023. URL: <https://www.w3.org/TR/WCAG22/>.

Appendix A. Functional requirements catalogue

Table 3 lists the 65 functional requirements obtained through the elicitation process described in Section 3.1, together with the module to which each one belongs.

Table 3

Functional requirements by code and module.

Code	Module	Description
RF01	EHR management	The system must allow users to create, view, update, and manage each patient's electronic health record, progressively accumulating information at each consultation.
RF02	EHR management	The system must display a summary view of the patient at the start of a consultation, including sex, age, relevant history, active conditions, and current treatment phase.
RF03	EHR management	The system must preload previously recorded family and personal history, requesting only the missing information in subsequent consultations.
RF04	EHR management	The system must allow clinical information to be organized and filtered by multiple parameters: date, diagnosis, age, sex, ethnicity, history, and test type.
RF05	EHR management	The system must support reusable templates for recording consultations, allowing standard fields to be prefilled with minimal interaction.
RF06	Test results lookup	The system must allow patients to view all their clinical test results from the platform.
RF07	Test results lookup	The system must present results in adaptable language: clinical terminology for professionals and understandable language for patients.
RF08	Test results lookup	The system must record each result linked to its collection date, allowing the patient's clinical evolution to be analyzed.
RF09	Test results lookup	The system must include a reliability indicator for each laboratory result, allowing the physician to flag a value as suspicious without deleting it from the record.
RF10	Appointment scheduling	The system must allow patients to book medical appointments directly from the app or website, without travelling.
RF11	Appointment scheduling	The system must show appointment availability in real time and confirm bookings immediately.
RF12	Remote monitoring	The system must remotely record and display vital signs: heart rate, temperature, blood pressure, respiratory rate, and ECG.
RF13	Remote monitoring	The system must adapt the update frequency to the patient's condition: real-time, hourly, or every 6 hours.
RF14	Remote monitoring	The system must allow patients to actively report symptoms, including adverse effects or lack of medication effectiveness.
RF15	Remote monitoring	The system must display monitoring data in a unified dashboard with trend charts, color alerts, and alphanumeric records.

RF16	Alerts and notifications	The system must send automatic reminders to patients for medication intake and medical follow-up visits.
RF17	Alerts and notifications	The system must notify the physician when a clinical parameter changes significantly, sending an alert to their mobile device.
RF18	Alerts and notifications	The system must notify patients when their test results are available, including the interpretation and a recommended action.
RF19	Alerts and notifications	The system must support multiple notification channels (push, email, and SMS), selectable by the user.
RF20	Alerts and notifications	The system must issue an alert when a test has not been reviewed by the treating physician within a defined period.
RF21	Teleconsultation	The system must offer a teleconsultation module through video call or chat, without the patient travelling to the hospital.
RF22	Teleconsultation	The system must allow the physician to monitor patients remotely and issue recommendations or treatment adjustments.
RF23	Imaging (RIS/PACS)	The system must upload, store, and retrieve diagnostic images linked to the health record.
RF24	Imaging (RIS/PACS)	The system must share diagnostic images with specialists at other institutions.
RF25	Audit and traceability	The system must log all CRUD events with a timestamp and user identification.
RF26	Audit and traceability	The system must record the role and level of care of the professional who modified each clinical data item.
RF27	Audit and traceability	The system must include a feedback module for reporting improvement suggestions.
RF28	Role management	The system must manage differentiated roles: administrator, physician, nurse, assistant, and patient.
RF29	Role management	The system must require the patient's explicit informed consent for the handling of their clinical information.
RF30	Security and privacy	The system must implement an international security protocol to protect data in transit and at rest.
RF31	Security and privacy	The system must ensure that sensitive clinical information is accessible only to authorized clinical staff.
RF32	Security and privacy	The system must require explicit consent before sharing information with third parties.
RF33	Availability	The system must maintain a minimum availability of 99.999%, equivalent to no more than 5 minutes of downtime per year.
RF34	Availability	The system must operate 24/7 without maintenance windows that interrupt emergency care.
RF35	Performance and latency	The system must support 150-180 concurrent users without noticeable degradation.
RF36	Performance and latency	Latency must allow biomedical data to be received and displayed in real time.

RF37	Performance and latency	The system must answer EHR queries within a time that allows its use during a 15-minute medical consultation.
RF38	Usability	The interface must be intuitive and require no prior technical training.
RF39	Usability	The interface must follow the logical flow of the medical consultation: reason for visit, current illness, examination, diagnosis, and treatment.
RF40	Usability	The system must be available as a mobile app and a web page.
RF41	Interoperability	The system must be implemented under HL7 FHIR R4 as the single protocol for clinical data exchange.
RF42	Interoperability	The system must allow the exchange of clinical information with other healthcare provider institutions.
RF43	Interoperability	The system must support integration with biomedical devices through open standards.
RF44	Scalability	Storage must scale according to the number of institutions, providers, and active patients.
RF45	Scalability	The system must be deployed in the cloud with native FHIR support, redundancy, and automatic backups.
RF46	Regulatory compliance	The system must comply with current Colombian regulations: Resolution 866/2021, Law 2015/2020, and Resolution 1888/2025.
RF47	Regulatory compliance	The system must keep disease codes updated according to international nomenclatures.
RF48	Maintainability	The system must be administered by the institution without depending on external vendors.
RF49	Maintainability	The system must incorporate new devices and updates without architectural restructuring.
RF50	Multichannel accessibility	The system must operate under limited connectivity, supporting SMS for critical notifications.
RF51	Multichannel accessibility	The system must be accessible from phones, tablets, and computers through an adaptive interface.
RF52	Imaging (RIS/PACS)	The system must implement a PACS server to manage, store, and distribute diagnostic images, validating the identity of the requester.
RF53	Imaging (RIS/PACS)	The system must generate a UID for each patient and diagnostic image, preventing information mix-ups.
RF54	Imaging (RIS/PACS)	The system must support the DICOM standard for image acquisition, storage, and transmission.
RF55	EHR management	The system must register unidentified emergency patients with temporary placeholder data, with a mechanism for later linkage to their record.
RF56	EHR management	The system must manage the HIS (textual information) and the RIS (images and visual results) in an integrated way.

RF57	Infrastructure	The system must implement a hybrid architecture with local priority for critical real-time operations.
RF58	Infrastructure	The system must have chained asynchronous backups and a replica at a physically separate site.
RF59	Audit and traceability	The system must record the arrival, collection, and dispatch times of each diagnostic order to compute latencies.
RF60	Alerts and notifications	The system must generate automatic alerts when latency exceeds the expected time, with legally valid reports.
RF61	Data quality	The system must homogenize captured data through standardized format rules.
RF62	Data quality	The system must implement validation controls that prevent proceeding with incomplete fields or incorrect formats.
RF63	Data quality	The system must guarantee the integrity of clinical data, preventing unauthorized modification of critical records such as allergies.
RF64	Monitoring and metrics	The system must provide centralized log analysis and real-time monitoring with automatic alerts for anomalies.
RF65	Scalability	The system must be designed for scalability from the outset, so that it can grow without architectural redesign.