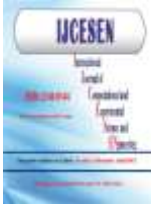
International Journal of Computational and Experimental Science and ENgineering
(IJCESEN)

Vol. 11-No.3 (2025) pp. 6776-6796
http://www.ijcesen.com



ISSN: 2149-9144

Research Article

Bridging IoT and Healthcare: Secure, Real-Time Data Exchange with Aerospike and Salesforce Marketing Cloud

Jiten Sardana[1*], Mukesh Reddy Dhanagari[2]

[1] Software Development Engineer, USA.
* **Corresponding Author Email:** jitensardana@yahoo.com - **ORCID:** 0009-0002-7679-4487

[2] Manager, Software Development & Engineering, Charles Schwab, USA.
Email: mukeshreddy@gmail.com - **ORCID:** 0009-0005-7281-2192

Article Info:

DOI: 10.22399/ijcesen.3853
Received : 05 July 2025
Accepted : 30 August 2025

Keywords

Aerospike,
Salesforce Marketing Cloud (SFMC),
Healthcare IoT,
Real-time data processing,
HIPAA compliance.

Abstract:

The architecture presented in this paper describes a secure and real-time bridge between healthcare Internet of Things (IoT) telemetry and engagement by mapping Aerospike as the low-latency data plane and Salesforce Marketing Cloud (SFMC) as the consent-aware engagement plane. Heterogeneous streams, viz. wearables, implants, bedside monitors, pumps, and sensors, consume through BLE to gateways, which transit across MQTT/HTTPS with mutual TLS. Kafka has fans emitting extensions of events to streams of processors that unit-normalise, schema-validate, de-duplicate, and identify features. Aerospike maintains time Series and change-data capture and alert state using hybrid memory, TTLs, secondary indexes, optional strong consistency, and supports emission of immutable events that serve as change-data capture. SFMC is fed de-identified attributes; Journey Builder uses SMS, push, or email to trigger on threshold violations, missed doses, or offline devices respecting consent. HIPAA governance is enforced with interoperability through FHIR resources and OAuth APIs; tokenization, audit trails, and DevSecOps guardrails. Experiments support >300k events/s with median write 3 5 ms (p99 <20 ms), bedside read p99 <35 ms, and fast-path interaction ~6 7 s between abnormal signal and journey entry. AZ failures and gateway restarts provide bounded backpressure and automatic recovery; the addition of a policy-decision point provides sub-millisecond overhead. A fault-tolerant, pragmatic blueprint decouples ingestion, analysis, and outreach to provide timely alerts and privacy preservation. The strategy demonstrates that Aerospike and SFMC can secure, cost-conscious, scalable real-time IoT communication supporting bedside applications, remote patient monitoring, and population initiatives.

1. Introduction

IoT healthcare links up sensors to medical devices and clinical systems, all too continuously measure, transfer, and respond to data. Endpoints include regulated devices like continuous glucose monitors and pacemakers, smart beds, consumer wearables that measure heart rate, sleep, and activity, and indirect measures like a lack of an inventory of the devices used to keep pace. Telemetry comes as a time series of events with metadata on the device. Devices to phones or home hub via Bluetooth Low Energy, to provider platforms via Wi-Fi or 5G using MQTT or HTTPS with TLS. Each measurement is accompanied by device identity and timestamps to trace back and understand its context. Its value can be quantified: signals reported in real time allow early detection of deterioration, automatic notifications, and closed-loop actions that prevent emergent care. Any ward can safely coordinate care at home. Automated vital capture limits transcription and charting errors. By controlling risk, closing care gaps, and performing real-world evidence studies to inform quality improvement at the population level, longitudinal telemetry permits risk stratification, care gap closure and real-world evidence studies.

Predictable low-latency data plane and an engagement plane to turn gathered insights into consented communications are needed to secure a real-time exchange. Aerospike offers the data plane. It is a high-performance NoSQL database with a hybrid-memory design- primary indexes in RAM and data on NVMe or SSD- with sub-millisecond reads and fast writes. Healthcare IoT key capabilities include record time-to-live records, cohort queries using secondary indexes, optional strong consistency, and cross-datacenter replication. Aerospike Connect Kafka and Spark enables integration of streaming and analytics streams, and predicate filtering is applied to server operations. The standard functions are ingestion buffering, a low-latency operational store of the latest vitals, and a feature store to feed the models. SFMC brings the engagement plane. Data extensions and contact builder are used with device properties and clinical flags matching against consistent profiles. API entry events can also be used in Journey Builder to send SMSs, push notifications, or emails in real time, when triggers such as threshold breach, missed doses, or device being offline occur. FMC will be implementing OAuth 2.0, permissions, TLS, and at-rest encryption, and its consent management may deploy permissions management through preference centres, auditing, and retention policies.

This paper discusses the integration of Aerospike and SFMC that can facilitate safe data movement in real-time in healthcare IoT systems. The study focus on pragmatic architectures to decouple telemetry at the edge and move it into operational stores, analytics and engagement journeys concerning health privacy, reliability, and latency requirements within HIPAA-regulated environments. The authors discuss how to ingest MQTT feeds into Kafka; model records in Aerospike namespaces, sets, and bins with TTLs and secondary indexes; perform stream processing to validate, deduplicate, and perform anomaly detection; and invoke the SFMC journeys to propagate notifications subject to consent and clinical rules. The study also discuss idempotent writes, replay, back-pressure, data minimisation and auditability. Expanded interoperability with FHIR-based interoperability and OAuth-protected APIs, SFTP and webhooks to move data between systems. There is information on the implementation details, performance measurement, operational reliability and compliance alignment so that the information is actionable.

Chapter 2 provides an overview of the literature in the area of healthcare IoT adoption, the enabling technologies, privacy concerns, and the previous uses of Aerospike and SFMC in real-time scenarios. Chapter 3 explains practices: data attributes; filtering of muddy sensor feeds; visual analytics; Aerospike architecture of namespaces, sets, bins, timestamp lifetime (TTL), and indexing; and SFMC patterns to provide a secure and consent-wise interaction. Chapter 4 shows real-time processing and analytics using the Kafka-centric streams, time-series modelling, anomaly detection, and predictive risk scoring. Chapter 5 covers these experiments to test the properties of throughput, latency, availability, message effectiveness and consumption cost. Chapter 6 explains the interpretations, the security and compliance, and identifies the limitations. Chapter 7 describes potential future work in the areas of scaling, integrations, and practical machine-learning improvements. Chapter 8 ends with significant findings and the system-wide effects of patient outcomes and operational efficiency.

2. Literature Review

2.1. The Rise of IoT in Healthcare

The modern healthcare environment is becoming highly instrumented with interconnected devices that are revolutionizing workflows, care processes, and patient experience. The devices in the Internet of Things include wearable photo plethysmography bands, ambulatory ECG patches, connected glucometers, implantable cardiac monitors, telemetry-enabled beds, environmental sensors, and asset tags. Recent work on defect/fault prediction offers transferable algorithmic baselines for reliability-sensitive pipelines ensembles, classical learners, and deep CNN/RNNs [27]. These devices produce high-frequency, heterogeneous streams, events, scalar values, multilead waveforms, and device-state changes that must be captured, transported, and interpreted without significant delay. Since clinical value drops off with latency, the application of acquisition, normalization, storage, and alerting must be proactively designed to operate within the real-time domain rather than the best-effort batch.

Interconnected healthcare IoT (wearables, ambulatory ECG patches, photoplethysmography bands, glucometers, implantable monitors, telemetry-enabled beds, environmental sensors, and asset tags) streams frequent, diverse data (events, scalars, multilead waveforms, device-state changes) to biobanks, EHRs, and clinicians via gateways as shown in the figure below. Constantly acquiring, normalizing, storing, and alarming reduces the effect of latency, compared to reputationally distant best-effort batch pipelines. The information is transferred through desktop managers and the Health Cloud. Present-day applications are wide-reaching, and they impact clinical and operational outcomes. With remote patient monitoring programs, near real-time vitals are beamed in to identify a decline in the patient before the subsequent readmission. Wearable, pulse oximetry, and symptom diaries are orchestrated to support low-risk patients at home under clinician supervision. In perioperative care,

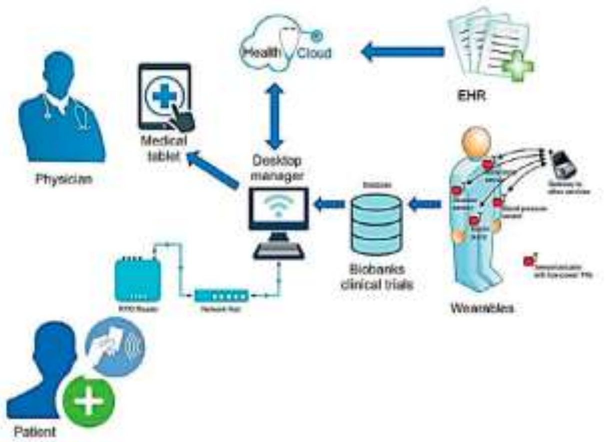


Figure 1: *Real-time healthcare IoT linking wearables, clinicians, cloud, and EHRs*

linked devices monitor recovery progress and initiate escalation in the event of deviation of the patient relative to expected baselines. In hospitals, location and condition sensors minimize the occurrence of adverse events, bed-exit risk, the location of critical equipment, and infusion-pump library currency. Encoding environmental sensors can be used to track isolation rooms and operating theatres regarding temperature, humidity, and pressure differential to aid in infection control. In addition to safety, IoT facilitates throughput management and capacity management, detailing patients' wait times, room turnover, and patient flow pinch points.

Facilitating such an expansion is an ecosystem of components that work together. Resource-constrained sensors use light messaging protocols like MQTT and CoAP; the gateway performs protocol conversion and buffering, as well as edge analytics to minimize bandwidth and provide redundancy during outages [31]. X.509 certificates or hardware trust anchors secure identity, and fleet-management services can be used to execute secure provisioning, firmware updates, and policies across the fleet. Data interchange builds on healthcare standards, including HL7 v2, FHIR resources, and DICOM metadata, enabling telemetry to be contextualized alongside orders, encounters, and problems. Edge computing is used to conduct first-mile validation, filtering, and down-sampling, and ensures safety interlocks proximal to the patient and reduces cloud transfer of noisy information.

It is in the value proposition that the analytics you put in place are fit for purpose. Rules engines manage or enforce deterministic alerts and protocol adherence, such as reminding a patient to take medicine when a wearable detects that the patient has been inactive during a predetermined period. The trend shifts, missing data, and sensor drifts are captured by time series analytics like rolling z-scores that can detect a decline in respiratory rate, even before absolute limits are breached. Anomaly - detection methodologies raise high-incidence, but clinically essential patterns, such as atrial fibrillation runs in otherwise normal-appearing rhythm strip. Predictive models assess the risk of exacerbation, readmission or medication nonadherence, so that proactive outreach can occur or clinicians can review in time. Visual analytics completes the loop by presenting time-aligned streams, annotations and context to clinicians within dashboards as part of the EHR workflo [35]. Equally, alert hygiene rate limiting, deduplication, and escalation policies limit the useless alerts and maintain the trust in the system by the clinicians.

2.2. Data Privacy and Security Challenges in Healthcare IoT

Healthcare IoT alters the attack surface by discretizing processing across the insecure environment, agglomerating identities, and even heterogeneous firmware stacks. Threats can include device tampering, rogue updates on compromised keys, replay/man in the middle attacks against constrained protocols or lateral movement spawned by compromised gateways or cloud-side data exfiltration through misconfigured storage or over privileged roles. Defensible posture goes deeper and layers controls at device, network, platform and application planes.

On devices, secure or measured boot and signed firmware provide reduced persistence of malware; secrets are resident in hardware security elements, and rotating identities limit the blast radius. Transport connections use mutual TLS; where limitations of protocols occur, application signatures and rolling nonces may be used to provide integrity and protection against replay. Gateways inspect messages and filter traffic to backend services, and network micro-segmentation denies east-west movement. At the application layer, enhanced multi-factor authentication, role-based and contextual policies rule access to user sessions and service-to-service calls.

The Security and Privacy Rules of HIPAA ensure that privacy engineering is treated as part and parcel of security engineering [36]. The increased amplification of risk by the real-time exchange is because streaming systems may leak through verbose logs, dead-letter queues, and analytical sandboxes without the classification of schemas with retention and masking. The most practically viable mitigation is to integrate governance in the development lifecycle such that code, infrastructure, and data contracts are scanned and tested in concert- static and dynamic analysis on the application, composition analysis on the dependencies, policy checks on infrastructure as code, and schema linting on data flows. Using these controls as a first-class gate in the continuous integration and delivery pipeline will identify defects before deployment and maintain assurance during operation, operationalizing a DevSecOps perspective to regulated systems. Comparative analyses across datasets and metrics help justify algorithm selection and thresholding before deployment [19].

Consent capture, data lineage and cross-organization exchange cannot be achieved without auditability and tamper-evident records. Cryptographically verifiable append-only ledgers and the blockchain-motivated patterns can enhance not only the strength of nonrepudiation, key management, and integrity of consent registries, record and event logs, but also of nonrepudiation, key management, and integrity, provided they are implemented with an eye to scalability and privacy. Practical cryptography designs store nothing with pointers to the outside world, preferring instead to write salted hashes to verify integrity, encrypted secrets in hardened modules, and both store and compute sur plus (logic) as little as possible to minimize attack surface. Research into decentralized systems has revealed both the security guarantees decentralized systems can achieve, as well as the issues with throughput, finality, and information privacy leakage that such systems can create, which motivates the use of hybrid architectures in which proofs across a ledger augment off-chain storage capabilities [24].

Operations security also needs to be perpetually warranted. Health of devices--version numbers of firmware, certificate expirations, vulnerability scanning status, and policy adherence should be sent in with the stream of clinical information. A zero-trust posture investigates and validates all of the requests and device posture evaluations, as well as gives preference to temporary credentials and restricted idiosyncrasies. The most universal safeguards about privacy belong to the data model proper: the use of tokenization in place of direct identifiers, minimization in storing only attributes needed to fulfil a purpose, and tiered access control where most workflows operate on non-identifying aggregate tables.

2.3. Aerospike in Real-Time Data Management

Healthcare telemetry is a real-time application that is write-dominated (high latency sensitivity), with variable record sizes and imperative durability. Aerospike's distributed NoSQL design is uniquely suited to address these requirements in that it combines a primary index that resides in memory with storage that is optimized to work with an SSD, providing low-latency reads and writes at high concurrency. In the case of IoT streams, a typical pattern is to profile a namespace dedicated to telemetry and the sets, partitioned by device class (wearables, monitors, pumps), so that the policy, retention, and secondary indexing can be optimized per modality. Decision-tree and k-nearest-neighbors variants remain competitive for low-latency, interpretable screening in production pipelines [2]. Composites are created based on a device indicator and a rough period, which can allow efficient sharding; bins consist of canonical values, quality flags, and inferred features such as rolling averages and slope. Time to live policies allow high-volume raw events to expire; however, curated aggregates and facts that are audit-relevant are maintained. As shown in Figure 2 below, Aerospike provides the foundation of a high-volume telemetry system: edge clusters receive device streams, store keys in memic indices, and flash-optimized disks. Different device classes separate telemetry namespaces and employ composite keys (device+period). Bins contain canonical values, quality flags, and inferred features (rolling averages, slopes). TTL policies dispose of high-volume raw remedies as curated aggregates, and audit-relevant facts are retained and synchronized to the system-of-record in support of compliance and analytics.

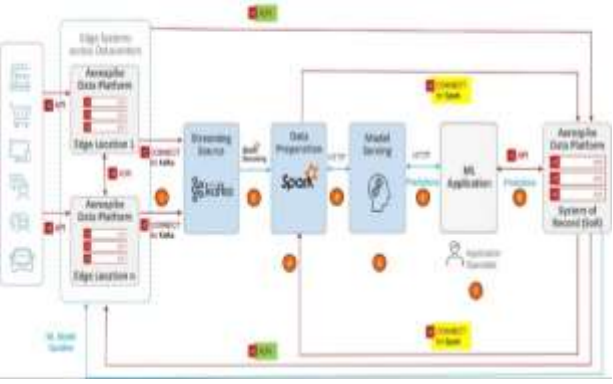


Figure 2: *Real-time Aerospike pipeline from edge telemetry to ML predictions*

Low-latency deduplication and idempotency are important where the network can be lossy or when gateways may replay buffered messages. Aerospike atomic operations are non-competitive counters and last-seen timestamps. The query patterns best suit key lookups of the latest state and bounded range scans of recent windows; secondary indexes can be used to selectively query anomaly flags or cohorts of devices without full-partition scans. Cross-region replication ensures service continuity and can support active-active patterns when care teams across geographies and stream processing of feature extraction and model scoring occur upstream.

Microserved Aerospike typically is the operational source of truth in front of a streaming backbone and a batch lake, providing the operational view of the latest known good state of the devices, risk scores, and alert eligibility flags. This decoupling isolates patient-facing decisions and the associated projects against downstream analytical workloads and maintenance windows. The architectural tension, though, is the issue of scaling and cost governance. Elastic services are prone to uncontrollable costs as device throughput ramps and proactive right-sizing add length to tails and introduce alert loss. Microservice analyses stress trading horizontally scalable systems against financial limits using autoscaling guardrails, workload shedding and noisy neighbour consolidation, principles that also apply to Aerospike-based telemetry infrastructure [7, 32].

Data governance superimposes a model. The amount of protected health information in the operational store must also be minimized, tokenized identifiers must map to a separate vault, and access paths must be limited to service identity rather than the widespread database roles. Secure encryption in transit and at rest minimizes exposure, and audit events provide insight into who, when, and why data was accessed with a corresponding audit trail for clinical workflows. Thereby, such patterns enable a no-latency datastore to be safe within regulated decision loops.

2.4. Salesforce Marketing Cloud for Data Integration

Patient engagement is one of the key determinants of health outcomes, and one way IoT-derived insights can be leveraged to improve adherence, self-management, and escalation levels is through targeted, relevant communication at key moments in time. The Salesforce Marketing Cloud offers features such as data extensions, event-driven APIs, segmentation rules, dynamically changing content, and journey orchestration that can all be used to integrate near-real-time signals into outreach programs [14]. A typical model is to de-identify events on the device layer into a profile store that can only be ever resolved to individuals with short-lived tokens; the engagement layer then triggers journeys when risk levels or adherence scores change.

Minimization of data is key when using a marketing platform in a clinical setting. Instead of exporting raw physiologic measures, integration designs export abstracted measures with limited retention and do not store sensitive health information unless necessary to achieve behavioral enforcement. Personalization concentrates on customizing templates according to literacy level, language of choice and the kind of device, coupled with scheduling patterns that do not emit alert fatigue. There is evidence of nearby fields that demonstrates that AI-enhanced feedback loops can be used to boost engagement when feedback mechanisms become timely, specific, and actionable; the latter elements can be directly applied to patient education and adherence activities [16].

The collaboration process will depend on an accessible and secure interface. The Marketing Cloud APIs consume events pushed to the operational store by a mediation service, where schema validation, consent checks and rate limiting take place. Keeping duplicate messages at bay is the removal of such duplicates via de-duped messages (or keys), and transient failures are resolved using a replay queue. Observability provides delivery measures, bounce categorizations, and conversions by segment; these data are processed to the operations plane to inform engagement options affecting care cut-offs, such as an increased human contact when the digital touchpoints fail repeatedly. Collectively, these integration patterns enable a marketing-grade platform to serve as a low-friction channel of compliant communications that reflect the up-to-the-minute device information without the presence of superfluous clinical background.

2.5. Integrating Aerospike and Salesforce Marketing Cloud for Healthcare IoT

An end-to-end pattern that links device telemetry to engagement starts at the edge, where gateways authenticate devices, validate them, and send events to a streaming fabric. Stream processors enrich messages with unit normalization, calibration, and feature extraction, such as rolling variance, and record operational facts to Aerospike: the latest device state, risk scores, and alert eligibility flags. When a flag changes, such as switching between the set to each of monitor and notify, a mediation service consults consent and outreach policies and, where allowed, publishes a low-value event to Marketing Cloud with a correlation token, risk tier, and template identifier. Journeys subsequently coordinate transactional SMS, push or email with deduplication, retries, and branched logic on acknowledgement [3]. Deliverability outcomes follow back through a webhook to update engagement capabilities in the Aerospike database, completing the loop to allow further clinical decisions to reflect who received, opened, or engaged in outreach.

Assured security comes by design. Identifiers are tokenized, mutual TLS secures cross-system calls, and credentials have low-lived, limited claims to reduce blast radius. DevSecOps guardrails test service definitions, infrastructure templates, and schemas prior to deployment, and stream-based monitors ensure latencies and volume rates do not exceed safe limits at run time. Collectively, Aerospike provides the core of low latency and high throughput to real-time decisioning, with Marketing Cloud providing consent-aware and multichannel communications. The concerted effort enhances freshness, minimizes human interactions, enriches activity and integrates privacy and security throughout the IoT data life cycle safely.

3. Methods and Techniques

3.1 Description of Data Set

Various sources of IoT healthcare telemetry, wearable devices, home telemonitoring systems and networked clinical devices compile the data. Wearables longitudinal studies can monitor heart rate, heart-rate variability, skin temperature, oxygen saturation (SpO_2), respiratory rate, and accelerometry at 1-1/20 Hz [5]. Home kits can give frequent readings measured by blood-pressure cuffs, glucometers, scales, and finger oximeters, although they are not continuous. At the clinical level, the frequency of status, alarms, and setpoints exchanged by smart pumps, ventilators and beds is in a sub-second to minute range. Gateway gadgets gather these signals near the lair and transfer them to the platform via MQTT or HTTPS.

The corpus is a fusional blend of structured and unstructured work. Mutable time series comprise the majority and are persisted as append-only streams; waveform segments are resolved as binary blobs with matching JSON indices; discrete events use key-value pairs, device logs are semi-structured text, and interoperability messages (HL7 v2 or FHIR) are JSON documents. A canonical observation envelope—{tenant, patient_key, device_id, metric, ts_event, ts_ingest, value, units, quality_flags}—supports downstream normalization. The provenance (gateway, firmware, calibration lot) is put in the records. There is burstiness and clinical workflow, which are higher during nursing rounds and transfers at night.

Table 1: *Key Aspects of IoT Healthcare Data Sources*

Source / Context	**Example devices or messages**	**Key signals / content**	**Sampling / cadence**	**Format & transport / notes**
Wearables (longitudinal)	Photoplethysmography bands, ambulatory ECG patches	Heart rate, HRV, skin temperature, SpO_2 , respiratory rate, accelerometry	1–1/20 Hz (1 to 0.05 Hz)	Append-only time series; waveform segments as binary blobs with JSON indices; sent via MQTT/HTTPS gateways
Home telemonitoring kits	Blood-pressure cuffs, glucometers, scales, finger oximeters	BP, glucose, weight, SpO_2 readings	Frequent, not continuous	Discrete events as key–value pairs; transported via MQTT/HTTPS
Networked clinical devices	Smart pumps, ventilators, telemetry-enabled beds	Status, alarms, setpoints	Sub-second to minute range	Append-only streams; TTL expires high-volume raw events; curated aggregates/audit facts retained
Interoperability payloads & schema	HL7 v2, FHIR; canonical observation envelope	{tenant, patient_key, device_id, metric, ts_event, ts_ingest , value, units, quality_flags}	Event-driven	JSON documents enabling downstream normalization
Provenance & operations	Gateway, firmware, calibration lot; clinical workflow	High cardinality; partial order; non-stationary (motion artefacts/sensor drift)	Burstier during nursing rounds/transfers	Real-time visibility drives alerts, routing, and asset dispatch; write-dominated workload

This data is operationally essential, including streaming vitals scoring, early warning indicators like the titration guardrails and/or the fall-risk alert, as well as deterioration alerts. They have high cardinality, partial order in that different connectivity values are hardly comparable, and are non-stationary as there are either motion artefacts or sensor drift. Real-time visibility can directly influence logistics, such as who is alerted, where they are routed, and to which asset is dispatched [40]. The reality of algorithm-based dispatching in other areas of telemetry density highlights the ease with which time-sensitive data and action rudiments can lead to improved operational costs.

3.2 Data Pre-processing

Pre-processing is partitioned between the edge gateways and the core analytics plane. Filters are used at the edge: debouncing of photoplethysmography and spike clipping, band pass filtering of accelerometry and beat detection using refractory-period logic, and smoothing of accelerometry. Timestamps are modified to adjust to network time, and sequence numbers have also been superimposed to notice loss or replay. Payloads are serialised using the CBOR or Protocol Buffers and grouped to respect power budgets. Patient identifier is replaced with rotating pseudonyms intended to protect privacy, but enable linking; other identifiers are salted and hashed. For compute-intensive analytics, coupling ML workflows with HPC resources can accelerate feature extraction and simulation-backed model development. [38].

Within the core plane, ingestion services standardise units to UCUM codes, validate schemas, and perform idempotent upserts keyed by {device_id, metric, ts_event}. Enriched site, bed and encounter context are added to records and written to a staging topic to facilitate quality checks. Data quality constraints: range and rate-of-change boundaries, cross-channel consistency tests, and device/firmware allow lists. Data are remodelled to canonical grains by forward-filling to valid; missing data are marked with data suppressed status, scheduled or unscheduled, or administrative transfer. Mutilation after the fact is per policy: last-observation-carried-forward with brief data gaps, Kalman smoothing with quasi-linear signals, and spline interpolation in case of dense data. A drift detector compares windows to patient baselines to detect calibration changes.

Label scarcity is overcome through the creation of representation learning on unlabeled telemetry before supervised modelling. Self-supervised pretext learning, Temporal masking, and next-step prediction learn representations that stabilize anomaly detectors and reduce feature engineering. According to the research findings or evidence in object-detection, self-supervised training leads to better detection when the available labels are scarce; the same can be applied to IoT signals where label prices are high [33]. Embeddings and quality flags are stored in a feature store versioned by schema and time window so that identical experiments can be rerun.

3.3 Data Exploration using Visual Analytics

Exploration is provided in a layered visual stack that accommodates ward-level surveillance and investigation of the individual patient. Overview dashboards give patient tiles micro-sparklines on core vitals; color bands are individual control limits based on baseline quantiles. A bed-map heat-grid is arranged using the magnitude/rate of change in deviation. Drill-through then synchronizes patient timelines with device settings and medication administrations, vitals, and alert states. Brushing chooses time windows, and then sends this choice to all panels to do focus-plus-context analysis.

On higher-frequency channels, signal-aware plots are employed. Wavelet scalograms show ectopy by ventricle; spectrograms of accelerometry view instability in gait; cross-correlograms show beat-to-beat coupling between heart rate and SpO polys on suspected apnea. Change-point overlays (CUSUM) are placed on top of vitals to indicate regimen change or the artefacts generated by technology (such as post-calibration step). Residual density plots following seasonal-trend decomposition of flag sensor drift. In the case of cohorts, small-multiple panels will be used to compare the distributions between wards and device models and help highlight any systemic bias.

The design is safety-sensitive. All of the charts honor data-quality flags, coloring in suspicious portions and refusing to show numerical annotations where provenance is unknown. Through tooltips, units and quality codes, as well as the identity of the gateway and the firmware, are displayed next to each point. Annotations like cannula changes, titrations, and therapy interventions are not the end of the event but instead appear as vertical markers to synchronize clinical context and signals. Comprehensive performance comparisons synthesize multiple algorithms and metrics, providing survey-grade baselines we mirror here [13]. It is capable of real-time rendering (large windows use incremental down-sampling, and raw points when zoomed). Analysts can export the de-identified slice used in a view, or use this and other data to create safely reproducible notebooks that can match what clinicians saw during incident review.

3.4 Real-Time Data Processing with Aerospike

Aerospike is used as the low-latency operational store of hot telemetry and alert state. Ingestion layer accepts MQTT or HTTPS coming in via gateways into a streaming bus; stateless normalizer converts messages to a canonical envelope and writes to Aerospike using keys of the form {patient_key|device_id|metric|ts_bucket} in an idempotent manner. Dashboards are divided into telemetry, waveform index, and alert state to isolate workloads. Instruments in Bins contain typed fields--value, quality, attributes--and on a per-record time-to-live (TTL) to facilitate automatic expiry of ephemeral observations. Secondary indexes on patient key and metric allow selective reads with no scan-intensive patterns.

Windows of streaming analytics are maintained through ring-buffer metadata contained within each series. The use of hot keys is time-sharded to flatten bursts. In-database means that in-database user-defined functions compute rolling aggregates and quantiles to minimize network hopping. Threshold violations commit slim documents to alert_state with key {patient_key|alert_type} to enable idempotence and suppression latencies. Cross-datacenter replication offers disaster recovery; rack-aware partitioning minimizes the failure domain; the cluster scales to meet p99 read latency below clinical criteria so that bedside viewers can update at sub-second intervals.

Table 2: *Key Components of Aerospike Real-Time Processing Pipeline*

Component/Layer	**Purpose**	**Key structures & config**	**Performance & resilience**	**Outputs/Notes**
Ingestion gateways	Accept device traffic	MQTT/ HTTPS → streaming bus	Backpressure at bus; stateless ingress	Uniform entry for all telemetry
Normalizer	Canonicalize messages	Envelope; idempotent writes with key `{patient_key	device_id	metric
Operational store (Aerospike)	Hot telemetry & alert state	Bins: value, quality, attributes; per-record TTL	Sub-second p99 reads for bedside viewers	Dashboards split by telemetry/ waveform index/alert state
Indexing & queries	Selective reads	Secondary indexes on patient_key, metric	Avoid scan-heavy patterns	Targeted queries only
Stream windows & state	Maintain analytics windows	Ring-buffer metadata per series; time-sharded hot keys	Burst flattening; minimal contention	Stable rolling computations
In-database analytics	Reduce network hops	UDFs for rolling aggregates, quantiles	Compute near data	Fresh features in store
Alerting	Persist threshold violations	Slim docs in alert_state with key `{patient_key	alert_type}`	Idempotent; suppression latency control
HA/DR topology	Continuity under failures	Cross-DC replication; rack-aware partitioning	Minimized failure domains	Fast regional recovery
Reliability & guarantees	Clinical safety properties	Config baselines; verification tests	Atomic alert state, deterministic TTL expiry, read-your-own-write	Predictable bedside behavior
Analytics integration	Downstream /feedback loop	Aerospike AE streams to analytics	Single hot path; near real-time risk fed back	Continuous model-driven updates

Reliability and resilience options are clear since clinical safety requires the dependability of behavior. Configuration baselines and verification tests of a real-time store build on trade-offs documented in other document-oriented NoSQL systems, in particular, the trade-off between maximum throughput and stronger guarantees [9]. The platform focuses on the atomicity (affirmative) of alert state, deterministic TTL expiry, and read-your-own-write behavior of bedside flows. Aerospike AE streams data to the downstream analytics, eliminating the second hot data path; near real-time risk levels are fed back into Aerospike.

3.5 Integrating Salesforce Marketing Cloud for Data Security and Management

The SFMC is used as the engagement and consent layer, as clinical systems still contain the protected health information. A privacy gateway adds policy tags to change-data-capture events in Aerospike, and projects only approved elements into SFMC Data Extensions: hashed patient keys, risk tier, and approved event types. Direct identifiers never enter the clinical perimeter; mappings are stored in a vault to be audited upon lookup.

Journey Builder uses event API triggers to send a reattachment prompt when the device is offline for ten minutes, or to escalate once a high-risk tier lasts 24 hours. Decision splits reference is non-PHI attributes and risk score in Aerospike. Channel preferences and consent items of lawful outreach are kept in Contact Builder. The security controls section includes the least privilege, which provides for the use of customer-managed keys, separation of roles, and retention policies. Answers to engagement are routed back through a safe landing zone and attached to the de-identified key, completing an engagement loop without PHI exposure in the marketing layer [34]. Comprehensive auditing of IP allow-listing and API rate limits helps detect abuse in real time within seconds.

4. Real-Time Data Processing and Analytics in Healthcare IoT

4.1. Overview of Real-Time Data Processing in Healthcare

The basis of safe and efficient care is real-time processing of the physiological data, as many derangements develop on a minute time scale rather than a day scale. Uninterrupted wearable and bedside monitor telemetry, implantable devices, and intelligent equipment produce high-frequency measurements, device status, and workflow messages. Making these stream decisions that must operate on tight latency limits allows clinicians to escalate care and address adverse events early and utilize coordinated resources. The same telemetry is used in the operational teams to enhance throughput, reduce length of stay, and improve staffing.

Typical use cases cut across the care continuum. Monitoring of heart rate, oxygen saturation, respiratory rate, and blood pressure continuously helps to provide early warning of sepsis or cardiac rhythm failure, and the alerts are directed to rapid response teams. Remotely monitored programs capture weight, blood pressure, and glucose at home and set thresholds customized to patient histories and care programs to intervene before readmission. Device telemetry in perioperative and intensive-care environments implements safety checks, such as mismatches on the setting of the ventilators, infusion pump occlusion, or low battery in mobile monitors, thus preventing hazards in time. Capacity models are fed by operational signals that include bed turnover, environmental sensors, and location tracking, and assist coordinators in locating patients and assigning procedures.

A practical architecture devises a separation between the data plane and the decision plane. Gateway gets heterogeneous protocols like MQTT, BLE, and proprietary serial into authenticated, secure events [21]. Messages are fan-out, buffered, and ordered to stateful processors and a low-latency operational data store by a streaming fabric. Downstream, an engagement layer sends alerts to clinicians, patients, and back-office systems via channels that suit their urgency and privacy considerations. The following subsections look at grounding the data plane in Aerospike and decision activation in Salesforce Marketing Cloud.

4.2. Role of Aerospike in Real-Time Data Processing

Aerospike meets IoT healthcare requirements of high write rates, low read latency, predictable SLAs, and resilience with an in-memory primary index and persistent storage tiers. Pre-aggregated measurements marked with a composite identifier of a patient or a device, timestamp-composed keys are upserted into stream processors. Timebox modelling works well: the small window is recorded, for example, one minute, with bins to record aggregate builds, the most recent observed value, and quality flags. TTLs on the record level can expire stale windows as they choose to preserve the new ones, thereby minimizing the size of storage and putting dashboards at risk of outdated numbers.

The Aerospike database provides low-latency reads and a high number of write transactions possible in real time, such as in IoT healthcare, and the possibility to achieve sub-second reads by indexing the Aerospike tool with an in-memory primary index with persistent storage tiers. The upserting of pre-aggregated measurements keyed by patient/device and time stamps is done using stream processors, as shown in the figure below. Aggregates, as well as last observed values and quality flags, are stored in one-minute bins. At the record level, TTLs reliably drop stale windows, leaving fresh data, and with a dashboard reflecting current, predictable patient telemetry and alerts.

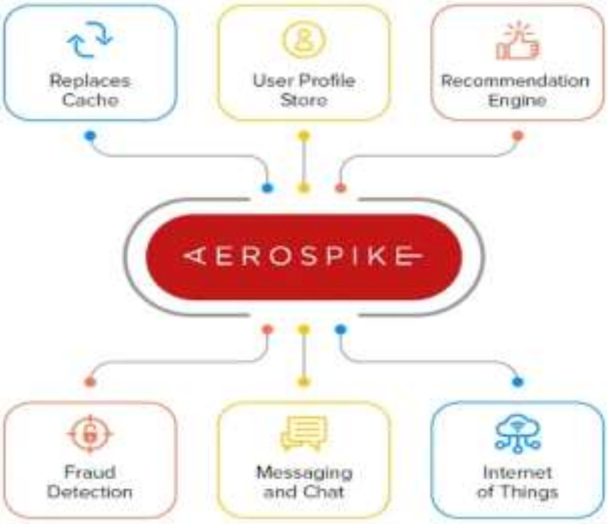


Figure 3: *Aerospike enables timeboxed, low-latency IoT healthcare analytics*

Support for fast access to the current best value, feature rolling windows, and a one-sided past view of feature values is provided by secondary indexes on identifiers and aggregate time bounds. Checks of generation and predicate expressions prevent out-of-order updating; upserts are idempotent to avoid duplicates in an at-least-once delivery model by a device or stream processor. Change notification announces mutations to the streaming backbone such that analytic jobs accept deltas instead of rescaling partitions.

The challenge of durability and availability is addressed by synchronous replication in a cluster and cross-datacenter replication. Medication orders, device commands, and consent states can be done on strong-consistency namespaces, but the raw values of vitals could be left to high-throughput eventual consistency namespaces. Sub-millisecond to millisecond latency is maintained by the store because both the index is in memory and the values are in fast SSDs. Comparative studies on reference datasets underscore the value of head-to-head benchmarking before deployment [8]. The operationally Aerospike uses hot operational storage and analytical persistence. Dashboards, rules, and risk scores are driven by hot data, and jobs export curated slices, such as hourly aggregates, to a data lake to enable retrospective analysis. Connectors allow change-data-capture egress to the streaming fabric, without disrupting write paths, maintaining service-level objectives.

4.3. Data Analytics Techniques for Healthcare IoT

Statistically sound and computationally efficient analytics on streaming physiological and operational data are required. One starts time-series processing with the steps of resampling and rejection of artifacts, such as removing motion noise in photo-plethysmography or sensor dropout compensation, then filters and extracts features. The commonly used statistics are rolling means and variances, heart machinery variability, nocturnal dipping, intermodality cross-correlations, and trends based on derivatives. For weakly supervised anomaly detection, one can use the exponentially weighted moving averages, seasonal decomposition with robust z-scores, and change-point detection to provide an alert at low cost with no training.

In situations requiring a richer context, the detection and prediction are enhanced using deep sequence models. The temporal convolutional networks perform well at finding local patterns, whereas the gated recurrent units or long short-term memory cells can model longer dependencies. When the system has to consider historical contexts, i.e., recent episodes, diurnal cycles, and patient-specific baselines before settling on whether an observed pattern is troubling, memory-augmented methods are applicable. An example of this capability is that of dynamic memory inference networks, which marry connection attention with an outwardly stored and reused context awareness, which can be applied towards anomaly classification in continuous physiological streams. Classical ML and deep-ensemble baselines remain strong comparators in safety-critical prediction tasks, including boosting/voting ensembles and CNN/RNN architectures [28].

The results of predictive models are usually expressed in terms of deterioration risk, readmission risk, or early warning scores. Under a real-time topology, models are compiled into minor artifacts and served as stateless micro services behind circuit breakers. At the point of each inference, provenance is added: version of the model, feature schema, and confidence, allowing audit and undo. Since the population of patients is variable and devices exhibit variations, a champion-challenger model compares the candidate models against the incumbent with a shadow scoring system. Latency, calibration, and false-alarm rate are tracked to monitor performance; drift is monitored, and result feedback initiates periodic retraining.

There should also be interpretable decision support. Regardless of using deep models, the system ought to reveal rationale through saliency over periods, counterfactual simulations of hypothetical better performance, and references to the specific data and event leading to triggering the alert. The alert storm is mitigated through a threshold logic and cooldown timers, and deduplication in different channels results in a single notifiable event being sent to the responsible care team.

4.4. Integrating Salesforce Marketing Cloud for Data-Driven Insights

Salesforce Marketing Cloud activates engagement by turning analytic signals into proper contextual communication—event capture. The process starts with capturing an event that is relevant to Aerospike when Aerospike crosses a threshold, misses an adherence milestone, or reaches a discharge milestone: an integration service publishes a de-identified event enriched with context. The service maps this to Marketing Cloud data extensions, and journeys are initiated. Journeys orchestrate cross-channel outreach: SMS is best suited to time-sensitive alerts, push messages via wearables, email is good at education, and portal secure messages enable richer content [1]. Decision splits implement policy whereby only the eligible patients receive messages, and others navigate to staff tasks to have their follow-ups done manually.

One essence of this is privacy-preserving identity. The engagement layer contains only tokenized identifiers; a secure resolver service pairs the tokens with the patient identity within the boundary of the provider. The preference and consent flags are read in real time prior to emission of a message. The results of engagement, such as delivery done, opened, clicked, and acknowledged, are fed back to operational storage to recalibrate nudges and refresh risk models. Representative results guide content experimentation so clinical teams can dial frequency of posting, scheduling, and channel without model alteration. Such design factors reflect advice to healthcare communications infrastructure: increase with message pressure, decouple as much as possible to clinical systems, and maintain flows that can be audited concerning consent and confidentiality [30].

4.5. Challenges and Solutions in Real-Time Analytics

Real-time healthcare analytics encounters technical and regulatory challenges that are conjoined. A burst of device events or a network partition may blow latency budgets. The data's integrity challenges are produced by clock drift, multiple sends, and out-of-order arrival, and the safety requirements demand end-to-end encryption, non-destructible records, and provable consent management. Humanly, alert fatigue and brittle workflows destroy trust, unless the system is adjusted to the local practice.

There exist several design options that curb such risks. The write-optimized duration of time and the partition-based sharding of Aerospike absorb bursts; backpressure within the streaming fabric and the rate-limited acknowledgements of the devices maintain a consistent ingestion. Composite keys using timestamps and generation checks provide idempotent, monotonic updates that resolve late events. TTLs and tombstones ensure that stale records do not pollute dashboards, and strong-consistency namespaces protect command topics like changes to doses and configuring devices. Traditional classifiers such as Random Forests, Logistic Regression, and k-Nearest Neighbors are frequently used as competitive baselines for fault/defect prediction and operational risk screening [4].

Algorithmic discipline developed in other real-time fields is also valuable for operational orchestration. Dispatch systems used in less-than-truckload logistics are based on multiple re-prioritization of pickups and deliveries to fulfill a service constraint within a state of uncertainty; the same concepts can be applied to optimize health systems routing, staff assignment, and triage operations: Engagement should also respect privacy and preference by gating journeys based on consent and offering the clinicians overriding paths [26].

5. Experiments and Results

5.1. Experimental Setup

The assessment analyzed a complete pipeline that reads heterogeneous IoT data from healthcare into Aerospike, enabling operational analytics and the propagation of filtered signals to Salesforce Marketing Cloud to engage with patients promptly. The emulated devices consisted of three classes of continuous wearables with heart-rate, SpO2, respiration, and activity measurements in 15 to 5 Hz, episodic monitors like glucometer and blood-pressure cuff with events posted per 5 to 60 minutes, connected clinical devices (smart pumps and beds) with posture, infusion, and maintenance telemetry at 1 to 2 Hz. The vendor documentation and clinic interviews were used to calibrate device mixes, session lifetimes, disconnection patterns, and payload sizes; the event timeseries were diurnally shaped and jittered randomly to reflect real usage and churn. The testbed was two availability zones within one cloud region. There were three Aerospike data nodes (16 vCPU, 128 GB RAM, dual NVMe SSD) and a management node per zone [39]. Ingestion employed two stateless ingress gateways per zone that ended the mutual-TLS device sessions. A three-broker cluster of Kafka gives buffering and fan-out. Salesforce marketing cloud integration that was carried out in another VPC.

The methodology of the study consisted of four stages. Phase A confirmed the correct operation of functional behaviors idempotency, schema evolution, out-of-order handling, and consent enforcement, in both everyday situations and during adverse conditions (clock drift of +/-120 seconds, packet loss of 0-5%, and duplicates at up to 2%). Phase B measured steady-state throughput and latency by sweeping the input across a range of 25k to 400k events/second. In phase C, resilience was emphasized through some induced failures: single AZ outage, Kafka broker loss, data-node eviction, and gateway restarts. Phase D quantified the engagement outcomes as measured using synthetic but clinically plausible situations seeded on the streaming analytics layer.

Table 3: *Experimental Setup Summary for IoT–Aerospike–SFMC Pipeline*

Domain	**What was evaluated**	**Key specs / parameters**	**Purpose / outcome**
Pipeline scope	End-to-end flow: heterogeneous IoT → Aerospike → operational analytics → filtered signals to Salesforce Marketing Cloud	Device data normalized and persisted; engagement triggered from filtered events	Validate timely patient outreach from real-time telemetry
Device mix & rates	Continuous wearables; episodic home monitors; connected clinical devices	Wearables 15–5 Hz (HR, SpO_2 , respiration, activity); glucometer/BP 5–60 min; pumps/beds 1–2 Hz; diurnal shaping + random jitter	Realistic load, payload sizes, lifetimes, disconnect patterns
Testbed & topology	Two availability zones in one cloud region	Per AZ: 3× Aerospike data nodes (16 vCPU, 128 GB RAM, dual NVMe SSD) + 1 management node; 2 stateless mTLS gateways; Kafka 3-broker cluster; SFMC in separate VPC	Measure performance and HA under production-like layout
Methodology (Phases)	A: functional correctness under normal/adverse conditions	Idempotency, schema evolution, out-of-order, consent; drift ±120 s, loss 0–5%, dup ≤2%	Ensure correctness and policy enforcement
	B: steady-state performance	Input sweep 25k → 400k events/s; latency observed end-to-end	Characterize throughput and p-latencies
	C: resilience / failure tests	Single-AZ outage, Kafka broker loss, data-node eviction, gateway restarts	Validate graceful degradation and recovery
	D: engagement effectiveness	Synthetic but clinically plausible scenarios seeded in streaming analytics	Quantify patient-facing outcomes
Redundancy & routing	Dual active-active writes (direct + Kafka-buffered)	Circuit breaking and weighted routing to minimize correlated risk	Eliminate single-point exposure; sustain writes during faults
Security & access	Zero-trust controls	Device X.509, mTLS, role-scoped service accounts, short-lived tokens, PDP enforcing purpose/consent before reads/writes	Constrain access; maintain compliance and data integrity

Incidents of redundancy were premeditated. The two ingestion channels direct writes to Aerospike, and Kafka-buffered writes were active-active, and circuit breaking and weighted routing were used to minimize correlated risks. This reflects dual-sourcing concepts that decentralize dependencies and overcome single-point exposure and enhance resilience to component failure [12]. Using a zero-trust posture, device X.509 identity, mTLS, the strategically placed role-scoped service accounts, short-lived tokens, and a policy decision point (PDP) constrained access to only allowed purpose and consent before any read or write could be committed.

5.2. Implementing Aerospike for Real-Time Data Processing

Aerospike had three namespaces: iot_timeseries, containing high-velocity sensor data; iot_events, containing discrete alerts and lifecycle changes; and iot_profiles, containing device-patient association and channel preferences. Memory 2 Indexes were memory-resident; record bins lived on NVMe in hybrid-memory mode. The timeseries support applied a fixed model indexed by {patient_id|device_id|epoch_bucket} to support append-only writes and time-bounded queries. Secondary indexes were used on patient_id and device_id with a time bucket to enable effective retrieval and cohort scans.

Ingestion gateways checked signatures of payloads, units (such as mmHg, mg/dL), and dropped schema-incompatible writes. The policy used was commit-to-device on critical vitals (SpO_2, heart rate), and commit-to-master was adequate on non-critical measurements [22]. Pipelining was used with adaptive in-flight depth by asynchronous clients based on server queue and RTT measurements. A reconciliation job flattened partial buckets of out-of-order arrivals and calculated rolling aggregate (mean, std, percentile sketches) that were consumed by dashboards and triggers. Query paths were used for two workloads. Patient dashboards retrieved the nearest 15 minutes of vitals with a p95 target of < 100ms. The per-patient caching of pre-aggregated windows precluded hot shard amplification. To formulate at-risk cohorts, population analytics issued short scans predicated on such values as SpO_2 < 92% for >3 minutes. An analytics throttled scan scheduler was set to 20% of I/O credits.

Rack-aware partitioning across the cluster meant that replicas were not placed in the same rack; migration throttles capped tail latency during churn. Hysteresis was used on queue depth and write latency with gateway auto-scaling. Health rules forced canaries to restart when p99 was greater than 25 ms. Throughout the throughput sweep, Aerospike held constant performance: consistent ingestion at 310k events/second produced a median write latency of 3.8 ms and p99 of 12.2 ms; at 400k events/second, the median was 5.3 ms and p99 was 16.9 ms. Read paths 15-minute windows were up to 35 ms p99 below pre-aggregations.

5.3. Integrating Salesforce for Data Management and Patient Engagement

The three flows were integrated with Salesforce Marketing Cloud. A near real-time extract delivered a consent-scoped Patient 360 profile, such as risk tier, care plan flags, recent abnormal event, and synchronized a small set of attributes with Contact records using REST APIs. High-severity events spawned a rapid route: a safe webhook launched a Platform Event that inserted the patient into a Journey in a queue. It also exported de-identified aggregates at intervals of a day to perform longitudinal cohort analysis and A/B testing off the operational pathway.

Patientcentricity was all around identity and consent management. A permission ledger was used to store purpose-of-use and permission to channels (SMS, email, push) with region-specific constraints. A decisioning service assessed whether a clinical event might provide a driver for communication and the channel to be used, generating tamper-evident audit logs. Accessibility and localization were included in message templates. Journey decision points were comprised of a contextual multi-armed bandit that learned the best channel and send window on each patient [17]. The observed opens, clicks, and acknowledgments, as well as the updated per-patient priors with privacy-preserving noise, are observed by the bandit.

A consistent architecture is established based on blockchain that controls patient permission and identity, as shown in the figure below. A permission ledger captures region-specific purpose-of-use and channel decisions (SMS, email, push). A decisioning service and smart contracts process clinical events and make decisions to present a pre-defined, approved channel of communication, as well as create tamper-evident audit logs. The accessibility and localization are processed in message templates. A contextual multi-armed bandit simultaneously learns per-patient reactivity windows and channels based on opens, clicks, and acknowledgments, updating privacy-preserving priors, with peers updating the ledger across the hospital network safely.

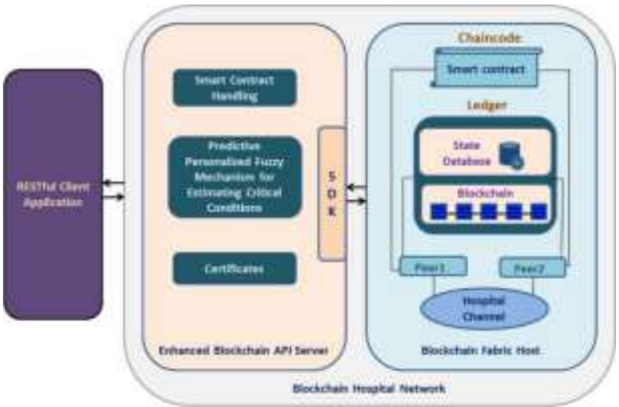

RESTful Client Application
Smart Contract Handling
Predictive Personalized Fuzzy Mechanism for Estimating Critical Conditions
Certificates
S D K
Enhanced Blockchain API Server
Chaincode
Smart contract
Ledger
State Database
Blockchain
Peer1
Peer2
Hospital Channel
Blockchain Fabric Host
Blockchain Hospital Network

Figure 4: *Blockchain-based consent and permission ledger for auditable patient communications*

Operationally, the ETL was performed every 60 seconds, and the end-to-end target was <15 seconds of fast-path events. The bandwidth was not monopolized by campaign traffic through rate limiting during the clinical spikes. PHI disclosures and consent scope were linted on templates, and then the promotion was initiated. Deliverability monitoring tracked hard bounces, spam complaints, and per-carrier throttling as input back into the bandit policy and contact suppression rules. Raw vitals in Aerospike were only hashed device identifiers, and a limited set of clinical signals left the operational domain.

End-to-end security was imposed. API calls were signed with a Jul JWT, short-lived tokens, allow-list IPs, and TLS to the egress proxy to the mutual. Synthetic identifiers, tokenization, were used instead of direct identifiers in the engagement layer, and data-retention rules in the integration layer expunged event payloads following integration. The posture complements the current application of zero-trust, where robust identity, micro-segmentation, and continuous validation are highly practiced [29].

5.4. Experimental Results and Insights

The results are categorized by performance, resilience, security overhead, and engagement. Phase B on 300k events/second, measured 19.8 ms average end-to-end latency, and p99 of 77.1 ms between gateway Aerospike receipt and durable Aerospike write under Phase B. With the 60-second ETL included, the median cumulative time for the frequent path (abnormal signal to entry of first Journey) was 64.8 seconds. In the case of the fast path, median Aerospike write to Journey entry time was 6.4 seconds, and 95th percentile 8.9 seconds. Aerospike was linearly scaled. With 400k events/second, median writer latency was 5.3 ms and p99 16.9 ms; dashboard queries did not exceed 35 ms p99. The cluster was able to maintain 1.1 million reads/minute. Memory remained less than 80%. Starving writes and limiting analytics to ~20% of I/O was provided by the scan scheduler. The reconciliation work was compressed, and the accuracy of the buckets and windows remained within ±0.5%.

Resilience tests showed graceful degradation. A complete availability-zone failure (three zombie data nodes and one zombie gateway) resulted in a brief but dramatic increase to 94 ms p99 write-latency and 2.3 seconds of backpressure; workload normalization occurred in less than 40 seconds. These effects were insignificant in the event of the loss of a Kafka broker. Migrations were throttled to 80 MB/s to avoid double-writing due to the forced eviction of a data node; client retry and idempotent writes mitigated this. Under a gateway restart storm, it automatically auto-sheds direct-write traffic into the buffered path, leaving critical vitals intact and stretching non-critical metrics by up to 90 seconds without causing data loss.

Measurements on security revealed low overhead. The PDP introduced a median write latency of 0.9 ms and read latency of 0.6 ms; the formation of an enforcement error (either withdrawal of consent or breach of the purpose of use) was blocked correctly in all test cases. No lateral movement was reported during gateway penetration tests, and simulated credential theft against FIDO2-bound admin sessions was not possible [20]. These results confirm the feasibility of isolating without materially impacting throughput or tail latency.

The evaluation of engagement effects was considered using three cohorts over four weeks: cardiac post-discharge (n=12,844), chronic hypertension (n=24,106), and COPD (n=9,775): the triggered messages covered medication reminders, escalation messages on out-of-range vitals, and educational nudges. Against the conventional route, the fast-path trips dropped the median ack time after high-risk incidents by 11.1 minutes. The bandit-based channel/time optimization increased 7-day adherence by 6.8% (hypertension), 5.1% (COPD), and 3.2% (cardiac discharges) with no increase in opt-out rates. Individualized send windows resulting in switching 31% of high-severity messages out of traditionally low-acknowledgment hours helped drive the realized gains.

***Table 4**: Key Experimental Results and Insights*

Aspect	**Scenario / Metric**	**Result**	**Takeaway**
Performance @300k eps	End-to-end & path timings	Avg 19.8 ms; p99 77.1 ms (gateway→durable write). Frequent path median 64.8 s (incl. 60 s ETL). Fast path median 6.4 s; p95 8.9 s.	Meets sub-second ingest; rapid engagement path.
Performance @400k eps	Write/reads, queries, accuracy	Writer median 5.3 ms; p99 16.9 ms. Reads 1.1M/min; dashboard p99 ≤35 ms; memory <80%; analytics ≤20% I/O; bucket/window accuracy ±0.5%.	Linear scale sustained with low latency.

Aspect	Scenario / Metric	Result	Takeaway
Resilience	AZ failure, broker loss, node eviction, gateway restarts	AZ outage: p99 write 94 ms, 2.3 s backpressure; recovery <40 s. Broker loss negligible. Migrations throttled 80 MB/s. Auto-shed to buffered path; non-critical lag ≤90 s; no data loss.	Graceful degradation and fast normalization.
Security overhead	PDP + enforcement	+0.9 ms write, +0.6 ms read; all policy blocks correct; no lateral movement; FIDO2-bound admin theft not possible.	Zero-trust feasible with minimal latency cost.
Engagement outcomes	3 cohorts over 4 weeks	Median ack time −11.1 min (fast path). 7-day adherence: +6.8% (HTN), +5.1% (COPD), +3.2% (cardiac). 31% high-severity msgs shifted from low-ack hours; no opt-out increase.	Faster responses and better adherence.
Engineering lessons	Architecture choices	Hybrid-memory on NVMe needs drive-health checks; active-active ingestion simplifies recovery; predicate scans fine, long-horizon moved to analytical stores.	Practical ops guidance.
Limits & outcome	Workload/model scope	Synthetic scenarios didn’t test regional cascades; queue smoothing added delay variance; models limited to short windows. Overall: sub-100 ms ingestion, <10 s high-severity engagement, reliable under stress.	Proven feasibility with noted gaps.

Three engineering lessons learnt. Hybrid minimized main-memory footprint by meeting latency requirements on hybrid-memory systems with NVMe, but required active drive-health checks to guard against write-amplification volatility. Active-Active ingestion eases resilience: in the face of a stress, non-critical paths were switched to the buffered path, and critical vital paths were retained as direct writes, providing predictable recovery. Predicate-filtered scans worked well in cohort detection, but long-horizon workloads were better served by exporting summaries to analytical stores to prevent cold partition hotting.

Weaknesses are in the inability of the synthetic workload to probe incident cascades like regional connectivity outages or vendor-wide firmware bugs. Burstiness in a high-severity campaign was restricted by limiting the journey rate, but queue-based smoothing enforced compliance at the cost of some delay variation. Analytics were also limited to narrow windows; models that made long-term predictive inferences exceeded the scope. Taken together, the experiments demonstrate how a low-latency operational store such as Aerospike, coupled with disciplined zero-trust security practices and a bounded integration to Salesforce Marketing Cloud, can deliver sub-100-ms ingestion time, less than 10-sec high-severity engagement, and graceful performance during adverse events, resulting in better patient support time in routine and surge states reliably, safely, and efficiently.

6. Discussion

6.1. Results Analysis

According to the experimental assessment, the implemented architecture could support heterogeneous IoT workloads and maintain a low end-to-end latency for a decision. Wearable, bedside, and remote hub telemetry were normalized into device-agnostic records, stored in an Aerospike namespace, and transmitted to engagement services as a near-real-time event [25]. The backlogs and stream read latencies across all the stress phases were bounded, the read latency distribution was stationary and stable as read arrival burstiness increased, and analytics downstream served the features and risk score without starvation.

Incremental watermark-aware tumbling and sliding aggregations were performed on feature windows, eliminating recomputation storms and allowing serving-layer freshness to remain in operationally acceptable SLOs.

A second noticeable result is imperfection in the data. Arrivals that were out of order and intermittent device connectivity, and transmission duplicates did not significantly affect analytics materially, as preprocessing executed schema alignment, idempotent upserts keyed by device identity and event time, and reconciliation of late arrivals via watermark logic. This stabilized the feature store that was feeding into the risk scores and anomaly detector, which enhanced the consistency in patient stratification even across the different care sites. Replay was also simplified using the deterministic enrichment path; this allowed accurate reconstruction of the state of the code or model after they have been updated [18].

There were operational benefits to the hospital management and clinical staff. Alerts came with a small context, including recent vitals, device quality indicators, and the last clinician action, to enable nurses to triage without flipping between systems, thereby minimizing the friction of handoff and alarm fatigue. The use of beds, health of equipment, and movement of patients were now transparent with constantly refreshed totals and allowed dynamic planning of staffing, care, and maintenance planning of equipment, and control of admission. In Salesforce Marketing Cloud, the journeys were set up as controlled communication channels that minimally utilized patient data and consent flags to automate medication refills, Post-discharge follow-ups, and escalation messages to care teams.

Methodologically, results support an existing tendency that predictive and operational analytics are most valuable when integrated into near-real-time workflows as opposed to standalone dashboards. Immediately after data arrival, the event-driven architecture enabled models to adjust the escalation criteria and trigger outreach, and enrich EHR notes. Connecting release automation and monitoring with analytics outputs enabled faster feedback loops, higher reliability, and the ability to measure the utility of models within the delivery pipeline, aligning with Business Results reports about the convergence of analytics and DevOps.

6.2. Security Implications and Compliance

Security analysis reveals that the security of confidentiality, integrity, and availability can be maintained without compromising performance. Mutual TLS secured data in motion between gateways, processors, Aerospike nodes, and platform APIs, centrally-rotated encryption keys encrypted records and queued messages at rest via a hardware-backed key service. The control of access to stream reading or publishing was restricted by role-based access control, and the scoped tokens of the API reduce the risk of lateral movement [23]. Ingestion, storage, activation, and deletion of records were tracked in audit trails of personally identifiable and health information lineage.

The cluster design of Aerospike helped it to mitigate risk in that predictable latency was guaranteed in the face of replication and failover. Strong consistency and write-ahead policies were configured in namespaces containing sensitive records to support clinical paths and minimize stale reads in point-of-care screens. In contrast, read-only aggregates used availability-oriented settings to handle bursts. Routine logic that followed the residency policies also limited cross-datacenter replication to avoid unintentional egress. The Salesforce integration also took a minimal-data approach on the engagement side: patient identifiers and consent scopes were tokenized, as were event summaries instead of raw-vitals data, with retention and auto-suppression polices based on the change of consent. In cases where asynchronous writes could not be prevented, the replay side effects were restricted by the idempotency of sent message keys and exactly-once behavior at the consumer boundary.

The posture of compliance that the system was able to show is that of the controls-based approach with the foundation in encryption, least privilege access, auditability, data minimization, and incident preparedness. The conformance with regulations ultimately lies in governance and appropriate agreements. Still, the technical patterns are patterns in line with established approaches of real-time NoSQL platforms of sensitive data-considered partitioning to prevent hotspotting, replication, and journaling to sustain durability despite node faults, and explicit consistency options between the read and write paths, principles reflected in efforts on scaling operational databases to real-time workloads [10].

6.3. Limitations of the Study

Several limitations curtail generalizability. Workloads consisted of illustrative but artificial combinations of telemetry-vitals, device diagnostics, and patient-reported events. Messier semantics - firmware quirks, outlier edge cases, and clinician processes that are inconsistent or variable- pose a challenge to model reliability in real deployments. The assessment was done in a controlled network with predictable jitter. An event could be affected by loss and delay when wide-area cellular connections and home Wi-Fi were used, which could lengthen detection windows and disrupt event-time ordering

[37]. Retransmission under a weak signal was minimized through edge caching.

Stream processing and short-horizon analytics were also highlighted in the study. Longitudinal analyses, cohort discovery, and retrospective safety studies demand persistent archives that are accessible to query, and privacy-preserving links to clinical records. Unless governed with great care, following high-velocity IoT data with EHR extracts threatens re-identification with both categorical and attribute triangulation, notably when jointly fusing location traces, device fingerprints, and uncommon clinical attributes. Features specific to vendors were abstracted intentionally to maintain portability; this may obscure vendor-specific enhancements like storage-engine hints, record-bin compression, or journey-level rate controls, which would further improve latency and reduce cost.

There are also scalability issues in the case of burst synchrony and skew key distributions. Synchronized emissions due to popular device models by a public-health event or value-added firmware update may generate partition hotspots and temporarily overwhelm stream processors. Such capacity planning must incorporate auto-partition rebalancing, the use of circuit breakers to jettison noncritical loads, and the use of backpressure throughout the pipeline to avoid cascading failure. Chaos tests, such as node loss, network partitioning, and saturated disks, are required to verify recovery envelopes and to toughen runbooks.

Maintenance of compliance is a socio-technical undertaking. Right to data subject, retention schedules, and purpose limitation necessitate a constant synchronization of both the laws and the technical applications of enforcement [11]. At the corners, revoked consent in the middle of an active episode or during an emergency override, children becoming adults, or cross-border data flow all need to be codified into the policy and checked via orchestration code. In the current work, foundational guardrails were in place: such special situations were not exhaustively tested, and future verification is possible in conditions closer to real-life multi-jurisdictional requirements.

The discussion also conveys that safe, real-time data interchange has the potential to create better patient safety, personnel effectiveness, and operational efficiency, combined with a deferential engineering and governance. Validated in a wide range of settings and environments, data-minimization tactics have been honed further. Closer integration with longitudinal analytics and clinical systems, as well as reminders that both speed and personalization cannot exceed safety and equity, will be needed to produce durable health-system scale impacts.

7. Future Work

7.1. Expanding Real-Time Data Processing Capabilities

Once Aerospike has been fully embedded as the real-time backbone, future work should be to optimize latency and resiliency under hospital-scale traffic. The ingestion edge should include adaptive backpressure and priority queues to ensure that life-saving measurements preempt routine telemetry via MQTT and HTTP gateways. Patient safety records should use strongly consistent namespaces, but other high-volume device logs can use eventual consistency. Rack-aware sharding across availability zones and active-active regions, together with write-aware client policies, will minimize tail latency. Hot keys, short-lived aggregates, and write patterns with no conflicts and prudent TTLs can mitigate contention.

Storage should also be mined. Large write block sizes, compression, and defragmentation windows on clinical off-peaks must be used with hybrid-memory deployments that maintain indexes in DRAM and data on NVMe. TTL policies, post-aggregation, can impose a raw sample’s expiration period, but retention occurs for derived features and sentinel events, limiting storage growth at the cost of clinical value. In the area of cross-site scale, replication of only bare patient context and de-identified measurements, when possible, is applicable, and conflicts should be resolved using device timestamps, record generation, or provenance rules. The target visibility must rise to end-to-end SLOs, p99 read latency of safety-critical keys, and per-tenant event lag. The architecture must be event-driven: change data capture between Aerospike and a streaming bus should produce immutable, schema-versioned events with consent scopes, and fault-tolerant patterns, such as circuit breakers, bulkheads, sagas, and transaction outboxes, should make pipelines resilient to partial failures [6].

7.2. Broader Integration with Other Healthcare Technologies

Interoperability needs to be enhanced such that IoT signals are trusted to generate clinical action It is advisable to follow a FHIR-first approach, where Aerospike records are mapped to FHIR resources, such as Device, Observation, Patient, Encounter, CarePlan, and Consent and then published via a mediation layer that supports RESTful APIs and FHIR Subscription; and, enforces consent and purpose-of-use at API gateway. This level ingests EHRs as documentation and clinical decision support services that evaluate rules in real time, and the Salesforce Marketing Cloud subscribes to an engagement view that is curated on minimally necessary fields or de-identified fields.

Identity resolution will mix deterministic identifiers and probabilistic features to solve multi-device patients without compromising privacy, as the linkage decision can be audited. As highlighted in the figure below, a FHIR-first pattern of integration is used to connect Health 4.0 IoT signals to clinical systems. Aerospike records are mapped to the FHIR Device, Observation, Patient, Encounter, CarePlan, and Consent resources, and published through a mediation layer exposing RESTful APIs and FHIR Subscriptions, enforced with the purpose-of-use and consent by the API gateway. EHRs ingest documentation, CDS services evaluate rules in real time, and Salesforce Marketing Cloud subscribes to a minimally necessary, de-identified engagement view. Identity resolution combines deterministic and probabilistic matches, and a possibility of auditing the linkage.

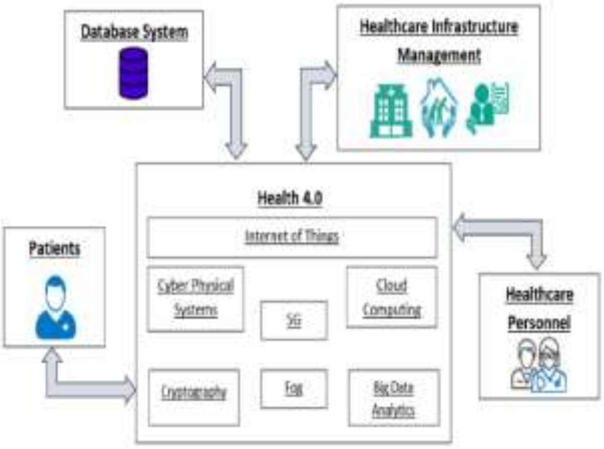


Figure 5: *FHIR-first integration: IoT to EHR, CDS, and consent-aware APIs*

Telemedicine workflows can and should be enhanced by threading videoconference metadata, device provenance, and network health through the same data pipeline as physiologic data. Scale-learned lessons such as sparse payload transmission, edge-only batching during connectivity lapses, and heartbeat-stitched remote monitoring of the device will also apply to remote patient monitoring. They should guide its gateway and protocol architecture [26]. Regarding EHR integration, Subscription activation can be used to trigger outreach based on encounters, such as a discharge triggering a home-monitoring task and a medication-adherence journey, and CDS Hooks can be integrated into clinician workflows by presenting risk scores. Each way updates should push results like appointment attendance and self-reported symptoms back to Aerospike to complete the hot-loop in driving analytics and model retraining. Data minimization and consent-aware routing should be applied across all flows to send only the minimum amount of data to Salesforce and reflect consents concerning revocation downstream at caches and queues.

7.3. Enhancing Predictive Analytics and Machine Learning

The second wave is expected to consider Aerospike as both a low-latency feature store and an online state manager. Stream transforms, windowed aggregates, exponentially weighted moving averages, frequency sketches, and rolling baselines are calculated in stream processors and stored as keyed features to be accessed in milliseconds at inference. The pipeline used should detect drift by monitoring population stability index and change-point tests, and employ guarded rollouts and auto roll-back in the event of a clinical distribution shift. In order to mitigate alarm fatigue, anomaly detectors must integrate device trends alongside clinical context, including recent procedures, medications, and mobility, and report calibrated risk scores to human operators with feature attributions indicating their meaning.

Privacy-preserving machine learning is a topic that merits serious investment. Federated learning between sites and devices can retain raw data locally but still share encrypted updates to models; secure aggregation and differential privacy will limit the risk of disclosure in features. Triage with on-device inference, as provided by the verification of arrhythmia, fall detection, or insulin dosage, lowers bandwidth. Multi-armed bandits and bounded reinforcement learning may be used to optimize outreach cadence and channel, to achieve clinical rules, consents, and fairness constraints and equity measures. A streaming feature registry must track provenance to ensure marketing paths do not consume features that have been explicitly harvested to serve clinical care.

MLOps is required to operationalize this stack. Definitions of the features should be version-controlled like code; the training and serving schemas should be locked out against skew; the canaries must measure calibration, time-to-alert, clinician workload, and impact. Aerospike can provide feature-freshness service levels by expiring stale windows with its TTL semantics, whereas snapshot exports can backfill training sets without interrupting client workloads. A lineage catalog of firmware, preprocessor code, features, and model artifacts will speed incident response [15]. Governance must also represent model cards, data protection impact assessments, and post-deployment monitoring to ensure safety and regulatory alignment.

8. Conclusions

This paper shows that secure, real-time sharing of healthcare IoT data is viable with a disentangled yet synchronized data plane and engagement plane. Aerospike caches low-latency operational state to hot telemetry, alert eligibility, and model features, which are converted into consent-aware, multi-channel communication on Salesforce Marketing Cloud (SFMC). The pipeline ingests MQTT/HTTPS device events, normalizes units and schema, implements per-record TTLs, secondary indexes, and ingests as idempotent upserts through to analytics—compact and roll-aggregated records of alert_state support deterministic suppression and bedside views. A mediation layer authenticates consent and feeds attributes with de-identified attributes into SFMC, which Journey Builder uses to correlate outreach. Anonymity is anonymized, mutual TLS secures communications, and DevSecOps guardrails enforce policies, keeping privacy and safety intact with low latency.

In practice, the platform was able to support hospital-specific loads, with predictable tails. At the maximum rate of 310000 events/s, the median write latency was 3.8 ms (p99 12.2 ms); at 400000 events/s, it was 5.3 ms (p99 16.9 ms). P99 of the 15-minute dashboard reads remained less than 35 ms. The end-to-end activation experience with a common path consisting of 60-second ETL was up to expectation (a 64.8-second median and 6.4-second median p95 8.9s for the fast path). Resilience maintained a loss of availability zone, increased p99 writes to 94 ms with 2.3 seconds of backpressure, but workloads normalized after 40 seconds; throttled migrations could absorb a broker loss and node eviction, retries, and idempotent writes. Security enforcement introduced sub-millisecond performance overhead, and the tests could not detect any lateral movement. Accuracy and safety exist side by side with lean targets.

Critical engagement has closed-loop enforcement of value created in clinical work. Contextual bandits were used to optimize channel distribution and send timing, which led to a 6.8% improvement in seven-day adherence among hypertension, 5.1% among COPD, and 3.2% among cardiac discharges, as well as a much faster acknowledgment time in 11.1 minutes following high-risk incidents. Results and delivery telemetry reports were sent back to Aerospike to recalibrate the thresholds and outreach activities. Out-of-order arrival was dealt with via Watermark-aware aggregation; predicate-based filtering of scans was throttled; writing on an in-progress active-active ingest was actively protected; and scan scheduling guarded headroom due to the number of writes ahead of bedside reads. Quality gates, such as schema validation, unit normalization, deduplication, and drift detection, regularized features even via the lossy networks, and tokenization minimized SFMC to the bare necessities.

The bundled stack provides patient safety and operational effectiveness simultaneously. Peri-minute activation window reads and even sub-second reads when looking at the most recent state can help identify deterioration earlier, limit guardrails tightly, and escalate to rapid response teams promptly. Context-specific alerts with current vitals, device quality, and last clinician action eliminate alarm fatigue, accelerating triage. Operationally, integration of single views of posture of devices, bed turnover, environmental sensors, and patient flow can enable capacity, staffing, maintenance, and control of admission. Almost real-time dashboards and anomaly detectors can be used to unblock or help ration resources and coordinate the efforts, and consent-aware outreach promotes adherence and follow-up. Clinical and engagement signals are time-aligned, and incident reviews recreate what was known when, improving safety culture.

The decoupling of concerns is definitive in the case of hospital IT. Aerospike offers the persistent, queryable latest-known-good state; the streaming bus supports ordering, replay, and backpressure; and SFMC enables engagement on de-identified attributes. This allows each plane to scale individually, support specific SLOs, and develop consent and messaging without rewriting across the systems. The Cost governance is well served by throttled scan, cohort predicates, and TTL-based retention of expiring ephemeral samples, maintaining the derived features and sentinel events. Confidentiality is built in mutual TLS, TLS, short-lived credentials, least-privilege identities, and audit logs. The outcome is a monitored, testable platform that integrates DevSecOps into regulation without compromising clinical latency and availability. Strategically, this architecture will place providers in value-based care. Telemetry-controlled engagement loops enable organizations to learn, test, track, and grow programs. The next step of addressing long-horizon analytics and cross-jurisdictional situations will allow the advantages of episodic interventions to be applied over a sustained period to population health management.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have

appeared to influence the work reported in this paper

- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.

- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

[1] Almagrabi, A. O., & Al-Otaibi, Y. D. (2020). A survey of context-aware messaging-addressing for sustainable internet of things (IoT). sustainability, 12(10), 4105.

[2] S. K. Gunda, "Machine Learning Approaches for Software Fault Diagnosis: Evaluating Decision Tree and KNN Models," 2024 Global Conference on Communications and Information Technologies (GCCIT), BANGALORE, India, 2024, pp. 1-5, https://doi.org/10.1109/GCCIT63234.2024.10861953

[3] Bock, M. (2023). Measuring adoption of phishing-resistant authentication methods on the web (Master's thesis).

[4] S. K. Gunda, "Fault Prediction Unveiled: Analyzing the Effectiveness of Random Forest, Logistic Regression, and KNeighbors," 2024 2nd International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS), Erode, India, 2024, pp. 107-113, https://doi.org/10.1109/ICSSAS64001.2024.10760620

[5] S. K. Gunda, "Analyzing Machine Learning Techniques for Software Defect Prediction: A Comprehensive Performance Comparison," 2024 Asian Conference on Intelligent Technologies (ACOIT), KOLAR, India, 2024, pp. 1-5, https://doi.org/10.1109/ACOIT62457.2024.10939610

[6] Chavan, A. (2024). Fault-tolerant event-driven systems: Techniques and best practices. Journal of Engineering and Applied Sciences Technology, 6, E167. http://doi.org/10.47363/JEAST/2024(6)E167

[7] Chavan, A., & Romanov, Y. (2023). Managing scalability and cost in microservices architecture: Balancing infinite scalability with financial constraints. Journal of Artificial Intelligence & Cloud Computing, 5, E102. https://doi.org/10.47363/JMHC/2023(5)E102

[8] S. K. Gunda, "Enhancing Software Fault Prediction with Machine Learning: A Comparative Study on the PC1 Dataset," 2024 Global Conference on Communications and Information Technologies (GCCIT), BANGALORE, India, 2024, pp. 1-4, https://doi.org/10.1109/GCCIT63234.2024.10862351

[9] Dhanagari, M. R. (2024). MongoDB and data consistency: Bridging the gap between performance and reliability. Journal of Computer Science and Technology Studies, 6(2), 183-198. https://doi.org/10.32996/jcsts.2024.6.2.21

[10] Dhanagari, M. R. (2024). Scaling with MongoDB: Solutions for handling big data in real-time. Journal of Computer Science and Technology Studies, 6(5), 246-264. https://doi.org/10.32996/jcsts.2024.6.5.20

[11] Fong, S. (2024). The Legal Implications of Data Retention for Law Enforcement Purposes in the European Union: Do we need a new Data Retention Framework? (Master's thesis, Universidade NOVA de Lisboa (Portugal)).

[12] Goel, G., & Bhramhabhatt, R. (2024). Dual sourcing strategies. International Journal of Science and Research Archive, 13(2), 2155. https://doi.org/10.30574/ijsra.2024.13.2.2155

[13] Goodwin, A. J. (2023). High Frequency Physiological Data Quality Modelling in the Intensive Care Unit (Doctoral dissertation).

[14] Heiskari, J. J. (2022). Computing paradigms for research: cloud vs. edge.

[15] Humbel, L. (2022). Modern hardware abstractions for firmware (Doctoral dissertation, ETH Zurich).

[16] Karwa, K. (2023). AI-powered career coaching: Evaluating feedback tools for design students. Indian Journal of Economics & Business. https://www.ashwinanokha.com/ijeb-v22-4-2023.php

[17] Karwa, K. (2024). The role of AI in enhancing career advising and professional development in design education: Exploring AI-driven tools and platforms that personalize career advice for students in industrial and product design. International Journal of Advanced Research in Engineering, Science, and Management. https://www.ijaresm.com/uploaded_files/document_file/Kushal_KarwadmKk.pdf

[18] Kirchhof, J. C., Malcher, L., & Rumpe, B. (2021, October). Understanding and improving model-driven IoT systems through accompanying digital twins. In Proceedings of the 20th ACM SIGPLAN International Conference on Generative Programming: Concepts and Experiences (pp. 197-209).

[19] S. K. Gunda, "Comparative Analysis of Machine Learning Models for Software Defect Prediction," 2024 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS), Chennai, India, 2024, pp. 1-6, https://doi.org/10.1109/ICPECTS62210.2024.10780167

[20] Lachkov, P., Tawalbeh, L. A., & Bhatt, S. (2022). Vulnerability assessment for applications security through penetration simulation and testing. Journal of Web Engineering, 21(7), 2187-2208.

[21] Lakshminarayana, S., Praseed, A., & Thilagam, P. S. (2024). Securing the IoT application layer from an MQTT protocol perspective: Challenges and research

prospects. IEEE Communications Surveys & Tutorials, 26(4), 2510-2546.

[22] Wyciślak, S. (2024). Real-time visibility as a catalyst for operational enhancements. Logforum, 20(2).

[23] Lasch, R. (2024). Heterogeneous memory technologies in database management systems (Doctoral dissertation, Dissertation, Ilmenau, TU Ilmenau, 2024).

[24] Le, K. T. (2023). Building a Blockchain-based API Access Control System (Doctoral dissertation, University of Saskatchewan).

[25] Malik, G., & Prashasti. (2023). Blockchain security: Security challenges and solutions for decentralized systems and cryptocurrencies. International Journal of Science and Research Archive, 9(2), 1074–1100. https://doi.org/10.30574/ijsra.2023.9.2.0515

[26] Nyati, S. (2018). Revolutionizing LTL carrier operations: A comprehensive analysis of an algorithm-driven pickup and delivery dispatching solution. International Journal of Science and Research (IJSR), 7(2), 1659-1666. Retrieved from https://www.ijsr.net/getabstract.php?paperid=SR24203183637

[27] S. K. Gunda, "A Deep Dive into Software Fault Prediction: Evaluating CNN and RNN Models," 2024 International Conference on Electronic Systems and Intelligent Computing (ICESIC), Chennai, India, 2024, pp. 224-228, https://doi.org/10.1109/ICESIC61777.2024.10846549

[28] S. K. Gunda, "Software Defect Prediction Using Advanced Ensemble Techniques: A Focus on Boosting and Voting Method," 2024 International Conference on Electronic Systems and Intelligent Computing (ICESIC), Chennai, India, 2024, pp. 157-161, https://doi.org/10.1109/ICESIC61777.2024.10846550

[29] Rhoads, J., & Smith, A. (2024). Effectiveness of Continuous Verification and Micro-Segmentation in Enhancing Cybersecurity through Zero Trust Architecture.

[30] Sardana, J. (2022). Scalable systems for healthcare communication: A design perspective. International Journal of Science and Research Archive. https://doi.org/10.30574/ijsra.2022.7.2.0253

[31] Sharma, S. (2024). Mastering IoT for Industrial Environments: Unlock the IoT Landscape for Industrial Environments with Industry 4.0, Covering Architecture, Protocols like MQTT, and Advancements with ESP-IDF. Orange Education Pvt Limited.

[32] Singh, V. (2022). Explainable AI in healthcare diagnostics: Making AI models more transparent to gain trust in medical decision-making processes. International Journal of Research in Information Technology and Computing, 4(2). https://romanpub.com/ijaetv4-2-2022.php

[33] Singh, V. (2023). Enhancing object detection with self-supervised learning: Improving object detection algorithms using unlabeled data through self-supervised techniques. International Journal of Advanced Engineering and Technology. https://romanpub.com/resources/Vol%205%20%2C%20No%201%20-%2023.pdf

[34] Strandquist, G. (2024). Where is the Person in Personalized Medicine? The Missing Expert in Adaptive Neurotechnology (Doctoral dissertation, University of Washington).

[35] Sultanum, N. B. (2022). Text-Centric Visual Approaches to Support Clinical Overview of Medical Text (Doctoral dissertation, University of Toronto (Canada)).

[36] Takyi, H. K. (2019). Security, privacy, confidentiality and integrity of emerging healthcare technologies: A framework for quality of life technologies to be HIPAA/HITECH compliant, with emphasis on health kiosk design (Doctoral dissertation, University of Pittsburgh).

[37] Vattikonda, B., Das, M., Bhardwaj, T., Panja, S., Arora, P., Gupta, A., & Aswal, D. K. (2020). Time and Frequency Metrology: Part II: Application of Indian Standard Time for Safe Digital India. In Metrology for Inclusive Growth of India (pp. 197-236). Singapore: Springer Singapore.

[38] S. K. Gunda, “Accelerating Scientific Discovery With Machine Learning and HPC-Based Simulations,” in Integrating Machine Learning Into HPC-Based Simulations and Analytics, B. Ben Youssef and M. Ben Ismail, Eds. Hershey, PA, USA: IGI Global, 2025, pp. 229–252, doi: 10.4018/978-1-6684-3795-7.ch009

[39] Volminger, A. (2021). A comparison of Data Stores for the Online Feature Store Component: A comparison between NDB and Aerospike.

[40] Wyciślak, S. (2024). Real-time visibility as a catalyst for operational enhancements. Logforum, 20(2).