SiteScribe: An Evidence-Centric, Multi-Tenant Change Order Intelligence Platform for Construction Projects

Muhammed Murat Yanaşoğlu[1,2]*
Sergazy Nurbavliyev[3]

[1]Department of Artificial Intelligence Engineering, AI and Informatics Faculty, Near East University, 99138 Nicosia, Cyprus

[2]Mered Technology, Nicosia, Cyprus

[3]University of Utah, Department of Mathematics, 155 S 1400 E, Salt Lake City, UT 84112, USA

*Corresponding Author: Muhammed Murat Yanaşoğlu. Email: eng.muratyanasoglu@gmail.com ORCID: 0009-0007-4722-8079

Contributor: Sergazy Nurbavliyev. Email: drsergazy@gmail.com ORCID: 0009-0001-4901-5620

ABSTRACT: Site logs, request for information (RFI) documents, plan revisions, contracts, and photographs taken in the field constitute large volumes of heterogeneous evidence associated with the construction of buildings, roads, and highways, among other projects. In the construction industry, construction evidence artifacts can be located across multiple, disconnected channels, which makes capturing, retracing, and auditing Change Orders (COs) time consuming and difficult. In this paper we describe **SiteScribe**, a collaborative, evidence-focused, multi-tenant platform with the ability to consolidate evidence collection, signal identification, event triaging, and CO construction automation into a single process. **SiteScribe** uniquely features role-based and project-specific access control, evidence identification, segregation, and optional AI-based semantic enhancement for evidence identified and packaged for export, with secured evidence export by means of PDF/ZIP combined with a SHA-256 manifest for evidence-based COs. **SiteScribe** promotes stakeholder communication and project governance by means of comments, notifications, webhooks, approvals, and evidence COs sustained within a project's historical record. **SiteScribe**'s architecture employs a modular design for each functional area (interface, domain logic, persistence, and external systems) to facilitate secure, scalable, and maintainable operation across local and cloud-sourced systems. The proposed solution addresses the primary industry challenge of converting fragmented field evidence into defensible contractual outputs, thereby enhancing transparency and minimizing operational friction. This work contextualizes **SiteScribe** as a construction digital evidence change management solution, as well as a basis for further studies on decision latency, quality of traceable evidence, and workflow reliability.

KEYWORDS: Construction Informatics; Change Order Management; Evidence Traceability; Multi-Tenant Systems; Role-Based Access Control; Document Intelligence; AI-Assisted Workflow; Retrieval-Augmented Systems; Export Integrity; Construction Technology.

---

1. Introducing a Problem

1.1 Construction Services' Background and Industry Context

The construction delivery environment is document intense and reliant on various time-critical decisions. Cognitive load is produced whilst executing construction processes and recording site diary entries, RFIs, revision drawings, contracts, and photos. However, these documents are processed through different media channels, i.e., emails, shared drives, messaging services, ad hoc reporting, and more. Because of this, documents are processed through different media channels, i.e. emails, shared drives, messaging services, ad hoc reporting, and more. There is a continued gap between the availability of evidence, and the readiness to make decisions based on the evidence, mainly when it concerns change orders (COs) [53], [54], [58].

The documentation and communication gap signals cost increases, time delays, and managerial burden growth related to change order management [54], [55], [56], [57]. Due to this gap, construction services lose time and money. In the eyes of management, this causes documentation delays and reduces the quality of disputed documents. Construction services explain delayed responsive documentation by the need to consolidate evidence prior to obtaining a reasonable and defensible change order narrative.

1.2 Defining the Problem

There are modern document systems and advanced project management systems. However, the problem remains structural. Evidence is captured, yet without any operational transformation to make it traceable, change orders (CO) are still processed as unstructured. Defining the problem in the existing systems shows:

1. weak evidence-to-claim linkage,
2. drafting quality inconsistency across teams,
3. limited visibility in approval and revision phases and
4. auditing of exported contractual packages [53], [54], [58] is not possible.

From a system's standpoint, this is more than a user interface problem; this is an issue of workflow architecture encompassing data modeling, access governance, retrieval, lifecycle control, integrity-preserving output generation, and various forms of operationalized workflow data.

1.3 Motivation

The vast potential of transformer-based language modeling and retrieval techniques is the most recent source for the operationalization of unstructured evidence at scale [1], [2], [5], [6], [7]. More specifically, retrieval-augmented techniques and embedding-based semantic search show promise in the enhancement of evidence retrieval and drafting support contextualized within a framework that is integrally unified with the project data [5], [6], [7], [8]. However, these emerging capabilities, in the context of construction contracts, a highly accountable domain, must be paired with governance, security, and traceability [13], [14], 19], [20] of a robust nature.

As a result, the principal driving force behind this effort is the design and implementation of a framework in which evidence can transition from raw ingestion to auditable CO exports within a secure, multi-tenant architecture, with an optional AI augmentation rather than an AI dependency.

1.4 Aim and Scope of This Study

This paper presents **SiteScribe**, a construction project operational platform for CO (contractor) intelligence, integrated with evidence-based functionalities. The functionality scope includes:

1. evidence ingestion and normalization,
2. signal detection and event triage,
3. structured CO drafting and lifecycle editing,
4. role-based governance and auditability,
5. export integrity (PDF/ZIP/manifest) packaging, and
6. optional AI retrieval and enrichment.

This study does not address the legal dimensions of contract documents, focusing instead on architect/engineering design. AI is viewed as a contextual and productivity-enhancing tool, while workflow determinism and continuity are maintained regardless of AI availability.

1.5 Research and Engineering Objectives

The primary goal is to operationalize evidence-to-CO translation, reducing the process gap. The secondary goals are to:

1. implement organizational/project scope and role-based access restrictions,
2. improve the evidence-CO output traceability,
3. harmonize CO drafting, artifact, and review process standards,
4. ensure export packages contain integrity metadata of a (cryptographic) integrity and
5. facilitate repetitive publication and dissemination workflows as per archival and dissemination practices [45], [46], [49], [52].

1.6 Summary of Contributions

This research offers the following contributions to knowledge and practice:

1. End to end model of workflows and evidence centric approaches for CO management in construction,\
2. Modular full stack design and architecture integrating domain services with access control and enforcement\n
3. Mixed structured deterministic and optional AI operational paradigm\n
4. Integrity exporting model using SHA 256 metadata with manifests [22], and
5. Technical baseline ready for publication and aligned with the principles of transparency, reproducibility, and open science [45],[49],[50]

1.7 Roadmap of the Paper

This paper is organized as follows. The second section is the Literature Review, Section 3 is the formal statement of the problem, Section 4 is the statement of the requirements and what follows the requirements is organized into major sections to enhance the navigability of the paper. Thus, Section 5 is about System Design and Implementation, Section 6 is about Security, Governance and Dependability, Section 7 is about Evaluation and Discussion, the concluding section is about the Strategy of Reproducibility and reproducibility and archival practice and the paper concludes in Section 8.

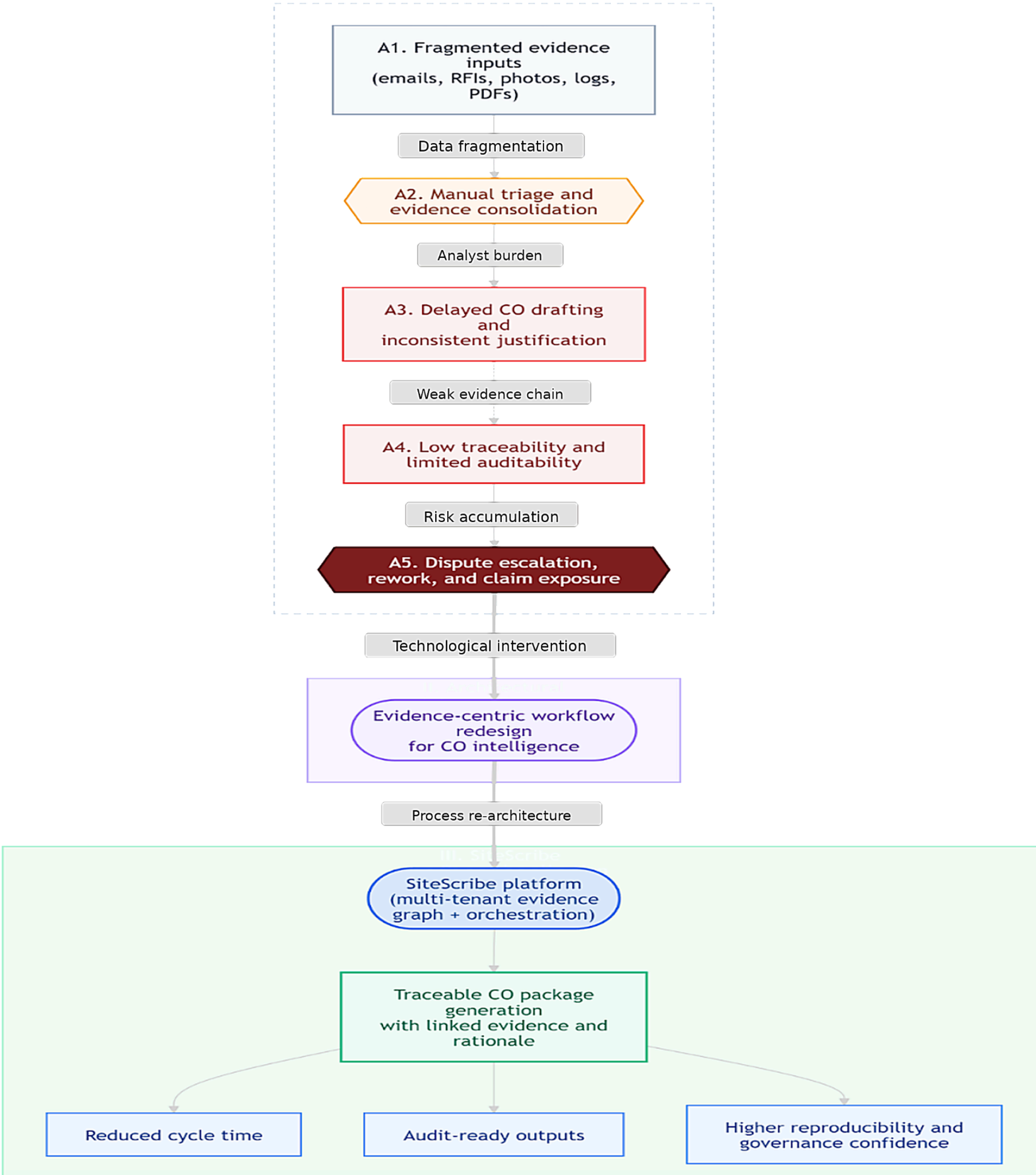

Figure 1. From fragmented evidence handling to evidence-centric CO operationalization.

2. Literature Review

2.1 Current State of Change Order Research in Construction

The adverse impacts of changes on costs and delays in construction projects is well documented. Research on change orders shows that in particular, when scope changes are poorly documented at certain critical points in time, and in poorly defined phases of the decision-making cycle, impactful changes occur [54], [55], [56], [58]. Various studies across different geographies and types of projects show structural gaps in information flow, fragmentation of communication, volatility in changes, and ambiguities in contracts [53], [54], [57], [58].

A repeated theme in the literature is the operational burden changes pose in the absence of change, as well as the absence of a process focused on capturing data that allows field documentation to be transformed into contractual documents that can be contractually transformed into evidence [53], [54], [55]. This supports the further need for systems that are designed to provide a clear linkage to evidence and the structure of the change order generated artifacts.

2.2 Digital Documentation and Evidence Gap in Traceability

Although project teams are using digital applications and tools, data is often left spread out across disconnected, non-integrated data channels. In practice, this creates an Aggregation tax, where teams must manually integrate disparate logs, files, and messages in order to create a coherent CO story.

From a systems viewpoint, this illustrates a missing layer between document repositories and decision artifacts: a workflow engine that can normalize evidence, hold context, and provide traceable outputs. Existing process-heavy approaches provide storage, but not operational synthesis into the current state of the CO lifecycle [53], [54], [58].

2.3 AI and Retrieval Literature Relevant to SiteScribe

The foundation of scalable language representation and reasoning was established by transformer-based modeling [1], [2], [11]. Instruction alignment and contemporary LLM evolution led to operational assistant quality improvements [3], [4].

Retrieval-centric approaches, particularly for evidence-based systems, are of interest. Retrieval-Augmented Generation (RAG) and dense retrieval techniques mitigate the dependence on a reactive response in a purely parametric way while facilitating grounded context injection [5], [6]. Methods for sentence-embedding and vector search provide tools for semantic retrieval at the evidence project corpus [7], [8].

Regarding the contexts of multimodal construction, such as site imagery, the vision-language techniques are applicable for descriptive enhancement and contextual understanding [9], [10] [12].

2.4 Security and Governance Literature for Production Systems

CO workflows have legal and financial repercussions, making governance a necessary design consideration. The OWASP and NIST guidance reference layered defenses for identity management, access control, input and output data validation, anti-abuse mechanisms and secure configuration controls [13], [14], [15], [19], [20], [21] [16].

In terms of protocol-level interoperability and secure integration, the behavioral standards established for HTTP semantics, JWT, HMAC, JSON, cookies, and TLS [23], [25], [26], [27], [29], [32] apply directly to the use of webhook signatures, authenticated sessions, and secure export flows of systems such as **SiteScribe**.

2.5 Reproducibility and Archival Practice

To claim more than a prototype, the software artifacts must be versioned, archived, and made citable. The repository service persistent-identifier workflows allow for Production Systems Identification (PID) and versioned release citations, as well as a citable lineage for the project at the concept level [45], [46], [47], [48]. The FAIR guidance focuses on the long-term accessibility and reusability of the information and the quality of metadata pertaining to the information [49], [50], [51], [52].

It is for this purpose that the current work is organized as not only a system description, but also as a model of an engineering artifact that is publication ready.

2.6 Synthesis and Identified Research Gap

The mentioned comprehensive bodies of work establish strong independent foundations in specific areas of interest: construction change-order impact and process challenges [53], [54], [55], [56], [57], [58], AI/retrieval methods for unstructured context [1], [5], [6], [7], [8], security and governance frameworks for production systems [13], [19], [20], [23], [32], and reproducibility and archival standards [45], [49], [52].

There is however a practical integration gap where very few end-to-end implementations exist that incorporate evidence normalization + event triage + structured CO lifecycle + export integrity + multi-tenant governance + optional AI augmentation into one operational platform. **SiteScribe** aims to fill this gap directly.

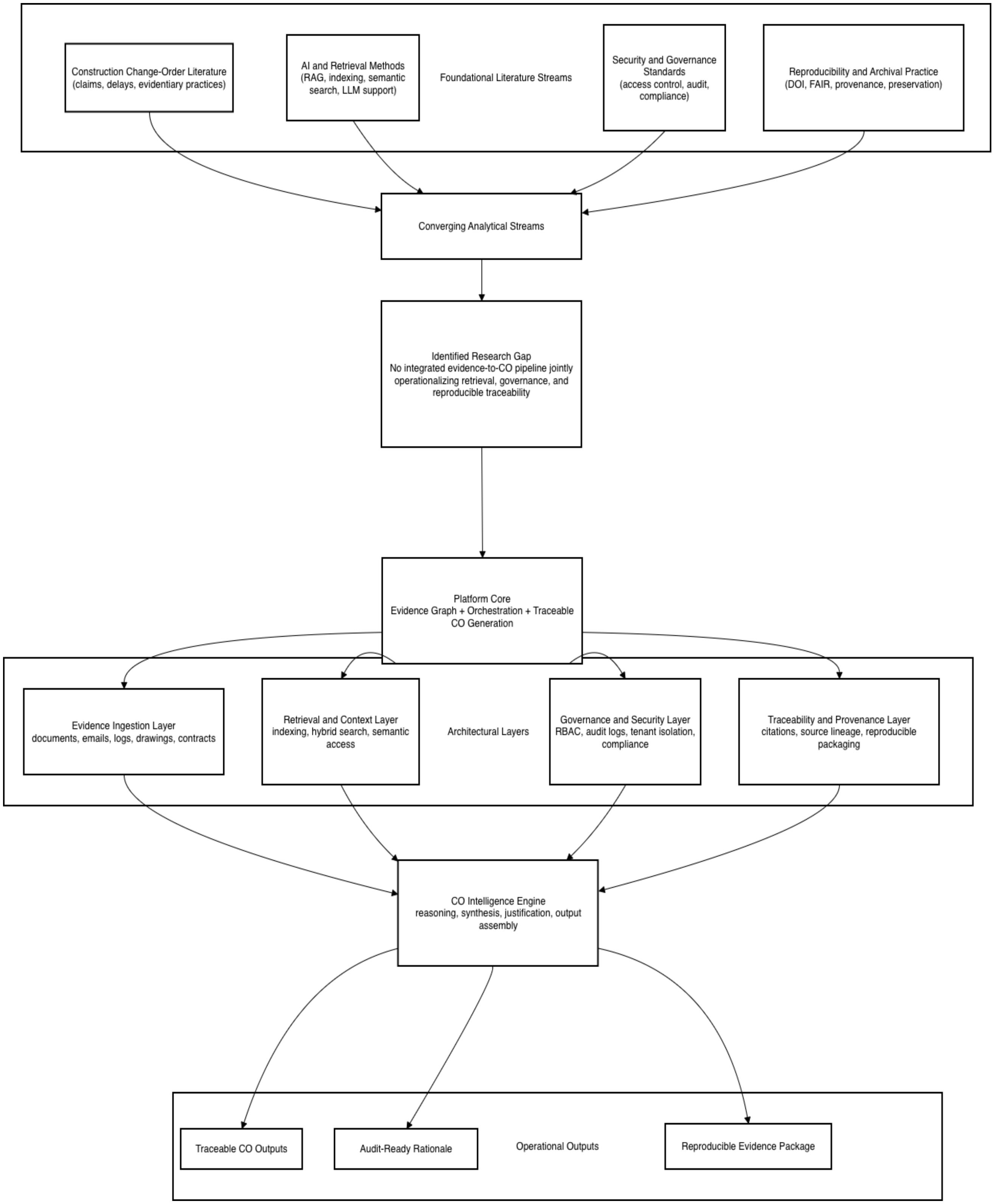

Figure 2. Literature streams converging into the identified research gap and **SiteScribe** positioning

Table 1. Comparative View of Literature Streams and **SiteScribe** Alignment

Literature Stream	**What Prior Work Emphasizes**	**Remaining Gap**	**SiteScribe Alignment**
Construction CO studies [53]-[58]	Causes/impacts of change orders, delays, and cost pressure	Limited implementation-level pipeline for evidence-to-CO traceability	End-to-end evidence-centric operational workflow
AI/LLM/RAG [1]-[12]	Context modeling, retrieval, generation, multimodal enrichment	Often lacks domain governance and contractual workflow constraints	Optional AI augmentation within governed workflow
Security/standards [13]-[32]	Identity, access, protocol and control baselines	Not construction workflow-specific by default	Multi-tenant RBAC, validation, secure integration model
DOI/FAIR ecosystem [45]-[52]	Persistent citation and metadata quality	Often disconnected from software workflow evaluation papers	DOI-oriented dissemination and reproducibility path

3. Problem Formulation

3.1 Operational Setting

Let a construction project produce a time-ordered stream of heterogeneous evidence artifacts:

$$E = \{e_1, e_2, \dots, e_n\}$$

Each evidence item e_i may include:

1. structured metadata (type, timestamp, author, project scope),
2. optional binary payload (PDF/image),
3. optional extracted text representation.

The target output space is a set of change-order artifacts:

$$C = \{c_1, c_2, \dots, c_m\}$$

where each c□ must be:

1. project-scoped and access-governed,
2. evidence-linked and reviewable, and
3. exportable in an integrity-verifiable package.

This formulation reflects the practical gap identified in Sections 1 and 2: evidence exists, but evidence-to-CO transformation is weakly operationalized [53], [54], [58].

3.2 Entity-Level Formal Definitions

For the purposes of this study:

1. Evidence item e_i: canonical project record with optional content and extracted representation.
2. Signal s□: machine- or rule-derived indicator that an evidence item may imply scope/cost/schedule change.
3. Event v□: triage unit grouping one or more signals for decision workflow.
4. Change Order c□: structured contractual artifact derived from event context and evidence support.
5. Export package x_q: generated artifact bundle containing summary document, evidence attachments, and manifest integrity metadata.

Evidence-to-CO provenance relation:

$$P(c_j) \subseteq E$$

meaning every CO should be traceable to a supporting evidence subset.

3.3 Objective Function

The platform objective is to minimize operational friction in evidence-to-CO conversion while preserving governance and integrity:

$$min\ \alpha\, L_{decision} + \beta\, L_{traceability} + \gamma\, R_{governance} + \delta\, R_{integrity}$$

where:

1. L_decision: latency from relevant evidence availability to first usable CO draft,
2. L_traceability: traceability loss (missing/weak evidence linkage),
3. R_governance: risk of unauthorized or unscoped actions,
4. R_integrity: risk of unverifiable export artifact composition.

Weights α, β, γ, δ are context-dependent and organization-specific.

3.4 Constraints

The optimization target above is subject to the following hard constraints:

1. Tenant isolation constraint
   Any mutation or retrieval action must be organization- and project-scoped [13], [19], [20].
2. Role constraint Action a on resource r is allowed only if role u_r satisfies minimum policy threshold [14], [35].
3. Validation constraint
   Inputs must pass type/size/identifier checks before persistence [15], [18].
4. Integrity constraint
   The export package must include deterministic manifest fields and cryptographic digests [22].
5. Availability constraint
   Core workflow must remain operational even when AI features are disabled or unavailable [5], [21].

3.5 System Boundary and Assumptions

In-Scope:

1. evidence ingestion and transformation,
2. signal/event workflow,
3. lifecycle of CO drafting and editing,
4. governed export generation.

Out-of-Scope:

1. legal adjudication of claim validity,
2. automatic contracts interpretation as binding legal advice,

3. enterprise-scale distributed consensus across independent bodies.

Assumptions:

1. users will be authenticated and membership will be resolved,
2. evidence timestamps are available or can be inferred,
3. the semantics of the storage provider are dependable at the level of the object,
4. external delivery endpoints (email/webhook) are reachable when configured.

3.6 Failure Modes and Error Surface

Representative failure classes include:

1. evidence uploads that are malformed,
2. text that is extracted and is missing from PDFs that are of low quality,
3. signal detection that is falsely positive or negative,
4. role limited actions that result in constrained permissions,
5. failure of outbound delivery (email/webhook),
6. optional AI not responding.

The formulation needs to include the possibility of graceful degradation: workflow correctness should not be dependent on the completion of AI or the success of external delivery [21], [23], [32].

3.7 Problem-to-Solution Mapping

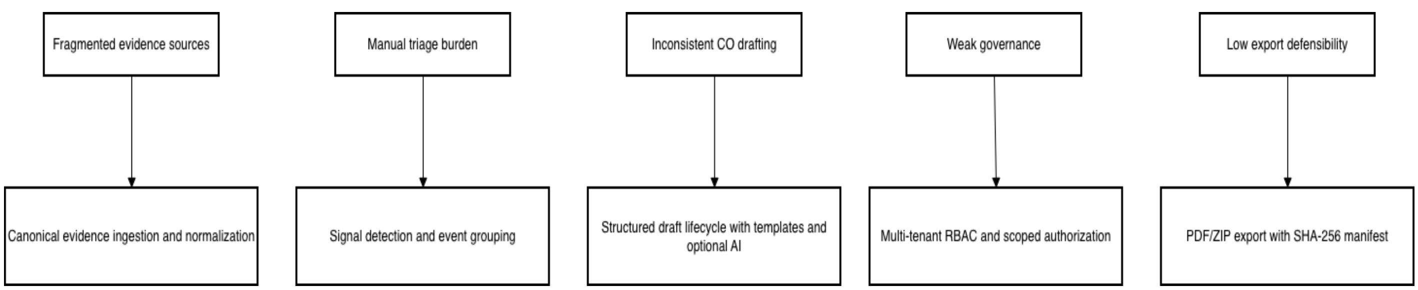

Figure 3. Core problem dimensions and corresponding **SiteScribe** solution mechanisms.

3.8 Measurable Formulation Outputs

To support later methodology and evaluation, the formulation yields the following measurable outputs:

Table 2. Problem-formulation output variables and measurable indicators.

Output Variable	Description	Indicative Metric
$Tdraft$	Time to first CO draft from relevant evidence	median hours
$Qtrace$	Evidence-link completeness in CO content	percentage linked sections

Output Variable	Description	Indicative Metric
$Gauth$	Authorization correctness	blocked unauthorized action rate
$Iexport$	Export integrity consistency	valid manifest verification rate
$Rreview$	Review process efficiency	number of review cycles

3.9 Transition to Methodology

Given this formalization, the methodology section will define:

1. how evidence is transformed into searchable and triage-ready representations,
2. how deterministic and optional-AI decision layers are composed,
3. how lifecycle governance is enforced at server-side boundaries, and
4. how evaluation metrics map to the optimization objective defined above.

4. Requirements Engineering

4.1 Requirements Engineering Approach

The construction constraints identified in change-order workflows [53], [54], [58] combined with the problem formulation in Section 3 has led us to develop an initial requirements model. We adopt a layered approach to the requirements: [42], [43], [44]

1. Domain requirements (construction and CO lifecycle needs evidence)
2. System requirements (capabilities of the functional workflow)
3. Assurance requirements (security, governance, and auditability)
4. Operational requirements (deployability, maintainability, and extensibility) [13], [19], [20], [33], [36] [38], [39], [40], [41]

Each requirement must map to a module family and a validation strategy, and thus, each requirement has been designed with a strong emphasis on traceability.

4.2 Stakeholder-Centered Requirement Sources

Requirements are sourced from the following stakeholder roles:

1. Project Manager (PM): needs rapid triage, draft consistency, and approval visibility.
2. Field/Documentation roles: need evidence ingestion and retrieval with minimal friction.
3. Organization Owner/Admin: needs to have governance, integrations, and audits they can trust.
4. Review/Commercial: need exports that can be defended and links to supporting evidence.
5. Technical operators: require deployment security and behavior cycle.

Actor needs, along with the identified domain, corresponds with the change-order literature and the governance practices of production software. [54], [56], [58], [14], [21].

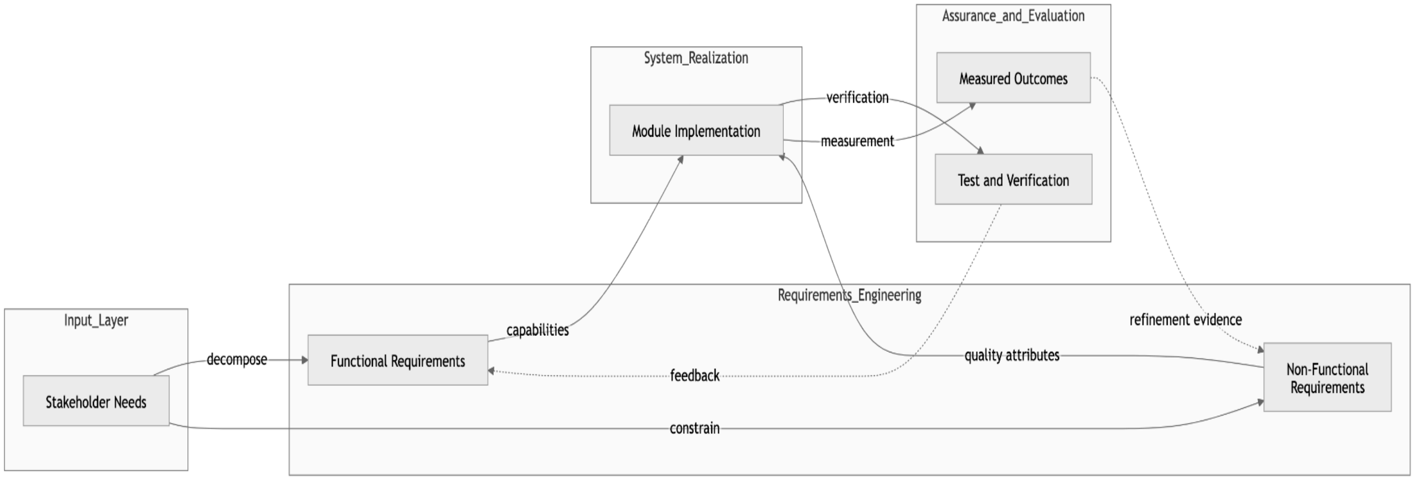

Figure 4. Requirements traceability flow from stakeholder needs to measurable outcomes.

4.3 Functional Requirements (FR)

The core functional requirements are defined below.

Table 3. Functional requirements set for **SiteScribe**.

ID	Functional Requirement	Rationale	Priority
FR-1	The system shall support project-scoped evidence ingestion for PDF, image, and text-based records.	Evidence normalization is the entry point of all downstream logic.	Must
FR-2	The system shall extract and persist textual representations and chunked segments when available.	Required for retrieval, triage context, and optional AI enrichment.	Must
FR-3	The system shall detect potential change signals from evidence and create triageable events.	Reduces manual discovery burden for PM workflows.	Must
FR-4	The system shall enable event-to-CO draft generation via template and optional AI-assisted paths.	Standardizes draft creation and accelerates drafting cycle.	Must
FR-5	The system shall provide structured CO editing (scope, clauses, assumptions, line items, status).	Supports contractual completeness and revision control.	Must
FR-6	The system shall support collaboration artifacts (comments, review signals, notifications).	Improves communication continuity during review cycles.	Should

ID	Functional Requirement	Rationale	Priority
FR-7	The system shall generate export artifacts (PDF + ZIP + manifest) with deterministic metadata.	Required for defensibility and package integrity verification.	Must
FR-8	The system shall support scheduled export execution and configurable delivery endpoints.	Supports recurring reporting and operational automation.	Should
FR-9	The system shall provide lexical and semantic search over project evidence and CO records.	Enables rapid retrieval in high-volume evidence contexts.	Should
FR-10	The system shall support org/project collaboration features (invites, role assignment, webhook management).	Needed for multi-tenant team operations.	Must

4.4 Non-Functional Requirements (NFR)

Table 4. Non-functional requirements and quality attributes.

ID	Non-Functional Requirement	Design Target	Priority
NFR-1	Security	Enforce authentication, authorization, validation, and secure headers.	Must
NFR-2	Tenant Isolation	Prevent cross-organization and cross-project data leakage.	Must
NFR-3	Traceability	Ensure evidence-to-CO provenance and auditable lifecycle transitions.	Must
NFR-4	Integrity	Provide hash-verifiable export manifests for generated packages.	Must
NFR-5	Availability	Maintain deterministic core workflow when AI services are unavailable.	Must
NFR-6	Maintainability	Modular architecture with clear domain/integration boundaries.	Should
NFR-7	Extensibility	Support incremental feature growth (integrations, metrics, policy).	Should
NFR-8	Usability	Keep operational workflows understandable for mixed technical roles.	Should

ID	Non-Functional Requirement	Design Target	Priority
NFR-9	Reproducibility	Support release/version archiving and DOI-oriented publication workflow.	Should

4.5 Mapping Requirements to Constraints

Requirements carry over hard constraints discussed in Section 3.

1. Governance constraints: FR-1 to FR-10 are subject to role and scope execution checks [14], [19], [20].
2. Validation constraints: FR-1/FR-2 are subject to input control restrictions [15], [18].
3. Integrity constraints: FR-7/FR-8 are subject to hash-based manifest generation [22].
4. Protocol constraints: the integration operations of FR-8/FR-10 are subject to secure HTTP, JSON, and signing [23], [26], [27], [32] [24], [28], [30], [31].

4.6 Strategy for Requirement Prioritization

The following operational risk model is applied to the prioritization.

1. Must: foundational workflows and governance control (FR-1..5, FR-7, FR-10, NFR-1..5).
2. Should: retrieval improvement, collaboration enhancements, extendability controls (FR-6, FR-8, FR-9, NFR-6..9).
3. Could (in the future): advanced domain intelligence and enhanced compliance and benchmarking functionalities.

This strategy demonstrates the minimum viable research system prioritizing safety with the added components for technical optimization.

4.7 Acceptance Criteria and Verification Hooks

Each critical requirement category is linked to a verification hook:

Table 5. Acceptance and validation mapping for key requirement clusters.

Requirement Area	Acceptance Condition	Verification Hook
Evidence ingestion	Invalid MIME/oversize payloads are rejected; valid payloads persist successfully.	Input validation tests + action-level checks
Event triage	Detection run produces event/signal output for qualifying evidence windows.	Detection unit tests + integration checks
CO lifecycle	Draft can be created, updated, and status-managed with role policy enforcement.	Server action tests + role-path verification
Export integrity	Manifest includes deterministic metadata and hash fields for package artifacts.	Manifest/hash tests
Access governance	Unauthorized actions are blocked across project/org boundaries.	Authorization path tests

4.8 Interpretation of requirements and associated risks

Some of the potential risks associated with requirement risks are:

1. variability of the domain across forms of the contract,
2. vagueness of the term “sufficient” evidence linkage,
3. organizational review semantics that are not captured in the generic workflow states, and
4. user’s expectation of reliance on optional AI outputs.

Mitigation strategy: retain deterministic baseline behavior and make clear the distinction between “required workflow output and optional ai features [5], [21].

4.9 Methodology and Architecture Section Transition

The baseline requirements outline the methodology and architecture:

1. What must be done is defined by the FR/NFR sets.
2. What may not be violated is defined by the constraints.
3. What must be shown to be verifiably true is defined by the acceptance criteria.

Thus, the following sections are organized based on these requirements and their associated architecture, pipeline, and evaluation framework.

5. System Design and Implementation

5.1 Methodology Overview

SiteScribe has been developed as an evidence-centric workflow system. This means that each stage of the workflow system has an evidence-based structured state transition which is consumable by the subsequent stage. As such, the workflow system has been implemented using a layered engineering approach:

- Normalize different project evidence into canonical records
- Identify and group evidence into triage events
- Create structured change orders from the context of the events (template-based and AI enriched options)
- Control edits and state transitions using role-constrained server-side actions
- Provide outputs as integrity-proofed packages (PDF+ZIP+manifest)

These techniques are consistent with retrieval-grounded AI workflows, operational software modularity, and secure system constraints.

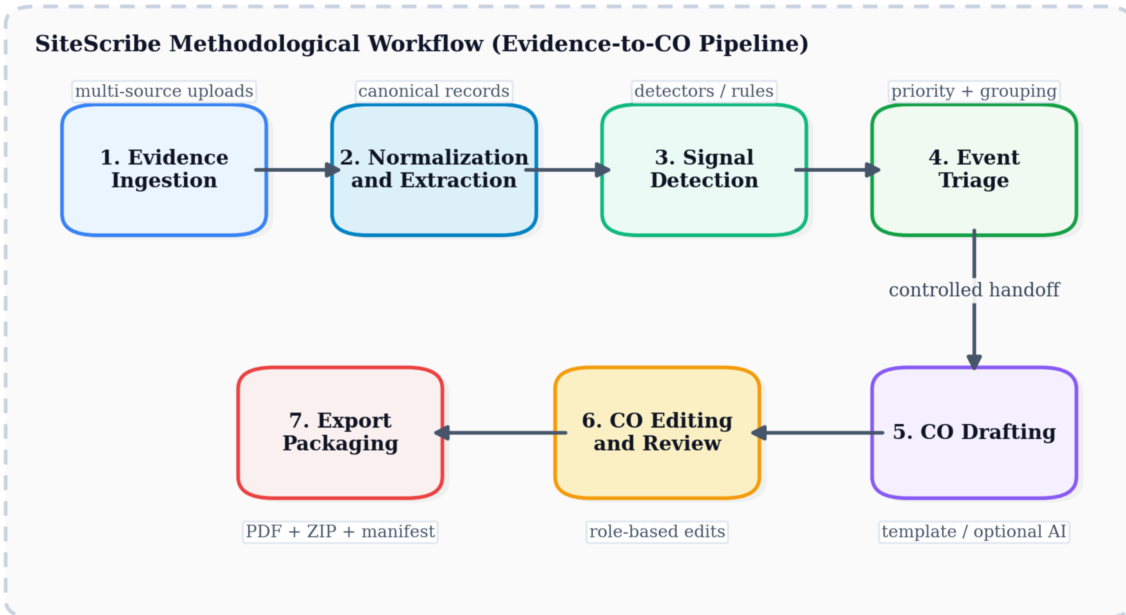

Figure 5. Methodological workflow of SiteScribe from evidence to export.

5.2 Logical Architecture

Utilizing modular architecture (full-stack), the platform integrates components such as the Next.js App Router and server actions [33], Auth.js/NextAuth [34], [35] credential/session middleware, and a Prisma-MySQL persistence layer [36], [37].

Primary architectural strategies include:

- server-side execution of business-critical authorization checks,
- UI components are non-deterministic and utilize results from actions,
- domain logic is routed into specialized service modules (lib/*), and
- infrastructure adapters (storage/email/webhooks/AI) are put into isolation to allow for substitute adjustments.

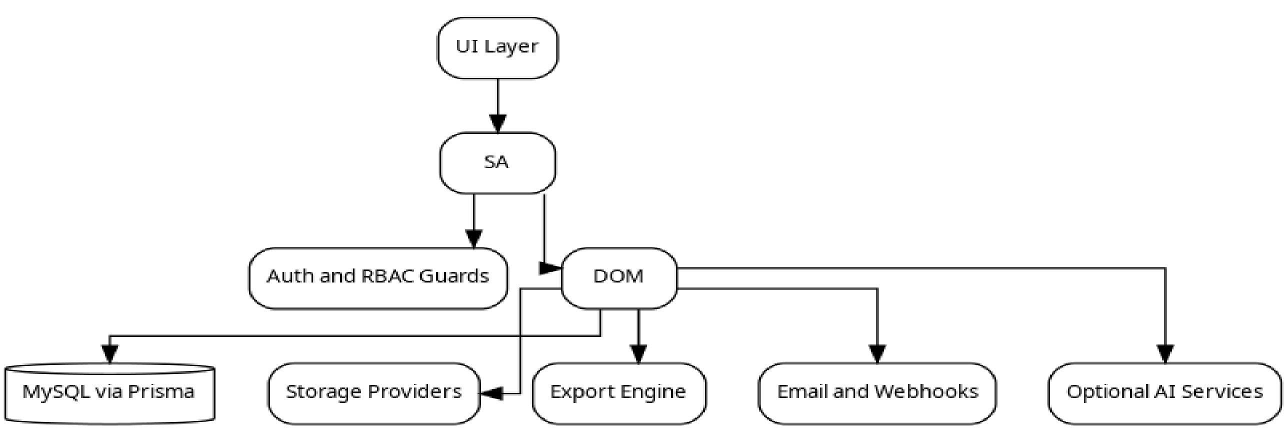

Figure 6. Logical architecture and integration boundaries.

5.3 Data Model and Multi-Tenancy

SiteScribe opts for organization-first tenancy. Each organization is the primary tenant, and all operational records (evidence, events, COs, exports) are scoped to the project level, with projects deeply nested under the organization.

Role governance is strictly hierarchical and follows the order: VIEWER < SUBCONTRACTOR < FIELD < PM < OWNER.

This architecture enables:

- isolation of tenants,
- mutations of policies in a consistent manner, and
- actions in the lifecycle that are auditable.

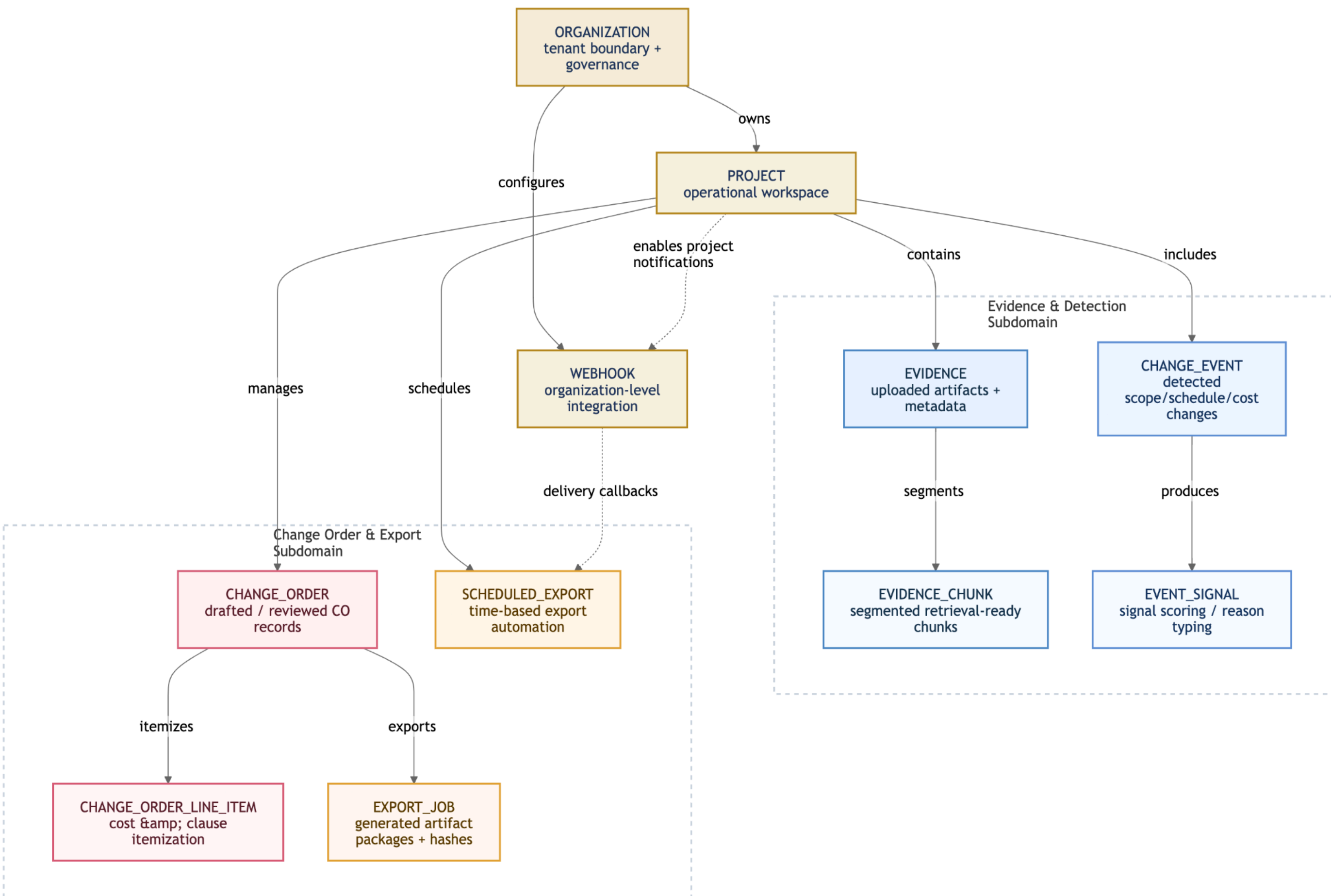

Figure 7. Core domain entities and multi-tenant data relationships.

5.4 Core Pipeline Implementation

5.4.1 Evidence Ingestion and Preprocessing

Prior to persistence, actions that upload evidence check the following: the file type, the file size, and the completion of the textual fields. The contents of a PDF, for example, are subject to extraction and division to facilitate retrieval. In addition, a hash of the file is computed for the sake of preserving the integrity of the export [22].

5.4.2 Creation of Events and Signal Detection

An event signal and event grouping are created as detected by the rule-driven detector from the most recent evidence collected. Optional AI enrichment may increase the level of accuracy on score/reason/type attributes [5], [21].

5.4.3 Lifecycle of Change Order Drafts

A CO draft is created by the system from the selected event which is structured and allows for titre, narrative, clause, assumption, exclusion, line item, and status edits to be made. Actions on comments or approvals can be used for collaborative governance.

5.4.4 Delivery and Export

The generated export composes the following:

a summary PDF,

a ZIP file of evidence, and

a JSON file of the manifest containing deterministic metadata and SHA-256.

The export may be downloaded, emailed, or created as a scheduled export job.

Table 6. Module-to-Workflow Mapping in Section 5

Workflow Stage	Representative Implementation Modules	Primary Output
Ingestion	app/actions/evidence.ts, lib/storage.ts, lib/validation.ts	canonical evidence record
Extraction/Chunking	lib/pdf-extract.ts, EvidenceChunk persistence	retrieval-ready text chunks
Detection/Triage	app/actions/signals.ts, lib/detect-signals.ts	events and event signals
CO Lifecycle	app/actions/change-order.ts, lib/co-draft.ts	structured CO artifact
Export	app/actions/export.ts, lib/export-package.ts, lib/export-job.ts	PDF+ZIP+manifest package
Governance/Audit	lib/auth-server.ts, lib/rbac.ts, lib/audit.ts	policy-compliant, auditable transitions

5.5 AI Augmentation Layer (Optional)

User discretion plays a role with AI in **SiteScribe**. If AI is toggled off, the core workflow will still be fully functional. AI functionality assists with:

- Summarization of evidence
- Suggestions of evidence type
- Semantic search embeddings

- Enrichment of signals
- Enrichment of CO narrative
- Description of photos

With AI, **SiteScribe** adheres to a bounded augmentation principle, meaning there are areas AI can enhance, like speed and the quality of context, but will not replace any policy related critical functions, or deterministic export functions [5], [7], [21].

5.6 Collaboration and Integration Features

Alongside the core functionality of evidence-to-CO, **SiteScribe** is built with operational collaboration and integration modules, like

- In-app notifications for workflow events
- Webhook delivery for external automation
- Role-aware invitation and membership management
- Friend-based 1:1 messaging for team coordination

Supporting layers were implemented with these functions, and while assisting operational continuity, they do not alter the central decision provenance model.

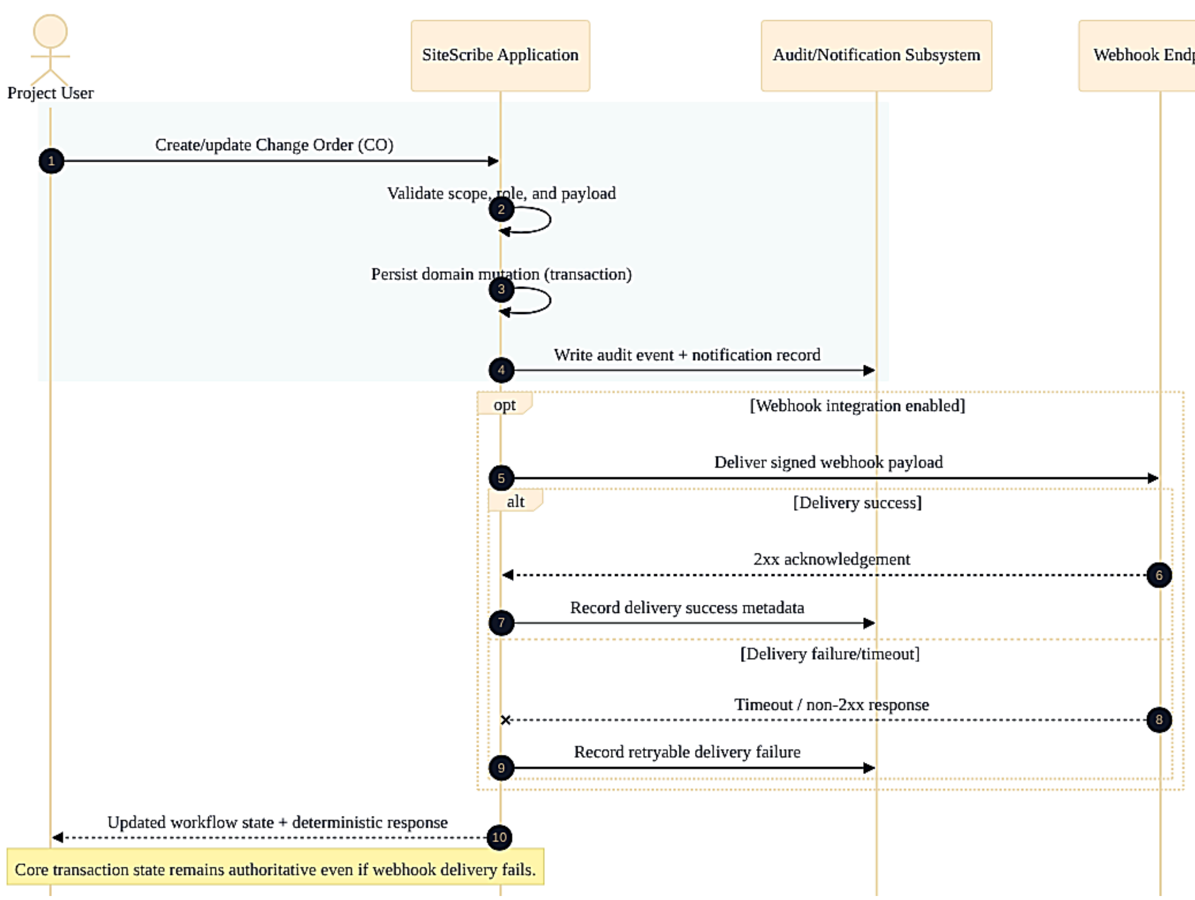

Figure 8. Collaboration/integration event flow after core workflow actions.

6. Security, Governance, and Reliability

6.1 Security-by-Design Principles

Using the defense-in-depth model, **SiteScribe** deploys security measures such as identity, authorization, input, and transport processing, as well as operational endpoint [13], [14], [19], [20].

Guiding principles include:

- Server-centric supervision for all actions important to any policy
- Role hierarchies and project/org scoping comply with least-privilege
- All external input requires prior validation
- All exportable legal documents must have integrity-preserving outputs
- Optional component failure (for example, AI/outbound services) leads to graceful degradation

6.2 Authentication and Session Governance

Session-based identity is insufficient to constitute authorization and server-side session checks. All operations with the potential to alter the state must pass role and scope checks [34], [35], [19].

Key governance points include:

- Protected operational routes require authenticated access
- Action ownership is determined by session-to-user resolution
- Production secret requirements and secure cookie posture
- Bounded session lifecycle assumptions

These controls mitigate risks associated with impersonation and unauthorized actions at the application boundary [13], [19].

6.3 Authorization and Isolation of Multi-Tenants

Employee boundaries as per tenant are managed by **SiteScribe** via organization/project based guards.

Authorization decisions are based on whether or not a person belongs to an organization or is part of a project ownership (lineage) or is listed as a participant (with the) requisite role to (the) action class. Horizontal role ordering (recipient FX < Pivot < Field (role) < (Project) Manager < Owner) affects evidence uploading, signal (classification) triaging, CO editing, and admin (role) functions.

This (role ordering) is a design to contain cross-tenant data breaches, privilege escalation, or other cross-tenant abuse of services (e.g., multi-tenant system) [14], [20], [21].

6.4 Safe Linking, and Validation & Sanitization of Inputs

All outward facing persistent values and queries undergo validation/normalization.

Controls enacted are (in the) shape of identifiers, (with) limits on types (and) counts (of values). Validation (of) URLs (with) a default of other protocols (that) model pathways to notification links.

Front-end evidence upload (interface) uses a size (set) and a class of safe payloads to limit and mitigate (the) attack surface [15], [18].

This layered validation approach meets an OWASP (Open Web Application Security Project) (standard for) validation (and) defense against injection [15], [18] [16].

6.5 Transport, Header Policies, and Endpoint Hardening

Protection via security headers, including, but not limited to, (Content Security Policy) CSP, and (other) browser enforced policies (and) controls are applied at (the) framework (level) to minimize (the) attack surface at the client (end) [17], [23].

Requested/replied token behaviors, and payload formatting, follow consistent 'Request For Comments', or RFC, guidelines [23], [25], [27], [29], [32].

The scheduled export endpoints utilize shared-secret protection and timing-safe comparison methods to reduce simple timing side-channel attacks [26].

6.6 Integrity and Auditability Guarantees

For CO exports, **SiteScribe** creates deterministic package metadata, which includes:

- file-level manifests,
- SHA-256 digests,
- artifact identifiers,
- timestamps, and
- Audit logging traces key lifecycle events (create, update, and status changes). This provides responsiveness and accountability for workflow audits.

Audit logs, combined with export integrity, enhance defensibility when the situation calls for a lot of documentation [22], [53], [58].

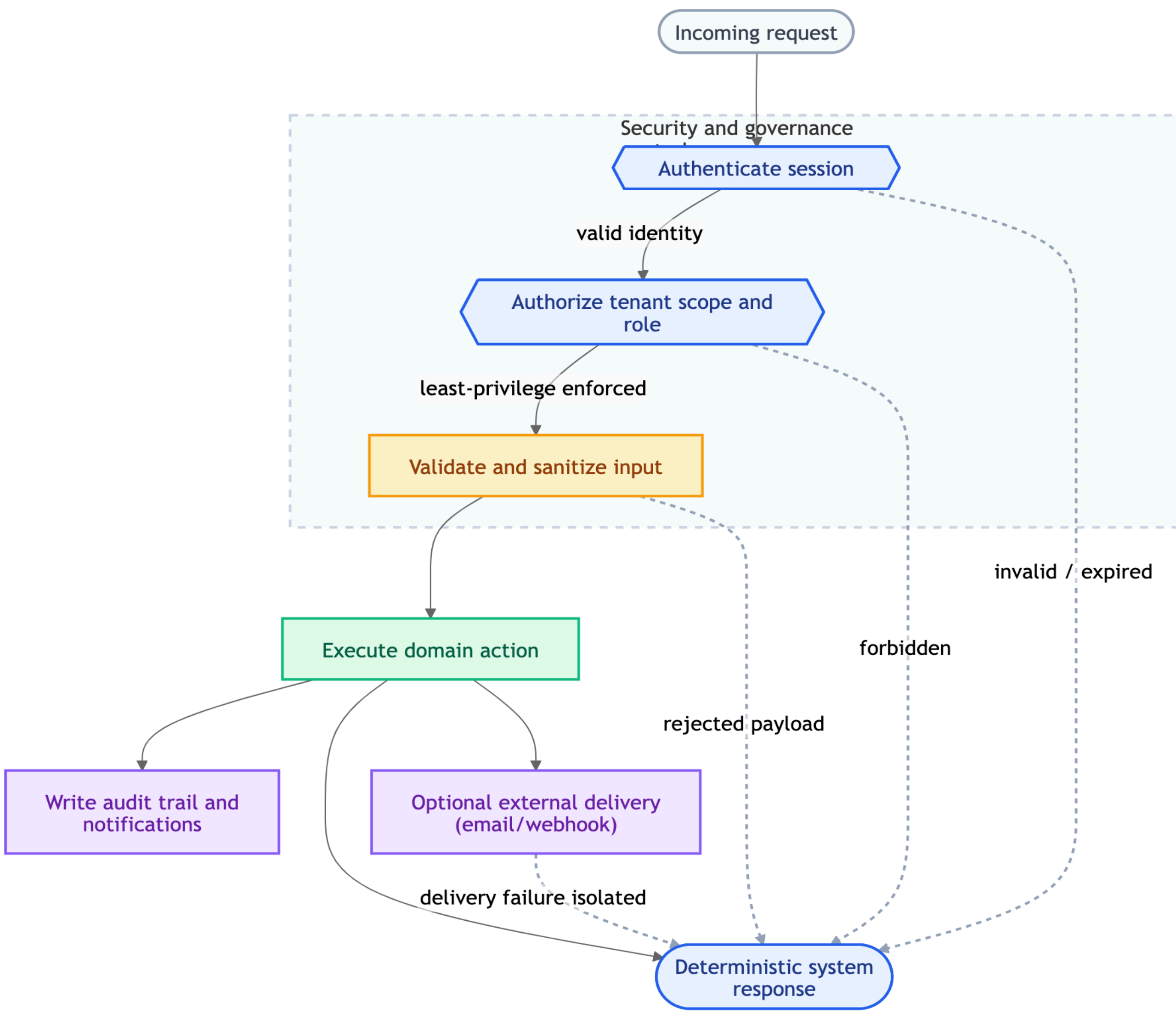

Figure 9. Security and governance control flow for server-side operations.

6.7 Reliability Model and Failure Handling

Reliability, assuming partial failure, is regarded as controlled continuity.

Example failure types and response strategies:

AI service is unavailable: continue a deterministic workflow that lacks AI enhancement.

Email or webhook failure: maintain the primary transaction state and display the results of the delivery status.

If there is a limitation to extraction (for example, the PDF text is poorly scanned), the evidence artifact and metadata will be kept, regardless of text quality.

For permission-denied actions, the system will return a non-success response and there will be no side-effects that persist.

Invalid input: reject before domain mutation.

This model ensures core system correctness does not depend on optional integrations.

6.8 Security and Reliability Control Mapping

Table 7. Security and reliability control mapping for **SiteScribe**.

Control Area	Mechanism in SiteScribe	Security/Reliability Goal	Reference Basis
Authentication	server-side session checks	authenticated action boundary	[19], [34], [35]
Authorization	org/project + role guards	tenant isolation and least privilege	[14], [20]
Input safety	validation/sanitization utilities	reject malformed or unsafe inputs	[15], [18]
Transport/browser hardening	CSP and related headers	reduce client-side exploitability	[17], [23]
Endpoint protection	secret-gated scheduled route	protect automated execution path	[26], [32]
Integrity assurance	SHA-256 manifest exports	verifiable artifact consistency	[22]
Accountability	audit log persistence	non-repudiable workflow history	[20], [53]
Degraded-mode reliability	optional AI/integration fallback	continuity under partial service loss	[21]

7. Evaluation and Discussion

7.1 Evaluation Objectives

Overall **SiteScribe**'s impact on the evidence-to-CO lifecycle will be assessed on four key dimensions:

1. Workflow efficiency (time-to-draft, time-to-review decision),
2. Traceability (evidence linkage completeness),
3. Governance (boundary behavior of authorizations),
4. Artifacts reliability (export integrity and delivery of artifacts).

These dimensions are directly related to the optimization variables described in Section 3 and the requirement constraints articulations provided in Section 4.

7.2 Evaluation Questions

The evaluation will answer the following questions:

RQ1: To what extent has the time necessary to generate the first draft of an available CO been reduced by **SiteScribe**?

RQ2: How does **SiteScribe** impact evidence-to-CO traceability and the workflows related to the traceable manual aggregation?

RQ3: Is there consistency in the enforcement of role and tenant boundaries in workflow-critical actions?

RQ4: Are the export packages produced by the system verifiable and reproducible in an operational sense?

RQ5: What influence does the optional AI augmentation to retrieval and drafting have on the utility (positive and negative) and on the workflow deterministic stability?

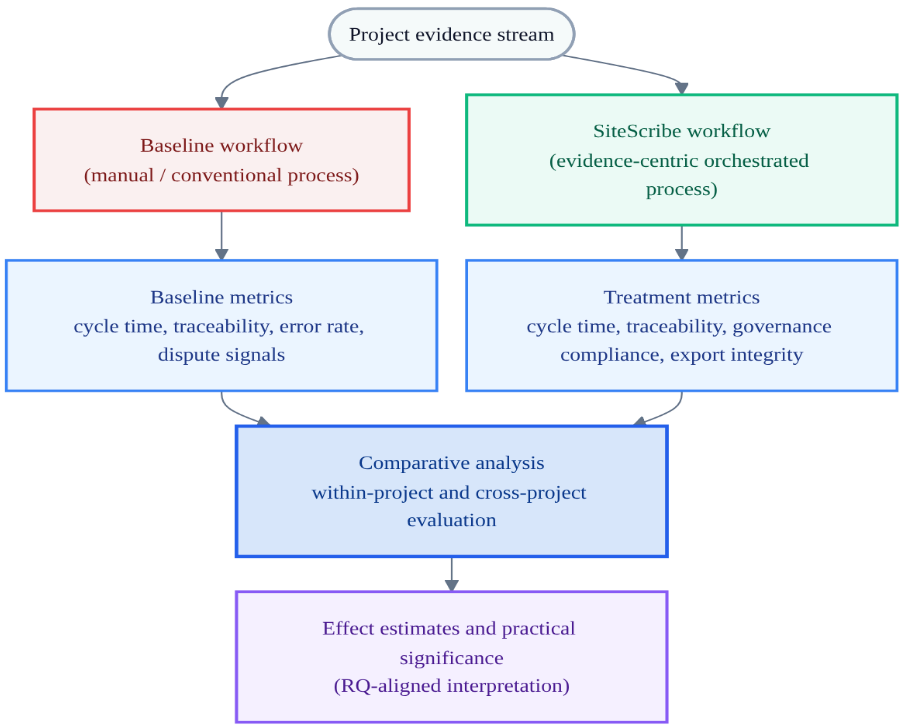

Figure 10. Comparative evaluation design: baseline versus **SiteScribe**-assisted workflow.

7.4 Evaluation Metrics

Table 8. Core evaluation metrics and research-question mapping.

Metric ID	Metric Name	Definition	Unit	Primary RQ
M1	Time-to-First-Draft	Duration from first relevant evidence to first CO draft creation	hours	RQ1
M2	Review Cycle Count	Number of revision loops before approval/rejection	count	RQ1

Metric ID	Metric Name	Definition	Unit	Primary RQ
M3	Traceability Coverage	Ratio of CO sections with explicit evidence backing	%	RQ2
M4	Authorization Correctness	Unauthorized actions correctly blocked	%	RQ3
M5	Export Integrity Pass Rate	Export packages with valid manifest hash checks	%	RQ4
M6	Delivery Reliability	Successful outbound delivery over total attempts	%	RQ4
M7	Search Effectiveness	Precision@k for evidence retrieval tasks	score	RQ5
M8	AI Dependency Stability	Core workflow completion rate when AI disabled/unavailable	%	RQ5

7.5 Measurement Protocol

7.5.1 Sources of Instrumentation

Data collection should integrate: persisted records containing workflow timestamps, chronological audit logs of actions, metadata of exports, verification hashes, authorization response actions, and retrieval task annotations for Precision@k.

7.5.2 Data Collection Windows

To reduce volatility: use data aggregation windows of one week, keep the onboarding phase separate from the steady-state phase, and for duration metrics report the median and the interquartile range.

7.5.3 Statistical Approach

For the recommended analysis, the following should be included: temporal variables that are skewed should use a non-parametric approach, effect sizes should be included in the analysis, exclusion of significance, primary metrics should include confidence intervals, stratified analysis should be conducted by role class and by the complexity of the project.

7.6 Strategy for Validation

Multi-layered verification should support the validity of the evaluation.

Unit-level validation: for the RBAC logic, behavioral scoring of signals, manifest and hash generation, and the routines of input validation.

Validation on the action level: for behavioral mutations and retrievals of projects.

Validation of the scenario: for workflow end-to end, from evidence uploads to the generation of exports.

Validation for resilience: for behavior in the absence of AI and failure in outbound deliveries.

This strategy separates the functional correctness from integrated volatility and is in line with secure software assurance practice.

7.7 Anticipated Results Framework

The design of architecture and control suggests the following expected outcomes:

1. Decrease in the average time to receive the first draft (M1).

2. Increase in coverage, and therefore improvement, in traceability (M3).
3. Excellent correctness in authorizations (M4 approaching ceiling).
4. Reliable, or near enough to be considered so, verification of records (M5).
5. Decrease in completions when AI is operating in a suboptimal state (M8).

These predictions are engineering conjectures, and not empirical claims. This is until controlled field measurements are undertaken.

7.8 Discussion

SiteScribe can be improved as workflow system and not just through the current software iterations.

The improvement of the platform lies in the ability to integrate ingestion, governance, drafting, and export integrity rather than in the singular improvement of each module. Strong security and reliability controls are as much a part of the measurable productivity and defensibility outcome as the module improvements.

The AI module will only improve the system if considered in isolation and will only provide marginal utility through drafting assistance. Integrating AI should not create the expectation for improved workflow completions as this could lead to an overestimate the role of the AI in the output of the system. [5], [7], [21].

8. Conclusion and Future Work

8.1 Conclusion

This study presented **SiteScribe** as the first evidence-based multi-tenant change-order intelligence construction platform. **SiteScribe** addresses fragmentation of documentation, delays in decision making, and efficient traceability issues that increase the construction industry's contractual and operational risks [53], [54], [58].

The engineering narrative was built out of the context and the definition of the gap was formed in the first two sections, then formal problem definition was articulated in Section 3, derivation of Section 4 constituted the requirements, while Section 5 encompassed the architecture and the plan for implementation, questions of reliability and security were discussed in Section 6, the evaluation framework was in Section 7, and finally, issues of reproducibility

The main conclusion here is that change-order support is robust when there is an integrated system that entails the following:

1. Deterministic execution of workflows;
2. Hierarchical organization/project scoping and governance role;
3. Evidence-to-output traceability;
4. Integrity preserving export semantics; and
5. Core operational correctness is preserved without AI, but rather with the support of operational context.

By adopting this approach, **SiteScribe** ensures operational continuity, irrespective of model or provider changes, while also remaining adaptable to retrieval-based contextual support [5],[7],[21].

8.2 Consolidated Contributions

From an academic-engineering viewpoint, the contributions include:

A grounded formalization of the problem in the domain of evidence-to-change-order transformation with multi-tenant constraints.

A requirement-to-architecture traceability chain that articulates stakeholder needs and measurable behavior of the systems.

A secure-by-design implementation posture that exhibits the integration of RBAC, validation gates, scoped mutations, signed/traceable export pathways, and auditability [13], [14], [19], [20].

A design framework for evaluation that focuses on and measures the impact of a workflow (time, traceability, resilience, correct authorization, and resilience), rather than impacts of isolated components.

A reproducibility and archival model that meets the criteria of the FAIR principles and software citation guidelines [45], [49], [52].

The contributions aim at narrowing the gap between the prototype level AI tooling and the production level contract-support systems.

8.3 Practical Implications

The implication of the findings for practice is that in organizations, evidence governance and workflow determinism should be prioritized before integrating high-level AI. In operational contexts, the most significant improvements are usually a result of systematic evidence capture, policy-compliant access behavior, and defensible export production.

Within the scope of the specific framework, the drafting and semantic retrieval augmented intelligence offer acceleration and contextual enhancement, whereas legal and contractual certainty relies on the traceability, completeness, and integrity assurances.

8.4 Limitations

Although this framework is relatively comprehensive, a number of limitations still exist.

Field-level results may differ due to the maturity of the organization, the contract regime, and the documentation discipline.

Discretionary AI actions may vary as a result of model updates and changes (provider side) [3], [4], [11].

The need for confidentiality may limit publication of raw datasets from the project.

The volatility of the consolidation ecosystem (external APIs, document types, channels of integration, delivery formats) may affect ongoing interoperability.

As a result, the evaluation claims outlined should be viewed as a preliminary framework and expectations in terms of engineering to be validated for a number of deployments in the real-world scenario.

8.5 Future Work

Future efforts should be directed to empirical expansion as well as to the improvement of particular aspects of the technology.

Field studies of various types of projects in different jurisdictions over an extended period of time in order to measure the sustained impact.

Extension of causal analysis to evaluate the impact of specific functions in isolation from the effects of onboarding or novelty.

Development of sophisticated multimodal evidential frameworks to enable more effective image and document cross-referencing [9], [10].

1. Policy-aware explanation interfaces that improve auditor and manager trust in system-generated suggestions.
2. Automated compliance templates for contract-specific export structures and regional governance regimes.
3. Benchmark datasets and open protocol suites that improve comparability of construction intelligence systems while preserving privacy.

8.6 Final Remarks

SiteScribe demonstrates that modern construction intelligence systems can be designed as accountable workflow platforms rather than isolated AI utilities. The architecture and publication strategy presented in this paper establish a path toward citable, reproducible, and operationally credible deployment in document-intensive construction decision environments.

The next stage is comprehensive empirical execution with persistent-identifier-linked release cycles, enabling stronger evidence for adoption decisions in both academia and industry.

References

[1] T. B. Brown et al., "Language Models are Few-Shot Learners," in Advances in Neural Information Processing Systems (NeurIPS), 2020. [Online]. Available: https://papers.nips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html

[2] L. Ouyang et al., "Training language models to follow instructions with human feedback," in Advances in Neural Information Processing Systems (NeurIPS), 2022. [Online]. Available: https://arxiv.org/abs/2203.02155

[3] A. Chowdhery et al., "PaLM: Scaling Language Modeling with Pathways," 2022. [Online]. Available: https://arxiv.org/abs/2204.02311

[4] OpenAI, "GPT-4 Technical Report," 2023. [Online]. Available: https://arxiv.org/abs/2303.08774

[5] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in Advances in Neural Information Processing Systems (NeurIPS), 2020. [Online]. Available: https://arxiv.org/abs/2005.11401

[6] Y. Gao et al., "Retrieval-Augmented Generation for Large Language Models: A Survey," 2023. [Online]. Available: https://arxiv.org/abs/2312.10997

[7] G. Izacard et al., "Unsupervised Dense Information Retrieval with Contrastive Learning," 2021. [Online]. Available: https://arxiv.org/abs/2112.09118

[8] H. Su et al., "Instructor Embedding: One Embedder, Any Task," 2022. [Online]. Available: https://arxiv.org/abs/2212.09741

[9] A. Radford et al., "Learning Transferable Visual Models From Natural Language Supervision," in Proc. ICML, 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[10] J. Li et al., "BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models," 2023. [Online]. Available: https://arxiv.org/abs/2301.12597

[11] H. Touvron et al., "Llama 2: Open Foundation and Fine-Tuned Chat Models," 2023. [Online]. Available: https://arxiv.org/abs/2307.09288

[12] Google DeepMind, "Gemini: A Family of Highly Capable Multimodal Models," 2023. [Online]. Available: https://arxiv.org/abs/2312.11805

[13] OWASP Foundation, "OWASP Top 10:2021," 2021. [Online]. Available: https://owasp.org/www-project-top-ten/

[14] NIST, "Security and Privacy Controls for Information Systems and Organizations (SP 800-53 Rev. 5)," 2020. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final

[15] OWASP Foundation, "OWASP API Security Top 10 (2023)," 2023. [Online]. Available: https://owasp.org/API-Security/editions/2023/en/0x11-t10/

[16] OWASP Foundation, "Application Security Verification Standard (ASVS) 4.0.3," 2021. [Online]. Available: https://github.com/OWASP/ASVS/releases/tag/v4.0.3

[17] OWASP Foundation, "Software Assurance Maturity Model (SAMM) v2.1," 2020. [Online]. Available: https://owaspsamm.org/model/

[18] OWASP Foundation, "OWASP Top 10 for LLM Applications 2025," 2025. [Online]. Available: https://genai.owasp.org/resource/owasp-top-10-for-llm-applications-2025/

[19] NIST, "Zero Trust Architecture (SP 800-207)," 2020. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-207/final

[20] NIST, "Secure Software Development Framework (SP 800-218)," 2022. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-218/final

[21] NIST, "Artificial Intelligence Risk Management Framework (AI RMF 1.0)," 2023. [Online]. Available: https://www.nist.gov/itl/ai-risk-management-framework

[22] NIST, "Cybersecurity Framework (CSF) 2.0," 2024. [Online]. Available: https://www.nist.gov/cyberframework

[23] R. Fielding et al., "HTTP Semantics," RFC 9110, IETF, 2022. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9110

[24] M. Nottingham and P. McManus, "HTTP Caching," RFC 9111, IETF, 2022. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9111

[25] M. Jones, J. Bradley, and N. Sakimura, "JSON Web Token Best Current Practices," RFC 8725, IETF, 2020. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8725

[26] JSON Schema Organization, "JSON Schema Draft 2020-12," 2020. [Online]. Available: https://json-schema.org/draft/2020-12

[27] C. Bormann and P. Hoffman, "Concise Binary Object Representation (CBOR)," RFC 8949, IETF, 2020. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8949

[28] A. Rundgren, B. Jordan, and S. Erdtman, "JSON Canonicalization Scheme (JCS)," RFC 8785, IETF, 2020. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8785

[29] Y. B. et al., "HTTP Message Signatures," RFC 9421, IETF, 2023. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9421

[30] M. West and J. Wilander, "Cookies: HTTP State Management Mechanism (rfc6265bis draft)," IETF Draft, 2024. [Online]. Available: https://datatracker.ietf.org/doc/draft-ietf-httpbis-rfc6265bis/

[31] M. Jones et al., "OAuth 2.0 Authorization Server Issuer Identification," RFC 9207, IETF, 2022. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9207

[32] Y. Sheffer, D. Benjamin, and A. G. et al., "Recommendations for Secure Use of TLS and DTLS," RFC 9325, IETF, 2023. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9325

[33] B. Beyer et al., "The Site Reliability Workbook," Google, 2022. [Online]. Available: https://sre.google/workbook/table-of-contents/

[34] Google Cloud, "DORA State of DevOps Report 2023," 2023. [Online]. Available: https://cloud.google.com/devops/state-of-devops

[35] Open Policy Agent, "OPA Documentation," 2024. [Online]. Available: https://www.openpolicyagent.org/docs/latest/

[36] CISA and NSA, "Kubernetes Hardening Guidance," 2022. [Online]. Available: https://www.cisa.gov/resources-tools/resources/kubernetes-hardening-guidance

[37] GitHub, "The State of Open Source and AI in 2024 (Octoverse)," 2024. [Online]. Available: https://octoverse.github.com/

[38] NIST, "Microservices-based Applications System Security Guide (SP 800-204C)," 2022. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-204c/final

[39] V. Lenarduzzi et al., "Software engineering for AI-based systems: a systematic mapping study," Information and Software Technology, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950584923001000

[40] A. B. M. Moniruzzaman et al., "A review of software quality models for modern software systems," Journal of Software: Evolution and Process, 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/smr.2345

[41] M. Fokaefs et al., "A study on refactoring practices in modern software repositories," in Proc. SANER, 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9054820

[42] M. Felizardo et al., "Guidelines for the search strategy to update systematic literature reviews in software engineering," Information and Software Technology, vol. 127, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S095058491930223X

[43] M. J. Page et al., "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews," BMJ, 2021. [Online]. Available: https://www.bmj.com/content/372/bmj.n71

[44] D. Mendez, M. Wagner, and M. Kalinowski, "Requirements engineering for AI-enabled systems: industrial challenges and practices," in Proc. RE, 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9121430

[45] Zenodo, "About Zenodo," 2024. [Online]. Available: https://about.zenodo.org/

[46] Zenodo, "GitHub Integration," 2024. [Online]. Available: https://help.zenodo.org/docs/github/

[47] International DOI Foundation, "DOI Handbook," 2023. [Online]. Available: https://www.doi.org/doi_handbook/

[48] FORCE11, "Software Citation Checklist for Authors," 2020. [Online]. Available: https://www.force11.org/software-citation-checklist

[49] FAIR4RS Working Group, "FAIR Principles for Research Software (FAIR4RS)," 2022. [Online]. Available: https://doi.org/10.15497/RDA00068

[50] ACM, "Artifact Review and Badging Version 1.1," 2020. [Online]. Available: https://www.acm.org/publications/policies/artifact-review-and-badging-current

[51] A. Katz et al., "Recognizing the value of software: a software citation guide," F1000Research, vol. 9, 2020. [Online]. Available: https://f1000research.com/articles/9-1257

[52] UNESCO, "UNESCO Recommendation on Open Science," 2021. [Online]. Available: https://www.unesco.org/en/open-science/about

[53] J. S. Ramadhan and M. Waty, "Impact of Change Orders on Cost Overruns and Delays in Large-Scale Construction Projects," Engineering, Technology & Applied Science Research, vol. 15, no. 1, pp. 20291-20299, 2025. [Online]. Available: https://etasr.com/index.php/ETASR/article/view/9449

[54] M. A. Al-Mutairi et al., "Effects of change orders on the budget of the public sector construction projects in the kingdom of Saudi Arabia," Results in Engineering, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2590123023007557

[55] F. Antoniou and A. V. Tsioulpa, "Assessing the Delay, Cost, and Quality Risks of Claims on Construction Contract Performance," Buildings, vol. 14, no. 2, 2024. [Online]. Available: https://www.mdpi.com/2075-5309/14/2/333

[56] P. Dahlin, M. R. Mougouei, and M. E. M. Eriksson, "Drivers of cost and time overruns: A client and contractor perspective," Organization, Technology and Management in Construction, vol. 13, no. 1, 2021. [Online]. Available: https://sciendo.com/article/10.2478/otmcj-2021-0006

[57] A. K. Shukla and N. S. Beniwal, "Delay Analysis of Infrastructure Construction Projects in India," Journal of the Institution of Engineers (India): Series A, vol. 106, 2025. [Online]. Available: https://link.springer.com/article/10.1007/s40030-025-00899-5

[58] M. B. Ahmadzai and K. Ye, "A mixed-method investigation of the root causes of construction project delays in Afghanistan," Heliyon, 2025. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/39906858/

Appendix A. Functional Requirement to Verification Matrix

Requirement Group	Verification Method	Primary Evidence	Acceptance Threshold
FR-1 to FR-3 (Ingestion and Normalization)	Unit + integration tests	Structured logs, validation reports	> 95% valid ingestion success on benchmark set
FR-4 to FR-6 (Signal Detection and CO Drafting)	Scenario tests + expert review	Event traces, draft snapshots	Correct event assignment and complete draft fields
FR-7 to FR-8 (Export and Integrity)	Deterministic replay + hash check	Manifest files, digest comparison	100% hash consistency in repeated exports
FR-9 to FR-10 (Collaboration and Integration)	API contract tests + fault injection	API test reports, retry logs	No unauthorized mutation, graceful retry behavior

Note: This appendix table operationalizes Section 4 acceptance hooks by mapping functional requirement clusters to concrete verification artifacts and measurable thresholds.

Appendix B. Evaluation Metrics Operationalization Table

Metric ID	Metric Name	Formal Definition	Data Source	Interpretation Caution
M1	Time-to-first-draft	Median time from first evidence upload to first CO draft state	Event timestamps	Sensitive to onboarding effects

Metric ID	Metric Name	Formal Definition	Data Source	Interpretation Caution
M2	Draft completion ratio	Number of required fields auto/pre-filled divided by total required fields	Draft snapshots	Role-dependent field policies may affect comparability
M3	Traceability coverage	CO statements with linked evidence objects divided by total CO statements	Link graph + audit trail	Link presence does not always imply semantic adequacy
M4	Authorization correctness	Authorized actions accepted + unauthorized actions rejected divided by total policy checks	Access control logs	Requires stable role taxonomy
M5	Export integrity success	Exports with verifiable manifest hash divided by total exports	Manifest verification logs	External tampering outside system scope is excluded
M6	Integration reliability	Successful webhook/API deliveries divided by attempted deliveries	Delivery logs	Third-party outages may dominate failures
M7	User correction burden	Median manual edits per draft before approval	Version history	Team workflow style can bias values
M8	Degraded-mode completion	Workflows completed while AI unavailable divided by total degraded-mode runs	Resilience test logs	Depends on scenario realism

Note: Metrics in this appendix should be interpreted jointly; single-metric optimization can produce misleading conclusions about contractual readiness.

Appendix C. Source Code and Project Repository

The complete implementation of the SiteScribe platform is available in an open source repository and is meant to encourage transparency, reproducibility and additional academic peer review of the SiteScribe platform.

The repository includes the SiteScribe core components such the SiteScribe SiteScribe multi-tenant system, the SiteScribe evidence based data management system, SiteScribe change order intelligence modules, and the backend systems of the platform at the time of the development of the SiteScribe platform. In addition, the repository contains all the examples, instructions, and diagrams that can be used to deploy and customize the SiteScribe system to a specific real-world construction project situation.

Access to the implementation provides a basic understanding of the system architecture and workflow of the proposed framework. There is a possibility to review the system modules, duplicate the experimental configurations and reconfigure the system for other purposes throughout the construction industry or academic research.

The project repository is accessible at the following URL:

GitHub Repository:
https://github.com/muratyanasoglu/SiteScribe