A Standardized Workflow Framework for Building Construction Project Management: Phase-Linked Controls for Reliable Delivery

Atul Prakash Lad

Florida, USA
atulpplad@gmail.com

Abstract:
Building projects routinely experience avoidable cost growth, schedule slippage, rework, and closeout delays because project management (PM) workflows are fragmented across phases, decision authority is inconsistently enforced, and teams often build from drifting or unofficial information. This paper proposes a standardized, building-focused workflow framework that converts common PM activities into a reliability system built on enforceable controls rather than informal best practices. The framework is operationalized through (1) six workflow modules spanning preconstruction through turnover, written as repeatable purpose-to-handoff procedures; (2) an artifact dictionary that defines a minimum viable set of control instruments (logs, registers, gates, and evidence indexes) with required fields and control rules; and (3) a failure-mode control matrix that models recurrent "traceability breaks" as auditable breakdown points where downstream work proceeds without stable decision records or verifiable closure evidence. Each failure mode is mapped to a minimum control pack composed of a required log entry with ownership and timestamps, an approval gate, a decision cadence, escalation thresholds, and closure-evidence requirements. The framework is tool-agnostic and can be implemented in spreadsheets or PMIS platforms. By linking phase outputs to next-phase controls and making governance and version control explicit, the proposed approach aims to reduce decision latency, limit unpriced change accumulation, improve inspection closure discipline, and stabilize turnover readiness under typical building delivery constraints.

Keywords: Construction project management; standardized workflow; governance; control artifacts; traceability; failure modes; change management; procurement scheduling; RFI management; submittal control; document control; closeout readiness.

I. INTRODUCTION

A. Problem: fragmented PM workflows: delays, rework, change chaos, closeout slippage

Building construction projects do not break only because the engineering is hard. They break because project information and decisions move through a weak pipeline [13].

A building is delivered through hundreds of small commitments made over months: a scope interpretation during preconstruction, a procurement decision that shifts lead time, an RFI response that clarifies a detail, a submittal approval that locks a product, a field directive that quietly becomes extra work, and an inspection closure that determines readiness for occupancy. When these commitments are not governed by a consistent workflow, the project behaves like a system with loose bolts. It can still move, but it rattles, wastes energy, and eventually something critical slips.

In most building projects, the Project Manager (PM) is expected to be the integrator [2], [3]. The PM sits at the intersection of owner intent, design documents, subcontractor scope, procurement timing, schedule logic, quality inspections, cost status, and closeout deliverables. Yet in practice, the workflow is often fragmented into separate lanes that do not share a single operating logic:

- Budget control is managed in one place (baseline budget and revisions).
- Schedule control is managed somewhere else (critical path and lookaheads).
- Procurement and buyout operate in another lane (RFPs, leveling, awards, subcontracts).

- Construction administration operates in another lane (RFIs, submittals, coordination, BIM).
- Change control is treated as a periodic paperwork event instead of a managed lifecycle.
- Quality and compliance become reactive (inspections are chased after readiness is already late).
- Closeout is pushed to the end even though it depends on records created from day one.

This fragmentation creates predictable failure patterns because the project loses traceability. A decision is made, but it is not linked to the cost baseline. A drawing is revised, but procurement continues using an older version. A submittal is approved, but installation proceeds from a superseded shop. A verbal directive is given, but it is not converted into a documented change event with pricing and schedule impact. These are not abstract communication problems. They are workflow failures: the project lacks stable controls and decision gates that keep information reliable over time [14], [17].

The downstream impacts are visible and measurable across building projects:

1)   Delays driven by unresolved questions and late approval: RFIs and submittals exist to convert ambiguity into buildable clarity. When the RFI and submittal process is unmanaged (no log discipline, no priority rules, no response expectations), critical path activities stall while the team waits for answers or approvals. The schedule impact is rarely isolated. A single delayed submittal can prevent procurement, which delays fabrication, which compresses installation, which increases overtime, which increases errors, which expands punch list volume. When decisions arrive late, the project pays for time twice, first in delay, then in acceleration [15], [18].

2)   Rework and quality drift from inconsistent controls: Quality problems are often treated as field-level issues, but many are workflow-level issues. Without a defined inspection and test planning approach (inspection hold points, checklists, closure logs), defects are discovered late, when access is limited and correction requires rework. Research links weak information flow and weak quality planning to increased defects and rework [13], [14]. Even conservative literature recognizes rework as a significant cost driver in construction [14]. The PM's workflow, not just field labor performance, influences whether errors are caught early (lower cost) or late (higher cost).

3)   Change chaos from undocumented directives and weak governance: Change is unavoidable in building construction [17]. The failure is not the existence of change. The failure is uncontrolled change. When change events are not captured early, priced consistently, reviewed for schedule impact, and approved before execution, the project accumulates invisible liabilities. These liabilities surface as claims, disputes, strained relationships, and budget shocks. Verbal directives and "start now, we'll figure it out later" instructions are especially damaging because they separate the physical work from the contractual record. That separation is where disputes are born [4], [5].

4)   Procurement failures that silently destroy the schedule: Procurement and buyout are often framed as administrative tasks, but they are schedule-critical engineering of time. The procurement schedule is not a nice-to-have document. It is the bridge between the critical path and material reality. If long-lead items are not identified early and tracked through a long-lead register (or equivalent), the project may keep progressing on paper while fabrication, shipping, and approval constraints are already pushing the project off milestones [2], [22]. Once the long-lead window is missed, the schedule can rarely be recovered without cost and quality consequences.

5)   Closeout slippage because closeout was not built into the workflow: Turnover and closeout are often treated as the last chapter, but they are the result of every chapter. Punch lists, as-builts, O&M manuals, training, warranties, and occupancy approvals depend on earlier decisions: what was approved in submittals, what was changed by RFIs, what was installed versus what was specified, what inspections were passed, what test reports were collected, and what corrective actions were documented. When closeout is not planned as a structured workflow with a turnover package index and closure gates, the project experiences document chaos at the exact moment when time is least available and stakeholder pressure is highest [2], [3].

Across these issues, the pattern is consistent: fragmented workflows create predictable failure modes, and those failure modes translate into schedule delays, rework, budget instability, and extended closeout [14], [15]. The PM is usually blamed for these outcomes, but the deeper cause is that many organizations do not provide PMs with a standardized, phase-linked workflow framework that defines [2], [3]:

- the minimum required control artifacts,
- the authority chain (who initiates, reviews, approves, records), and
- the control rules (logs, gates, cadence, escalation thresholds).

A building project is not only built in concrete, steel, and glass. It is built in decisions, approvals, records, and controlled handoffs [13]. When that decision system is fragmented, the building inherits the fragmentation.

B. Research Aim & Contribution:

This paper proposes a Standardized Workflow Framework for Building Construction Project Managers that treats PM work as a disciplined system of control artifacts, governance, and phase-to-phase handoffs. The objective is not to restate common PM duties. The objective is to standardize how those duties are executed so that project decisions remain traceable from preconstruction through turnover [9].

1)   Research aim: The aim is to develop a building-focused, end-to-end PM workflow framework that:
- Organizes the building lifecycle into a unified six-phase spine for building projects (defined later in the paper).
- Defines governance and accountability for the three highest-risk administrative channels in building projects: RFIs, submittals, and change orders, including who initiates, who reviews, who approves, and who records.
- Converts common failure modes into enforceable workflow controls so the framework is operational, not aspirational. In this paper, "control" is expressed through defined artifacts and rules: logs and registers, approval gates, meeting cadence, escalation thresholds, and closure evidence requirements.
- Provides a standardized artifact dictionary that anchors PM work in consistent terminology and purpose (budget and schedule baselines, procurement controls, subcontract scope instruments, and the core administration and closure logs).

2)   Contribution: **This paper contributes four primary elements:**
a)   A unified phase spine with explicit cross-phase linking logic: outputs from one phase become controls for the next, so the workflow functions as a connected system rather than isolated responsibilities.
b)   A governance model that is audit-friendly and contract-aligned: standardized decision authority and traceability for RFIs, submittals, and change orders to support defensible records, consistent approvals, and controlled implementation [4], [5], [6].
c)   A failure-mode to control-mechanism map: high-frequency failure modes linked to specific control mechanisms (logs, gates, cadence, escalation rules, closure evidence requirements), forming a practical reliability system.
d)   A minimum viable toolkit of control artifacts: core artifacts formalized as control levers, not paperwork, including budget baseline and revisions, critical path prioritization, procurement schedule controls, subcontract scope instruments, and the administration, quality, and closeout registers that maintain traceability.

The intended outcome is a standardized, building-focused PM workflow framework that can function as:
- a training reference for developing PMs,
- an organizational consistency standard,
- a governance template for RFIs, submittals, and changes, and
- a reliability model designed to reduce delays, rework, uncontrolled changes, and closeout slippage by design [14], [15], [16].

If a building project is a chain of commitments, this framework is the method for keeping that chain unbroken.

II. BACKGROUND AND PRIOR FRAMEWORKS

A. Standard Phase Models (PMBOK/ CMAA / textbooks)

Across authoritative project management and construction management references, one point is consistent: projects are managed through phases, and within each phase the PM applies recurring control domains (scope, time, cost, quality, risk, procurement, and communications) [1], [3]. Differences across sources are mainly terminology and packaging, not the underlying logic.

1) General project management lifecycle (PMBOK-style view): General project management frameworks commonly describe a lifecycle as initiate → plan → execute → monitor/control → close [1], [27]. This view maps to construction at a high level, but building delivery adds a structural reality: execution is a multi-party production system (many firms, trades, and delegated roles), not a single organization performing work end-to-end. Because responsibility is distributed, construction relies more heavily on disciplined coordination, formal approvals, and record-based control than many centralized industries.

2) Construction management lifecycle (CMAA and construction management textbook view): Construction management sources typically describe building delivery in phase groupings closer to how projects are actually produced [2], [3]:

- Pre-project planning / feasibility (business case, early scope intent, preliminary budget and schedule)
- Design development (documents mature from concept to buildable packages)
- Procurement / bidding / buyout (selecting trade partners; locking scope, pricing, and risk allocation)
- Construction execution plus contract administration (field production plus formal decision/document management)
- Closeout / turnover (testing, punch, documentation, training, occupancy readiness)

This framing matters because controls must be staged. Early intent must be translated into enforceable scope, executable procurement commitments, field-ready approvals, and verifiable completion records.

3) Why phase models alone are not sufficient for workflow reliability in buildings: Phase models describe when major work clusters occur. They do not consistently specify how to keep information reliable as it moves between organizations and between phases. In building projects, failures concentrate at handoffs where a decision must be converted into a controlled record that downstream parties can safely act on [13]. Examples include:

- Scope intent translating into bid packages, then subcontract scope, then installed work.
- Design decisions translating into submittal approvals and procurement commitments.
- Field questions translating into formal clarifications that govern installation and inspection.
- Closeout requirements depending on records created progressively, not reconstructed at the end.
  For this paper, phase-based thinking remains the correct foundation. The limitation is not the idea of phases.

The limitation is that phase models often stop before defining the operational mechanics that protect traceability across handoffs. That missing layer is the focus of the next sections.

B. Governance norms in building projects: authority and records for RFIs, submittals, and change documentation

In building construction, "governance" is not a management preference. It is embedded in how projects are structured through contracts, delegated responsibilities, and formal communication channels [4], [7]. The practical purpose is simple: ensure that key decisions affecting design intent, product acceptance, and scope adjustments are made by the appropriate parties and captured in a retrievable record [4].

Within this environment, three channels function as the core administrative mechanisms that convert evolving conditions into controlled direction:

- RFIs formalize design clarifications so field execution does not rely on informal interpretation [6].
- Submittals formalize product and fabrication acceptance, so procurement and installation proceed based on documented conformance [2], [3].

- Change documentation formalizes scope evolution, so time and cost impacts are authorized and reflected in the contract record [4], [5].

The unifying requirement across these channels is record integrity [9]. On complex building projects, the ability to defend acceptance, payment, compliance, and turnover depends on what can be shown in the project record, not what was assumed in conversation [4]. When authority and records are not aligned, decisions may exist in practice but remain unstable as a basis for procurement, installation, inspection, and turnover.

To keep terminology consistent in this paper, the following definitions are used (as commonly applied in construction practice):

- Critical path: the sequence of activities that determines overall project duration.
- Procurement schedule: the set of required awards and purchasing dates needed to meet the construction schedule.
- Scope sheet: a checklist of work items and requirements expected within a trade's scope.
- Request to award: a formal request to authorize subcontract/supplier award based on scope, commercial terms, and requirements.
- RFP: a request issued to obtain bids/proposals from subcontractors or suppliers.
- Boilerplate: subcontract general requirements that apply across all trades.
- SOW: trade-specific requirements defining the subcontractor's deliverables and responsibilities.

These are not "extra paperwork." They are governance instruments that keep decisions executable and defensible across a multi-party delivery system.

C. Gap: existing guidance is strong in parts, but weak as an integrated operating workflow:

Prior frameworks are valuable, but they tend to provide components rather than a single, execution-ready workflow that stays stable across phases and across organizations [1], [3]. The gap is not that the industry lacks concepts. The gap is that projects often lack a standardized way to connect concepts into day-to-day control.

Three gaps are common:

1)   Descriptive models without operational linking rule: Phase models and knowledge-area guidance describe what must be managed. They less often define the linking rules that govern handoffs across the lifecycle, such as required inputs/outputs, decision gates, and objective closure criteria that make a handoff safe [9]. Without these linking rules, projects can appear organized while critical decisions remain distributed across emails, meetings, and disconnected logs.

2)   Governance norms are recognized, but not consistently converted into routine control: Contracts and standard practice establish that RFIs, submittals, and change documentation should follow defined authority and recordkeeping patterns [4], [5], [6]. However, guidance is less consistent on how to operationalize these norms into repeatable routines that hold up under schedule pressure: what must be logged, what must be escalated, what constitutes "closed," and what evidence is required for downstream reliance.

3)   Tool availability does not equal workflow standardization: Digital systems can store and route information, but tools typically reflect the process a team chooses to run [11], [12]. Where the underlying workflow is not standardized, implementation quality varies by project team and individual PM habits [28]. In that environment, two projects can use the same software and still produce very different levels of decision traceability and control.

This background establishes the need for a building-focused workflow that is not merely a list of responsibilities or phases, but a defined system of handoffs, authority, and records that can be executed consistently and reviewed objectively [9].

III. METHODOLOGY

This study uses a structured synthesis method to develop a standardized workflow framework for building construction project managers. The method is intentionally practical and audit-friendly: it combines authoritative references (standards and textbooks) with peer-reviewed research to identify recurring workflow structures, recurring breakdown patterns, and the control mechanisms that stabilize execution [1], [9]. The output of the method is a unified framework that connects four elements into one operating model: lifecycle phases, control artifacts, governance (RACI), and reliability controls [1].

A. Literature synthesis approach

The literature synthesis was designed to answer two questions:

- How do standard sources define the building project lifecycle and the PM's responsibility domains?
- What workflow breakdowns and control practices are reported repeatedly across building projects?

Sources were grouped into two tiers to separate "accepted structure" from "evidence of recurring failure and control."

1)   Tier 1: Authoritative references: This tier establishes widely accepted phase models, recurring PM control domains, and contract administration norms that shape expected roles and decision channels [2], [4]. These sources were used to anchor the framework's lifecycle structure and governance expectations.

2)   Tier 2: Peer-reviewed research: This tier was used to identify and cross-check recurring patterns reported in the research literature, including common causes of workflow instability (for example, decision latency, coordination gaps, uncontrolled change progression, weak quality closure discipline) and the interventions commonly associated with improved reliability (for example, disciplined logs and registers, decision gates, planning cadence, inspection and closure controls) [14], [15].

Instead of summarizing each source independently, a structured extraction template was applied to reduce repetition and improve traceability of the synthesis [9]. For each source, relevant content was coded into four buckets:

- Lifecycle and responsibility definitions: what is expected to be controlled, and when it becomes relevant.
- Governance expectations: who initiates, reviews, approves, and records decisions in the formal channels.
- Control mechanisms: artifacts and rules that stabilize work (logs/registers, gates, cadence, escalation, closure criteria).
- Recurring breakdown patterns: commonly reported failure modes, root causes, and typical impacts.

This approach supports synthesis across sources because no single reference provides a complete, end-to-end operating workflow that integrates governance and reliability controls in one model [1], [3].

B. Practice-informed inputs:

In addition to literature, the framework was refined using practice-informed inputs derived from typical building project management documentation and workflow artifacts. These inputs are treated as generic practice patterns, not as project-specific case study data.

The practice-informed component contributed three types of inputs:

- Terminology alignment: common definitions used in practice for core PM terms and instruments (for example, baseline budget versus revisions, critical path concepts, procurement schedule timing, RFP and award steps, subcontract structure such as boilerplate versus scope of work, and the core administration logs).
- Typical workflow sequences: recurring operational sequences that PM teams follow to move from intent to commitment to execution to closure (for example, bid solicitation and leveling, scope clarification, award and buyout, approvals and procurement, installation readiness, inspection closure, and turnover documentation).
- Operational behaviors that drive reliability: repeatable behaviors that determine whether a workflow holds under schedule pressure (for example, log discipline, response deadlines, escalation triggers, and "no-go" gates that prevent uncontrolled execution).

Practitioner observations were used to test the framework for implementability and to ensure the workflow can be executed with common project systems (whether software-based or spreadsheet-based) [1]. To maintain confidentiality and general applicability, practice inputs are presented without identifiers, project data, proprietary procedures, or organization-specific role labels.

C. Framework construction logic: phases, artifacts, governance, and reliability controls

The framework was constructed using a four-layer logic model. Each layer is necessary; none is sufficient alone.

1)   Layer 1 - Phase spine: A unified lifecycle spine was selected based on convergence across standard references and practical building delivery sequencing [1], [3]. The phase spine provides a stable structure for organizing PM responsibilities and defining handoffs.

2) Layer 2 - Artifact dictionary: A minimum set of control artifacts was defined to stabilize decisions and maintain traceability across the lifecycle [9]. Artifacts were selected using a minimum-viable principle: if an artifact is required to prevent recurring breakdowns or to establish reliable downstream reliance, it belongs in the baseline toolkit; otherwise it is treated as optional.

3) Layer 3 - Governance model: For the administrative channels most sensitive to schedule, claims, and rework exposure (RFIs, submittals, and change events), a RACI (Responsible, Accountable, Consulted, Informed) structure was defined to prevent decision drift and record ambiguity [16], [18]. The RACI assigns who executes, who provides final accountability, who must be consulted, and who must be informed for the decision to become actionable and traceable [1].

4) Layer 4: Reliability overlay: Finally, recurring failure modes were converted into enforceable controls using a consistent mapping structure: failure mode → root cause → impact → required control mechanisms [9], [10]. Controls are expressed as operational mechanisms that can be executed and audited (logs/registers, gates, cadence routines, escalation thresholds, and closure evidence rules) [9]. This layer ensures the framework is operational, not descriptive.

D. Limitations

This paper is a synthesis-based framework study. Its strengths and limits are stated directly.

- Not a statistical performance study. The paper does not claim quantified causal effects based on controlled empirical testing within this manuscript.
- Context sensitivity. Building projects vary by delivery method, contract structure, regulatory environment, and team maturity. The framework is designed to be adaptable, but governance assignments and approval gates must be aligned with the governing contract and delivery method [4], [7].
- Implementation maturity varies. Some teams implement controls through a project management information system; others rely on shared spreadsheets and repositories. Outcomes depend on discipline and enforcement, not the mere existence of templates [9].
- Scope boundary. The framework is targeted to building construction (commercial and residential). Other project types may share mechanisms but are not the focus.

Future validation path. A next step is empirical validation through pilot deployment across multiple building projects using operational metrics that indicate workflow health (for example, cycle times and backlogs for RFIs and submittals, aging of change events, long-lead adherence, inspection closure rates, and closeout completeness at substantial completion) [16], [18].

IV. PROPOSED END-TO-END PM WORKFLOW FRAMEWORK

This section defines the proposed framework as an operating system for building construction project management. It keeps four elements connected at all times:

- Phases: when decisions are made and handed off
- Artifacts: where decisions are recorded and controlled
- Governance: who has authority to initiate, review, approve, and log decisions
- Controls: the rules that keep the system reliable (logs, gates, cadence, escalation, closure evidence)

The framework is minimum viable but complete. It does not attempt to catalog every document a project might generate. It defines the smallest phase-linked set of controls that keeps cost, schedule, quality, and turnover traceable from the first baseline through final acceptance [9].

A. Unified Phase Map:

Phase sequence used in this paper:

Preconstruction & Planning → Procurement & Buyout → Construction Administration → Cost & Change Management → Quality / Inspections / Compliance → Turnover / Closeout / Post-Construction [2], [3].

Each phase is defined by (1) purpose, (2) PM decision focus, (3) phase-critical controls, and (4) outputs that become controls for the next phase [1]. (Formal artifact definitions are standardized in Section 5)

1) Phase A - Preconstruction and Planning:

a) Purpose: convert intent into a controlled baseline that can be bought, built, and inspected [2].

b)   *PM decision focus:* what is included/excluded, what is assumed, what drives time, what must be protected early.

c)   *Phase-critical controls:*
- Baseline budget (original budget baseline)
- Critical path logic (what truly governs duration)
- Procurement assumptions (early procurement schedule assumptions and long-lead candidates)
- Risk and constructability record (what could break execution if not resolved early) [10]

d)   *Outputs that become next-phase controls:*
- Bid package and trade breakdown strategy
- Long-lead priority list tied to schedule logic
- Scope intent translated into bid-ready boundaries (later expressed through scope sheets)

2)   *Phase B. Procurement and Subcontractor Buyout:*
a)   *Purpose:* convert documents into executed commitments: scope, price, schedule obligations, and risk allocation [4], [7].

b)   *PM decision focus:* award strategy, scope completeness, exclusions, qualifications, commercial and schedule terms.

c)   *Phase-critical controls:*
- RFP packages issued with scope boundaries.
- Scope sheets used to prevent scope gaps.
- Proposal review discipline (coverage, exclusions, inconsistencies)
- Procurement schedule and long-lead register tying awards and orders to schedule reality [22].
- Award authorization step where required by the project's approval structure (for example, "request to award")

d)   Outputs that become next-phase controls:
- Executed trade commitments that define scope boundaries for administration and change
- Procurement commitments (lead times, submittal requirements, fabrication timelines)
- Budget commitments that feed controlled budget revisions

3)   *Phase C - Construction Administration:*
a)   *Purpose*: operate the decision pipeline that keeps field production unblocked while preserving record integrity [2], [9].

b)   *PM decision focus:* prioritization of questions, approval readiness, coordination decisions that prevent rework.

c)   *Phase-critical controls:*
- RFI log with priority, due dates, and distribution discipline
- Submittal register with revision control and approval status
- Coordination cadence (standing routines that resolve constraints before they hit production) [20], [21]
- Document control rule for "current set" and revision visibility [9], [12]

d)   *Outputs that become next-phase controls:*
- Approved submittals that become installation authority
- RFI responses that may trigger change evaluation.
- Coordination decisions documented as actionable direction.

4)   Phase D - Cost and Change Management:

a)   Purpose: convert scope evolution into controlled contractual change while protecting budget integrity and reducing claim exposure [4], [5].

b)   PM decision focus: what is in-scope vs change, pricing discipline, time impact relevance, approval pathway.

c)   Phase-critical controls:
- Change log that treats change as a lifecycle (not a one-time form) [17]
- Budget revisions updated as commitments and approvals occur
- Forecast and contingency discipline (what is being consumed, why, and with what approval)
- Change documentation package that supports approval and later auditability

d)   Outputs that become next-phase controls:
- Approved change authority aligned to field execution.
- Updated baselines (budget and, where applicable, schedule)
- Written record linking physical work to contract authorization.

5)   Phase E - Quality, Inspections, and Compliance:

a)   Purpose: close quality and compliance requirements in a traceable way before they become late-stage emergencies [9].

b)   PM decision focus: required inspections and tests, hold points, closure evidence, correction priority.

c)   Phase-critical controls:
- Inspection and test planning (what must be checked, when, and by whom)
- Inspection log with corrective action ownership and closure evidence
- Evidence discipline (photos, reports, and acceptance records stored consistently)
- Regulatory milestone tracker where occupancy readiness depends on documented approvals

d)   Outputs that become next-phase controls:
- Closed inspection records and acceptance evidence
- Reduced downstream punch volume through earlier closure [14]
- Compliance documentation ready to populate turnover requirements

6)   Phase F - Turnover, Closeout, and Post-Construction:

a)   Purpose: deliver a complete building as a verified package: physical completion plus documentation, training, and warranty readiness [2], [3].

b)   PM decision focus: what constitutes completion, what remains open, and what must be delivered before acceptance/occupancy.

c)   Phase-critical controls:
- Punch list log with ownership, due dates, and verified closure.
- Turnover package index used to confirm completeness of required deliverables.
- Training/commissioning completion records where applicable
- Post-occupancy follow-up plan when included by contract or owner requirements.

d)   Outputs (completion controls):
- Verified handover package aligned to contract requirements [4]
- Objective closure evidence supporting acceptance and occupancy readiness
- Reduced post-turnover disruption through structured follow-up discipline

Key property of the phase map is that phases are not treated as silos. Each phase produces outputs that become enforceable controls for the next phase [9].

B. Cross-phase linking rules:

Most building-project failures concentrate at interfaces, where a decision exists but is not converted into a controlled input for the next workflow step [9], [13]. The framework therefore uses explicit linking rules to force continuity.

1) Baselines precede commitments:
- Control: buyout and major commitments must reconcile to the established budget and schedule logic.
- If broken: commitments drive the baselines, and control becomes reactive.

2) Intent must be transformed into trade boundaries:
- Control: scope intent is converted into bid-ready scope sheets and procurement packages.
- If broken: scope gaps surface later as RFIs, change events, and disputes over responsibility.

3) Procurement is a schedule control:
- Control: the procurement schedule and long-lead register are tied to what governs time.
- If broken: long-lead misses create delays that appear only after recovery options are gone.

4) Approved submittals define installation authority [2], [3]:
- Control: installation and inspection reference the approved submittal set with revision control.
- If broken: wrong-product or wrong-revision installation increases rework and acceptance risk.

5) Every RFI response is screened for change impact:
- Control: RFIs that alter scope, cost, or time are captured as change events immediately.
- If broken: directives accumulate without contract alignment, and costs surface late.

6) Approval gates precede change execution [4], [5], [17]:
- Control: field execution follows written change authority, not informal direction.
- If broken: unapproved work creates claims, payment friction, and baseline instability.

7) Inspection closure feeds turnover readiness [2], [3], [9]:
- Control: closed inspections and test evidence populate turnover requirements continuously.
- If broken: closeout becomes a document scramble rather than a controlled verification step.

8) Records remain usable only if ownership and status are explicit [9]:
- Control: each decision-channel item has an owner, due date, status, and closure evidence standard.
- If broken: decisions exist but cannot be relied on downstream. These linking rules act as the framework's continuity wiring: they prevent the project from "forgetting" decisions as the work moves across phases and parties.

C. Governance model:

The governance premise is simple: workflow speed and defensibility improve when authority is explicit and the recordkeeper role is non-negotiable [4], [9]. The RACI below is written in typical building-project terms and can be adapted to match contract structure.
Role abbreviations
- OWN = Owner / Owner's Representative
- DOR = Designer of Record (architect/engineer)
- PM = General Contractor Project Manager
- PE = General Contractor Project Engineer
- DC = General Contractor Document Controller

- SUPT = General Contractor Superintendent
- SUB = Subcontractor PM/Foreman
- QA = Quality lead (GC or third party, depending on project)

Governance enforcement rule: every RFI, submittal, and change event must have an owner, a due date, and a status in a controlled log [9]. Items that are not logged are not treated as controlled decisions [9].

Table I - III summarizes governance assignments (RACI) for RFIs, submittals, and change events to keep decision ownership explicit and auditable [4]-[6].

TABLE I.   RACI: RFIs [4], [6]

Activity	R	A	C	I
Draft and submit RFI (include exhibits, proposed solution when possible)	SUB	PM	PE/DC, SUPT	QA
Internal screen (completeness, duplication check, priority, due date)	PE/DC	PM	SUPT	SUB
Provide design clarification/response	DOR	DOR	PM (as needed)	PE/DC, SUPT, SUB, QA, OWN (if impact)
Distribute response + update log and current-set record	PE/DC	PM	—	SUPT, SUB, QA, OWN (if impact)
Screen response for change impact (scope/cost/time)	PE/DC	PM	DOR	OWN (as required)

TABLE II.   RACI: Submittals [4], [2]

Activity	R	A	C	I
Prepare and submit submittal package	SUB	SUB	PM/PE (req.)	SUPT, QA
Contractor review (completeness, coordination, compliance)	PE/DC	PM	SUPT, QA	SUB
Designer review/approval	DOR	DOR	PM (as needed)	PE/DC, SUPT, SUB, QA
Owner review (only if contract requires)	OWN	OWN	PM	PE/DC, SUB
Log control, revision control, and distribution	PE/DC	PM	—	SUPT, SUB, QA, OWN (if required)

TABLE III.   RACI: Change Events and Change Orders [4], [5]

Activity	R	A	C	I
Identify change event (directive, condition, request, RFI impact)	SUPT/SUB/PE	PM	—	OWN
Prepare scope narrative + pricing backup	SUB (trade pricing) + PM/PE (compilation)	PM	DOR (intent alignment)	OWN
Review change for design intent and scope consistency	DOR	PM(process owner)	SUB	OWN
Approve change (contract authority)	OWN	OWN	PM	PE/DC, SUPT, SUB
Issue executed change document + update baselines and logs	PE/DC	PM	—	OWN, DOR, SUPT, SUB
Update budget revisions and forecast discipline	PE/DC	PM	—	OWN (as required)

D. Framework principles:

These principles define how the system is run under real schedule pressure. They are implementation rules, not aspirations.

- Single source of truth is enforced: "current set" revision control is treated as a control, not a convenience [9], [12].
- Logs are treated as live controls: each log entry must carry owner, due date, next action, and closure evidence [9].
- Decision speed is managed as measurable risk: backlog and overdue items are tracked and escalated using defined thresholds [10].
- Work release requires prerequisites: approvals and gates are prerequisites for execution, not after-the-fact documentation [4], [5].
- Baselines remain stable unless changed through control: revisions occur through documented approval pathways and are time-stamped [9].
- Controls stay minimum but non-negotiable: artifacts are used as levers that prevent predictable breakdowns, not as paperwork volume.
- Closure is verified, not assumed: inspections, punch, and turnover deliverables close only with objective evidence [9].

V. ARTIFACT DICTIONARY: STANDARDIZED PM CONTROL TOOLKIT

This section defines the minimum viable set of control artifacts used by the proposed framework. Each artifact is treated as a control instrument, not paperwork. Its job is to keep decisions usable over time by making them (1) traceable, (2) enforceable through clear fields and rules, and (3) stable enough to support cost, schedule, quality, and turnover [9].

Artifact definition format:

- Purpose (why it exists)
- Inputs (what it relies on)
- Minimum content (fields that must exist)
- Outputs / use (what it controls)
- Control rule (the non-negotiable condition for reliability)

A. Budget Controls:

1) Original Budget: The initial cost baseline used to plan and control spending [1]. It aligns to the executed owner–contractor contract value at award and is structured in the same way the work will be procured and managed [4].

a) *Purpose*: Establish a stable baseline for commitments, variance tracking, and forecasting discipline.

b) *Inputs:* Contract value, estimate breakdown, bid package plan, early schedule logic, allowances/alternates, major constraints.

c) *Minimum content:*

- Cost categories or codes aligned to procurement packages
- Allowances and alternates stated explicitly
- Contingency definition (authorized uses and approval authority)
- High-impact assumptions and exclusions only

d) *Outputs / use:* Becomes the reference for buyout commitments, budget revisions, and cost-to-complete forecasting.

e) *Control rule:* No commitment is issued unless it is visibly tied to a baseline budget category.

2) Budget Revisions: Controlled updates to budget line items triggered by formal commitments or formally authorized value changes [5].

a) *Purpose:* Keep forecast accuracy aligned to executed commitments rather than informal expectations.

b) *Inputs:* Executed subcontracts, approved change orders, approved allowance usage, verified actual costs.

c) *Minimum content:*

- Revision ID and date

- Line item adjusted and reason code (buyout, CO, allowance, reforecast)
- Dollar delta (+/−)
- Approval reference (who authorized and the document reference)

d)   Outputs / use: Updated forecast and cost-to-complete, early warning for overruns, defensible owner reporting.

e)   Control rule: Every revision must link to an executed subcontract, approved change order, or approved allowance use. No silent adjustments [5], [9].

B. Schedule and Procurement Controls:

1)   Critical Path: The sequence of activities that governs project duration. Delays to critical activities delay completion unless mitigated [1].

a)   Purpose: Establish decision priority so limited time is spent on what controls completion.

b)   Inputs: Baseline schedule logic, durations, procurement lead times, inspection and test requirements.

c)   Minimum content:

- Critical and near-critical activities clearly identified.
- Key logic ties driving completion.
- Milestones tied to turnover or occupancy targets.

d)   Outputs / use: Priority rules for RFIs, submittals, procurement actions, inspections, and change decisions.

e)   *Control rule:* Any item affecting a critical activity must be flagged as schedule-critical in the log it belongs to.

2)   Procurement Schedule: A time-based register that converts schedule logic into procurement actions by defining required award and purchasing dates [2].

a)   Purpose: Prevent long-lead and buyout timing failures from silently controlling the schedule.

b)   Inputs: Critical path milestones, submittal durations, fabrication durations, shipping/logistics constraints, lead-time assumptions.

c)   Minimum content:

- Package name and responsible party.
- Required-on-site date.
- Required submittal date.
- Required award/PO date.
- Lead time and fabrication assumptions.
- Status (bidding, negotiating, awarded, in production, shipped, delivered)

d)   Outputs / use: Buyout sequencing, long-lead visibility, schedule risk reporting.

e)   Control rule: Procurement dates are derived from the construction schedule. They are not guessed independently [2].

C. Procurement and Buyout Artifacts:

1)   Request for Proposal (RFP): A formal request to subcontractors or suppliers to submit technical and commercial proposals for a defined scope [2], [3].

a)   Purpose: Produce comparable proposals and surface scope ambiguity early.

b)   Inputs: Drawings/specifications, scope sheets, phasing constraints, schedule requirements, compliance requirements.

c)   Minimum content:

- Clear scope boundary (inclusions and exclusions)
- Proposal format requirement (pricing structure, alternates, unit rates if needed)
- Schedule and lead-time requirements
- Submittal requirements and review expectations
- Compliance requirements (insurance, bonding when applicable)

d)   Outputs / use: Proposals that can be leveled, clarified, and awarded with reduced downstream scope conflict.

e)   Control rule: "Bid the drawings" alone is not a control. An RFP must include a scope definition tool (scope sheets or equivalent).

2)   Proposal Review: A structured evaluation of received proposals for completeness, scope coverage, exclusions, and schedule risk [2].

a)   Purpose: Prevent award decisions that embed gaps and conflicting assumptions.

b)   Inputs: Proposals, scope sheets, RFP requirements, clarification record, bid tabs.

c)   Minimum content:

- Scope coverage checklist against the scope sheet
- Exclusions and qualifications list
- Clarifications required before award.
- Schedule and lead-time risk flags

d)   Outputs / use: Cleaned proposals, documented award rationale, clarified scope boundaries before commitment.

e)   *Control rule:* No award recommendation proceeds without a documented proposal review and recorded exclusions/clarifications.

3)   Scope Sheet: A trade-specific scope definition list used to align bids and prevent scope gaps [3].

a)   Purpose: Translate design intent into bid-ready, trade-boundary clarity.

b)   Inputs: Drawings/specifications, coordination requirements, project standards, testing and inspection requirements.

c)   Minimum content:

- Inclusions and explicit exclusions
- Interface responsibilities at boundaries
- Submittal requirements
- Testing and inspection responsibilities
- Required closeout deliverables from the trade

d)   Outputs / use: Comparable bids, stronger subcontract scope of work, fewer interface disputes.

e)   Control rule: Scope sheets must be issued consistently across bidders to make leveling meaningful.

4)   Request to Award: A formal request to authorize award of a subcontract or procurement package, used when project structure requires owner authorization before commitment [4], [7].

a)   Purpose: Ensure award decisions are transparent and formally authorized.

b)   Inputs: Bid tab, proposal review summary, clarifications, scope sheet alignment, procurement schedule impact.

c)   Minimum content:

- Recommended firm and scope summary
- Award amount
- Key exclusions and clarifications
- Lead-time and schedule implications
- Risk notes relevant to performance and compliance

d)   Outputs / use: Authorization record supporting contract compliance and auditability.

e)   *Control rule:* When award authorization is required, no commitment is executed before written authorization is obtained [4], [7].

D. Subcontract Package Controls:

1)   Boilerplate - Subcontract General Conditions: Standard terms and general requirements applied across subcontracts [4], [7].

a)   Purpose: Standardize contractual expectations and reduce inconsistent risk exposure.

b)   Inputs: Prime contract flow-down requirements, subcontract template, insurance and bonding requirements.

c)   Minimum content:

- Payment terms and coordination duties
- Schedule and reporting requirements
- Safety and compliance requirements

- Notice provisions, change and claims procedures

d)   *Outputs / use:* Consistent contractual governance across trades.

e)   *Control rule:* Boilerplate must match prime contract flow-down obligations [4].

2)   *Exhibits:* Attachments that convert general terms into project-specific commitments (schedule milestones, pricing structure, reporting requirements) [4], [7].

a)   *Purpose:* Make subcontract requirements executable and measurable.

b)   *Inputs:* Procurement schedule, pricing breakdowns, project standards, required reporting outputs.

c)   *Minimum content:*

- Schedule milestones or required delivery dates (when applicable)
- Pricing breakdown and schedule-of-values rules
- Required reports and submittal formatting expectations.

d)   *Outputs / use:* Enforceable project-specific commitments that reduce ambiguity in execution.

e)   *Control rule:* Negotiated clarifications must be captured in exhibits, not left in emails [4].

3)   *Scope of Work (SOW):* The subcontract section that defines what the subcontractor must furnish and install, including boundaries and required deliverables [4], [7].

a)   *Purpose:* Prevent scope gaps and interface failures that later convert into conflict and change exposure [16], [17].

b)   *Inputs:* Scope sheets, award clarifications, drawings/specifications, coordination requirements.

c)   *Minimum content:*

- Inclusions, exclusions, and boundaries
- Interface responsibilities
- Submittal responsibilities
- Testing and inspection responsibilities
- Closeout deliverables owed by the trade

d)   *Outputs / use:* Clear scope boundary that stabilizes administration, change evaluation, and closeout completeness.

e)   *Control rule:* SOW must be traceable to the issued scope sheet and the final clarification set used at award.

4)   *Certificates of Insurance (COIs):* Evidence that required insurance coverage is active and compliant with contract requirements [4], [7].

a)   *Purpose:* Verify risk transfer before work begins.

b)   *Inputs:* Subcontract insurance requirements, prime contract requirements.

c)   *Minimum content:*

- Coverage types and limits
- Additional insured requirements when applicable
- Policy effective dates and required endorsements.

d)   *Outputs / use:* Authorization to mobilize where insurance compliance is mandatory.

e)   *Control rule:* No mobilization proceeds without verified COI compliance when required by contract [4], [7].

5)   *Bonds (Performance and Payment, when required):*

Financial instruments guaranteeing performance and payment obligations when contractually required [4], [7].

a)   *Purpose:* Reduce default and payment risk exposure.

b)   *Inputs:* Bond requirements, subcontract value thresholds, contract risk posture.

c)   *Minimum content:*

- Bond type, value, surety details, effective date

d)   *Outputs / use:* Compliance record and risk mitigation before full notice-to-proceed.

e)   *Control rule:* When bonding is required, bonds must be in place before full work authorization [4], [7].

E. Construction Administration and Coordination Controls:

1) *Coordination Decision Log:* A controlled record of coordination issues, decisions, and closures. BIM (Building Information Modeling) may be a tool, but the control artifact is the decision record [11], [12].

a) *Purpose:* Reduce rework by resolving interface conflicts before installation [14].

b) *Inputs:* Trade drawings, coordination models (if used), schedule constraints, approved submittals.

c) *Minimum content:*

• Issue description (clash, constraint, interface ambiguity)

• Decision and assigned responsibility

• Due date and closure evidence reference

d) *Outputs / use:* Reduced field conflict, fewer RFIs driven by interface ambiguity.

e) *Control rule:* Coordination issues are managed like RFIs: logged, owned, due-dated, and closed with evidence [9].

2) *RFI Log:* The controlled register tracking each RFI from submission through response, distribution, and closure [6], [18].

a) *Purpose:* Prevent decision latency and preserve traceability of clarifications [9], [18].

b) *Inputs:* Field questions, design ambiguities, coordination conflicts.

c) *Minimum content:*

• RFI ID, title, system/location reference

• Question and proposed solution when appropriate

• Priority and critical-path flag

• Submitted date, due date, answered date

• Distribution record and closure confirmation

d) *Outputs / use:* Visibility into decision pipeline health and structured change screening.

e)   *Control rule:* Every answered RFI is screened for cost, time, and scope impact and linked to the change log when applicable [5], [17].

3) *Submittal Register:* The controlled register tracking each submittal through contractor review, designer review, resubmittal cycles, approval, and release [4].

a) *Purpose:* Protect procurement and installation from approval delays and wrong-version installation [14], [15].

b) *Inputs:* Subcontract submittal requirements, procurement schedule, vendor and trade submittals.

c) *Minimum content:*

• Submittal ID, spec section/system

• Required date tied to procurement schedule.

• Status (in prep, submitted, under review, revise and resubmit, approved)

• Revision history and current approved version reference

d) *Outputs / use:* Procurement release control and evidence trail for quality and turnover documentation.

e)   *Control rule:* Controlled items are not released for procurement without a documented approved submittal status [4].

4) *Document Control - Single Source of Truth:* A controlled approach ensuring the team builds from current, approved information with revision tracking [9].

a) *Purpose:* Prevent wrong work caused by outdated or uncontrolled documents.

b) *Inputs:* Issued drawings/specifications, revisions, approved submittals, RFI responses, field records.

c) *Minimum content:*

• Current set identification and revision history

• Controlled distribution rule

• Record of superseded documents and replacement references

d) *Outputs / use:* Reduced version confusion and defensible record of what governed work at time of execution [9].

e)   *Control rule:* "Current set" must be explicit. Use of uncontrolled documents is treated as a controllable quality and cost risk [9], [14].

F. Change and Quality Controls:

1) *Change Log:* The controlled register tracking change events from identification through pricing, approval, implementation, and baseline update [5], [17].

a) *Purpose:* Prevent change chaos and keep cost and schedule impacts visible and governable [16], [17].

b) *Inputs:* Impact RFIs, owner directives, field conditions, design changes, value proposals.

c) *Minimum content:*

- Change event ID, origin, scope narrative
- Status (pricing, review, approved, implemented, closed)
- Cost impact and schedule impact when applicable
- Approval reference and approval date
- Link to the related budget revision entry [5], [9].

d) *Outputs / use:* Controlled change lifecycle, reliable forecasting, reduced late-claim exposure.

e) *Control rule:* Every executed change must have an approval reference and a linked budget revision entry.

2) *Inspection and Test Plans (ITPs):* A structured plan defining required inspections and tests, acceptance criteria, hold points, responsible parties, and closure evidence [9].

a) *Purpose:* Reduce defects and close compliance systematically rather than reactively.

b) *Inputs:* Code requirements, specifications, product approvals, project quality plan, third-party requirements.

c) *Minimum content:*

- Inspection or test item and acceptance criteria
- Hold points and responsible parties
- Required closure evidence (reports, photos, sign-offs)
- Corrective action and verification method

d) *Outputs / use:* Early defect capture and documented acceptance evidence for turnover readiness.

e) *Control rule:* Inspections are managed as closure workflows with evidence, not as isolated events.

G. Closeout Controls:

1) *Closeout Index: Turnover Package Index:* A structured inventory of required turnover deliverables (as-builts, O&M manuals, warranties, training records, test reports, and closeout forms) with ownership and status tracking [2], [3], [4].

a) *Purpose:* Prevent closeout delays caused by missing documents and unclear ownership.

b) *Inputs:* Subcontract closeout requirements, specifications, commissioning requirements, inspection closure records, approved submittals.

c) *Minimum content:*

- Deliverable category and description
- Responsible party (trade, vendor, contractor)
- Due date, status, and closure evidence reference
- Storage location reference for retrieval

d) *Outputs / use:* Complete turnover package aligned to contract requirements and reduced post-turnover disruption.

e) *Control rule:* The closeout index is started early and maintained continuously. It is not created at the end [2], [9].

This dictionary is designed to be portable. It can be implemented using spreadsheets and shared repositories, or within a PMIS platform with workflow automation. The tool is not the framework. The framework is the discipline: artifacts with minimum fields, enforced governance, and control rules that keep decisions stable and usable from baseline through turnover [9], [13].

VI. WORKFLOW MODULES (A–F)

These modules translate the framework into repeatable operating workflows [1], [2]. Each module is written as: Purpose → Entry inputs → PM workflow steps → Key artifacts used → Exit outputs / handoff. The intent is execution. A PM should be able to run these modules without improvising the fundamentals.

A.   Module A - Preconstruction & Planning:

1)   Purpose: Convert intent into a buildable plan by removing uncertainty early, when it is cheapest to remove [14].

2)   Entry inputs: Owner objectives and constraints, progressive design documents, site/logistics constraints, contract requirements, permitting/compliance needs, target milestones and budget intent.

3)   PM workflow (core steps):

- Define scope boundaries early (in/out, assumptions, constraints, phasing, decision priorities).
- Establish the baseline logic for time and identify schedule drivers that must be protected.
- Translate time logic into a procurement path: what must be awarded and approved early to prevent schedule loss.
- Perform constructability screening focused on buildability, access, sequencing, and inspection readiness, not aesthetics.
- Run value evaluations using explicit rules: lifecycle value, lead time risk, install complexity, and downstream coordination exposure.
- Build a risk register with owners and response actions, prioritizing items that can break procurement, inspections, or turnover [10].
- Define quality and compliance intent as planned closure: what evidence will be required later and how it will be captured from the start [9].
- Establish the decision governance stance for the project team: which channels are used, how decisions are logged, and how escalation works [4], [9].

4)   Key artifacts used (defined in Section V): Original Budget; Critical Path; Procurement Schedule; constructability/decision record; risk register; early closeout requirements list.

5)   Exit outputs / handoff: A controlled baseline (cost + time logic), procurement priorities aligned to schedule reality, scope boundaries ready for bid packaging, and a decision-ready setup that feeds Buyout.

B.   Module B - Procurement & Subcontractor Buyout:

1)   Purpose: Convert plans into enforceable commitments without embedding scope gaps, schedule misses, or commercial ambiguity [4], [7].

2)   Entry inputs: Baseline controls from Module A, current design documents, bid packages, procurement schedule requirements, qualification and compliance requirements.

3)   PM workflow (core steps):

- Structure bid packages to prevent trade boundary gaps and avoid overlapping responsibility.
- Issue RFPs with standardized scope definition and consistent proposal requirements [2], [3].
- Control bidder clarifications so all bidders price the same scope reality.
- Level bids using a scope matrix to normalize exclusions, alternates, and assumptions.
- Perform proposal completeness checks with explicit flags for lead-time and schedule exposure.
- Negotiate to close gaps and lock the final scope boundary and commercial rules that govern future changes.
- Execute long-lead actions early where schedule logic demands it, including submittal timing expectations [22].
- Obtain award authorization when the contract structure requires it before commitment [4], [7].
- Execute the subcontract package in a way that preserves what was negotiated and can be enforced.
- Conduct a handoff briefing that transfers scope boundaries, schedule commitments, and documentation expectations to the execution team.

4)   Key artifacts used (defined in Section V): RFP; Scope Sheets; Proposal Review; clarifications record; Procurement Schedule; Request to Award (when required); subcontract package instruments; COIs/bonds (when required).

5)   Exit outputs / handoff: Executed scopes aligned to the budget baseline, procurement dates protected, long-leads stabilized, and enforceable trade boundaries that reduce downstream decision churn.

C.   Module C - Construction Administration:

1)   Purpose: Operate the project's decision pipeline so questions convert into approved direction that can be built without wrong-version work or decision latency [9], [18].

2)   Entry inputs: Issued-for-construction information (current set), executed trade commitments, procurement schedule targets, communication and governance requirements.

3)   PM workflow (core steps):

- Stand up the required registers immediately and define ownership and due-date rules.
- Apply schedule-priority discipline: decision items that affect schedule drivers are flagged and managed as time risk.
- Enforce RFI quality at entry: context, location/system clarity, and completeness before submission.
- Control the RFI cycle: submit, track, escalate when overdue, distribute, and confirm downstream closure [6], [18].
- Control the submittal cycle: contractor review discipline first, then designer review, with resubmittal loops managed to closure [2], [4].
- Release for procurement/installation only when the controlling approval status is satisfied.
- Run coordination as a closure workflow: issues are logged, assigned, due-dated, and verified closed before installation.
- Enforce document control as a field safety and cost control: "current set" is explicit and version drift is treated as a defect precursor [9].
- Route impacts correctly: decisions with cost/time/scope impact are captured immediately as change events (not left as informal direction) [5], [17].

4)   Key artifacts used (defined in Section V; governance per Section IV): RFI Log; Submittal Register; coordination decision log; document control mechanism; meeting decision record; transmittal record.

5)   Exit outputs / handoff: Approved direction that is buildable and current, controlled releases for procurement/installation, closed coordination constraints, and decision records that feed Change/Cost control and Quality closure.

D.   Module D - Cost & Change Management:

1)   Purpose: Run change as a disciplined lifecycle so field execution stays aligned to contractual authority and the budget forecast stays credible [4], [5], [17].

2)   Entry inputs: Contract change requirements, executed trade commitments, RFI/submittal outputs, documented directives/conditions, baseline budget and schedule logic.

3)   PM workflow (core steps):

- Log change events immediately upon identification, even before pricing, to prevent invisible exposure [17].
- Classify each change event so scope and responsibility are evaluated consistently.
- Perform impact analysis beyond direct cost: schedule effect, downstream trades, and inspection/turnover implications.
- Request pricing with a clear scope narrative and required backup so review is objective.
- Negotiate and resolve scope disputes before approval so execution does not run ahead of authority.
- Enforce authorization gates: the project does not rely on memory-based direction where approval is required [4], [5].
- Maintain status visibility and aging discipline so change does not accumulate into late-stage budget shocks [16], [17].
- Upon approval, update the budget controls and communicate what is authorized for field execution [5], [9].
- Close the loop by identifying whether the change originated from a preventable workflow failure and reinforce the relevant control.

4)   Key artifacts used (defined in Section V; governance per Section IV): Change Log; pricing backup package; approved change authorization record; Budget Revisions; forecast/contingency tracking; correspondence record for traceability.

5)   Exit outputs / handoff: Approved change authority aligned to field execution, updated budget commitments and forecast accuracy, reduced claim exposure, and transparent cost-to-complete.

E.   Module E - Quality, Inspections & Compliance

1)   Purpose: Convert "installed work" into "accepted work" by running planned inspection closures with retrievable evidence [9].

2)   Entry inputs: Quality and compliance requirements, approved submittals, inspection/testing requirements, schedule milestones tied to readiness, identified risk areas.

3)   PM workflow (core steps):

• Establish the inspection strategy as an execution plan: what must be checked, when, and what closes it.

• Translate requirements into inspection points and hold points with clear acceptance criteria and evidence needs [9].

• Execute first-install verification to correct early and prevent repetitive defects at scale [14].

• Coordinate third-party and authority inspections through readiness, access, and documentation controls [3].

• Track observations as closure items: assign ownership, due dates, re-verification, and evidence capture.

• Link inspection outcomes to the governing approved information so acceptance is defensible [4], [9].

• Elevate occupancy-critical inspections and tests as schedule-critical closure items, not late-stage surprises.

4)   Key artifacts used (defined in Section V): ITPs/checklists; Inspection Log; observation/closure record (as used); evidence index; readiness tracker for regulatory milestones.

5)   Exit outputs / handoff: Closed inspections with retrievable evidence, reduced rework through earlier defect capture, and compliance records ready to populate turnover requirements [9], [14].

F.   Module F - Turnover, Closeout & Post-Construction

1)   Purpose: Deliver a complete, operable building package and close the contract without last-minute documentation collapse [2], [4].

2)   Entry inputs: Completed work, approved submittals, inspection closures, outstanding change items, contract closeout requirements, training/commissioning requirements when applicable.

3)   PM workflow (core steps):

• Maintain a live Closeout Index early enough that missing deliverables are detected while the team still has time [2], [9].

• Run punch as a managed closure system with ownership, deadlines, and verified close criteria.

• Govern as-builts as controlled outputs tied to approved changes and clarifications, not end-stage reconstruction [5], [9].

• Collect O&M manuals and warranties with completeness checks and indexing for retrieval.

• Plan and document training as a deliverable with attendance and sign-off evidence.

• Reconcile commercial closure items (final changes, releases, closeout requirements) without deferring unresolved issues [4], [5].

• Where applicable and contractually aligned, execute structured early-occupancy follow-up so performance stabilizes and warranty noise is minimized.

4)   Key artifacts used (defined in Section V): Closeout Index; punch list log; as-built governance record; O&M register; training record; warranty log; commissioning/acceptance records (when applicable); final change reconciliation.

5)   Exit outputs / handoff: Indexed turnover package delivered, training completed, warranties organized, punch closures verified, and acceptance/occupancy readiness protected [2], [4].

VII. FAILURE MODES AND RELIABILITY CONTROLS

This section converts the proposed workflow into a reliability system by defining predictable breakdowns as traceability breaks: points where downstream work proceeds without a stable controlling record (approved RFI/submittal/change) or without verifiable closure evidence (inspection/test/punch). Controls are written as minimum enforceable rules that can be audited through logs/registers, timestamps, approvals, and closure evidence [9].

A. Definition used in this paper

A traceability break occurs when (1) the controlling decision is missing, late, superseded, or ambiguous, or (2) closure evidence is incomplete or not retrievable [9]. Each traceability break is paired with a minimum control pack: the smallest set of enforceable artifacts and rules that contains the break and prevents recurrence [9].

B. How to read the matrices (Tables IV–V)

Each row identifies:

• the traceability break (what fails),

- the primary origin module(s) (A–F),
- a leading signal that reveals the break early (audit-friendly), and
- the minimum control pack that contains it (artifacts from Section 5, executed through Module routines in Section 6) [9].

TABLE IV.   TRACEABILITY BREAKS AND MINIMUM CONTROL PACKS (MODULES A–C) [2], [9], [18]

#	Traceability break (what fails)	Primary origin (Module)	Leading signal (audit-friendly)	Minimum control pack (enforceable)
1	Boundary ambiguity (trade gaps/overlaps survive buyout)	B	Repeated "scope boundary" RFIs; recurring scope disputes at same interface	Scope Sheets + scope matrix used for all bidders; Proposal Review includes interface check; award gate: no award with unresolved exclusions/interfaces; subcontract SOW must include boundary responsibilities; "boundary-walk" meeting note filed
2	Schedule-to-procurement disconnect (critical work not protected by award/PO dates)	A–B	"Required-on-site" dates slip without recovery plan; long-lead items discovered late	Procurement Schedule derived from CPM logic; long-lead subset flagged; required submittal-by and award/PO-by dates recorded; weekly procurement checkpoint; escalation rule: award/PO slips beyond defined threshold trigger escalation to decision authority
3	Assumption drift at award (price accepted but scope assumptions unstable)	B	Post-award clarification surge; early change events cluster in same packages	Proposal Review must close clarifications list before award; award memo/Request to Award records assumptions; subcontract exhibits capture unit-rate/allowance rules where applicable; award gate: "no unresolved clarifications" requirement
4	RFI queue saturation (questions exist but answers arrive too late to protect production)	C	RFI aging trend rising; critical activities waiting for clarification	RFI Log with owner + due date + priority + critical-path flag; RFI quality standard (location/system + clear question + exhibit + proposed solution when appropriate); weekly RFI aging review; escalation rule: schedule-critical RFIs overdue beyond threshold escalate to decision authority
5	Submittal rework loop (revise/resubmit cycles delay procurement and create churn)	C	High resubmittal rate; missing information recurring across same trade	Submittal Register includes required-by dates tied to Procurement Schedule; contractor completeness checklist before designer review; resubmittal reason codes tracked; weekly backlog review; release gate: no procurement/installation authority without current approved submittal
6	Version conflict (teams build from superseded information)	C	Field references "older set" emails/screenshots; inspection finds mismatch vs approved documents	Document Control as Single Source of Truth with explicit "current set"; controlled distribution/transmittals; superseded docs withdrawal practice; install authority rule: only current approved submittal/RFI response governs work; field confirmation step for high-risk revisions

TABLE V.   TRACEABILITY BREAKS AND MINIMUM CONTROL PACKS (MODULES D–F) [4], [5], [9], [17]

#	Traceability break (what fails)	Primary origin (Module)	Leading signal (audit-friendly)	Minimum control pack (enforceable)
7	Change screening failure (impactful RFI/submittal does not become a change event)	C–D	Extra work starts with no change ID; later pricing becomes disputed	Rule: every scope/time/cost-impact RFI/submittal triggers change screening; Change Log entry created at identification (even pre-pricing); scope narrative required; correspondence capture standard; weekly change review includes "unlogged impact" check
8	Unauthorized change execution ("do now, paper later")	D	Work installed without written authorization reference; pay apps contested	No-go gate: extra work requires documented authorization unless contract-defined emergency notice applies; field directive form (or CCD-style equivalent where allowed) logged; change ID required before billing; budget revision linkage mandatory after approval
9	Change aging and decision blockage (priced changes stall, harming schedule/cash flow)	D	Large "pending approval" queue; aging beyond thresholds; stacking of unresolved COs	Change Log aging buckets (e.g., 0–14/15–30/31+ days or project-defined); decision-ready change package standard (scope + backup + time impact when applicable); weekly decision cadence; escalation thresholds tied to aging and critical path exposure
10	Inspection closure drift (work done but acceptance evidence missing)	E	Inspections passed but evidence not retrievable; repeat inspection requests	ITPs define hold points + closure evidence; Inspection Log with owner + corrective action + verification evidence; pre-inspection readiness checklist; compliance items on occupancy path flagged schedule-critical; escalation for missing closure evidence
11	Closeout index failure (turnover deliverables not built progressively)	F	O&M/as-builts missing late; closeout becomes end-phase scramble	Closeout Index started early with owners/due dates/storage location; rolling closeout submittals by trade; monthly closeout readiness review starting mid-project; turnover completeness threshold gate defined for substantial completion
12	Punch re-open and demobilization drift (items "closed" but fail verification)	F	Re-open rate rising; trades demobilize with critical items open	Punch Log with categories (life-safety/functional/cosmetic) + owners + due dates; verification protocol (who confirms closure); re-open reason code; demobilization gate (no demobilize with open critical categories); weekly punch cadence with escalation for repeat offenders

C. What Section VII contributes (research value)

Section VII converts the proposed workflow into a reliability system by defining "traceability breaks" and mapping each break to (1) its origin module, (2) an early leading signal, and (3) a minimum enforceable control pack [9]. The contribution is that workflow fragmentation is operationalized into auditable failure signatures, not generic advice [9], [13]. Each control pack is written as a verifiable mechanism (log entry, approval gate,

cadence checkpoint, or closure evidence rule), enabling teams to detect breakdowns early and prevent recurrence with minimum overhead [9].

VIII.   DISCUSSION

This section interprets the framework as a reliability system, not a collection of "good PM habits." The central claim is that building projects become more dependable when management actions are converted into verifiable state changes: a decision becomes binding, a scope becomes owned, a version becomes current, an inspection becomes closed, and a turnover item becomes retrievable [9], [11]. The framework improves outcomes because it reduces the project's ability to operate on ambiguity [9].

A. Why this framework improves reliability

Reliability in building delivery is mostly an information problem disguised as a production problem [13]. Work fails downstream when upstream intent is not translated into enforceable constraints, and when decisions do not persist as "current, auditable truth" [9], [11]. The framework improves reliability by hardening three failure-prone interfaces: handoffs between phases, handoffs between roles, and handoffs between versions [1], [9].

1)   Phase-to-phase handoffs are converted from "knowledge" into constraints: Many breakdowns are not caused by poor execution inside a phase. They emerge when phase outputs remain descriptive instead of becoming the next phase's operating limits. Examples include:

* planning assumptions that remain unstated rather than becoming bid-level scope boundaries,
* lead times treated as vendor facts rather than schedule-derived "award-by" commitments,
* responses issued as text but not converted into cost/time screening actions,
* inspections completed as events but not converted into indexed acceptance evidence.

The framework's integration logic prevents phase outputs from evaporating by requiring each phase to export a control-ready artifact that the next phase must consume [9]. In effect, the project acquires continuity: planning produces baselines that procurement must honor, procurement produces commitments that administration must govern, administration produces decisions that change control must authorize, and quality produces evidence that closeout can deliver [2], [4]. The result is a workflow that is less dependent on memory, personalities, and informal continuity.

2)   Governance is treated as a throughput constraint, not an org chart: In most building projects, the most damaging delays are not physical. They are decision delays. When ownership and authority are vague, the system develops predictable symptoms: unanswered questions accumulate, work starts under informal direction, and changes are priced late or disputed [17], [18]. The framework addresses this by making governance operational:

* every decision object (RFI, submittal, change event, inspection closure) is a tracked unit with an owner, due date, status, and closure requirement [9],
* accountability is declared through role logic (RACI) rather than assumed through titles,
* escalation is defined as a threshold rule, not a social negotiation.

This shifts governance from "communication etiquette" to a controllable pipeline [9]. Reliability improves because decisions arrive before they are needed, rather than after the work has already adapted around the missing decision.

3)   Authoritative project information reduces wrong-work more than "better effort" does: Wrong-work rarely happens because teams are careless. It happens because the project allows multiple competing truths: superseded drawings, outdated submittals, informal meeting outcomes, and email-based direction [9], [11]. The framework treats the "current binding set" as a risk control:

* approvals define installation authority,
* direction is linked to downstream screening,
* revision control preserves defensible traceability.

This reduces rework not by adding bureaucracy, but by shrinking the space in which outdated information can survive [9], [14].

4)   The framework's differentiator: enforceable control packs instead of advice: Many PM references describe what teams should do. Section VII is the conversion layer that defines what teams must be able to prove. A control pack is intentionally minimal, but enforceable: it includes at least one verifiable element (a

log entry, an approval gate, a cadence checkpoint, or a closure-evidence rule) [9]. This is what makes the framework operational: it does not rely on "strong PMs" to remember best practices. It designs failure resistance into the workflow itself [9].

B. Fit with Lean/IPD/Last Planner

A practical way to position the relationship is: Lean/Last Planner/IPD improve the production engine; this framework stabilizes the control plane that feeds it. In building work, production reliability collapses when the field is starved of decisions, approvals, and current information [13], [18]. That starvation is administrative and governance-driven, not craft-driven.

1) *Lean alignment (constraint visibility and dependable handoffs):* Lean emphasizes reducing variability and removing constraints before they block work [19], [21]. The framework supports that intent by making constraints measurable and actionable through required registers (procurement, RFIs, submittals, coordination issues, changes, inspections, closeout) [21], [22]. Lean tools are amplified when constraint removal is not an informal effort but a governed workflow with owners and thresholds.

2) *Last Planner alignment (commitments based on readiness, not optimism):* Lookahead planning only works when the "make-ready" process is real [20], [21]. The framework complements Last Planner by supplying readiness gates [20], [21]: commitments are made only when enabling information exists (approved submittals, resolved questions, authorized change direction, cleared constraints). This reduces the classic failure mode of making promises on work that is not actually executable.

3) *IPD alignment (Integrated Project Delivery):* IPD can reduce disputes by aligning incentives and promoting early collaboration, but collaboration does not automatically create clarity. The framework is compatible because it formalizes traceability and decision authority [9]. It functions with or without IPD because it does not depend on contract philosophy. It depends on auditable governance and phase-linked controls.

Key takeaway: delivery models can improve collaboration and production planning, but they do not automatically prevent decision backlog, version drift, or undocumented direction [9], [17], [18]. This framework targets those specific breakdowns with enforceable mechanics.

C. Practical adoption challenges

Adoption difficulty is rarely caused by lack of software. It is caused by behavioral enforcement costs and role-load imbalance.

1) *"We have logs" without enforcement:* Teams often maintain registers but treat them as reporting, not control [9]. Gates are bypassed; status is stale; closure evidence is not demanded. The mitigation is not more documentation. It is minimum viable enforcement:
- select a small non-negotiable set of controls (highest ROI first),
- assign true ownership to each control object,
- define escalation thresholds so backlogs cannot grow silently,
- require closure evidence as a completion condition, not a preference.

2) *Decision channel overload and unclear authority:* Many projects allow broad initiation of RFIs and changes, but concentrate approval into a narrow bottleneck, producing predictable aging [17], [18]. Mitigation requires structural clarity:
- explicit RACI for the decision objects (RFI, submittal, change event, inspection closure),
- priority rules tied to schedule exposure (critical path or near-critical),
- fast-track paths for schedule-critical decisions while maintaining traceability.

3) *Low document-control maturity:* When teams operate from scattered files and informal distribution, the project will build from the wrong set regardless of intent [9], [12]. Mitigation can be lightweight but strict:
- define what qualifies as the current binding set,
- require links between approvals and installation authority,
- require controlled distribution and withdrawal of superseded content,
- maintain a decision record so "truth" does not live only in inboxes.

4) *Resistance to "more process":* Controls can be perceived as paperwork unless the value is concrete. The mitigation is to implement controls as failure-prevention, not compliance theater:

- start with controls that prevent expensive recurrence (procurement timing discipline, decision aging control, change authorization gates, inspection closure evidence) [14], [16],
- demonstrate avoided failure modes using the Section VII mapping,
- expand only after the minimum set is running reliably.

5) Contract and owner constraints: Authorization requirements vary by contract, but the framework can be calibrated [4], [7]:

- align approval gates and notice rules at startup [4], [5],
- flow down governance and closeout requirements into subcontract exhibits early,
- treat the framework as a template that is configured once, then executed consistently.

The fastest path is "progressive hardening." Start with visibility (logs that are truly current), then introduce gates (no-go rules for irreversible actions), then cadence (routine backlog reviews), then escalation (threshold-based intervention) [9], [10]. This sequence reduces friction while still moving toward enforceable reliability. The framework improves building-project reliability by converting cross-phase intent into enforceable controls, converting governance into a managed decision pipeline, and converting approvals and inspections into retrievable, current, audit-ready truth; Lean/Last Planner/IPD amplify the system when they operate on top of this structured control spine, and adoption succeeds when minimum viable controls are enforced with clear ownership and threshold-based escalation [9], [20].

IX. CONCLUSION

This paper presented a standardized, building-focused workflow framework that treats project management as a controlled decision system rather than a collection of disconnected responsibilities [1], [2]. The framework connects lifecycle phases, a minimum viable control toolkit, explicit governance for high-risk decision channels, and failure-mode-based reliability controls into one operating model [9].

The practical implication is straightforward: when commitments are made traceable through controlled artifacts and enforced gates, projects reduce the probability of delay, rework, uncontrolled change growth, and late-stage closeout instability [14]-[17]. Because the framework is defined as an execution system (not a software choice and not a theory label), it can be implemented across delivery methods by calibrating authority assignments and approval thresholds to the governing contract [4], [7], [8].

The contribution is an audit-friendly workflow structure that can be taught, repeated, and evaluated using observable artifacts and closure evidence [9]. Future validation should test implementation across multiple building projects using workflow health metrics (decision cycle time, backlog aging, long-lead adherence, inspection closure rates, and closeout completeness) to quantify performance impact [18].

ACKNOWLEDGMENT

REFERENCES:

[1] Project Management Institute, A Guide to the Project Management Body of Knowledge (PMBOK® Guide), 7th ed. Newtown Square, PA, USA: PMI, 2021.

[2] Project Management Institute, Construction Extension to the PMBOK® Guide, 3rd ed. Newtown Square, PA, USA: PMI, 2016.

[3] Construction Management Association of America, Construction Management Standards of Practice. Vienna, VA, USA: CMAA, 2015.

[4] The American Institute of Architects, AIA Document A201™–2017: General Conditions of the Contract for Construction. Washington, DC, USA: AIA, 2017.

[5] The American Institute of Architects, AIA Document G701™–2017: Change Order. Washington, DC, USA: AIA, 2017.

[6]   The American Institute of Architects, AIA Document G716™–2004: Request for Information (RFI). Washington, DC, USA: AIA, 2004.

[7]   ConsensusDocs, ConsensusDocs 200: Standard Agreement and General Conditions Between Owner and Constructor (Lump Sum). Alexandria, VA, USA: ConsensusDocs, 2017.

[8]   Engineers Joint Contract Documents Committee, EJCDC C-700: Standard General Conditions of the Construction Contract. Washington, DC, USA: EJCDC, 2018.

[9]   International Organization for Standardization, ISO 9001:2015, Quality Management Systems—Requirements. Geneva, Switzerland: ISO, 2015.

[10] International Organization for Standardization, ISO 31000:2018, Risk Management—Guidelines. Geneva, Switzerland: ISO, 2018.

[11] International Organization for Standardization, ISO 19650-1:2018, Organization and Digitization of Information about Buildings and Civil Engineering Works, Including BIM—Information Management Using BIM—Part 1: Concepts and Principles. Geneva, Switzerland: ISO, 2018.

[12] International Organization for Standardization, ISO 19650-2:2018, Organization and Digitization of Information about Buildings and Civil Engineering Works, Including BIM—Information Management Using BIM—Part 2: Delivery Phase of the Assets. Geneva, Switzerland: ISO, 2018.

[13] M. P. Gallaher, A. C. O'Connor, J. L. Dettbarn, Jr., and L. T. Gilday, Cost Analysis of Inadequate Interoperability in the U.S. Capital Facilities Industry, NIST GCR 04-867. Gaithersburg, MD, USA: National Institute of Standards and Technology, 2004.

[14] P. E. D. Love, Z. Irani, and D. J. Edwards, A rework reduction model for construction projects, Construction Management and Economics, vol. 18, no. 4, pp. 479–490, 2000, doi: 10.1080/01446190050024897.

[15] S. A. Assaf and S. Al-Hejji, Causes of delay in large construction projects, International Journal of Project Management, vol. 24, no. 4, pp. 349–357, 2006, doi: 10.1016/j.ijproman.2005.11.007.

[16] A. S. Hanna, J. S. Russell, L. E. Nordheim, and M. J. Bruggink, Impact of change orders on labor efficiency for electrical construction, Journal of Construction Engineering and Management, vol. 128, no. 4, pp. 283–291, 2002, doi: 10.1061/(ASCE)0733-9364(2002)128:4(283).

[17] W. Ibbs, S. Wong, and Y. H. Kwak, Project change management system, Journal of Management in Engineering, vol. 17, no. 3, pp. 159–165, 2001, doi: 10.1061/(ASCE)0742-597X(2001)17:3(159).

[18] A. S. Hanna, E. J. Tadt, and G. C. Whited, Request for Information: Benchmarks and Metrics for Major Highway Projects, J. Constr. Eng. Manage., vol. 138, no. 12, pp. 1347–1352, 2012, doi: 10.1061/(ASCE)CO.1943-7862.0000554.

[19] L. Koskela, Application of the New Production Philosophy to Construction, Technical Report No. 72. Espoo, Finland: VTT Technical Research Centre of Finland, 1992.

[20] G. Ballard, The Last Planner System of Production Control. Ph.D. dissertation, School of Civil Engineering, University of Birmingham, Birmingham, U.K., 2000.

[21] G. Ballard and G. Howell, Shielding production: essential step in production control, in Proc. 6th Ann. Conf. Int. Group for Lean Construction (IGLC), 1998.

[22] R. Vrijhoef and L. Koskela, The four roles of supply chain management in construction, in Proc. 8th Ann. Conf. Int. Group for Lean Construction (IGLC), 2000.

[23] AACE International, Recommended Practice 29R-03: Forensic Schedule Analysis. Morgantown, WV, USA: AACE International, (latest rev.).

[24] American Society for Testing and Materials, ASTM E1105, Standard Test Method for Field Determination of Water Penetration of Installed Exterior Windows, Skylights, Doors, and Curtain Walls by Uniform or Cyclic Static Air Pressure Difference. West Conshohocken, PA, USA: ASTM International, (latest rev.).

[25] American Architectural Manufacturers Association, AAMA 502, Voluntary Specification for Field Testing of Newly Installed Fenestration Products. Schaumburg, IL, USA: AAMA, (latest rev.).

[26] U.S. Army Corps of Engineers, EM 385-1-1, Safety and Health Requirements Manual. Washington, DC, USA: USACE, (latest rev.).

[27] International Organization for Standardization, ISO 21500:2021, Project, Programme and Portfolio Management—Guidance on Project Management. Geneva, Switzerland: ISO, 2021.

[28] Project Management Institute, Pulse of the Profession (latest annual report). Newtown Square, PA, USA: PMI, (latest ed.).