

Chapter 3 - RESEARCH METHODOLOGY

3.1 Introduction

Building directly upon the **integrated framework synthesized at the conclusion of Chapter 2**, this chapter details the methodology used to design, develop, and evaluate the AI-enhanced document and contract governance system. This research operationalizes that conceptual model by employing a methodology guided by **Design Science Research (DSR)** principles.

The methodological framework translates the conceptual blueprint into a functional artifact. It integrates three complementary perspectives.

1. **Research Logic:** The DSR paradigm provides an overarching structure for the systematic development and evaluation of the artifact in order to solve real-world challenges at SGU.
2. **Architectural Design:** The TOGAF Architecture Development Method (ADM) aligns the system's architecture with SGU's strategic and operational requirements.
3. **Technical Development:** An iterative/agile software development life cycle (SDLC) guides the hands-on development of the proof of concept (PoC) prototype.

This multi-layered approach ensures that the research is theoretically grounded, methodologically rigorous, and executed with practicality.

Throughout this process, the methodology incorporates **ISO 15489** and **ISO 23081** standards to establish robust record and metadata governance, as well as **COBIT 2019** to reinforce IT controls. The technical implementation leverages SGU's existing **Google Workspace for Education Plus** infrastructure, which is enhanced with the **Google Vision API, Vertex AI, and Gemini AI**. These technologies ensure that the

final solution is technologically feasible, aligned with governance principles, and tailored to the institutional context.

The remainder of this chapter is structured as follows:

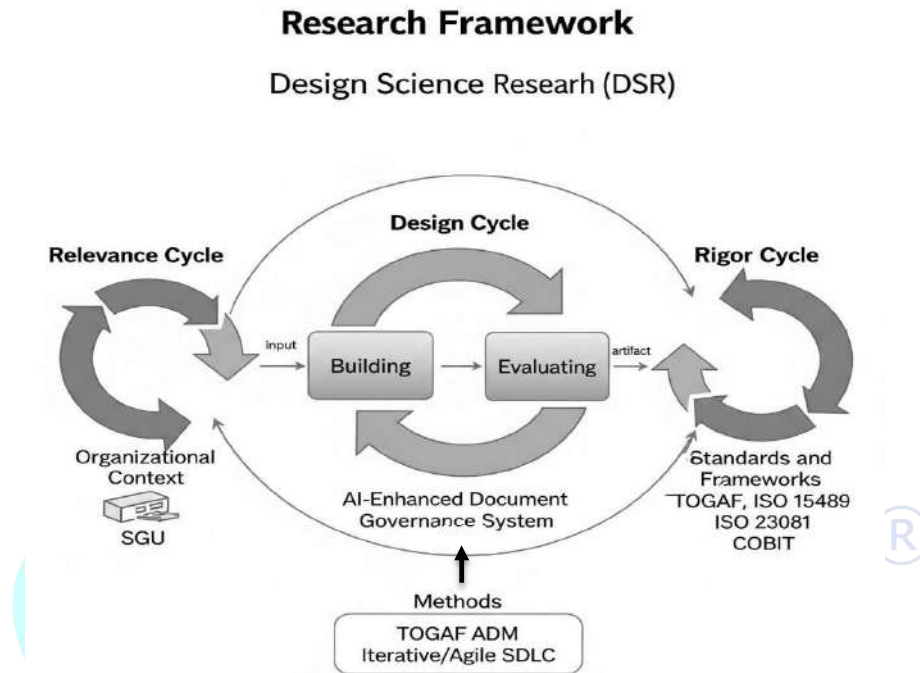
- Section 3.2 presents the research framework and its layered integration of DSR, TOGAF, and SDLC.
- Section 3.3 describes the process of identifying problems and eliciting requirements.
- Section 3.4 and 3.5 details the design and development activities, including system architecture, prototyping, and AI integration.
- Section 3.6 outlines the evaluation and validation strategy, aligning success criteria with the frameworks and standards discussed in Chapter 2.
- Section 3.7 discusses the ethical considerations guiding the research.

3.2 Research Framework

This study's research framework integrates Design Science Research (DSR), the Open Group Architecture Framework's (TOGAF) Architecture Development Method (ADM), and an iterative/agile software development life cycle (SDLC) into a layered, complementary methodology. This combination ensures that the design and implementation of the AI-enhanced document and contract governance system at Swiss German University (SGU) is grounded in academic rigor and practical feasibility.

As explain in previous Section 2.12, Figure 3.1 illustrates the step-by-step integration of these methodological components: Research Framework. The diagram shows how the Relevance Cycle of DSR is anchored in SGU's institutional context and feeds into the Design Cycle. The Design Cycle combines TOGAF ADM Phases B, C, and D with an iterative Agile SDLC prototyping process. The Rigor Cycle operates in parallel to ensure that internationally recognized frameworks and standards, such as ISO 15489, ISO 23081, and COBIT, are systematically embedded into the architectural design and development processes. Each element of the diagram represents a methodological stage or influence elaborated on in subsequent subsections, providing a coherent flow from

problem context to prototype evaluation. The detailed diagram for each cycle interaction illustrated in Figure 3.2 and Figure 3.3.



3.2.1 Overarching Research Logic – Design Science Research (DSR)

This study adopts Hevner’s Design Science Research (DSR) framework as its overarching research logic. The framework operates through three interrelated cycles—**Relevance, Rigor, and Design** (see Figure 2.5)—to ensure that the research is firmly grounded in the needs of stakeholders, adheres to sound methodology, and contributes to existing knowledge.

The Rigor Cycle embeds established frameworks and standards throughout the development process to ensure methodological robustness and technical quality. For this study, it involves drawing upon the knowledge base justified in Chapter 2. These include ISO 15489 for ensuring the authenticity, reliability, and usability of records; ISO 23081 for metadata design and management; and COBIT for IT governance, access control, and compliance monitoring. Additionally, the cycle incorporates cutting-edge,

cloud-based AI tools, such as Google Vision API for optical character recognition (OCR), Vertex AI for intelligent classification, and Gemini AI for semantic search, to enhance automation, compliance, and retrieval efficiency.

The Relevance Cycle is anchored in the operational realities of Swiss German University. Document fragmentation, poor retrieval efficiency, and the absence of contract deadline monitoring have been identified as persistent issues in these departments. This cycle ensures continuous stakeholder engagement, validating that the evolving artifact aligns with the institution's needs and operational constraints.

The Design Cycle — The design cycle involves the creation, iterative refinement, and evaluation of the PoC. TOGAF's Architecture Development Method (ADM), specifically Phases B (Business Architecture), C (Information Systems Architecture), and D (Technology Architecture), is integrated to ensure strategic alignment and governance compliance of the system architecture. Development is executed through an iterative agile software development life cycle (SDLC), which enables incremental delivery, usability testing, and rapid adjustments based on feedback from end users in HR and finance.

The application of this framework in the research is not linear but is characterized by a dynamic interplay between the cycles. To illustrate this, the process is visualized through two key interactions:

1. The Foundational Interaction: Relevance and Rigor

The initial phase of the research is driven by the interaction between the **relevance cycle** and the **rigor cycle**. Real-world problems at SGU (Relevance) are framed and understood through the established knowledge base (Rigor). This approach ensures that the system requirements are both practical and theoretically sound, as illustrated in Figure 3.2. Detailed explanation of the Relevance Cycle presented in Section 3.3.

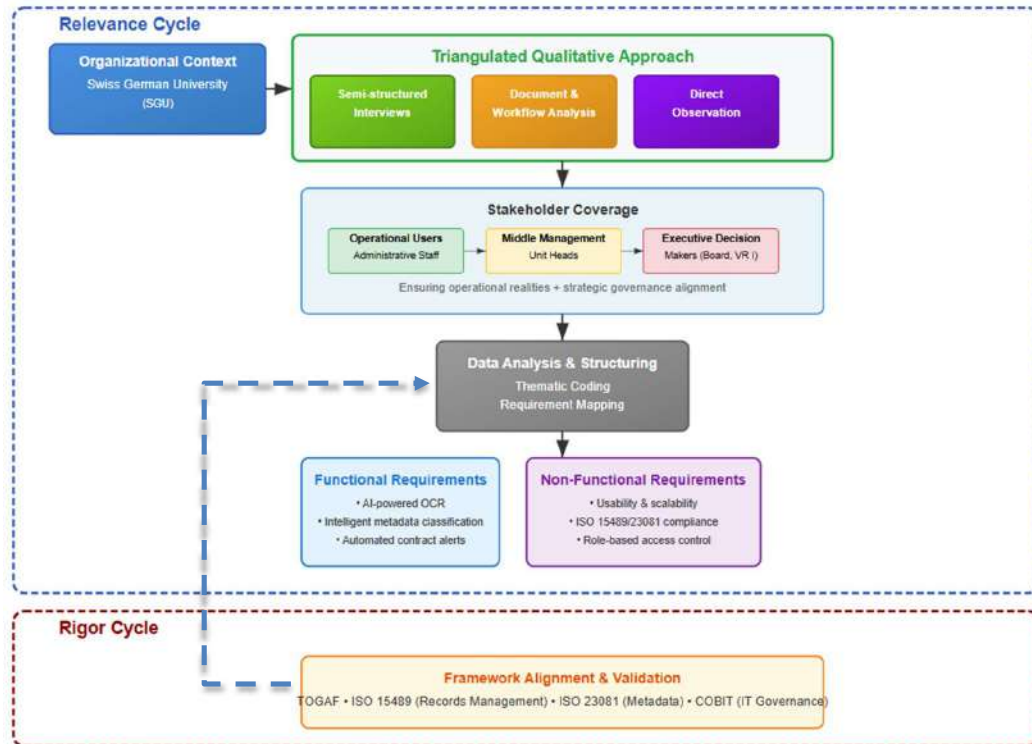


Figure 3.2 Interaction of the Relevance and Rigor Cycle in Problem Framing

2. The Core Process: Design and Rigor

The creation of artifacts is guided by the continuous interaction between the design cycle and the rigor cycle. As the prototype is built iteratively (Design), it is constantly checked against the architectural principles and governance standards in the knowledge base (Rigor). This process ensures that the final artifact is a robust, compliant solution, as illustrated in Figure 3.3. Detailed explanation of Design Cycle presented in Section 3.4.

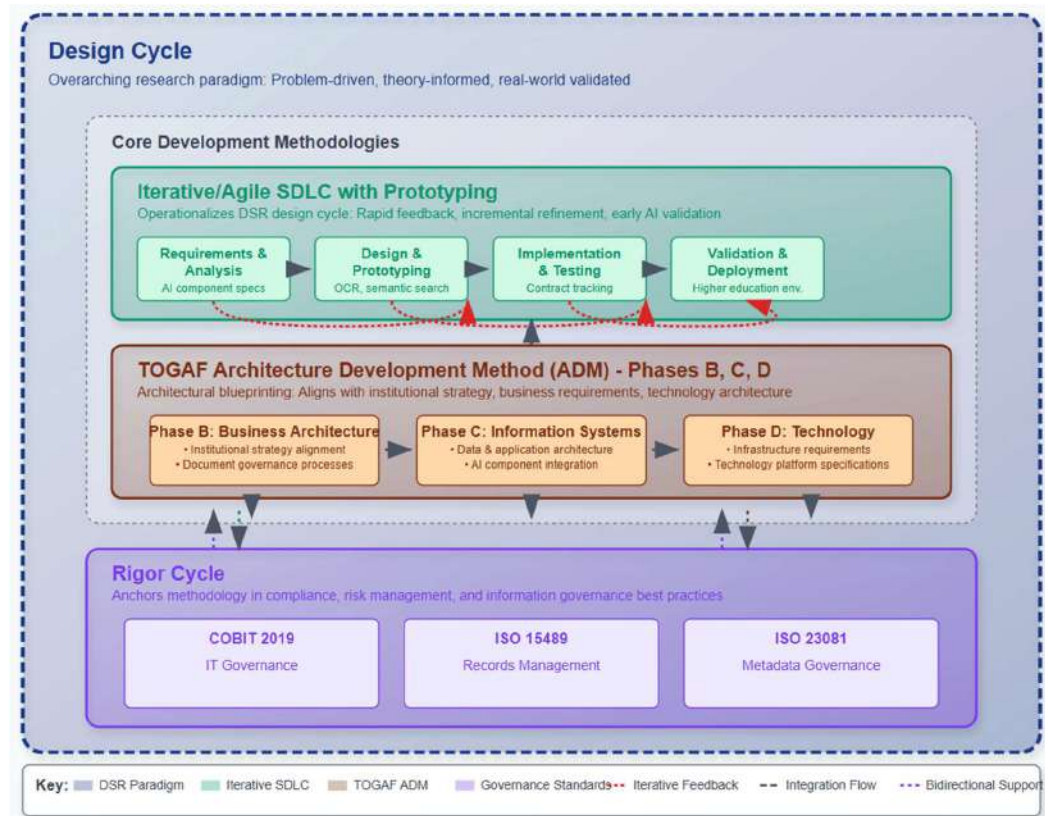


Figure 3.3 Interaction of the Design and Rigor Cycles in Artifact Development

Integrating these three cycles achieves a necessary synthesis of practice and theory in the research methodology. The Relevance Cycle connects the project to SGU's practical needs. The Rigor Cycle builds the solution on established knowledge. The Design Cycle drives iterative creation and validation. Together, these cycles ensure that the final artifact is a contextually appropriate solution and a meaningful contribution to the field of AI-enhanced document governance.

3.2.2 Architectural Alignment – TOGAF ADM

Within the DSR framework, the **TOGAF Architecture Development Method (ADM)** is the formal process used to execute the architectural design activities of the **Design Cycle**. The ADM acts as the architectural backbone that translates the business needs identified in the Relevance Cycle into a coherent, governance-compliant system design.

For this research, the scope of the TOGAF ADM is **limited to Phases B (Business Architecture), C (Information Systems Architecture), and D (Technology Architecture)**. These three phases represent the core design engine of the framework and are the most relevant for developing a new artifact, such as this proof of concept (PoC) system.

The other ADM phases were excluded because they fall outside the PoC scope:

- **Phase A (Architecture Vision)** was effectively completed during the problem analysis and requirements elicitation of the DSR Relevance Cycle, which established the strategic vision for the solution.
- **Phases E (Opportunities and Solutions), F (Migration Planning), G (Implementation Governance), and H (Architecture Change Management)** focus on large-scale, enterprise-wide implementation, migration, and the long-term governance of a deployed system. Since the goal of a PoC is to prove feasibility rather than manage a full organizational rollout, these implementation-focused phases are considered future work that would follow a successful PoC.

3.2.2.1 Phase B - Business Architecture: This phase translates the institutional challenges identified in the Relevance Cycle, such as fragmented document storage, inefficient retrieval processes, and a lack of contract deadline monitoring, into structured business requirements. The business architecture ensures that the proposed solution addresses the workflows, roles, and governance policies of the HR and Finance departments. It also aligns the system's objectives with SGU's broader strategic goals, enabling institutional buy-in.

3.2.2.2 Phase C - Information Systems Architecture: This phase addresses the data and application architectures. The data architecture design incorporates ISO 15489 principles for record authenticity and ISO 23081 standards for metadata governance. This ensures that all documents and contracts are captured, described, and stored for long-term usability. The application architecture defines how various software components, including Google Workspace for Education, Google Vision API, Vertex

AI, and Gemini AI, interact to deliver optical character recognition (OCR), classification, metadata management, and semantic search requirement.

3.2.2.3 Phase D - Technology Architecture: This phase defines the technology stack and deployment model. The PoC is developed within SGU's existing Google Workspace for Education infrastructure to optimize cost-effectiveness and adoption. Cloud services, APIs, and AI tools are integrated into a modular architecture that supports scalability. This enables expansion beyond the HR and finance departments in later phases without disrupting the core system.

Integrating the TOGAF ADM into the DSR process ensures that the PoC is a strategically designed artifact and not just a functional prototype. This architectural discipline provides a clear blueprint that guides each iteration of the Agile SDLC by balancing the need for rapid development with the stability and compliance required in a university governance context.

3.2.3 System Development Process – Iterative/Agile SDLC

The practical implementation of the AI-enhanced document and contract governance system is driven by an iterative/agile software development life cycle (SDLC) approach. This approach operationalizes the architectural outputs from the TOGAF ADM into tangible system components and aligns with the iterative nature of the **DSR Design Cycle**. The Agile SDLC was chosen for its flexibility and emphasis on continuous feedback, which are essential for developing a PoC with evolving requirements.

The development process consists of a series of short, incremental sprints, each of which focuses on delivering a functional part of the system. As illustrated in Figure 3.5, each sprint follows the core Agile phases.

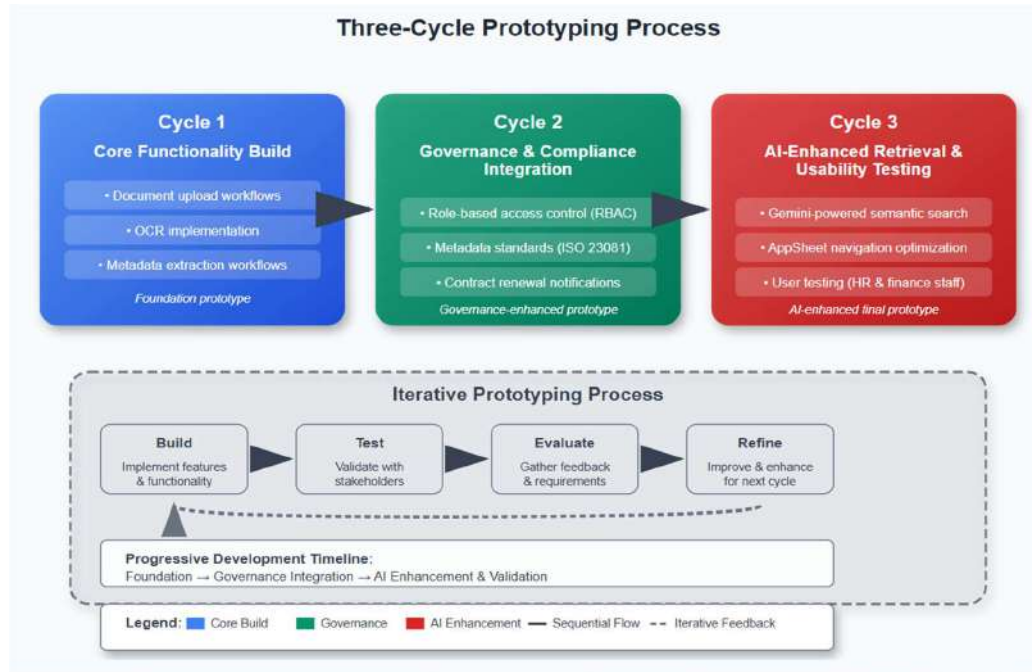


Figure 3.4 The Iterative/Agile SDLC Process

The core delivery mechanism of this process is prototyping. Each sprint produces an operational prototype reviewed by SGU's HR and Finance department stakeholders. This ensures that the evolving system remains technically robust and aligned with the day-to-day needs of its end users through continuous *"build-test-refine"* iterations.

Importantly, this Agile development process is not operated in isolation. It is anchored in the structured guidance of the TOGAF ADM and the governance principles of the ISO and COBIT frameworks. This ensures that, even as development is rapid and adaptive, the strategic alignment and compliance requirements established in the Rigor Cycle are never compromised. Combining Agile prototyping with enterprise architecture discipline enables the PoC development to balance innovation speed with institutional governance needs successfully.

3.2.4 Framework and Standards Integration

The research framework strategically integrates governance and compliance standards into the design and development process. This ensures that the resulting proof-of-concept (PoC) system is technically functional and compliant with recognized records

and information management best practices. These standards form the core of the DSR Rigor Cycle and are integrated into the phases of the TOGAF ADM and the iterations of the Agile SDLC, influencing decisions at every stage.

The specific "*job*" of each standard within this research is summarized in Table 3.1.

Table 3.1 Role of Integrated Governance and Compliance Standards

Standard	Core Principle	Application in the PoC
ISO 15489	Records Management: Ensures the authenticity, reliability, integrity, and usability of institutional records.	Informs the business and information architectures (TOGAF Phases B & C), guiding requirements for document capture, secure storage, and controlled access.
ISO 23081	Metadata for Records: Defines the schema and requirements for metadata that supports discovery, lifecycle management, and compliance.	Operationalized through the automated metadata schema applied by Vertex AI. It ensures all documents are tagged consistently for enhanced search and governance.
COBIT 2019	IT Governance & Control: Provides objectives for access control, compliance monitoring, and risk mitigation.	Guides the implementation of Role-Based Access Control (RBAC) policies and the design of audit log mechanisms to align the PoC with SGU's IT governance needs.

Practical Integration Across the Development Lifecycle: The integration of these standards is dynamic and iterative. They are not a final checklist but are actively used throughout the DSR process:

- In the relevance cycle, they shape problem framing and requirements elicitation to ensure that governance is a core design driver.
- In the design cycle, they are embedded into TOGAF ADM's architectural blueprints and influence agile sprint priorities (e.g., access control before broad feature expansion).
- Finally, they provide the formal criteria for evaluating the artifact's compliance and metadata integrity during the evaluation phase.

By embedding these standards directly into the DSR–TOGAF–Agile framework, the research ensures the final artifact is a compliant, sustainable, and governance-aligned solution.

3.3 The DSR Relevance Cycle: Problem Analysis and Requirements Elicitation

The relevance cycle grounds this study in the real-world conditions of the Swiss German University (SGU) refer to Section 2.2, ensuring that the system design addresses existing operational issues. The core activities of this cycle involved **problem diagnosis** and a structured **requirements elicitation** process.

3.3.1 Problem Diagnosis: Root Cause Analysis

To gain a deeper understanding of the challenges at SGU, a formal root cause analysis was conducted using an **Ishikawa (fishbone) diagram**. This methodological tool, which is presented in Chapter 1 (Figure 1.1), was used to systematically investigate and categorize the underlying causes of inefficiencies in the governance of non-academic documents.

A detailed analysis of each category is provided below:

- **Methods (Processes):** The investigation revealed a significant lack of standardized processes. Document management practices were inconsistent across departments and lacked formal procedures for naming conventions, version control, and metadata application. The contract monitoring process was entirely manual and ad hoc, relying on memory or personal calendar reminders, both of which are unreliable and high-risk methods.
- **Machines (Technology):** Although SGU has a powerful platform in Google Workspace, its advanced features were found to be underutilized. The primary issue was technological fragmentation: documents were stored in many disconnected locations, such as personal Google Drives, local computer desktops, and email inboxes. This created severe data silos and made unified searching impossible. There was no centralized repository or automated system to enforce governance.
- **Manpower (People):** The analysis revealed an overreliance on the tacit knowledge of individual staff members. The location and history of critical documents were often known only by one or two people, which created a significant vulnerability during staff turnover or leave. Reliance on personal

knowledge rather than a systematic process was a major contributor to information loss.

- **Materials (Information/Data):** The materials, or the documents themselves, were found to be poorly managed. The most critical issue was the lack of consistent, structured metadata. Without proper tagging and classification, documents were effectively invisible to any systematic search, which made retrieval inefficient and time-consuming.

This detailed analysis confirmed that the issues at SGU are systemic problems, not isolated incidents. Three primary, interlinked root-level problems were identified:

1. **Fragmentation and loss of institutional knowledge.**
2. **Inefficient document retrieval.**
3. **Absence of proactive contract deadline monitoring.**

These formally diagnosed problems provide the foundation for the next activity in the Relevance Cycle: determining the specific requirements for a new system designed to solve these problems.

3.3.2 Requirements Elicitation Approach

After the initial problem analysis, the next critical activity in the DSR relevance cycle was the formal requirements elicitation process. This process was designed to establish a traceable link between the institutional challenges identified at SGU and the specific requirements for the proposed system (refer to Figure 3.2). This ensures that the artifact is based on the actual needs of users and governance principles.

Elicitation Strategy: To ensure that the collected requirements were comprehensive and accurate and not biased by a single perspective, a triangulated qualitative approach was employed. **Triangulation increases the validity of findings by cross-verifying data from multiple sources.** This approach provides a more complete and robust understanding of user needs and operational realities. The strategy combined three complementary techniques:

- **Semi-structured interviews:** This was chosen to capture the *"what people say."* It's the best way to understand users' perceptions, opinions, frustrations, and their stated needs and desires. The semi-structured interviews with key stakeholders from HR, Finance, and selected administrative units to capture current pain points, latent needs, and aspirational system features.
- **Document and workflow analysis:** This was chosen to capture the *"what people are supposed to do."* It provides the formal, official picture of how processes are designed to work. It helps identify gaps between the official process and what actually happens. This analysis of existing non-academic document management processes, including storage practices, metadata conventions, approval workflows, and contract renewal methods.
- **Direct observation:** This was chosen to capture the *"what people actually do."* This is often the most powerful technique because it reveals the informal workarounds, hidden bottlenecks, and tacit knowledge that people don't mention in interviews.

To ensure the elicited requirements reflected both operational realities and strategic governance objectives, stakeholder engagement was intentionally broad, including operational users, middle management, and executive decision-makers. The data was then analysed and organized into two categories: **functional requirements** (what the system must do, e.g., AI-powered OCR) and **non-functional requirements** (system qualities, e.g., usability and ISO compliance).

This rigorous elicitation process yielded a comprehensive, prioritized list of functional and non-functional requirements. This list served as direct input for the architectural design activities of the **DSR Design Cycle**.

3.3.3 Requirements Validation and Alignment

After the elicitation process, each requirement was systematically validated to ensure methodological rigor and coherence. This critical step, guided by the **DSR Rigor Cycle**, ensures that the proposed system solves SGU's operational problems and aligns with established best practices.

This validation was achieved through a two-layer alignment process.

1. Alignment with Frameworks and Standards: Each requirement was cross-validated against the internationally recognized frameworks and standards discussed in Chapter 2.

- TOGAF ADM: This ensured that the system's requirements were coherent with SGU's enterprise architecture vision.
- ISO 15489 and ISO 23081: Guided the creation of authentic records and a consistent, structured metadata schema.
- COBIT 2019 informed the governance controls for access management, permissions, and compliance.

2. Traceability to Research Questions and Objectives: Each requirement was mapped directly to the study's research questions (RQs) and objectives (ROs). This maintains a clear, traceable link between every feature of the artifact and the research's core goals.

This comprehensive alignment is the foundation of the requirements summary in Table 3.2. This table consolidates all functional and non-functional requirements and shows their direct mapping to the identified problems, governing frameworks, and research questions.

Table 3.2 Consolidated Requirements with Alignment to Frameworks, Standards, RQs and Ros

Identified Problem	Requirement	Framework / Standard	RQ & RO
Functional Requirements			
Fragmented storage across departments causing data silos	Centralized document repository	TOGAF ADM (Phases B–D), ISO 15489	RQ1, RQ2, RO1
Inefficient retrieval due to inconsistent metadata	AI-powered OCR and metadata classification	ISO 23081	RQ1, RQ2, RO1
Lack of intelligent search requirement	Semantic search for documents and contracts	ISO 23081	RQ1, RQ2, RO1

Identified Problem	Requirement	Framework / Standard	RQ & RO
Missed contract renewals due to absence of monitoring	Automated contract deadline tracking with calendar integration	ISO 15489, COBIT	RQ1, RQ3, RO2
Manual upload process prone to errors	Guided upload workflow with document type classification	TOGAF ADM, ISO 23081	RQ1, RQ2, RO1
Dependence on staff knowledge for locating documents	Role-based access search within department scope	COBIT, ISO 15489	RQ1, RQ3, RO2
Non-Functional Requirements			
Risk of unauthorized access	Role-Based Access Control (RBAC) tied to SGU accounts	COBIT, ISO 15489	RQ2, RQ3, RO2
Inconsistent compliance practices	Governance alignment	ISO 15489, ISO 23081, COBIT	RQ2, RQ3, RO2
Scalability for future university-wide deployment	Cloud-ready architecture with scalable storage	TOGAF ADM (Technology Architecture)	RQ2, RO1
Data loss risk in storage	Redundant storage with version control	ISO 15489	RQ2, RQ3, RO2
Lack of auditability	Activity logs for all document upload and access	COBIT, ISO 15489	RQ2, RQ3, RO2

3.4 DSR – Design Cycle

The design cycle is the operational core of the DSR framework. There, the conceptual foundations of the research are transformed into a tangible artifact. The cycle receives the **requirements specification** from the relevance cycle and the **conceptual blueprint** from the rigor cycle as its primary inputs. The output is a functional, AI-enhanced document and contract (Refer to Figure 3.3).

The design cycle is not a linear build process, but rather an iterative **Agile SDLC**. This approach allows for rapid prototyping and continuous refinement based on feedback from HR and finance stakeholders (see Diagram 3.4). To ensure architectural discipline, each development iteration adheres to the **TOGAF ADM principles (Phases B, C, and D)**. This alignment guarantees that the artifact functions correctly, supports SGU’s compliance obligations, and adheres to ISO standards for records and metadata.

Given the scope of this research, the immediate objective is to deliver a functional **proof of concept (PoC)**, rather than a production-ready enterprise system. The PoC will focus

exclusively on the core governance features and AI-driven requirements identified in the previous section.

3.4.1 Architectural Design – TOGAF ADM

The architectural design of the artifact is a core activity within the DSR Design Cycle and is guided by the TOGAF Architecture Development Method (ADM). The scope is deliberately limited to Phases B, C, and D in order to align the proof of concept (PoC) with SGU's strategic requirements while remaining feasible within the research's constraints.

Figure 3.5 illustrates the layered architecture of the proposed system, mapping the design elements across these three TOGAF phases.

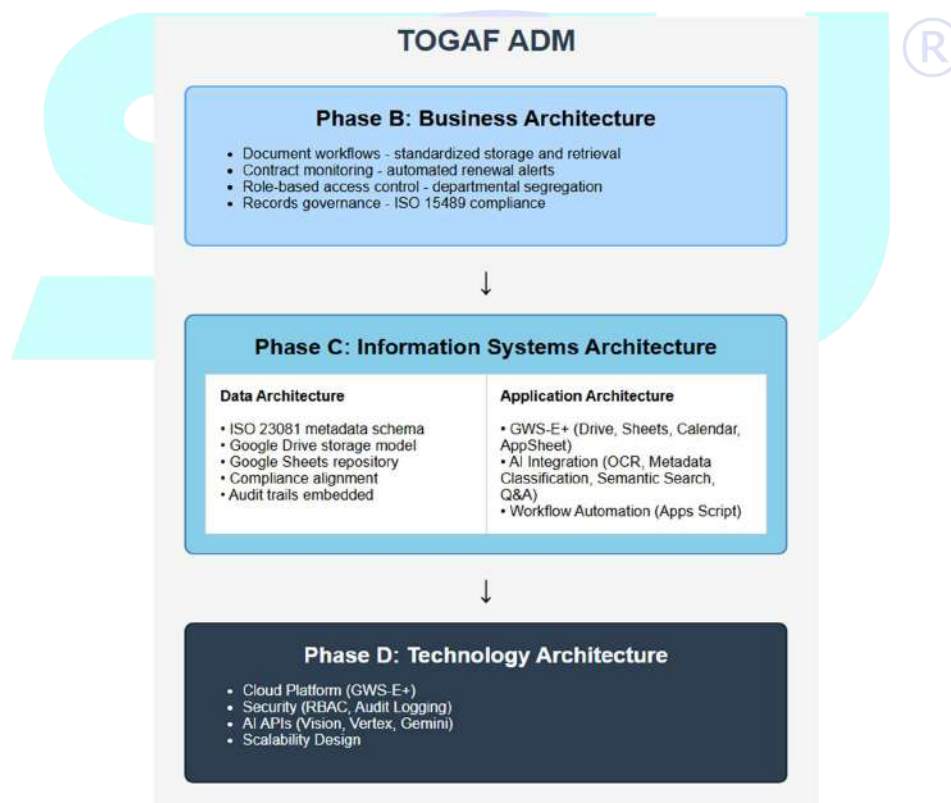


Figure 3.5 Layered System Architecture (TOGAF ADM Phases B-D)

The following subsections detail the key considerations and outcomes for each phase as shown in the diagram.

3.4.1.1 Phase B: Business Architecture

The Business Architecture phase is a critical first step in the TOGAF ADM. It translates the high-level strategic needs and problems identified in the DSR Relevance Cycle into a detailed operational model. The primary goal of this phase was to design target *"to-be"* business processes for document and contract governance at SGU that would directly solve the root causes of identified inefficiencies and risks.

This involved defining how the new system would reshape existing workflows and introduce new capabilities to meet governance and compliance requirements. The key design considerations were:

- **Standardizing Document Workflows:** The *"as is"* state was characterized by inconsistent and ad hoc workflows. The *"to be"* business architecture, however, defines a single, standardized process for all non-academic documents. The new workflow requires that all documents enter the system through a controlled ingestion point, the AppSheet interface, where they undergo automated classification and routing. This design directly addresses the problems of fragmented storage and inconsistent metadata by creating a single, governed pathway for all information.
- **Implementing Automated Contract Monitoring:** The analysis revealed a complete absence of a formal business process for contract monitoring. Phase B therefore involved designing a new business capability from the ground up. According to the *"To-Be"* process, upon ingestion, documents identified as contracts will have their key dates automatically extracted and renewal alerts scheduled in a centralized system (Google Calendar), eliminating the need for manual intervention. This new process is designed to eliminate the risk of missing deadlines.
- **Enforcing Formal Role-Based Access Control (RBAC):** The *"as is"* state relied on informal access rights and personal knowledge. The new business architecture formalizes access by defining clear roles, such as *"HR Staff," "Finance Head,"* and *"University Leadership,"* and mapping them to specific permissions based on the principle of least privilege. In accordance with ISO 15489 principles for record

integrity and COBIT security objectives, this design ensures that access to information is governed by an individual's official function rather than personal relationships.

The outcome of Phase B is a clear, governance-compliant operational model. It defines precisely how users in the HR and finance departments will interact with the platform, how documents will be managed throughout their lifecycle, and how compliance will be enforced. This detailed business architecture directly informs the Information Systems Architecture developed in Phase C.

3.4.1.2 Phase C – Information Systems Architecture

In this phase, the business architecture defined in Phase B is translated into a logical information systems architecture.

1. **Data Architecture:** This defines the information structure. The core element is the **metadata schema**, which is aligned with **ISO 23081** and includes fields such as Document Type, Department, and Contract Expiry Date. Data is stored in Google Drive, with metadata managed in Google Sheets for this PoC.
2. **Application Architecture:** This defines the functional components. The **Google Workspace** platform serves as the foundation, with **Google AppSheet** as the user interface. **AI integration points** are clearly defined: Google Vision API for OCR, Vertex AI for metadata classification, and Gemini AI for semantic search. **Google Apps Script** orchestrates the workflow automation between these components.

The outcome is a fully defined logical structure that ensures all system interactions are traceable, compliant, and efficient.

3.4.1.3 Phase D – Technology Architecture

This phase outlines the technical infrastructure required to implement the business and information system designs from Phases B and C. The focus is on providing a secure, scalable, and cost-effective environment that aligns with SGU's current IT policies and the scope of the proof of concept (PoC):

- **Core Platform:** The system is built on **Google Workspace for Education Plus (GWS-E+)** for its native integration, security, and cost-efficiency.
- **Security Architecture:** **Role-Based Access Control (RBAC)** is enforced at the account level, and all document activity is tracked via Google Drive's native audit logs.
- **AI Service Integration:** The three Google Cloud AI services are integrated via **Google Apps Script**, which acts as the orchestration layer.
- **Scalability Considerations:** For the PoC, **Google Sheets** is used as the metadata repository for rapid development. A future migration path to the more robust **Google Cloud Firestore** is defined for enterprise-scale deployment.

Integrating these three phases ensures the final product is a strategically designed solution. The following table summarizes the key outputs and design decisions made during each phase of the TOGAF ADM.

Table 3.3 Summary of TOGAF ADM Application for the PoC

TOGAF Phase	Core Objective	Requirement Fulfilled	Key Design Outputs & Decisions
Phase B: Business Architecture	Translate requirements into business processes.	• Centralized Repository • Automated Tracking • RBAC	• Standardized Document Workflow • New Contract Monitoring Process • Formalized RBAC Model
Phase C: Information Systems Architecture	Define the logical data and application structures.	• Metadata Classification • Semantic Search • Governance Alignment	• Data Architecture: ISO 23081-aligned metadata schema • Application Architecture: AppSheet UI, Apps Script backbone, AI integration points defined.
Phase D: Technology Architecture	Define the underlying technical infrastructure.	• Scalable Architecture • Auditability • RBAC	• Core Platform: Google Workspace for Education Plus • Security: RBAC at the account level; audit logs. • Scalability Path: Defined migration from Sheets to Firestore.

3.4.2 Prototyping with Agile SDLC (See Diagram 3.4)

The PoC artifact was developed using an adapted Agile SDLC process. This approach is a core part of the DSR design cycle and translates architectural blueprints into functional prototypes through structured internal sprints.

Due to the PoC nature of this research, the sprint cycles were driven by the predefined requirements from Table 3.2, and the research team conducted an internal review loop. This ensured that each iteration aligned with the technical specifications and governance standards before the final prototype was presented to end users for evaluation.

The development process was organized into three sequential cycles, detailed in table 3.4; each focused on building a specific layer of functionality.

Table 3.4 Prototyping Development Cycles

Cycle	Focus	Key Features Implemented
Cycle 1	Core Functional Implementation	• AppSheet interface for document uploads (DOCX, PDF) • Google Vision API integration for OCR. • Initial metadata extraction and storage in Google Sheets.
Cycle 2	Governance & Compliance Integration	• Role-Based Access Control (RBAC) at the Workspace level. • ISO 23081 metadata standards applied to classification. • Automated contract renewal notifications via Google Calendar.
Cycle 3	AI-Enhanced Retrieval & Optimization	• Gemini AI integration for semantic search and Q&A. • UI/UX refinements for the AppSheet interface. • Final internal verification of all AI and workflow rules.

This structured, cyclical approach enabled the systematic construction of the prototype. It ensured that foundational features were stable before adding more complex AI and governance layers. Each cycle concluded with an internal review to verify that the build complied with the architectural design and requirements before proceeding to the next cycle.

3.4.3 AI and Automation Integration Strategy

The system's central value proposition is the integration of AI and automation, which directly addresses the core challenges of inefficient retrieval and the absence of proactive contract monitoring. As a key part of the DSR Design Cycle, the integration

strategy prioritizes seamless interoperability within the Google Workspace environment and adherence to governance principles.

Figure 3.6 illustrates the end-to-end AI and automation workflow.

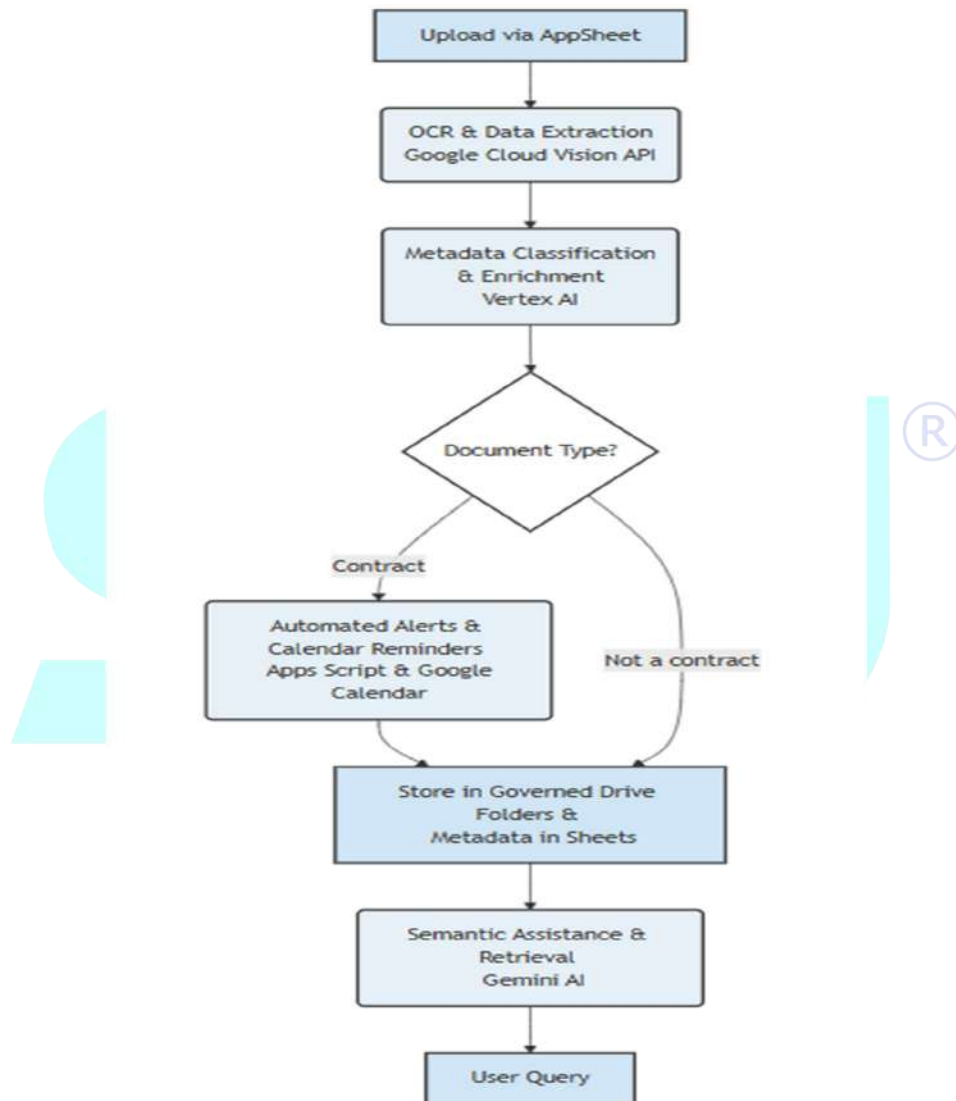


Figure 3.6 AI & Automation Workflow Diagram

The role of each core technology component in this workflow is detailed in Table 3.5.

Table 3.5 AI and Automation Component Roles

Component	Technology	Role in the Governance Workflow
1. OCR & Data Extraction	Google Vision API	Converts scanned or image-based documents into machine-readable text, making them searchable and ready for analysis.
2. Metadata Classification	Vertex AI	Automatically processes the extracted text to tag and categorize documents according to the ISO 23081-aligned metadata schema.
3. Semantic Retrieval	Gemini AI	Enables users to query document content using natural language, providing context-aware answers beyond simple keyword search.
4. Automation Backbone	Google Apps Script	Acts as the central orchestrator, managing the data flow between services, triggering AI calls, updating the metadata repository (Google Sheets), and creating contract renewal alerts in Google Calendar.
5. Security & Governance	Google Workspace Admin	Enforces Role-Based Access Control (RBAC) at the user account level, ensuring staff can only access documents relevant to their role and department, in line with COBIT and ISO principles.

This integrated approach embeds the system's intelligence within a governance-compliant, automation-driven architecture. The entire workflow directly implements the application and technology architectures designed in **TOGAF ADM Phases C and D**. This ensures that the final artifact is a coherent, strategically aligned solution.

3.5 Data & Metadata Model Design

A robust, governance-aligned data and metadata model provides a structured foundation for the entire AI and automation workflow. A key deliverable of the **DSR Design Cycle**, this model ensures the accuracy and consistency necessary for effective system operation.

3.5.1 Metadata Schema Design (ISO 23081 Alignment)

The metadata schema was developed in alignment with the principles of ISO 23081. It incorporates categories of descriptive, administrative, and preservation metadata, as well as additional fields specifically for contract management. This alignment detailed in table 3.6

Table 3.6 ISO 23081-Aligned Metadata Schema for PoC

ISO 23081 Category	Representative Fields in PoC	Purpose in Governance
Descriptive Metadata	• Document Title • Document Type • Department	Enables structured classification, discovery, and retrieval.
Administrative Metadata	• Owner • Access Level (RBAC) • Retention Category	Manages access rights, security, and document lifecycle.
Contract-Specific Fields	• Parties Involved • End Date • Renewal Terms	Provides the specific data points to power automated contract monitoring.

Validation rules embedded in the AppSheet upload form ensure that mandatory fields are completed and that controlled vocabularies are applied consistently.

3.5.2 Data Flow and Storage Decisions

The system's data flow is a sequential process that ensures each document is processed, classified, and stored properly:

1. Upload documents via the AppSheet interface.
2. OCR processing by the Vision API.
3. Metadata classification and enrichment by Vertex AI.
4. The governed file is stored in an RBAC-restricted Google Drive folder, and the metadata is written to Google Sheets.
5. Automated alerts are triggered by Apps Script for documents identified as contracts.

For the proof of concept (PoC), metadata is stored in Google Sheets to enable rapid development and seamless integration. In the future, the production system will migrate to a more robust database, such as Google Cloud Firestore, to ensure enterprise-grade scalability and performance.

3.5.3 Metadata Governance

To maintain the system's long-term integrity, **governance mechanisms** are embedded directly into the metadata model. These mechanisms are not arbitrary; rather, they are the **practical implementation of the non-functional requirements** defined in Table 3.2.

While functional requirements define what the system does (e.g., optical character recognition [OCR] and search), non-functional requirements define the essential qualities the system must have (e.g., security and auditability). The governance mechanisms are the specific features and processes designed to enforce these qualities. They are applied to the functional parts of the system to ensure that every operation is performed in a compliant, secure, and auditable manner.

This approach establishes a traceable link from the high-level requirement to the specific feature that implements it. Table 3.7 summarizes the alignment between each requirement, its corresponding governance mechanism, and the guiding ISO and COBIT standards.

Table 3.7 Metadata Governance Mapping from Requirements to Implementation

Non-Functional Requirement	Governance Mechanism	COBIT Domain & Process	Relevant ISO 23081 Principle(s)	System Implementation
Governance Alignment	Controlled Vocabularies	APO03 (Manage Enterprise Architecture): Ensures that the information architecture includes standards for consistent data definition across the enterprise.	Metadata must be structured and consistent to support interoperability.	Centralized, administrator-managed lists for Document Type, Department, Retention Code, etc., in AppSheet's backend.
Governance Alignment	Periodic Metadata Quality Reviews	APO11 (Manage Quality): Focuses on the continuous improvement of data and information quality through a defined Quality Management System (QMS).	Metadata must be maintained and updated throughout the record's lifecycle to remain accurate.	Scheduled governance officer audits to sample records for completeness, detect outdated values, and verify retention compliance.
Governance Alignment	Automated Field Validation	BAI09 (Manage Assets): Ensures the integrity and accuracy of information assets by implementing controls at the point of creation.	Metadata quality must be ensured at the point of capture to be reliable.	AppSheet form rules enforcing mandatory fields, date format checks, and drop-down lists for controlled fields.
Role-Based Access Control	Access Control & Metadata Visibility	DSS05 (Manage Security Services): Protects information against unauthorized access by managing user identity and logical access.	Metadata must reflect and enforce access and use restrictions of the record.	RBAC applied to documents and sensitive metadata fields, restricting visibility to authorized roles.
Auditability	Audit Trails	MEA03 (Monitor, Evaluate and Assess	Metadata changes must be	Automated logging of metadata

Non-Functional Requirement	Governance Mechanism	COBIT Domain & Process	Relevant ISO 23081 Principle(s)	System Implementation
		Compliance): Confirms compliance with external requirements by gathering and evaluating evidence, for which audit trails are a primary source.	documented to ensure the authenticity, reliability, and integrity of the record over time.	creation/modification events, including user ID, action, and timestamp.

3.6 System Evaluation and Validation Methods

This section details the methods used to evaluate the AI-enhanced document governance system. The evaluation phase is the final activity of the DSR design cycle. It is designed to rigorously answer research question 4 (RQ4) by assessing the prototype's usability, effectiveness, and compliance.

To establish a clear connection to the theoretical foundations of the research, the success criteria are derived directly from the principles and objectives of the guiding frameworks (TOGAF, COBIT, and ISO) rather than based on generic benchmarks. For instance, the usability criteria directly measure the "*usability*" requirement in ISO 15489, and the 100% success target for access control directly reflects the absolute security requirements in COBIT.

The evaluation employs a mixed-methods approach, combining expert validation, user testing, and direct system validation.

3.6.1 Expert Validation

The first layer of evaluation involved expert validation to assess the artifact's architectural soundness and adherence to core governance standards, such as TOGAF, ISO, and COBIT.

- **Participants:** A panel of five experts with backgrounds in business strategy, information systems, human resources, and quality assurance was assembled.
- **Method:** The evaluation consisted of structured interviews and a live system walkthrough, during which the experts could interact with the prototype and provide feedback.

- **Evaluation criteria:** The experts evaluated the system's architectural design, compliance with ISO and COBIT principles, scalability, and the reliability of its AI model integration.

3.6.2 User Testing

The second layer of evaluation involved user testing with SGU administrative staff to measure the prototype's real-world usability and effectiveness in solving identified problems.

- **Participants:** A group of administrative staff members from the HR and Finance departments at SGU participated in the study.
- **Method:** The testing used a task-based scenario approach. Participants were asked to complete realistic tasks, such as "*Upload a new contract*" or "*Find the policy on academic leave,*" while the researcher observed. After completing the tasks, participants filled out the System Usability Scale (SUS) questionnaire and took part in a brief qualitative interview.
- **Evaluation criteria:** The evaluation focused on overall usability (via the SUS score), task success rate, error rate, and qualitative feedback on the user experience.

3.6.3 System & Governance Validation

The final evaluation step was an objective, direct test of the system's ability to enforce non-negotiable compliance and security rules.

- **Method:** This involved executing predefined test scenarios (e.g., attempting to access unauthorized documents or checking whether an expiring contract triggers an alert) and conducting a direct audit of system outputs, such as access logs and metadata records.
- **Evaluation criteria:** This validation measured record integrity, metadata completeness, enforcement of role-based access control (RBAC), auditability, and reliability of automated alerts.

3.6.4 Data Analysis for Evaluation

Data from both evaluation layers were integrated to create a comprehensive assessment.

- **Quantitative data**, such as SUS scores, task completion rates, and expert ratings, were analysed using descriptive statistics to measure performance against the defined success criteria.
- **Qualitative data**, such as interview notes and open-ended feedback, were analysed using thematic analysis to understand the reasons behind the quantitative results and identify specific strengths and areas for improvement.

The entire evaluation plan, linking each method to its specific metrics, success criteria, and the research questions it addresses, is consolidated in Table 3.8. This table serves as the definitive blueprint for the evaluation conducted in this study.

Table 3.8 Evaluation & Validation Plan

Evaluation Method	What is Being Measured (Evaluation Criteria)	How it is Measured (Metric)	Definition of Success (Framework-Derived)	Aligns with
Expert Validation	Architectural & Governance Soundness: The degree to which the design aligns with TOGAF, ISO, and COBIT principles.	Expert panel rating (1-5 Likert scale) after a structured walkthrough.	• Architectural Fit (TOGAF): Avg. score $\geq 4.0/5$ • Compliance (ISO/COBIT): $\geq 85\%$ of controls rated "Implemented" with 0 critical non-compliances.	RQ2, RQ4 H3, H4
User Testing	Usability & Effectiveness: How easily and effectively end-users can complete core tasks to solve their problems.	• System Usability Scale (SUS) • Task Completion Rate • Qualitative Feedback	• Usability (ISO 15489): Avg. SUS score ≥ 70 (Acceptable) • Effectiveness (COBIT Value): $\geq 90\%$ completion on critical tasks • No critical (severity-1) usability issues.	RQ4 H1, H2
System & Governance Validation	Compliance & Reliability: The system's ability to correctly enforce non-negotiable governance and security rules in practice.	Execution of predefined test scenarios and audit of system outputs (logs, alerts, etc.).	• Access Control (COBIT): 100% RBAC enforcement in tests • Auditability (COBIT): 100% of events recorded in logs • Reliability (ISO 15489): 100% of contract alerts are sent correctly	RQ3, RQ4 H3, H4

3.7 Ethical Considerations

This section outlines the ethical safeguards implemented throughout the study. These safeguards reflect a commitment to responsible research conduct, especially given the involvement of human participants, institutional documentation, and artificial intelligence (AI). Ethical rigor was maintained to ensure participant rights, data protection, and responsible AI use within an institutional compliance framework.

This research's ethical approach aligns with key international standards and guidelines, as detailed below.

3.7.1 Participant Protection

- **Informed Consent:** All human participants engaged in interviews, focus group discussions, and usability testing were given clear and comprehensive explanations of the study's objectives, methods, risks, and benefits. Written invitations and consent forms were distributed in accordance with SGU's Ethics Policy and global best practices for human subject research. Participants retained the right to withdraw at any time without penalty. This process emphasizes autonomy, voluntariness, and comprehension, all of which are foundational to informed consent (Xu *et al.*, 2020).
- **Confidentiality and Anonymity:** To protect privacy, all personally identifiable information was removed from research data. All reports and publications anonymize participant responses, and any direct quotations are de-identified to prevent indirect identification.

3.7.2 Data and System Integrity

- **Secure Storage:** Research data were stored within SGU's institutional Google Workspace environment, following the controls outlined by ISO/IEC 27001 and U.S. NIST SP 800-122 standards for information security (Ajijola, Zavorsky and Ruhl, 2014).
- **Use of Non-Sensitive Data:** As outlined in Section 1.6 (Scope and Limitations), the development of the prototype utilized only non-sensitive, publicly available, or

synthetic documents. No live institutional data were used to mitigate privacy and confidentiality risks.

3.7.3 Responsible AI Integration

- **No autonomous decisions:** AI components (Google Cloud Vision, Vertex AI, and Gemini AI) were not granted autonomous decision-making authority. They were limited to supporting tasks such as classification and metadata extraction, which kept human users in full control (McKee, 2023)
- **Human-in-the-Loop Verification:** All AI-generated outputs were manually reviewed. This process mitigated errors and potential biases, ensuring trustworthiness and fairness (*AI Principles Overview - OECD.AI*, 2024).

3.7.4 Institutional Oversight and Approvals

The research project underwent academic clearance by SGU's research supervisors and was carried out in the university's sandbox development environment. To ensure alignment with institutional policies and safeguard ethical and operational boundaries, formal approvals were secured from the following:

- The Board of Management of SGU for the involvement of the Human Resources and Finance departments;
- The Vice Rector overseeing University Information Systems, who ensured compliance with institutional IT governance.
- The Head of the Information Systems and Technology (IST) Department for technical and academic endorsement;
- The heads of the HR and finance departments affirmed departmental cooperation and informed participation.

These multi-level authorizations ensured that the research complied with SGU's research policy, ISO 15489 records management principles, and broader institutional accountability frameworks.

3.8 Chapter Summary

This chapter meticulously outlines the rigorous research methodology that guided the design, development, and evaluation of the AI-enhanced document and contract

governance system. Embracing the Design Science Research (DSR) paradigm, the methodology followed a systematic process of creating and validating a practical artifact to solve the real-world governance challenges at SGU.

The research process began with the DSR Relevance Cycle, in which institutional problems were analysed and translated into a comprehensive set of functional and nonfunctional requirements. These requirements were validated and aligned with the core principles of the guiding frameworks (TOGAF, ISO 15489/23081, and COBIT), which were established in the DSR Rigor Cycle.

The central DSR Design Cycle detailed the architectural design and development of the proof of concept (PoC). The system's architecture was formally designed using the TOGAF Architecture Development Method (ADM), and the prototype was built using an iterative agile software development lifecycle (SDLC) on SGU's existing Google Workspace infrastructure, integrated with Google Cloud's AI services.

Finally, the chapter presented a robust, framework-driven evaluation strategy designed to answer the core research questions. This strategy combines expert validation, user testing, and direct system validation with specific, measurable success criteria derived from the guiding standards.

The findings resulting from this rigorous methodology will be presented and analysed in the next chapter.