DRAFT - IT Project Guidance

Evolvable Service Delivery Framework (ESDF) – Detail & Rationale

Version:

0.1

## Purpose

This document describes a coherent, extensible system for organising artefacts, documents, decisions, and models across the entire service lifecycle of a digital system. SELF provides a consistent structure from the earliest strategic conception through to operational delivery, continuous improvement, and eventual decommissioning.

It is intended for use by any organisation that must manage complex, long-lived services, and is designed to integrate naturally with government agency practices, investment processes, and operational accountability requirements. The sequential organisation framework addresses the longstanding problem of fragmentation caused by unstructured documentation and inconsistent handovers between phases, supporting services being deliberately enabled and sustained, not just delivered.

## Synopsis

Many digital services lack documentation structures that persist from early strategic sensing through to long-term operational maintenance and eventual closure. Artefacts are often misplaced, recreated, or abandoned entirely, causing wasted effort, strategic drift, and operational risk.  
The frameworkprovides a structured, durable, and extensible numbering model that reflects the full service lifecycle. It enables improved continuity, governance oversight, traceability, and seamless handovers between teams, projects, and operational phases over the life of the service. Although general in design, it fully supports agency environments where formal accountability, investment justification, and ongoing assurance are required.

## Contents

[Purpose 1](#_Toc195791704)

[Synopsis 1](#_Toc195791705)

[Contents 2](#_Toc195791706)

[Introduction 3](#_Toc195791707)

[Background of this work 3](#_Toc195791708)

[Context 3](#_Toc195791709)

[Background of issue 3](#_Toc195791710)

[Problem Statement 3](#_Toc195791711)

[H3 4](#_Toc195791712)

[H4 4](#_Toc195791713)

[Appendices 5](#_Toc195791714)

[Appendix A - Document Information 5](#_Toc195791715)

[Versions 5](#_Toc195791716)

[Images 5](#_Toc195791717)

[Tables 5](#_Toc195791718)

[References 5](#_Toc195791719)

[Review Distribution 5](#_Toc195791720)

[Audience 5](#_Toc195791721)

[Structure 5](#_Toc195791722)

[Diagrams 6](#_Toc195791723)

[Acronyms 6](#_Toc195791724)

[Terms 6](#_Toc195791725)

# Introduction

Long-lived services, particularly within complex organisations and government agencies, often originate from early strategic observations, evolve through layers of investment, solution engagement, and vendor activity, and persist into multi-decade operational responsibilities. The pathway from conception to decommission is intricate, fragmented, and frequently poorly structured.  
This document introduces the **Service Enablement and Lifecycle Framework (SELF)**, a practical and extensible structure for managing artefacts, decisions, models, and evidence across the entire lifecycle of a service — from its earliest detection as a future need, through to operational excellence, and finally to structured transition or retirement.

The need for a durable organisational system arises from years of observing that artefacts — whether strategic investment cases, technical architecture diagrams, procurement records, deployment specifications, or operational logs — routinely disappear into disorganised folders, isolated systems, or inconsistent personal conventions. In the absence of a unified system, teams tend to create ad hoc numbering and storage schemes that begin from their own immediate perspective or phase, resulting in multiple overlapping hierarchies — each with their own local "1, 2, 3" — for discovery, procurement, delivery, or operations.

Over time, this leads to confusion, duplication, fractured knowledge bases, and a loss of traceability across transitions. Decisions made early in discovery are often disconnected from delivery and operation realities downstream. Critical compliance, technical, and governance evidence becomes difficult to trace or validate. Service resilience and investment value are compromised by avoidable drift.

The **SELF** framework proposes a more structured and encompassing viewpoint, drawing from both national and international best practices, but extending beyond any single funding, procurement, or technical delivery model. It establishes a common artefact, decision, and traceability structure that spans the full-service lifespan — regardless of changes in team composition, technology choices, funding sources, or governance regimes.

By providing a clear, logical, and extensible sequencing model, the **Service Enablement and Lifecycle Framework** aims to tame complexity and make the management of services both more resilient and more predictable. Rather than terminating at the point of delivery, SELF recognises that the greatest continuity risks — and therefore the greatest value opportunities — occur across the quieter middle and late stages of service life, including support, maintenance, benefits realisation, improvement, transition, and closure.

SELF is designed to ensure that services are not only delivered effectively, but also deliberately enabled, maintained, evolved, and retired with full lifecycle accountability and operational integrity.

# Background

Many services and systems originate as informal ideas, sector observations, or perceived needs long before any formal funding, design, or development begins. They often pass through multiple, distinct phases — including early analysis, solution shaping, procurement engagement, implementation, operational scaling, and eventual reassessment or closure.

During this time, delivery teams, operational teams, vendors, technology platforms, and even policy frameworks often change. Without a consistent structural model governing the continuity of artefacts, decisions, and historical knowledge, critical context is lost between these phases.

Business cases become detached from the technical architectures they justified. Solution designs drift from the user requirements they were intended to meet. Operational practices are disconnected from the assumptions and governance obligations that underpinned the original service design.

The absence of a durable organisational framework leads directly to avoidable waste, duplicated analysis, inefficient rework, poor change management, and increased operational and compliance risk.

The **Service Enablement and Lifecycle Framework (SELF)** was developed to address this persistent and deeply structural challenge. It offers a cohesive, extensible model to preserve traceability, strengthen decision continuity, and support service integrity across all phases of the service lifecycle — from early sensing to final decommissioning.

# Context

Several lifecycle models and management frameworks are already in use across government, enterprise, and international sectors. Each provides valuable structure for specific aspects of service or system development, governance, or delivery:

* The Treasury’s **Better Business Case (BBC)** model structures investment decision-making from strategic case to detailed case development.
* **MBIE’s Procurement Lifecycle** governs transparent, fair engagement with external vendors and suppliers.
* The Government Chief Digital Officer’s (**GCDO**) **Service Design and Delivery Lifecycle** promotes discovery, validation, and agile service creation within the public sector.
* **ISO/IEC 15288** defines system life cycle processes from concept through to retirement, especially for engineering-focused environments.
* **PRINCE2** formalises project delivery stages, governance roles, and structured handovers.
* **TOGAF** provides methodologies for developing and governing enterprise architectures.
* **IT4IT** structures the IT value chain across portfolio, development, and operations activities.
* **Portfolio, Programme, and Project Management Maturity (P3M)** promotes scalable governance across multiple layers of initiatives.
* **Managing Successful Programmes (MSP)** focuses on coordinating related projects under programme structures to achieve strategic outcomes.
* The **Project Management Body of Knowledge (PMBOK)** provides a comprehensive, internationally recognised guide to project management practices.
* The **Scaled Agile Framework (SAFe)** enables agile principles to be applied at enterprise scale across multiple delivery teams.

While each of these frameworks supports important aspects of service conception, procurement, delivery, or operation, they are generally designed to address specific phases or perspectives rather than the entire, continuous service lifecycle.  
None are specifically intended to provide an enduring artefact, traceability, and documentation governance structure that persists logically from early sensing and strategic framing through operational maturity and final closure.  
In practice, services and projects span many of these phases without a unified organising system, leading to fragmentation, duplication, and the loss of decision continuity over time.

The **Service Enablement and Lifecycle Framework (SELF)** complements these established models by providing the missing lifecycle structure: a durable, extensible foundation for managing artefacts, models, and decision traceability across decades of service existence.  
A detailed comparison of SELF’s positioning against these established frameworks is provided in **Appendix B**.

# Issues

The lack of a coherent, project-spanning organisational structure leads to:

* Disorganised and duplicated documentation
* Strategic decisions being disconnected from implementation
* Difficult handovers between phases and teams
* Artefacts being lost between procurement and delivery
* Inconsistent governance and oversight of technical decisions

These issues are exacerbated in government where a competitive bid process may precede delivery, and where the distinction between business case, vendor selection, internal development, and long-term service management is often blurred or collapsed.

# Why a Structured Service Enablement Framework is Needed

To address longstanding fragmentation across the service lifecycle, the **Service Enablement and Lifecycle Framework (SELF)** proposes a structured numbering model that governs artefacts, decisions, and models from the earliest stages of sensing and framing through to final closure.  
It is designed to be:

* **Durable**: persists beyond the life of any given team, vendor, or delivery phase.
* **Extensible**: allows new artefacts to be introduced without disrupting existing structures.
* **Sequential**: ensures items sort predictably and logically in shared drives, repositories, or records systems.
* **Visible**: enables artefacts to be managed either compactly within folders or exposed fully for transparency and auditability.
* **Distributable**: supports phases or artefact sets being separated for remote development or governance, then reliably reassembled without confusion.
* **Contextual**: embeds lifecycle phase and intended purpose within each artefact’s identifier.
* **Complete**: spans the full continuum from early sensing and strategic framing through delivery, operation, improvement, and decommissioning.

By adopting SELF early, teams can consistently label folders, documents, deliverables, and tools in ways that preserve meaning across handovers, investments, audits, and operational transitions.  
This reduces architectural drift, strengthens strategic alignment, accelerates onboarding, and supports the long-term operational health of complex services.

# The Framework

The framework applies a structured prefixing system to organise artefacts, documents, and activities across the full service lifecycle.

To preserve order, extensibility, and durability over decades of service existence, the framework introduces a major prefix before the traditional numbering structure.

The framework id is developed as: **[Area].[Aspect].[Activity] and optionally [Artefacts]**

[01] Cross Cutting Management

Structures governance, management, and risk activities spanning the full service lifecycle.

[01.1] Cross-Cutting Governance

This Aspect provides decision frameworks, architecture reviews, privacy, security, and compliance oversight across all lifecycle stages.

**Optional Activities & Artifacts of this Aspect include:**Project Principles Registry, Governance Decision Registry, Privacy Registry, Security Registry, Accessibility Review, Compliance Registry, Issues Registry, Risks Registry, Contacts Registry.

[01.1.1] Preparation

[01.1.2] Principles

Maintain a catalogue of Organisation sourced Principles applicable to this work.

[01.1.2.1] Service Principles

[01.1.2.2] Support Principles

[01.1.2.3] Delivery Principles

[01.1.2.4] Solution Principles

[01.1.2.5] Enterprise Principles

[01.1.2.6] Integration Principles

[01.1.2.7] Data Principles

[01.1.3] Reviews

[01.1.3.1] Architectural Reviews

Maintain a journal of structured technical and design review checkpoints to guide solution integrity - typically conducted by architecture boards or advisory groups,.

[01.1.2] Logs

[01.1.2.1] Decision Log

Maintain chronological traceable log of governance decisions, architectural exceptions, approvals, policy clarifications, and escalation outcomes.  
May also track outcomes of Compliance and Assurance reviews that result in required actions or follow-ups. Chronological log of governance decisions, architectural exceptions, approvals, and policy clarifications made across the project.

[01.1.2] Registries

[01.1.2.01] Privacy Registry

Track obligations, impact assessments, and   
Tracks throughout the lifecycle across phases privacy obligations, data minimisation rules, consent dependencies, privacy control decisions and links to Privacy Impact Assessments (PIAs). This registry tracks existence, timing, and outcomes of reviews, not issues.

Note detailed privacy assessments should be recorded as having occurred, but stored in access-restricted system or folders.

[01.1.2.02] Security Registry

Capture security controls, threat models, assurance activities, and incident reporting.

[01.1. 2.03] Accessibility Registry

Capture equity of access and support for diverse needs. Include accessibility compliance, inclusion, and alt service channels.

[01.1. 2.04] Risk Registry

Central log of risks identified across all lifecycle phases. Tracks risk likelihood, impact, mitigation plans, escalation, and resolution history.

[01.1. 2.05] Compliance Registry

Maintain audit trail of mandatory compliance obligations, identified risks, raised issues reviews, outcomes.

[01.1. 2.06] Issue Register

Records problems raised (e.g. from C&A reviews or operational feedback), requiring investigation, mitigation, or governance action.

[01.1. 2.07] Contact Registry

Maintain authoritative contact lists for governance boards, assurance reviewers, and regulators. Maintains up-to-date references for individuals, agencies, vendors, reviewers, and other external stakeholders involved in cross-phase or cross-agency concerns.

[01.2] Cross-Cutting Management (Aspect)

Coordinates operational, delivery, resourcing, and stakeholder activities across all lifecycle stages.

[01.2.01] Preparation

[01.2.01.01] Document Storage

Select an organisation stakeholder accessible document store to persist artefacts, resources, outputs, and deliverables.

[01.2.01.02] Document Storage Organisation Methodology

Select a system for organising the shared document storage (i.e., this framework, or similar).

[01.2.01.02] Resources

Develop onboarding resources.

[01.2.01.02] Guidance

Develop onboarding guidance.

[01.2.02] Registries

Develop Registries to manage traceable objectives, resources, progress, achievements.

[01.2.02.01] Role Registry

Lists defined roles involved in the project, including decision-makers, approvers, contributors, and system actors. Supports clarity of responsibility across phases.

[01.2.02.02] People Registry

Maintain full inventory of people involved across agencies, vendors, or internal teams.

[01.2.02.03] Dates Registry

Track critical milestones, holidays, review points, contract deadlines, renewals, and transition events.

[01.2.02.04] Commitments Registry

Log formal and informal promises made to stakeholders, regulators, or partner agencies. Supports accountability and follow-up.

[01.2.02.05] Stakeholder Engagement Log

tracks individual or group-level consultation, including meetings, workshops, and formal feedback. Captures who was engaged, when, why, and what was discussed or agreed. Record structured consultation, feedback loops, and stakeholder agreements.

[01.2.02.06] Communication Registry

Logs outbound communications such as emails, status updates, newsletters, or announcements to one or many recipients. Focuses on information broadcast, not consultation.

[01.2.02.07] Dependency Registry

Captures technical, organisational, or policy dependencies that must be met or aligned with during the service lifecycle.

[01.2.02.08] Meeting and Event Log

Maintain a chronological record of key meetings, workshops, attendance and decisions.

[01.2.02.09] Deliverables Registry

Optional consolidated summary of formal artefacts committed or produced.

[01.2.02.10] Contract Registry

Centralise active contracts, scopes, terms, dates, and renewal alerts.

[01.2.02.11] Accessibility Registry

Record and address impediments to equitable access.

[01.3] Enterprise Risk and Compliance Management (Aspect)

Structures service risk identification, mitigation plans, compliance obligations, and audits.

[01.3.1] Risk Management Framework Application

Apply organisational risk frameworks to service lifecycle phases.

[01.3.2] Compliance Traceability Mapping

Link artefacts and activities to mandatory compliance obligations.

[01.3.3] Assurance Event Planning

Schedule and manage audits, security reviews, privacy assessments, and assurance reports.

[01.4] Structured Review Planning (Aspect)

Plans formal review checkpoints across the service lifecycle, including internal assurance reviews and external investment reviews such as Gateway Reviews, ensuring alignment with strategic, delivery, operational, and closure validation milestones.

[02] Environmental Sensing

Detects early needs, risks, and opportunities influencing service intent.

[02.1] Preparation

[02.2] Early Sector/Problem Sensing

Identifies early sector trends, emerging needs, threats, or opportunities relevant to strategic intent.

[02.2.1] Environmental Scan

Survey policy, technology, demographic, economic, and sectoral changes likely to influence service direction.

[02.2.2] Problem Statement Drafting

Frame emerging problems or gaps in current service landscapes requiring intervention or investigation.

[02.2.2.1] Problem Statement Document

Consider developing a document that can be referenced and summarised from other artefacts and outputs.

[02.2.3] Trigger Event Analysis

Assess incidents, mandates, reforms, or external events prompting service exploration or transformation.

[02.2.4] Opportunity Identification

Detect and document early opportunities for sector leadership, innovation, or service improvement.

[03] Strategic Framing

Shapes strategic direction, investment logic, and foundational objectives.

[03.1] Preparation

[03.2] Early Direction Setting

Establishes initial strategic purpose, alignment, and intended service outcomes.

[03.2.1] Strategic Intent Definition

Capture and formalise early mandates, sector alignments, or government priorities shaping the service.

[03.2.2] Strategic Fit Assessment

Evaluate proposed service alignment against agency strategy, sector needs, and public value principles.

[03.2.3] Outcome Framing

Define high-level intended outcomes and benefits before detailed solution or funding discussions.

[03.3] Strategic Financing Planning

Frames early thinking about financial feasibility, investment pathways, and funding structures.

[03.3.1] Investment Logic Mapping

Summarise why investment is needed, what it aims to achieve, and what evidence supports it.

[03.3.2] Indicative Funding Scenario Planning

Draft early high-level cost, funding source, and phasing assumptions to inform business case preparation.

[03.3.3] Constraints and Dependencies Identification

Document known financial, regulatory, policy, or sectoral constraints influencing feasibility.

[04] Project Financing

This section defines the process of identifying, securing, and structuring funding for the initiative. It includes both internal and external budget considerations, investment models, funding sources, and key financial governance artefacts.

**Outputs:**

Draft or confirmed budget models, funding scenarios, financial assumptions, and early investment plans used to inform Treasury or agency-level budget rounds.

[04.1] Preparation

Activities focused on engaging finance stakeholders, identifying budget envelopes, preparing investment briefings, and understanding funding constraints.

[04.2] Cost Estimation

Rough-order magnitude costing and refined cost models based on emerging solution understanding.

[04.3] Funding Strategy Development

Define how funding will be sought — internal baseline, capex bids, operating budget allocations, external grants, partnerships.

[04.4] Investment Logic Mapping (Optional)

Develop an investment logic map if required by governance frameworks (like BBC or internal models).

[04.4] Budget Submission Preparation

Prepare documentation, briefing notes, forecasts, and Treasury templates as required to seek funding.

[04.5] Funding Negotiation and Approval

Engage with finance teams, leadership, Treasury (if applicable) to advocate for the investment and negotiate adjustments.

[04.6] Funding Allocation and Baseline Setup

On funding approval, define how funding is allocated across phases and manage financial baselines for tracking.

[04.4] Financial Governance Planning

Set up cost tracking, reporting requirements, financial change management, benefits realisation tracking.

[05] Project Preparation

Frames project purpose, governance, and execution readiness.

[05.1] Preparation

[05.2] Project Framing

Defines the service project's scope, objectives, success criteria, and governance model.

[05.2.1] Purpose and Scope Definition

Articulates what the project what expected outcomes and boundaries are expected by stakeholders.

[05.2.2] Governance Structure Design

Establishes decision rights, escalation paths, steering groups, and governance roles.

[05.2.3] Success Criteria and KPIs

Drafts measurable success factors linked to strategic framing and sponsor expectations.

[05.3] Resourcing

[05.3.01] Determine Delivery Roles

[05.3.01] Advertise for Delivery Roles

[05.3.03] Interview

[05.3.04] Contract

[05.4] Project Resource

Prepares the structures, resources, and operational mechanisms needed to start delivery.

[05.4.1] Delivery Methods and Planning

Selects project delivery approach (e.g. waterfall, agile, hybrid) and plans early phase sequencing.

[05.4.2] Project Initiation and Resourcing

Confirms team roles, vendor onboarding (if any), initial delivery infrastructure, and induction activities.

[05.4.3] Stakeholder and Communications Planning

Develops internal and external communication strategies and stakeholder engagement plans.

[06] Solution Context Discovery

Discovers needs, capabilities, requirements, and high-level solution design.

[06.1] Preparation

[06.2] Context Discovery

Explores sector, organisation, user, and system context to inform problem understanding.

[06.2.1] Sector and Ecosystem Mapping

Identifies key organisations, initiatives, and environmental factors influencing the solution space.

[06.2.2] Stakeholder Mapping

Maps stakeholder groups, interests, pain points, and influence across service delivery.

[06.2.3] Systems Landscape Review

Documents existing systems, integration points, overlaps, and dependencies.

[06.3] Constraint Discovery

[06.3.1] Legal Constraints

[06.3.1.1] Disclosure Regulations

Discover the legal obligations to disclose information.   
Examples include local and/or national Official Information Acts. Implying that information is retained, categorizable for subsequent discovery, and rendering removing private information.  
Other examples include reporting information upon valid information requests from law enforcement.

[06.3.1.2] Privacy Regulations

Discover the legal obligations to protect the privacy of users.   
Examples include Privacy Act and Principles that dictate that the purpose of the collection of information is disclosed, that whom it is shared and with whom is disclosed, that incorrect information can be corrected, that personal identifiable information can be removed (by being reassociated to an anonymous user, so that it is not removed from a system).

[06.3.1.3] Security Regulations

Discover the legal obligations to ensure system remain available, are accessible by tampering

That information is encrypted at rest (e.g.: encrypted database) and in transit (e.g. HTTP/S) to ensure confidentiality and prevent tamperability, and all access to or change of information is permanently logged for auditability and traceability

[06.3.1.4] Accessibility Regulations

Discover the legal and/or regulatory obligations to make the service available to visually, and/or motor impaired users, and making the service accessible to the different official cultures of the countries where the service originates from and is delivered to.

[06.3.2] Organisation Policy Constraints

[06.3.2.1] Governance Constraints

Discover the governance boards that are required by the organisation.   
For example, this may include a Strategic Technical Review, a Technical Review Governance, and an Operations Review before a service is released for user access.

[06.3.2.2] Principle Constraints

Discover the organisation’s defined principles.   
For example, this may include Security, Privacy, Service, Procurement, Enterprise, Delivery, Quality, Data, System and Integration principles that the organisation’s governance boards check for alignment to.

[06.3.2.3] Pattern Constraints

Discover whether the organisation’s has development, integration, and/or migration patterns that are expected to be followed.  
Again, governance boards are expected to check for alignment.

[06.3.2.4] Technology Constraints

Discover whether the organisation manages a list of technologies that it permits, recommends, recommends against or prohibits the use of, under specific circumstances.

[06.4] Reference

[06.4.1] Reference Terminologies

Develop Reference Documentation of Terminologies

[06.4.01.01] Reference: Delivery Terms & and Acronyms Document

Develop a reference document that is the basis of terms used to delivery services – whether automated or manual.

[06.4.2] Delivery Guidance

[06.4.3] Discovery Guidance

[06.4.4] Definition Guidance

[06.4.5] Design Guidance

[06.4.6] Design Guidance

[06.4.7] Service Design Guidance

Develop or Reference existing service design guidance documents

[06.4.8] System Design Guidance

While it is inappropriate at this early stage to define how the service will be developed, it is appropriate to find, reference and or develop guidance on system design concepts.

Consider the following:

System Design Guidance: Design: Domain Design

System Design Guidance: Evolvability: System Capabilities

System Design Guidance: Discovery: Service Domain Naming

System Design Guidance: Performance: Caching

System Design Guidance: Performance: Data store Indexing

System Design Guidance: Evolvability: Data Schema Design Considerations

System Design Guidance: Interoperability: Messages to Consider

System Design Guidance: Tenancies versus Accounts

[06.5] Definition of Needs, Constraints, and Requirements

Documents business needs, user requirements, quality expectations, system capabilities, functional and non-functional requirements, and transitional conditions.

[06.5.1] Reference: Terminologies

Definition of terminologies used in industry, sector, organisation, department, et.

[06.5.1] Business Needs Definition

Captures strategic drivers, organisational mandates, and expected business outcomes.

[06.5.1] Delivery Channels

Examples, include considering whether to provide service via Mobile specific service client, falling back to a web client, falling back to impaired vision reader services, falling back to providing slower Phone based inquiries.   
Explains how the service supports users without access to the digital channel, such as phone, paper, or in-person options.

[06.5.2] User Persona Definition

Describes the human and system roles involved in the service, including user personas, administrative roles, and API clients.

[06.5.3] User Journey Mapping

Important for visualising pain points, goals, and system interactions.

[06.5.4] User Role Modelling: Definitions, Escalation, Delegation

Describe how roles relate to each other, including permission inheritance, delegation access, escalation logic, removal, processes.

[06.5.5] User Requirements Discovery

Captures expected user goals, interactions, pain points, and critical user journeys.

[06.5.6] Capability Modelling

Structures the required capabilities a service must deliver to achieve business and user needs.

[06.7] Market Analysis

[06.8] Options Analysis

[07] Solution Definition

[07.1] Preparation

[07.2] Definitions

[07.2.1] Terminologies

[07.2.1.1] Business Domain Terminology

[07.2.1.2] Delivery Terminology

[07.2.1.3] Infrastructure Terminology

[07.2.1.4] System Terminology

[07.2.1.4] System Integration Terminology]

[07.2] Requirements

[07.1.1] Business Requirements

[07.1.2] User Requirements

Define the requirements   
Examples of categories of system Users are Service Consumers, Service Providers, Customer Support Users, System Operators.   
Under each category are one or mor Roles of Users. For example under Service Consumers are service end users, but also service account admins, if the service is a SaaS that is consumed by multiple individual organisations, schools, etc.

[07.2.3] Quality Requirements

[07.2.4] Capability Requirements

[07.2.5] Define the Service Capabilities.

[07.1.3] System Non-Functional Requirements

Examples of non-functional service quality categories follow ISO-25010 guidance and are Functional Suitability, Performance Efficiency, Compatibility, Interaction Compatibility, Reliability, Security, Maintainability, Flexibility, Safety. Under each category are sub categories. Under each sub category are zero or more quality requirements defined to have SMART qualities.

Examples of non-functional service data qualities follow ISO-25012 guidance and are Accuracy, Timeliness, Consistency, Completeness, Credibility, Accessibility, Understandability, Conformity. Under each are zero or mor quality requirements defined to have SMART qualities.

Examples of non-functional service usage qualities follow ISO-25022 guidance and are Effectiveness, Efficiency, Satisfaction ,Freedom from Risk, Context Coverage. Each of these Categories contain sub categories Under each sub category are zero or more quality requirements defined to have SMART qualities.

[07.1.4] System Functional Requirements

Frames system-level behaviours, features, and processes needed to support capabilities.

Examples include requirements in one or more of the following categories, Discoverability, Routing, Access Control, Caching, Encoding, Rendering, Auditing.

[07.1.5] Transitional Requirements

Captures migration, legacy, interim-state, and governance constraints affecting delivery.

[06.4] Solution Description

Synthesises findings into a structured high-level design, options analysis, and traceability model.

[07.3.1] Solution Options and Analysis

Explores solution approaches, pros/cons, and alignment to needs and constraints.

[07.3.2] High-Level Architecture Framing

Sketches indicative structures of how the solution components fit together.

[07.3.3] Traceability Mapping

Links business needs, user requirements, capabilities, system features, and quality expectations for later procurement and delivery validation.

[08] Solution Design Response

Develops internal or vendor service proposals aligned to solution needs.

[08.1] Preparation

[08.2] Solution Response Investigation

Study available market options and alignment.

[08.3] Solution Response Preparation

Frames internal or vendor responses to documented needs, constraints, and requirements.

[08.3.1] Response Structuring

Structures the response to clearly map to the Solution Definition artefacts, ensuring traceability.

[08.3.2] Compliance Self-Assessment

Documents internal assessment of how well the proposed solution meets business, user, quality, capability, and transitional requirements.

[08.3.3] Risk Identification in Response

Highlights risks, assumptions, exclusions, and mitigation strategies included in the proposed solution.

[08.3.4] Commercial Modelling

Outlines indicative commercial terms, pricing models, licensing assumptions, and service commitments.

[08.4] Solution Alignment and Internal Review

Validates alignment of the response to strategic, user, and technical expectations before submission or finalisation.

[08.4.1] Internal Architecture and Technical Review

Assesses the coherence, scalability, security, and technical soundness of the proposed solution.

[08.4.2] Strategic and Business Alignment Review

Validates that the solution advances strategic objectives, delivers intended benefits, and respects known constraints.

[08.4.3] Readiness for Procurement or Build Decision

Confirms whether the response is robust enough to proceed to procurement, evaluation, or direct build (for internal delivery models).

[09] Procurement Preparation

Structures approach-to-market and prepares procurement artefacts.

[09.1] Preparation

[09.2] Procurement Strategy and Planning

Defines procurement objectives, thresholds, governance model, and method (e.g., RFP, RFQ, direct engagement).

[09.2.1] Procurement Threshold Analysis

Confirms applicable financial thresholds and required process formalities.

[09.2.2] Procurement Governance Setup

Establishes evaluation panels, probity oversight, approvals, and sign-off gates.

[09.2.3] Approach to Market Structuring

Defines approach format, timeframes, information expectations, and supplier interaction protocols.

[09.2] Tender Documentation Preparation

Prepares the formal documentation required for vendor engagement.

[09.2.1] Solution Architecture Documentation Finalisation

Prepares the final externally shareable Solution Architecture Document (SAD) tailored for procurement purposes.

[09.2.2] Tender Response Requirements Specification

Defines the structure, format, and minimum content required in vendor responses.

[09.2.3] Evaluation Criteria and Weighting Definition

Documents the evaluation criteria, scoring methodology, weightings, and mandatory requirements.

[09.2.4] Draft Contract and Terms Finalisation

Prepares draft contractual documents including Statements of Work (SOW), licensing conditions, support models, and service levels for review by respondents.

[09.3] Internal Procurement Approvals

Secures governance approvals to proceed to market.

[09.3.1] Probity and Legal Review

Confirms that documentation and processes meet fairness, transparency, and legal compliance expectations.

[09.3.2] Governance and Sponsorship Endorsement

Obtains endorsement from sponsor or delegated authorities to proceed to market issuance.

[10] Vendor Selection

Validates, evaluates, and selects preferred vendor solutions.

[10.1] Preparation

[10.2] Response Validation

Confirms responses meet minimum eligibility, completeness, and compliance requirements.

[10.2.1] Eligibility Screening

Checks that vendors meet mandatory conditions (e.g., deadlines, financial viability, required certifications).

[10.2.2] Completeness Review

Verifies that responses address all sections, formats, and artefacts requested.

[10.2.3] Alignment Pre-Check

Assesses whether proposals broadly align to business needs, user needs, capability expectations, and constraints before detailed evaluation.

[10.3] Response Evaluation

Scores and moderates vendor responses against published evaluation criteria.

[10.3.1] Scoring Coordination

Organises evaluators, captures individual scoring, and manages clarification requests if allowed.

[10.3.2] Moderation and Consensus

Conducts moderation meetings to resolve score differences and achieve evaluation team consensus.

[10.3.3] Compliance, Quality, and Risk Reviews

Validates scoring outcomes against compliance, non-functional requirements, service continuity, and transition risks.

[10.4] Recommendation and Governance Approval

Prepares recommendation reporting and secures governance endorsement of preferred vendor.

[10.4.1] Evaluation Reporting

Summarises evaluation findings, moderation outcomes, key differentiators, and risks in a recommendation report.

[10.4.2] Governance Decision and Recording

Secures formal approval to award or negotiate based on evaluation results, recording sponsor sign-off.

[11] Delivery Preparation & Enablement

Finalises contracts, resourcing, environment setup, and team onboarding to enable structured service delivery.

[11.1] Preparation

[11.2] Contract Finalisation

Finalises contracts, licensing agreements, and statements of work needed to formally authorise delivery activities.

[11.3] Resourcing

Secures internal staff and external vendor resources, aligning roles, availability, and responsibilities for delivery phases.

[11.4] Environment and Licensing Enablement

Prepares delivery and development environments, configures licensing, access, and systems needed for service implementation.

[11.5] Delivery Orientation and Training

Prepares delivery teams and stakeholders by providing inductions, tool access, role onboarding, and initial delivery training.

[12] Service Delivery Implementation

Establishes delivery automation, environment readiness, and system platform structures to support efficient service build.

The work is sufficiently complex as to be a system in its own right, with design, storage, configuration, and integration implementation considerations.

[12.2] Preparation

[12.1] System of Delivery Enablement

Implements automation pipelines, CI/CD processes, artefact packaging rules, and build-test-deploy environments for structured delivery.

[13] Service Implementation

Builds, integrates, configures, and deploys the core service functionality and interfaces for operational use.

[13.1] Preparation

[13.2] System Data Implementation

Establishes structured data models, controlled vocabularies, and initial system data loads.

[13.3] System Media

Prepares and integrates system-embedded media assets including text, images, audio, and compliance content.

[13.3.11] Logo Images

[13.3.12] Background Images

[13.3.21] Cookie/Tracking Disclosure

[13.3.22] Data Use Purpose, Correction and Sharing Disclosure

[13.3.23] System Terms & Conditions

[13.3.31] Labels (internal)

[13.3.32] Messages (external)

[13.3.41] Audio

[13.3.51] Culture-Language Packs

[13.4] Core Services Build

Develops and configures the foundational application components according to solution requirements.

[13.4.01] Component Layout

[13.4.01] Logical Schema

[13.4.02] Data Schema

[13.4.01] Service Domain

* + Maintainability Domain:
    - [13.4.01] Configuration
    - [13.4.02] Integrations – Diagnostics Storage
    - [13.4.03] Integrations – Data Storage
    - [13.4.04] Integrations – Caching
  + Configurability Domain
    - [13.4.05] System Settings
  + Access Domain
    - [13.4.06] Sessions
    - [13.4.07] Permissions
  + Endpoints Domain
    - [13.4.08] Routes

[13.4.02] Social Domain

* + [13.4.01] Users
  + [13.4.02] Relationships
  + [13.4.03] Groups
  + [13.4.04] Roles

[13.4.03] Registries Domain

* + [13.5.01] Registries

[13.4.04] Aspirations Domain

* + [13.6.01] Aspirations
  + [13.6.02] Milestones

[13.4.05] Work Domain

* + [13.7.01] Tasks
  + [13.7.02] Projects

[13.4.60]

[13. 05] Secure Service Integrations

Implements secure connections to authentication, messaging, storage, and analytics services.

[13. 06] Data Migrations

Develops data pipelines for external integrations, reporting, and warehousing needs.

[13. 06] Service Monitoring

Implements operational observability including health checks, alerting, and performance tracking.

[13. 08] Service Interoperability Enablement

Publishes APIs, SDKs, and sandbox environments to allow external systems to connect safely.

[13.09] Service Client User Interface

Develops the front-end user interfaces for web, mobile, or other digital interaction points.

[13.10] Service Registration and Discovery

Publishes metadata, service descriptions, and discovery artefacts to internal and external directories.

[14] Operational Readiness Assessment

Covers verifying the service's internal and external readiness for real-world launch, including quality assurance, compliance, user orientation, distribution planning, and evaluation.

[14.1] Service Orientation and Support Artefacts

Develops onboarding and support materials tailored to different user and stakeholder groups.

[14.2] Distribution and Uptake Planning

Plans how the service will be distributed to users and stakeholders, including access models and rollout scheduling.

[14.3] Marketing and Communications

Prepares internal and external communications campaigns to raise awareness and support launch success.

[14.4] Account Management

Prepares account structures, onboarding processes, and support mechanisms for ongoing customer management.

[14.5] System Quality Review

Conducts quality assurance activities to verify functional, usability, performance, and non-functional attributes.

[14.6] Service Compliance and Assurance

Confirms that legal, privacy, security, and accessibility obligations are met to an auditable standard.

[14.7] Service Evaluation

Assesses the service's internal readiness, residual risks, and completeness ahead of formal launch approvals.

[14.8] Gateway Review

Optionally support external reviews – e.g. a Gateway Reviews.

[15] Operational Service Management

Covers the day-to-day running, maintenance, evolution, and evaluation of the service once live, ensuring it remains effective, compliant, and valuable.

[15.1] Consumer Support Operations

Handles user inquiries, support tickets, issue triage, and customer communication.

[15.2] Provider Support Operations

Supports partner agencies, vendors, and intermediaries who deliver parts of the service.

[15.3] Live Operations and Scheduling

Manages routine operational jobs, automated tasks, environment health, and service availability.

[15.4] Incident Management

Manages unplanned interruptions, rapid diagnostics, mitigation actions, and recovery efforts.

[15.5] Problem Management

Identifies root causes behind repeated incidents and implements structural fixes to prevent recurrence.

[15.6] Change Management

Processes requests for enhancements or fixes, assesses impacts, coordinates approvals, and schedules releases.

[15.7] Maintenance

Executes regular patching, technical debt remediation, documentation updates, and minor service improvements.

[15.8] Continuous Improvement

Captures user feedback, evaluates emerging needs, and implements enhancements to sustain and grow service value.

[15.9] Operation Evaluation

Conducts periodic reviews of operational performance, effectiveness, resilience, and user satisfaction.

[16] Service Transition

Manages the planning, preparation, and execution of transferring, handing over, or retiring the service safely and accountably.

[16.1] Preparation

[16.2] Transition Evaluation

Assesses strategic intent, timing, risks, and options for transferring, handing over, or closing the service.  
Optionally align to Gateway Review requirements.

[16.4] Transition Preparation and Readiness

Prepares documentation, cleans up environments, fixes known issues, and ensures the service is fit for transition or closure.

[16.4] Transfer Target Identification and Vetting

Identifies and assesses potential recipients for service transfer, verifying capability, fit, and compliance readiness.

[16.5] Transition Execution

Carries out technical, operational, contractual, and communication activities needed to complete handover or shutdown.

[16.6] Post Execution Support

Provides optional limited-time support to recipients after transfer to ensure smooth stabilisation and handover completion.

[16.7] Transition Evaluation

Conducts a final review of the transition’s success, capturing lessons learned and unresolved actions.

[17] Closure and Exit

Finalises the retirement of the service through structured evaluation, decommissioning, and securing of data and compliance obligations.

[17.1] Preparation

[17.2] Final Evaluation

Assesses the effectiveness, efficiency, and strategic outcomes of the service lifecycle before closure.

Conducts formal closure assessments, providing inputs to closure reviews or Gateway-type assurance activities.

[17.3] Closure Planning

Defines exit strategy, schedules, data management, compliance requirements, and stakeholder communications for shutting down the service.

[17.4] Decommissioning and Exit

Executes the technical, operational, and contractual steps to securely close the service and hand over any retained obligations.

[17.5] Exit Evaluation

Captures final lessons learned, confirms data retention, and completes reporting required for formal closure.

Appendices

Appendix A - Document Information

Authors & Collaborators

* Sky Sigal, Solution Architect

### Versions

* 1. Initial Draft

### Images

**No table of figures entries found.**

### Tables

**No table of figures entries found.**

### References

**There are no sources in the current document.**

### Review Distribution

The document was distributed for review as below:

|  |  |
| --- | --- |
| Identity | Notes |
|  |  |
|  |  |
|  |  |

### Audience

The document is technical in nature, but parts are expected to be read and/or validated by a non-technical audience.

### Structure

Where possible, the document structure is guided by either ISO-\* standards or best practice.

### Diagrams

Diagrams are developed for a wide audience. Unless specifically for a technical audience, where the use of industry standard diagram types (ArchiMate, UML, C4), is appropriate, diagrams are developed as simple “box & line” monochrome diagrams.

### Acronyms

API

: [Application Programming Interface](#Term_ApplicationProgrammingInterface).

DDD

: Domain Driven Design

GUI

: [Graphical User Interface](#Term_ApplicationProgrammingInterface). A form of [UI](#Acronym_UI).

ICT

: acronym for Information & Communication Technology, the domain of defining Information elements and using technology to automate their communication between entities. [IT](#Acronym_IT) is a subset of ICT.

IT

: acronym for Information, using Technology to automate and facilitate its management.

UI

: User Interface. Contrast with [API](#Acronym_API).

### Terms

Refer to the project’s Glossary.

Application Programming Interface

: an Interface provided for other systems to invoke (as opposed to User Interfaces).

Capability

: a capability is what an organisation or system must be able to achieve to meet its goals. Each capability belongs to a domain and is realised through one or more functions that, together, deliver the intended outcome within that area of concern.

Domain

: a domain is a defined area of knowledge, responsibility, or activity within an organisation or system. It groups related capabilities, entities, and functions that collectively serve a common purpose. Each capability belongs to a domain, and each function operates within one.

Entity

: an entity is a core object of interest within a domain, usually representing a person, place, thing, or event that holds information and can change over time, such as a Student, School, or Enrolment.

Function

: a function is a specific task or operation performed by a system, process, or person. Functions work together to enable a capability to be carried out. Each function operates within a domain and supports the delivery of one or more capabilities.

Person

: a physical person, who has one or more Personas. Not necessarily a system User.

Persona

: a facet that a Person presents to a Group of some kind.

Quality

: a quality is a measurable or observable attribute of a system or outcome that indicates how well it meets expectations. Examples include reliability, usability, and performance. Refer to the ISO-25000 SQuaRE series of standards.

User

: a human user of a system via its UIs.

User Interface

: a system interface intended for use by system users. Most computer system UIs are Graphics User Interfaces ([GUI](#Acronym_GUI)) or Text/Console User Interfaces (TUI).

Appendix B: Suggested Use of Numbering

Documents, folders, deliverables, and records may be prefixed with the appropriate number (e.g., "06.12 Integration Architecture", "14.03 Lessons Learned"). Each major folder or phase group can contain subfolders (e.g., "06 Design" → "06.10 Solution Design", "06.20 System Design", etc.).

Appendix C: Mapping to ISO/IEC 15288 (to be developed)

Appendix D: Mapping to Treasury BBC, MBIE Procurement, and GCDO Lifecycle (to be developed)

Appendix B – Framework Comparison

The **Service Enablement and Lifecycle Framework (SELF)** is designed to build upon and extend the strengths of existing government, industry, and international frameworks without replacing them.  
Where existing models focus effectively on funding, procurement, project delivery, operational management, or architecture development, SELF provides the missing structural continuity across the entire service lifecycle.  
It enables artefacts, decisions, models, and accountabilities to persist logically and visibly from early sensing through strategic framing, procurement, delivery, operational life, and eventual closure or transition.

The following sections position SELF against several established standards, highlighting alignment, points of difference, and complementary value.

### Treasury Better Business Case (BBC)

The BBC model, developed for New Zealand government agencies, structures the progression of investment cases from high-level strategy through to detailed justification, helping ensure decisions are well grounded before funding approval.

#### Recognised Strengths

BBC is highly effective for structuring investment planning. It ensures strategic alignment, disciplined justification, and systematic progression from strategic to detailed cases. It aligns funding with intended benefits and ensures sponsor accountability at decision points.

#### Limitations for Lifecycle Structuring

BBC ends at the funding decision point. It does not address technical delivery, operational setup, governance continuity, change management, or eventual service evaluation and closure. Artefact organisation past investment is unsupported.

#### Summary of Alignment

The Service Lifecycle Numbering Framework builds on the early strategic alignment captured by BBC but provides continuity beyond funding approval, ensuring artefacts are systematically managed across delivery, operation, and retirement.

### MBIE Procurement Lifecycle

The Ministry of Business, Innovation and Employment (MBIE) procurement guidance defines best practice for government vendor engagement, ensuring transparency, fairness, and contract management discipline during external service acquisition.

#### Recognised Strengths

MBIE's procurement guidance strengthens agency capability in vendor engagement, tendering, evaluation, and contract management. It introduces discipline around fairness, transparency, and commercial compliance during external acquisition.

#### Limitations for Lifecycle Structuring

The MBIE model is bounded to the procurement window. It does not structure artefacts outside tender management, nor address service implementation, operational transition, or support artefact governance across phases.

#### Summary of Alignment and Comparison

This framework accommodates MBIE procurement within a broader structure, ensuring procurement outputs are properly placed, governed, and connected to the full service lifecycle before and after market engagement.

### GCDO Service Design and Delivery Lifecycle

The Government Chief Digital Officer’s lifecycle promotes user-centred service design within New Zealand government digital initiatives, encouraging discovery, validation, and iterative service improvement.

#### Recognised Strengths

The Government Chief Digital Officer's (GCDO) lifecycle promotes good service design principles, including user-centred design, discovery validation, and iterative delivery. It brings important culture shift toward agile service creation.

#### Limitations for Lifecycle Structuring

The GCDO model focuses on service iteration but does not provide a file/folder governance model for artefacts, nor does it bridge well into long-term operational and decommissioning phases. It can struggle to enforce document continuity.

#### Summary of Alignment and Comparison

The Service Lifecycle Numbering Framework embeds GCDO-aligned service design principles but extends them into a structured, lifelong artefact governance system that supports both agile and traditional delivery models over time.

### ISO/IEC 15288 (System Life Cycle Processes)

ISO 15288 defines international standards for system lifecycle processes, framing best practice from concept definition through disposal, with particular emphasis on engineering and systems management disciplines.

#### Recognised Strengths

ISO 15288 provides comprehensive system life cycle processes covering concept, development, production, operation, maintenance, and disposal. It is internationally recognised and robust for system engineering disciplines.

#### Limitations for Lifecycle Structuring

While ISO 15288 describes processes, it does not prescribe artefact structuring, folder organisation, or documentation governance. It leaves teams to create their own traceability models.

#### Summary of Alignment and Comparison

This framework complements ISO 15288 by offering a ready artefact organisation structure aligned to the full lifecycle stages described, improving practical implementability without conflicting with ISO principles.

### PRINCE2

PRINCE2 is a globally recognised project management framework that structures project governance through distinct stages, formalising approvals, change control, and structured handovers across delivery phases.

#### Recognised Strengths

PRINCE2 formalises project management through clearly defined stages, roles, governance gates, and change control mechanisms. It is a respected standard for structured project delivery.

#### Limitations for Lifecycle Structuring

PRINCE2 focuses on project delivery only. It does not address artefact organisation before project initiation (conceptual exploration) or after closure (operational handover, support evolution, or decommission).

#### Summary of Alignment and Comparison

This framework respects PRINCE2 project structures but extends coverage both upstream into pre-project ideation and downstream into enduring service operation, maintenance, and closure phases.

### TOGAF (The Open Group Architecture Framework)

The Open Group Architecture Framework (TOGAF) provides a high-level methodology for developing, governing, and maintaining enterprise architecture, ensuring solutions align with organisational capabilities and strategies.

#### Recognised Strengths

TOGAF offers extensive guidance for enterprise architecture, especially in strategic capability alignment, solution framing, and governance via the Architecture Development Method (ADM).

#### Limitations for Lifecycle Structuring

TOGAF is abstract and strategic. It does not prescribe artefact or document structures suitable for project-level or service-level folder organisation or operational continuity beyond architecture governance.

#### Summary of Alignment and Comparison

The Service Lifecycle Numbering Framework operates at a lower layer than TOGAF, supporting practical service construction, operation, and documentation management within or outside an enterprise architecture context.

### IT4IT (Open Group IT Management Framework)

The IT4IT standard proposes an end-to-end IT value chain model, offering toolchain-based views for managing portfolios, requirements, delivery, and operations across IT organisations.

#### Recognised Strengths

IT4IT provides a value-chain based approach for structuring IT delivery and operations, from portfolio through request, fulfilment, and support, with strong artefact and operational traceability at an IT service management (ITSM) level.

#### Limitations for Lifecycle Structuring

IT4IT focuses on operational flow and service management artefacts but is heavily toolchain-oriented and less suited for general business project artefact management, especially in government settings where bespoke artefact production is the norm.

#### Summary of Alignment and Comparison

This framework complements IT4IT by ensuring all artefacts — strategic, operational, technical, support — are managed across the entire lifecycle without being dependent on specific toolchain ecosystems.

### P3M (Portfolio, Programme, and Project Management Framework – Audit NZ)

The Portfolio, Programme, and Project Management Maturity Model used by Audit NZ encourages government agencies to build structured capability across layered initiatives, ensuring oversight scales appropriately from individual projects to full portfolios.

#### Recognised Strengths

P3M distinguishes work at portfolio, programme, and project levels, encouraging structure and governance across complex and layered initiatives. It supports scaling and complexity management.

#### Limitations for Lifecycle Structuring

P3M focuses on management layers, not artefact organisation inside projects or services themselves. It does not address the need for durable, enduring artefact traceability across operational or maintenance phases.

#### Summary of Alignment and Comparison

The Service Lifecycle Numbering Framework can be applied within P3M structures, providing the missing artefact continuity inside projects and programmes, and ensuring smooth knowledge transition between portfolio oversight and operational management.

### MSP (Managing Successful Programmes)

Managing Successful Programmes (MSP) provides best practice guidelines for managing multiple related projects under a programme structure, aiming to deliver complex strategic change across organisations.

#### Recognised Strengths

MSP focuses on coordinating related projects under a programme structure to deliver strategic outcomes. It strengthens alignment to strategic goals, benefit management, and dynamic adaptation as priorities shift.

#### Limitations for Lifecycle Structuring

MSP structures governance and outcome delivery across multiple projects but does not address how to manage artefacts within a single service or long-term operational continuity post-programme.

#### Summary of Alignment and Comparison

This framework complements MSP by addressing lifecycle documentation and operational continuity needs for individual services, even after programme closure, ensuring no loss of artefacts or system context.

### PMBOK (Project Management Body of Knowledge)

The Project Management Body of Knowledge (PMBOK) is a globally recognised guide for project management professionals, providing detailed process groups, knowledge areas, and governance standards across the project lifecycle.

#### Recognised Strengths

PMBOK provides a comprehensive guide for project management processes, covering initiation, planning, execution, monitoring, controlling, and closure. It emphasises structured governance and predictable delivery.

#### Limitations for Lifecycle Structuring

PMBOK focuses on project activities and deliverables but does not prescribe durable artefact structures post-project. It assumes project artefacts are archived or handed off without a persistent numbering or organisational model.

#### Summary of Alignment and Comparison

The Service Lifecycle Numbering Framework augments PMBOK by ensuring artefacts produced during project execution remain structured and usable across operational and maintenance phases, not just archived after closure.

### SAFe (Scaled Agile Framework)

The Scaled Agile Framework (SAFe) enables large enterprises to extend agile practices across multiple teams and portfolios, aligning strategic intent with incremental delivery of working solutions at scale.

#### Recognised Strengths

SAFe offers strong mechanisms for scaling agile practices across large enterprises, aligning portfolio planning with agile delivery teams. It promotes collaboration, responsiveness, and alignment between strategic themes and working code.

#### Limitations for Lifecycle Structuring

SAFe focuses heavily on agile cadence and flow but does not address long-term artefact organisation, service decommissioning, or persistent documentation continuity beyond agile delivery cycles.

#### Summary of Alignment and Comparison

The Service Lifecycle Numbering Framework complements SAFe by providing a durable artefact governance model that persists beyond agile iterations and programme increments, ensuring traceability even across decades.

Appendix C – Framework Comparison Chart

The following provides a comparative mapping between the Framework and several established lifecycle and governance standards, including PRINCE2, TOGAF, IT4IT, P3M, P3O, PMBOK, MSP, and SAFe. Each framework offers strengths within particular domains, but none fully cover the end-to-end service lifecycle breadth addressed here.

The following table maps phases from the Service Lifecycle Numbering Framework against the coverage provided by each reference model, highlighting the completeness of the framework while showing areas where others may omit or narrow their scope. An "X" indicates substantial alignment or equivalent treatment, while a blank entry indicates no comparable or systematic treatment found in that framework.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Lifecycle Phase | This Framework | Prince2 | TOGAF | IT4IT | P3M | P3O | PMBOK | MSP | SAFe |
| Strategic Governance | – | X | X | X | – | – | X | X | X |
| Enterprise Portfolio Management | – | X | – | X | – | – | X | X | X |
| Innovation Strategy and Pipeline | – | – | – | X | – | – | – | – | X |
| Problem and Opportunity | – | – | – | – | – | – | – | – | – |
| 00’s: Foundational and External Coordination | X | – | – | – | X | X | – | – | – |
| 01 Published | X | – | – | – | X | X | – | – | – |
| 02 Exchange | X | – | – | – | X | – | – | – | – |
| 03 Cross-Cutting Governance | X | X | X | X | X | X | X | X | – |
| 04 Cross-Cutting Management | X | X | X | X | X | X | X | X | – |
| 10’s: Strategic Framing, Procurement Preparation | X | X | X | X | X | X | X | X | – |
| 11 Early Framing and Planning | X | X | X | X | X | X | X | X | – |
| 12 Financing | X | X | – | – | X | – | X | X | – |
| 13 Solution Discovery, Definition, Capability Design | X | X | X | X | X | X | X | X | X |
| 14 Procurement Preparation | X | X | – | – | X | – | X | – | – |
| 20’s: Response Evaluation | X | X | – | – | X | – | X | – | – |
| 21 Tender | X | X | – | – | X | – | X | – | – |
| 22 Response Receipt and Registration | X | X | – | – | X | – | X | – | – |
| 23 Compliance Checking | X | X | – | – | X | – | X | – | – |
| 24 Clarifications and Corrections | X | X | – | – | X | – | X | – | – |
| 25 Evaluation Planning Finalisation | X | X | – | – | X | – | X | – | – |
| 26 Formal Evaluation Execution | X | X | – | – | X | – | X | – | – |
| 27 Moderation and Consensus | X | X | – | – | X | – | X | – | – |
| 28 Recommendation and Reporting | X | X | – | – | X | – | X | – | – |
| 30’s: System Development and Enablement | X | X | X | X | X | X | X | X | X |
| 31 System of Delivery | X | X | X | X | X | X | X | X | X |
| 32 System Data Implementation | X | X | X | X | X | X | X | X | X |
| 33 System Media | X | – | – | – | – | – | – | – | – |
| 34 Service Implementation | X | X | X | X | X | X | X | X | X |
| 35 Service Integrations | X | X | X | X | X | X | X | X | X |
| 36 Service Monitoring | X | X | X | X | X | X | X | X | X |
| 37 Service Interoperability Enablement | X | X | X | X | X | X | X | X | X |
| 38 Service Client UI | X | X | X | X | X | X | X | X | X |
| 39 System Registration and Discovery | X | X | X | X | X | X | X | X | X |
| 40’s: Customer Orientation and Enablement | X | – | – | – | – | – | – | – | – |
| 41 Service Orientation & Support Artefacts | X | – | – | – | – | – | – | – | – |
| 50’s: Distribution & Uptake | X | – | – | – | – | – | – | – | – |
| 51 Distribution & Uptake Planning | X | – | – | – | – | – | – | – | – |
| 52 Marketing & Communications | X | – | – | – | – | – | – | – | – |
| 53 Account Management | X | – | – | – | – | – | – | – | – |
| 60’s: Assessment and Readiness | X | X | X | X | X | X | X | X | X |
| 61 System Quality Review (QA) | X | X | X | X | X | X | X | X | X |
| 62 Service Compliance and Assurance | X | X | X | X | X | X | X | X | X |
| 63 Service Evaluation | X | X | X | X | X | X | X | X | X |
| 70’s: Post-Deployment Operation | X | X | X | X | X | X | X | X | X |
| 71 Service Consumer Support | X | X | X | X | X | X | X | X | X |
| 72 Service Provider Support | X | X | X | X | X | X | X | X | X |
| 73 Operations | X | X | X | X | X | X | X | X | X |
| 74 Change Management | X | X | X | X | X | X | X | X | X |
| 75 Maintenance | X | X | X | X | X | X | X | X | X |
| 76 Evaluation | X | X | X | X | X | X | X | X | X |
| 77 Continuous Improvement | X | X | X | X | X | X | X | X | X |
| 80’s: Transfer of Service | X | – | – | – | – | – | – | – | – |
| 81 Transition Evaluation | X | – | – | – | – | – | – | – | – |
| 82 Transition Preparation and Readiness | X | – | – | – | – | – | – | – | – |
| 83 Target Identification and Vetting | X | – | – | – | – | – | – | – | – |
| 84 Transition Execution | X | – | – | – | – | – | – | – | – |
| 85 Post Evaluation | X | – | – | – | – | – | – | – | – |
| 86 Post Transition Support | X | – | – | – | – | – | – | – | – |
| 87 Closure | X | – | – | – | – | – | – | – | – |
| 90’s: Closure & Exit | X | X | – | – | X | X | X | X | X |
| 91 Final Evaluation | X | X | – | – | X | X | X | X | X |
| 92 Decommissioning & Exit | X | X | – | – | X | X | X | X | X |

Appendix D – To Parse Through and Reuse

## [00] Whole of Service Lifecycle

### [03] Cross-Cutting Governance Concerns (Upwards):

This section addresses cross-phase decision-making and oversight. It includes records of architectural governance, review checkpoints, approvals, exceptions, and traceable decisions that guide the project across lifecycle stages. It includes periodic assurance activities such as Compliance and Assurance (C&A) reviews, which often raise issues requiring action. It includes records of architectural governance, review checkpoints, approvals, exceptions, and traceable decisions that guide the project across lifecycle stages.

Outputs include:

* [03.--] Identity Federation and Directory Integration: Describes how identities are managed across systems, including login providers, SSO, and identity handovers.
* [03.--] Role-Based Access Control (RBAC) Model: summarises how permissions are structured by role, to align with system functions and protect access.
* [03.--] Accessibility Assessment Summary: evaluates WCAG or NZ Government accessibility standards, identifying any barriers for users with impairments.
* [03.--] Equity and Inclusion Considerations: documents how the design addresses different user needs—e.g., digital literacy, rural access, language, cultural sensitivity.
* [04] Cross-Cutting Management Concerns (Inwards)

This section contains internally governed cross-cutting concerns — coordination, resourcing, and continuity — that apply throughout the lifecycle. While distinct from externally imposed governance, these structures are essential to delivery, assurance preparation, and team consistency.

This section supports coordination, delivery readiness, and organisational continuity across all lifecycle phases. It includes registries and administrative references that are not specific to any single phase, but are critical for aligning people, time, and responsibilities across agencies and vendors.

Outputs include:

## [12s] Strategic Planning, Budgeting, Financing and Setup

### [13] Early Framing and Planning

#### [13.1] Preparation (Concept Shaping)

Initial scoping activities that explore context, spark, or early drivers of the service. May include environmental scans, trigger events, or problem statements.

#### [13.2] Direction (Strategic Framing)

This section defines the early strategic positioning of the initiative. It captures agency mandates, sector priorities, early options, and framing from sponsors or ministers. This is not governance — it is upstream directional intent used to shape the purpose, scope, and expected alignment of what follows.

#### [13.3] Discovery (Early Exploration)

Initial research and exploration to understand stakeholders, possible futures, service gaps, or unstructured opportunity areas. Often precedes formal requirement definition.

#### [13.4] Definition (Needs and Constraints)

Captures what the initiative must achieve, who it must serve, and any known constraints (legal, ethical, technical, budgetary). Forms the foundation for the business case.

#### [13.5] Business Case Development (Framing, Iteration, Validation)

Collaborative shaping and refinement of the business case, including costings, benefit modelling, strategic fit, and investment logic.

#### [13.8] Outputs

Artefacts produced during early planning, including stakeholder analysis, preliminary business and user needs, and foundational inputs into the business case.

##### [13.81] Stakeholder Map

Identifies individuals, groups, and organisations affected by or involved in the service. Captures interests, influence, and engagement needs.

##### [13.82] Business Requirements

Defines what the service must achieve from an organisational and policy perspective, independent of implementation. Forms the basis for assessing relevance and value.

##### [13.83] High-Level Benefits

Summarises the intended positive outcomes, improvements, or efficiencies expected as a result of delivering the service.

##### [13.84] Cost Estimates

Initial cost modelling including indicative capital and operational costs. Used to support investment discussions and business case framing.

##### [13.85] Risk Register (Initial)

Early identification of strategic, operational, or delivery risks with proposed mitigations. Serves as a foundation for evolving risk governance.

##### [13.86] Draft Management Case

Describes delivery strategy, governance model, implementation timeline, and responsibilities. Used to test feasibility and readiness assumptions.

##### [13.87] Initial (unvalidated) User Requirements

Early articulation of user needs, expectations, pain points, and service interactions. Not yet validated, but helps guide discovery and design.

#### [13.9] Deliverables:

Endorsed outputs from early framing and planning, finalised for approval and use in downstream phases such as Financing and Procurement.

##### [13.91] Final Business Case Document (Signed)

Todo

##### [13.92] Benefits Realisation Strategy (Approved)

Todo

### [14] Financing (distinct from framing & planning)

Outputs   
Outputs include draft or confirmed budget models, funding scenarios, financial assumptions, and early investment plans used to inform Treasury or agency-level budget rounds.

#### [15.1] Preparation

Sets up internal roles, tools, reference structures and traceability methodology to support analysis, definition and solution design.

#### [15.2] Business Requirements Discovery & Definition

Formalise what the organisation needs to achieve, independent of technology choices.

#### [15.3] User Requirements Discovery & Definition

Capture what users need to accomplish, experience, or avoid, from their perspective.

#### [15.4] Workflow Scenarios & Journeys Discovery & Definition

Develop usage scenarios, workflows, and journey maps across roles and user types.

#### [15.5] Service Qualities & Non-Functional Requirement Discovery & Definition

Specify expectations for performance, reliability, accessibility, security, and other qualities (aligns with ISO-25010 thinking).

#### [15.6] Capability Identification and Structuring

Define the service and system capabilities required to meet the business and user needs (the bridge between what is needed and how it might be realised).

#### [15.7] System Functional Requirement Discovery & Definition

Organise the functional, non-functional, and transitional solution requirements into a coherent high-level design framing.

Note:  
At this stage, it is still solution-agnostic but structured enough to assess options.

#### [15.8] Transitional Service Delivery Requirement Discovery & Definition

Describes require governance, obligations, data and service migration aspects.

#### [15.8] Outputs:

##### [15.81] Solution Design - Reference / Guidance

Technical or business guidance documents that shape the approach to solution design. May include design principles, templates, or legacy learnings.

##### [15.82] Solution Design View - Background

Narrative context and strategic intent behind the solution. Captures sector goals, alignment rationale, and historical constraints.

##### [15.83] Solution Design View - Context

Describes the environment, systems, and relationships in which the service operates. Includes ecosystem mapping and interdependencies.

##### [15.84] Solution Design View - Constraints (legal, agreements, principles, patterns, technology, integrations)

Defines constraints that shape the solution space — including legal obligations, agreed standards, legacy system dependencies, and integration boundaries.

##### [15.85] Solution Design View - Information

Describes key data entities, flows, classifications, and stewardship across the solution. Supports later privacy, integration, and reporting design.

##### [15.86] Solution Design View - User Requirements

Summarises validated user needs and expectations, derived from discovery activities in [15.3-]. Serves as a concise reference for downstream design and procurement alignment. Serves as the human-centred basis for capabilities and functional design.

##### [15.87] Solution Design View - Capabilities Requirements

Summarises what the solution must enable, based on capability identification performed in [15.3-] and structured in [15.4-]. Used for design comparisons and traceability.

##### [15.88] Solution Design View - Quality Requirements

Defines the non-functional qualities the solution must meet, such as performance, reliability, scalability, usability, accessibility, and environmental constraints.

##### [15.89] Solution Design View - Selective Functional Summary

A concise summary of core business functionality the solution must support, informed by user and capability views but distinct from technical function listings.

##### [15.8a] Solution View - Transition Requirements

Specifies what must be true to move safely from legacy systems or states to the new service — including migration, changeover, and dependencies.

##### [15.8b] Solution Design View - Selective Traceability Summary (business-user-capabilities-selective functions)

Maps how business goals, user needs, and capabilities trace to key functional requirements — offering traceable justification for inclusion and evaluation.

##### [15.8c] Solution Design View - Integrations

Outlines required integrations with internal and external systems, including APIs, data exchanges, and authentication bridges.

### [16] Solution Architect Description (SAD)

## [20s] Solution Response Development

### [21] Response Preparation

### [22] Response Development

Covers the structured preparation of solution responses to the SAD and if external, the other Tender documents. It may involve internal solution teams, external vendors, or both, depending on the delivery model. Activities focus on producing an initial SDD that maps back to the business, user, capability, and quality requirements captured during discovery.

### [23] Response Validation and Alignment

This section captures internal validation activities against received SDDs (whether from internal teams or vendors).

It ensures proposed solutions meet the framed needs before proceeding to procurement evaluation or build initiation. Validation focuses on feasibility, alignment, risk exposure, and compliance with strategic, user, and system quality expectations.

## [30s] Tender and Response Management

Actions, artefacts and outputs produced include:

* RFP Package
  + Declaration of no Conflict of Interest
  + Vendor acknowledgement form
  + Solution Description
  + Business Requirements
  + User Requirements
  + Capabilities Requirements
  + Transitional Requirements
* Engagement Package
  + RFP Versions
  + Q&A Logs
  + Published Amendments
* Review Package
  + Review Process
  + Probity Process
  + Scoring Templates
* Reviewer Selection
* Approval (to publish RFP)

### [31] Procurement Preparation

This section includes all activities needed to prepare for and manage the procurement of a product or service. It involves documenting the approach to market, evaluation criteria, and governance processes required to engage external vendors.

Actions, artefacts and outputs produced include:

* organisation document templates
* collection and review of applicable organisation specific processes
* market research
* vendors to invite
* approach to market document
* evaluation criteria, weightings, scoring matrix
* tender response requirements
* draft contracts to accompany the RFP
* governance approval to go to tender from market

#### [31.1] Procurement Planning

Defines procurement strategy, objectives, governance responsibilities, and alignment with business case. May include MBIE templates, thresholds, and agency-specific procedures.

#### [31.2] Market Research and Approach to Market

Documents any early vendor scans, market sounding, or ROI/EOI processes. Describes rationale for using RFP, RFQ, or other methods.

#### [31.3] Procurement Documentation

Preparation of evaluation criteria, weighting, tender response requirements, and any draft contracts to accompany the RFP.

#### [31.4] Internal Procurement Governance

Records of review and approvals prior to market engagement — including legal review, probity sign-off, and cross-agency alignment where applicable.

### [32] Tender

This section covers the active phase of market engagement after procurement planning is complete. It includes the formal release of tender documents, handling of supplier clarifications, and the management of submitted responses.

#### [32.1] Preparation

Final activities prior to issuing the RFP, including packaging documents, internal approvals, and communication planning.

#### [32.2] RFP Issuance and Market Engagement

The release of the Request for Proposal or equivalent documents to selected vendors or the open market.

#### [32.3] Clarifications and Amendments

Tracks vendor questions and responses, as well as any formal amendments issued to the tender documents.

### [33] Response Evaluation

This section handles the structured review and scoring of responses received during the tender phase. It documents evaluation planning, team formation, scoring results, governance checks, and the final selection.

Actions, artefacts and outputs produced include:

* Moderation logs
* Evaluation memos,
* Draft Recommendation
* Evaluation Debrief Notes
* Final Evaluation Report

#### [33.1] Evaluation Planning

Defines evaluation criteria, weightings, conflict-of-interest processes, and review steps. May include probity protocols.

#### [33.2] Evaluation Team and Process

Details assigned reviewers, scoring methods, moderation steps, and record-keeping standards.

#### [33.3] Evaluation Results and Recommendation

Summarises scores, evaluation observations, preferred vendor justification, and draft recommendation to governance.

## [40s] System Development and Enablement

Outputs include:

* Code Repository
* Deployment Pipeline System
* Environments
* Automated Test Suite
* System Service
* Integration Documentation & SDK
* System Service Client (User Interface)
* Localised Media Packages (Images & Text)
* Discovery Subscriptions & Registration (Domain Name, Certs)

### [41] System of Delivery

This section defines the foundational infrastructure and transitional scaffolding required to support deployment. It includes automation, environment provisioning, deployment configuration, and artefact packaging. It represents the creation or configuration of a platform that enables continuous integration, test automation, versioning, and deployment. Often distinct from the system being delivered, this supports repeatable delivery and compliance with assurance requirements. If this system is not prepared first, later work in [41] and [42] risks inefficiency, delay, or rework.

Note:  
This system is often reusable across multiple system implementations and must precede [41] to avoid duplication or uncoordinated deployments. Dependencies include technical environment availability and early architectural scoping.

Dependencies:

* Cloud Services Subscription

Actions, Artefacts and Outputs include:

#### [40.1] Delivery Pipeline and Automation Setup

Defines the build, test, and deploy automation flows. Includes CI/CD configuration, artefact packaging rules, version control integration, and testing scaffolds.

#### [40.2] Environment Provisioning

Describes infrastructure creation (cloud or on-prem), environment setup, and infrastructure-as-code definitions to support deployment, testing, and demonstration.

#### [40.3] Transitional Requirements and Constraints

Outlines any known transitional delivery requirements, such as audit evidence, rollback requirements, pipeline compliance, version labelling, or non-standard controls.

#### [40.8] Outputs

Environment provisioning scripts, CI/CD configuration files, test runners, versioning documents, deployment rules, transitional constraints documentation.

### [42] System Data Implementation

This section establishes the structured data foundations required by the system prior to or in parallel with functional implementation. It includes the creation, sourcing, mapping, and versioning of core data definitions that underpin logic, form behaviour, analytics, and integrations. Data implementation precedes application logic in domains where the meaning of the system is encoded in classification schemes, reference data, or relational mappings.

This includes code lists, enumerations, controlled vocabularies, categorisations, identity types, address structures, qualification frameworks, and any domain-specific values with embedded semantics. Where external standards apply (e.g. NZ ethnicities, ISO 3166 countries, NZQA unit standards), it includes sourcing, aligning, and registering these appropriately.

Data structures must be implemented in a way that supports traceability, revision, and governance independent of the executing system.

*Note: This section exists separately from [42] System Implementation to acknowledge that system behaviour is often dependent on data logic. Data implementation may precede or iterate ahead of system delivery.*

#### [42.8] Outputs

* Versioned data dictionaries and controlled vocabularies
* Mapping sheets or transformation logic for external data
* Entity value registries (e.g. organisation types, funding levels)
* Scripts or automation to load initial values into environments

### [43] System Media

*This section addresses the planning, development, and integration of media artefacts that form part of the system itself — including content, legal text, graphics, icons, audio, video, or other embedded resources. These are distinct from documentation or training artefacts and are not external orientation materials; they are integral to the user experience of the system.*

System Media includes functional, informational, or compliance-related media required inside the application, such as terms of use, consent prompts, privacy notices, help overlays, multilingual labels, and embedded graphics. These artefacts are often neglected or delayed, despite being prerequisites for accessibility, legal compliance, and effective user communication.

Media elements must align with branding, accessibility, cultural expectations, and content governance standards. Coordination across legal, content, UX, and localisation teams is typically required. Translation workflows, screen content reviews, and content style guides also belong here.

Note: This section is distinct from [42] System Data, which addresses structural data models, reference data, and data behaviour; and from [49] Service Orientation & Support Artefacts, which covers onboarding and user-facing documentation after go-live.

#### 33.8 Outputs

* Media asset register
* Approved legal and policy text (terms, disclaimers, privacy statements)
* Localised content sets (e.g., te reo Māori, alternate languages)
* Accessibility-compliant screen text and overlays
* Embedded iconography and system images
* Translated UI strings for multilingual operation
* Approval records from legal, accessibility, or branding reviews

### [44] System Implementation

his section includes the implementation of the core system itself—whether through the configuration of a SaaS platform, custom software development done by a vendor or inhouse, or a combination thereof.

The implementation must conform to the design and transitional constraints established in earlier phases. In all cases, the system should be automated and reproducible where possible, avoiding manual configuration drift.

Implementation includes adding the Development View of the Solution Architecture Document (SAD)that has been being updated throughout prior phases.

It also includes the development of lower-level artefacts such as Technical Architecture Documents (TADs), interface specifications, and testable code modules.

Note:   
This work assumes that [41] System of Delivery and [44] Integration Planning have already prepared the delivery environment and clarified how this system will coexist or connect with others.

#### [44.8] Outputs

Finalised Development View in the SAD

Initial and updated TAD artefacts per component or integration point

Source-controlled configuration or codebase

Build pipeline and environment setup references

Technical reference documentation for testing and assurance

### [45] System Integrations

This section focuses on connecting the system to its required enterprise and external services, including authentication, messaging, analytics, and data exchange pipelines. These integrations enable the service to function within its operational ecosystem and avoid dependency bottlenecks post go-live.

While some integration planning may occur earlier (e.g. during [15] Design or [41] Transition Readiness), this section captures the deliberate execution of those tasks. It ensures not only technical connectivity but also operational preparedness — such as valid credentials, test data arrangements, monitoring hooks, and fallback behaviours.

Integration is typically undertaken after the core system is stable enough to expose integration points, but must be completed before broader Interoperability Enablement [45] is possible.

Note:   
Integration scope includes both direct technical interfaces and service-aligned business dependencies (e.g. links to agency portals, corporate sites, or shared document repositories).

#### [45.1] Integration Planning

Clarifies which integrations are required, confirms technical compatibility, identifies test data needs, defines security handovers, and sequences connection order based on criticality and availability.

#### [45.2] Technical Integration Execution

Implements and tests technical linkages — including authentication handoffs (SSO/IdP), system-to-system APIs, data pipelines (ETL or real-time), webhooks, and internal service bus connections.

#### [45.3] Dependency Stabilisation and Assurance

Verifies that all integrations perform as expected under simulated load, exception conditions, and degraded upstream scenarios. Includes retry strategies, monitoring checks, and governance for future updates or handovers.

#### [45.8] Outputs

Integration maps, test connection logs, credentials handover artefacts, retry/fallback specifications, data flow diagrams, and dependency escalation procedures.

#### [45.9] Deliverables

Signed-off integration test results, updated architecture views showing confirmed interfaces, and documented operating model inclusions (e.g. who manages tokens, certificates, routing).

### [46] System Monitoring

This section defines the operational observability structures necessary to verify the service is live, healthy, and behaving as expected. It includes the implementation of uptime tracking, response time measurement, alerting frameworks, logging pipelines, and threshold-based triggers that signal degraded or failing states. Monitoring may span multiple components, including the core system, its integrations, and supporting infrastructure. Implementation should align with non-functional requirements defined in earlier stages, and must be in place before enabling interoperability or wider service discovery.

Note: Reliable monitoring is a prerequisite for risk-managed system exposure. Without it, teams cannot guarantee SLA conformance, rapidly detect issues, or support incident response processes across agencies or vendors.

#### [46.8] Outputs

Monitoring dashboard specifications, health check endpoints, alerting rules, incident classification criteria, baseline thresholds, and integration with existing monitoring platforms (e.g., Azure Monitor, CloudWatch, Grafana, Prometheus).

### [47] System Interoperability Enablement

This section ensures that the service can be discovered, tested, and safely integrated by external systems or service clients. It includes publishing developer documentation, provisioning sandbox environments, releasing SDKs, defining API contracts, and managing external access controls.

#### [47.1] Delivery Planning

Details roles, resources, internal scheduling, vendor onboarding, and go-live criteria.

#### [47.2] Change Readiness and Communication

Prepares users and stakeholders for upcoming changes via communications, training, and behavioural alignment.

#### [47.3] Implementation and Integration Activities

Tracks technical setup, configuration, integration of systems, and environment readiness.

#### [47.4] Operational Transition Planning

Defines the steps to move from project to operations — including support model, knowledge handover, and ownership transitions.

#### [47.8] Outputs

Checklists, plans, support guides, cutover maps, readiness test outcomes.

#### [47.9] Deliverables

Approved release documentation, signed off transition readiness, and go-live approvals.

### [48] Service Client User Interface

### [49] System Registration and Discovery

This section focuses on making the system discoverable and navigable from the wider digital and public ecosystem. It includes the content, links, and metadata required for users, systems, and stakeholders to locate, understand, and engage with the service. This work is often overlooked but essential for uptake and correct usage, especially when multiple channels or enterprise platforms (e.g., corporate websites, service directories, government portals) must refer to the service.

System Discovery is distinct from Interoperability, which prepares the system for technical integration. Discovery focuses on visibility and orientation — ensuring the system is found, understood, and accessed appropriately. Activities may include publishing service pages on agency websites, aligning names and navigation with public expectations, and registering metadata with search services or federated directories.

Note: This stage often requires coordination with communications teams, translators, accessibility reviewers, and brand/comms governance groups. It may also involve design for fallback navigation (e.g., redirects from decommissioned services), and aligning the voice and style of external-facing materials.

#### [48.1] Planning

#### [48.8] Outputs

Public-facing media, updated corporate site pages, redirects from legacy services, SEO metadata, service registration with federated portals or search registries, and navigation mock-ups.

#### [48.9] Deliverables

Ensure links from System Discovery pages are in place,

Approved service listing on corporate/sector portals,

signed-off page content for external websites,

validated public-facing service artefacts (e.g., translated service overview PDFs, welcome kits, FAQs).

## [50s] Comms, Service Orientation and Customer Enablement

### [51] Service Orientation & Support Artefacts

This section captures the development of artefacts intended to support service understanding, ease of access, and confident use. It includes orientation materials designed for first-time or infrequent users, artefacts supporting non-digital interaction (e.g. community posters, print-ready handouts), as well as culturally or linguistically adapted resources. These artefacts are typically produced late in the implementation phase, once the service shape has stabilised, but are essential for uptake, equity, and successful handover to business-as-usual support models.

Note:   
This section is often mistakenly folded into training, operational support, or communications. It deserves separate treatment due to its cross-functional nature — spanning service design, education, content, accessibility, and cultural engagement. Many services fail to produce effective orientation materials, leaving users unsupported and staff overburdened with reactive help.

#### [51.1] Planning and Audience Identification

Identify the personas who require onboarding support,

Identify and analyse their likely pain points

identify, analyse and choose channels and formats appropriate for different groups (e.g. school admins vs. parents, students vs. teachers, regional iwi vs. general public)

#### [51.2] Artefact Development and Validation

Drafts and iteratively refines materials.   
Examples include animated explainer videos, visual walkthroughs, sample completed forms, translated welcome kits, user journey maps, and drop-in desk packs.

#### [51.3] Distribution and Handover

Coordinates the publishing, embedding, and hosting of artefacts across digital and physical environments.

Ensures frontline support teams have printable versions and update paths.

#### [51.8] Outputs

* Persona-specific onboarding kits
* Printable or distributable artefact sets
* Internal reference collections for training reuse
* Language translations and cultural adaptations
* Asset management logs for multimedia artefacts

#### [51.9] Deliverables

* Approved orientation packs, by audience
* Media artefact handover (with licensing and update notes)
* Communication assets embedded into support workflows

### [52] …

## [60s] Distribution Enablement

### [61] Distribution, Uptake and Relationship Planning

Covers Sales, Marketing Communications, Channel Development, Commercial Contracts and Renewals.

#### [61.1] Channel Strategy

Defines the mechanisms and platforms through which the service will be made available (e.g., internal portals, public directories, partner agencies, resellers if applicable).

#### [61.2] Signup, Subscription, Incentives and Termination Setup

Covers how users or organisations gain access (e.g., signup processes, registration systems, onboarding approvals, eligibility rules).

#### [61.3] Terms

Where applicable, defines the financial model, cost recovery, subscription pricing, free access rules, partnership funding agreements, or usage-based models.

#### [61.4] Pricing (Optional)

Includes pilot group onboarding, early access, staged rollout plans, and support readiness for first-wave adopters.

#### [61.5] Pre-Launch Enablement

Includes pilot group onboarding, early access, staged rollout plans, and support readiness for first-wave adopters.

#### [61.6] Launch Campaigns

Covers the planning and delivery of activities to announce availability, encourage initial signups, and set expectations for support, limits, or rollout pacing.

#### [61.7] Partnership & Channel Development (Optional)

Defines how external organisations may resell, refer, or bundle the service if applicable. Could include MOU templates, partnership management plans.

### [62] Marketing & Communications

Covers the structured planning, development, execution, and governance of service messaging, awareness, engagement campaigns, and media activities, both internal and external.

#### [62.1] Marketing Strategy

Defines the overall positioning of the service, key messages, target audiences, brand alignment, tone of voice, and differentiated value propositions.

#### [62.2] Messaging and Content Development

Prepares the actual media content — press releases, web copy, FAQs, blog posts, explainer videos, infographics, promotional artefacts, and translations — aligned to the marketing strategy.

#### [62.3] Channel Planning and Media Buys

Identifies what distribution methods will be used (e.g., agency newsletters, social media, government portals, third-party platforms) and negotiates or books any paid placements if applicable.

#### [62.4] Internal Communications

Ensures agency or sector staff are briefed, given talking points, FAQs, escalation procedures, and support materials so that public interactions are consistent and accurate.

#### [62.5] Campaign Execution

Covers launching campaigns, measuring early performance, adjusting messaging based on feedback, and handling public or partner engagement.

#### [62.6] Media and Issue Management

Prepares for public or media questions, drafts reactive Q&A documents, escalation pathways for misinformation, and issue handling playbooks.

### [63] Account & Relationship Management

Covers structured preparation and operational processes for managing inbound interest, subscriptions, purchases, or formal service onboarding — whether framed as "sales," "subscriptions," or another uptake model. Focused on ensuring professional, consistent, and auditable handling of service adoption.

#### [63.1] Sales/Uptake Planning

Defines uptake models (subscription, licensing, one-off acquisition), terms and conditions, eligibility criteria, pricing structures if applicable, and approval flows.

#### [63.2] Sales Enablement Resources

Creates internal sales kits, FAQs, case studies, ROI statements, subscription agreements, onboarding checklists, service brochures, and customer decision guides.

#### [63.3] Prospect and Lead Management

Defines how interest will be captured, tracked, and followed up — including CRM or intake systems, assignment of responsibility, and response timelines.

#### [63.4] Subscription and Onboarding Execution

Manages the formal acceptance of new customers/users, including identity checks, contract acceptance, account setup, technical onboarding, and early stage support.

#### [63.5] Account Management Setup

Prepares ongoing account management structures — including who owns relationship continuity, how issues are escalated, periodic check-ins, and optional upsell or service extension activities if relevant.

**[63] – Account Management**

* 51 – Channel Strategy
* 52 – Signup, Subscription, Incentives and Termination Setup
* 53 – Terms
* 54 – Pricing (Optional)
* 55 – Pre-Launch Enablement
* 56 – Launch Campaigns
* 57 – Partnership & Channel Development (Optional)

**63: Account Management**

* Uptake Planning
* Sales Enablement resources
* Prosepect and Lead Management
* Subscription and Onboarding Execution
* Account Management Setup

## [70’s] Assessment & Readiness

### [71] System Quality Review (QA)

This section supports structured verification that the implemented system meets the intended quality attributes. It includes functional verification, usability evaluation, performance testing, accessibility conformance, and other non-functional assessments required to ensure the system is fit for use prior to broader assessment or release decisions. While some quality checks may occur continuously within the delivery pipeline, others may require manual execution, reporting, or formal sign-off.

**Note:**   
QA should be viewed as distinct from compliance and assurance. QA confirms that the system works well. C&A confirm that it is allowed to operate.

#### [71.1] Quality Planning and Framing

Defines which quality attributes will be assessed, how testing will be conducted, and by whom. May draw from ISO-25010 categories (e.g. usability, reliability, efficiency), agency-specific standards, or prior defect trends.

#### [71.2] Quality Testing and Review Activities

Captures execution of planned tests or reviews. May include automated test runs, accessibility reviews, performance benchmarking, UI evaluations, and cross-device testing.

#### [71.3] Quality Reporting and Sign-Off

Summarises the results of QA activities. Includes test coverage reports, defect lists, summary dashboards, and any required sign-off from technical leads or QA managers.

#### [71.8] Outputs

QA planning documents, test scenarios, scripts, execution logs, automated pipeline results, manual test summaries, and stakeholder review notes.

#### [71.9] Deliverables

Quality Review Report, Signed QA Completion Statement, Accessibility Conformance Summary.

### [72] Service Compliance and Assurance

This section isolates the messy but essential C&A processes needed for approval to operate. It tracks formal assessments, remediation, and assurance activities across legal, security, and privacy domains.

#### [72.1] C&A Planning and Scoping

Defines the assurance boundaries, assessment types, and artefacts required.

#### [72.2] Review Coordination and Remediation

Organises assessment execution, tracks findings, and coordinates remediation and sign-off. Details roles, resources, internal scheduling, vendor onboarding, and go-live criteria.

#### [72.3] Change Readiness and Communication

Prepares users and stakeholders for upcoming changes via communications, training, and behavioural alignment.

#### [72.4] Implementation and Integration Activities

Tracks technical setup, configuration, integration of systems, and environment readiness.

#### [72.5] Operational Transition Planning

Defines the steps to move from project to operations — including support model, knowledge handover, and ownership transitions.

#### [72.8] Outputs

Checklists, plans, support guides, cutover maps, readiness test outcomes.

### [73] Service Evaluation

This section documents the system team’s internal assessment of whether the system, as built and configured, meets its intended design and readiness objectives. It includes structured reviews of solution completeness, alignment with user and business requirements, reliability under expected loads, and success against development and integration criteria. The output of this section supports formal governance steps such as CAB or C&A.

## [80’s] Post Deployment

### [81] Support

This section includes artefacts and structures needed to deliver live support, triage, and continuous user engagement.

#### [81.1] Support Planning and Service Design

Support model definition, onboarding of support roles, knowledge base preparation.

#### [81.2] Support Operations and Triage

Triage flowcharts, support logs, escalation records, user contact summaries.

#### [81.8] Outputs

Support playbooks, onboarding packs, response matrices.

#### [81.9] Deliverables

Signed-off support model, service desk go-live checklist, handover docs.

### [82] Operations

This section describes the day-to-day running of the service once live. It includes monitoring, scheduled operations, automation activities, and coordination of service performance.

#### [82.1] Operational Monitoring and Scheduling

Defines jobs, triggers, automated flows, monitoring rules, and system health indicators.

#### [82.2] Service Execution and Coordination

Describes how operations are conducted, who is responsible, how exceptions are handled, and reporting flows.

#### [82.8] Outputs

Logs, dashboards, automation maps, service inventory.

#### [82.9] Deliverables

Operations handbook, responsibility assignments, and escalation triggers.

### [83] Change Management

This section supports post-go-live enhancements, internal adjustments, or minor upgrades. It includes requests for change, approvals, and updates managed within a live or semi-live environment.

#### [83.1] Change Requests and Prioritisation

Logs of requested changes, internal enhancements, or vendor updates. Includes review and impact assessment.

#### [83.2] Change Approval and Coordination

Tracks governance decisions to accept, defer, or reject changes. Documents how and when changes will be applied.

#### [83.8] Outputs

Approved change requests, prioritisation logs, impact assessments.

#### [83.9] Deliverables

Endorsed change sets, version control updates, and change deployment records.

### [84] Maintenance

This section documents updates, patching, code maintenance, documentation refresh, and improvement cycles.

#### [84.1] Maintenance Planning

Covers technical debt registers, version plans, vendor patches, and internal continuous improvement.

#### [84.2] Update Execution and Review

Execution of patches or changes and post-change validation. Often coordinated with change control.

#### [84.8] Outputs

Maintenance logs, versioning sheets, improvement registers.

#### [84.9] Deliverables

Published updates, patch schedules, updated user guides.

### [85] Evaluation

This section includes structured reviews to assess the effectiveness, efficiency, and value of the service post-deployment. It supports lessons learned, strategic alignment, and future investment decisions.

#### [85.1] Post-Implementation Review

Captures reflection on implementation success, delivery challenges, and immediate stakeholder feedback.

#### [85.2] Benefits Realisation and Outcome Measurement

Assesses whether stated benefits have been achieved and if service objectives remain valid.

#### [85.8] Outputs

Review notes, outcome metrics, user feedback summaries, change recommendations.

#### [85.9] Deliverables

Formal evaluation reports, lessons learned documentation, benefits realisation updates.

#### [86] Continuous Improvement

This section documents improvement efforts undertaken during the mature service phase, including refinements, innovation pilots, user-driven enhancements, and emerging technology adoption.

##### [86.1] Improvement Backlog and Ideation

Log of candidate improvements, innovations, or enhancements sourced from users, analytics, or strategy.

##### [86.2] Evaluation and Selection

Frameworks or decisions used to prioritise and approve improvement efforts.

##### [86.8] Outputs

Improvement plans, decision notes, user feedback loops.

##### [86.9] Deliverables

Implemented change packages, updated service models, new capability releases

## [90’s] Transfer or Closure, & Exit

### Transfer of Service

#### [91] Transition Evaluation

Define why, when, and how the service will be transferred. Covers strategic intent, risks, governance, and preliminary stakeholder alignment.

#### [92] Transition Preparation and Readiness

Execute preparation activities. Update documentation, fix known defects, clean environments, tidy licensing, validate artefacts — making the service "presentable" for transfer.

#### [93] Target Identification and Vetting

Identify, approach, and assess potential recipients. May include internal agencies, private sector buyers, or even public sunset plans. Evaluate their capability, compliance, intent.

#### [94] Transition Execution

Coordinate technical, operational, contractual, and communications work to transfer ownership. Includes live handover, access migrations, operational independence cutover.

#### [95] Post-Transition Support and Closure

Optional support phase for handover assistance, troubleshooting, and then formal final closure of responsibilities.

### Closure of Service

#### [91] Final Evaluation

This section includes structured reviews to assess the effectiveness, efficiency, and value of the service post-deployment. It supports lessons learned, strategic alignment, and future investment decisions.

##### [91.1] Post-Implementation Review

Captures reflection on implementation success, delivery challenges, and immediate stakeholder feedback.

##### [91.2] Benefits Realisation and Outcome Measurement

Assesses whether stated benefits have been achieved and if service objectives remain valid.

##### [91.8] Outputs

Review notes, outcome metrics, user feedback summaries, change recommendations.

##### [91.9] Deliverables

Formal evaluation reports, lessons learned documentation, benefits realisation updates.

#### [92] Decommissioning, Closure and Exit

This section documents the managed wind-down of a system, including data migration, support withdrawal, stakeholder transition, and record archiving.

##### [92.1] Decommissioning Planning

Describes strategy for exiting the service, including timing, dependencies, approvals, and alternatives.

##### [92.2] Transition to Successor or Archive

Documents technical or operational handover, data archival, retention policies, and regulatory closure.

##### [92.8] Outputs

Exit checklists, legacy documentation, successor system links.

##### [92.9] Deliverables

Decommissioning sign-off, closure reports, and confirmed data retention records.

## Areas and Aspects

[01] Cross Cutting Management

[01.1] Cross-Cutting Governance

: governance oversight, mandates, and structures.

[01.2] Cross-Cutting Management

: Internal. Coordinate of roles, deliverables, budget, task management.

[01.3] Risk and Compliance Management

: risk identification, compliance tracking, and mitigation planning.

[01.4] Structured Review Planning

Plans formal review checkpoints across the service lifecycle, including internal assurance reviews and external investment reviews such as Gateway Reviews, ensuring alignment with strategic, delivery, operational, and closure validation milestones.

[02] Environmental Sensing

[02.2] Early Sector/Problem Sensing

: detect early needs, risks, and change drivers.

[03] Strategic Framing

[03.2] Early Direction Setting

: shape early strategic purpose, scope, and positioning.

[03.3] Strategic Financing Planning

: Define funding strategies, budget frameworks, financial alignment.

[04] Project Financing

This section defines the process of identifying, securing, and structuring funding for the initiative. It includes both internal and external budget considerations, investment models, funding sources, and key financial governance artefacts.

**Outputs:**

Draft or confirmed budget models, funding scenarios, financial assumptions, and early investment plans used to inform Treasury or agency-level budget rounds.

* **Preparation**
* Activities focused on engaging finance stakeholders, identifying budget envelopes, preparing investment briefings, and understanding funding constraints.
* **[04.1] Cost Estimation**  
  Rough-order magnitude costing and refined cost models based on emerging solution understanding.
* **Funding Strategy Development**  
  Define how funding will be sought — internal baseline, capex bids, operating budget allocations, external grants, partnerships.
* **Investment Logic Mapping (Optional)**
* Develop an investment logic map if required by governance frameworks (like BBC or internal models).
* **Budget Submission Preparation**
* Prepare documentation, briefing notes, forecasts, and Treasury templates as required to seek funding.
* **Funding Negotiation and Approval**
* Engage with finance teams, leadership, Treasury (if applicable) to advocate for the investment and negotiate adjustments.
* **Funding Allocation and Baseline Setup**
* On funding approval, define how funding is allocated across phases and manage financial baselines for tracking.
* **Financial Governance Planning**
* Set up cost tracking, reporting requirements, financial change management, benefits realisation tracking.

### [05] Project Preparation

#### [04.1] Project Framing

Define project purpose, scope, and high-level plan.

[04.2] Project Setup

Establish governance, stakeholders, and delivery foundations.

[06] Solution Definition

This Areas Artefacts include:   
Stakeholder Map, Requirements Source SME Catalogue, Business Requirements (Why [Threat/Opportunity], Change [Optimisation, Creation, Reduction]) Of [Market, reputation, value, service, quality cost recovery risk]), Personas, User (all stages) Requirements, Guidance Documents, Domains, Service Quality Requirements (ISO-25022,25012,25010) tiered by industry, sector, organisation, project, Capabilities Required, Transitional Project requirements.

[06.1] Context Discovery

Explore constraints, legacy, dependencies, and sector needs.

#### [06.3] Definition

Define business needs, user requirements, quality expectations, capabilities, and project transitional conditions.

**[06.4] Options Analyse**

TODO

**[06.5] Solution Description**

Describe the high-level solution concept and options.

[08] Service Response

**[08.1] Preparation**

*[Optional]* Internal system shaping or RFP response development.

[08.3] Response Alignment

validate alignment of response to strategy, needs, and constraints.

[09] Procurement Preparation

[09.2] Procurement Preparation

Prepares approach-to-market, procurement, evaluation method, RFP parts.

[09.2] Tender Management

Manages invitations, the RFP release, clarification requests.

[10] Vendor Selection

* **[10.2] Response Validation**  
  Validate internal or vendor response for eligibility, strategic, completeness, constraint, technical and delivery alignment.

[10.3] Response Evaluation

: manages moderation and scoring.

[10.4] Recommendation and Governance Approval

: final vendor decision

[11] Delivery Preparation

[11.2] Contract Finalisation

: finalises terms, agreements, contracts, statements of work to start work.

[11.3] Resourcing

: secure internal and/or vendor resourcing for service delivery execution.

[11.4] Environment and Licensing Enablement

: Enable hosts, infrastructure, platforms, products, licenses to begin delivery.

[11.5] Delivery Orientation and Training

Induct, tool familiarise, and train teams, vendors, and stakeholders.

[12] Delivery Enablement

* **[12.1] System of Delivery Enablement**   
  Establish confidential integration credential storage, code repository protection, automation pipelines, CI/CD, structures, processes, test data, running, and reporting.

[13] Service Implementation

[13.1] System Data Implementation

Develop reference data, master data structures.

[13.2] System Media

Content, Images, legal, messages -- for Service UI, corporate website, etc.

[13.3] Core Services Build

Solution Development.

[13.4] Secure Service Integrations

Automated configuration of service to connect to dependencies - services for diagnostics, integration credential storage, system data storage, media storage, monitoring, caching, authentication, messaging, search, geolocation, AI, etc.

[13.5] ETL

External supporting processes (e.g. ETL to data warehouse, batch, etc.)

[13.6] Service Monitoring

Set up health, logging, diagnostics, and observability tooling.

[13.7] Service Interoperability Enablement

Expose versioned DTO’ed APIs, external developer onboarding material (SDK, online documentation and trial)

[13.8] Service Client User Interface

Deliver web front end (e.g. SPA), mobile UI, etc.

[13.9] Service Registration and Discovery

Ensure service can be found, indexed, and linked externally (DNS, SEO, Corp Website, etc.).

[14] Operational Readiness Assessment

[14.1] Service Orientation and Support Artefacts

Prepare onboarding guidance, help materials

[14.2] Distribution and Uptake Planning

: plan launch channels and account migration/onboarding.

[14.3] Marketing and Communications

: promote service to internal and external audiences

[14.4] Account Management

: prepare for customer and partner accounts signup/refund/cancellation

**[14.5] System Quality Review**

: validate against ISO-25022, ISO-25012,ISO-25010 criteria

**[14.6] Service Compliance and Assurance**

: assess privacy, security, legal, and policy readiness

**[14.7] Service Evaluation**

: assess operational readiness for release or production cutover.

[15] Operational Service Management

**[15.1] Consumer Support Operations**

Manage helpdesk and user support flows.

**[15.2] Provider Support Operations**   
Support partner and intermediary service providers

**[15.3] Live Operations and Scheduling**   
Operate the service and manage job orchestration

**[15.4] Incident Management**   
Respond to service interruptions and outages.

**[15.5] Problem Management**   
identify and eliminate root causes of recurring incidents

**[15.6] Change Management**   
Govern change requests, approvals, and releases.

**[15.7] Maintenance**   
Apply patches, updates, and technical improvements.

**[15.8] Continuous Improvement**   
Gather feedback and incrementally improve the service

**[15.9] Operation Evaluation**

Cyclically review and assess service operation health.

[16] Service Transition

**[16.1] Transition Evaluation**

Decide when and why service transition or shutdown is needed.

**[16.2] Transition Preparation and Readiness**

Cleanup/hygiene, documentation, defects, dependency licensing & subscription.

* **[16.3] Transfer Target Identification and Vetting**
* Assess and approve new service owners if transferring.
* **[16.4] Transition Execution**
* Complete the handover or closure activities.
* **[16.5] Post Execution Support**
* Provide temporary support after transfer/closure.
* **[16.6] Transition Evaluation**
* Review success and lessons from transition or closure.

[16] Service Closure and Exit

[16.1] Final Evaluation

* Conduct a formal closure review and lessons learned.

[16.2] Closure Planning

* Plan secure and complete closure steps.

[16.3] Decommissioning and Exit

* carry out shutdown, data handover, and system disposal.

[16.4] Exit Evaluation

* Assess the outcomes and governance of the exit.